Massive Data Sets

Proceedings of an Workshop

Committee on Applied and Theoretical Statistics
Board on Mathematical Sciences
Commission on Physical Sciences, Mathematics, and Applications
National Research Council

National Academy Press
Washington, D.C. 1996

NOTICE: The project that is the subject of this report was approved by the Governing Board of the National Research Council, whose members are drawn from the councils of the National Academy of Sciences, the National Academy of Engineering, and the Institute of Medicine.
The National Academy of Sciences is a private, nonprofit, self-perpetuating society of distinguished scholars engaged in scientific and engineering research, dedicated to the furtherance of science and technology and to their use for the general welfare. Upon the authority of the charter granted to it by the Congress in 1863, the Academy has a mandate that requires it to advise the federal government on scientific and technical matters. Dr. Bruce Alberts is president of the National Academy of Sciences.
The National Academy of Engineering was established in 1964, under the charter of the National Academy of Sciences, as a parallel organization of outstanding engineers. It is autonomous in its administration and in the selection of its members, sharing with the National Academy of Sciences the responsibility for advising the federal government. The National Academy of Engineering also sponsors engineering programs aimed at meeting national needs, encourages education and research, and recognizes the superior achievement of engineers. Dr. William A. Wulf is interim president of the National Academy of Engineering.
The Institute of Medicine was established in 1970 by the National Academy of Sciences to secure the services of eminent members of appropriate professions in the examination of policy matters pertaining to the health of the public. The Institute acts under the responsibility given to the National Academy of Sciences by its congressional charter to be an adviser to the federal government and, upon its own initiative, to identify issues of medical care, research, and education. Dr. Kenneth I. Shine is president of the Institute of Medicine.
The National Research Council was organized by the National Academy of Sciences in 1916 to associate the broad community of science and technology with the Academy's purposes of furthering knowledge and advising the federal government. Functioning in accordance with general policies determined by the Academy, the Council has become the principal operating agency of both the National Academy of Sciences and the National Academy of Engineering in providing services to the government, the public, and the scientific and engineering communities. The Council is administered jointly by both Academies and the Institute of Medicine. Dr. Bruce Alberts and Dr. William A. Wulf are chairman and interim vice chairman, respectively, of the National Research Council.
The National Research Council established the Board on Mathematical Sciences in 1984. The objectives of the Board are to maintain awareness and active concern for the health of the mathematical sciences and to serve as the focal point in the National Research Council for issues connected with the mathematical sciences. In addition, the Board conducts studies for federal agencies and maintains liaison with the mathematical sciences communities and academia, professional societies, and industry.
Support for this project was provided by the Department of Defense and the National Science Foundation. Any opinions, findings, or conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

International Standard Book Number 0-309-05694-2

Additional copies of this report are available from:
Board on Mathematical Sciences
National Research Council
2101 Constitution Avenue, N.W.
Washington, D.C. 20418
Tel: 202-334-2421 FAX: 202-334-1597 Email: bms@nas.edu
Printed in the United States of America

Committee on Applied and Theoretical Statistics

Staff

Board on Mathematical Sciences

Commission on Physical Science, Mathematics, and Applications

PREFACE

In response to a request from the Chief of Statistical Research Techniques for the National Security Agency (NSA), the Committee on Applied and Theoretical Statistics (CATS) commenced an activity on the statistical analysis and visualization of massive data sets. On July 7-8, 1995, a workshop that brought together more than 50 scientists and practitioners (see appendix) was conducted at the National Research Council's facilities in Washington, D.C.

Massive data sets pose a great challenge to scientific research in numerous disciplines, including modem statistics. Today's data sets, with large numbers of dimensions and often huge numbers of observations, have now outstripped the capability of previously developed data measurement, data analysis, and data visualization tools. To address this challenge, the workshop intermixed prepared applications papers (Part II) with small group discussions and additional invited papers (Part III), and it culminated in a panel discussion of fundamental issues and grand challenges (Part IV).

Workshop participants addressed a number of issues clustered in four major categories: concepts, methods, computing environment, and research community paradigm. Under concepts, problem definition was a main concern. What do we mean by massive dam? In addition to a working definition of massive data sets, a richer language for description, models, and the modeling process is needed (e.g., new modeling metaphors). Moreover, a systematic study of how, when, and why methods break down on medium-sized data sets is needed to understand trade-offs between data complexity and the comprehensibility and usefulness of models.

In terms of methods we need to adapt existing techniques, find and compare homogeneous groups, generalize and match local models, and identify robust models or multiple models as well as sequential and dynamic models. There is also the need to invent new techniques that may mean using methods for infinite data sets (i.e., population-based statistics) to stimulate development of new methods. For reduction of dimensionality we may need to develop rigorous theory-based methods. And we need an alternative to internal cross-validation.

Consideration of computing environment issues prompted workshop participants to suggest a retooling of the computing environment for analysis. This would entail development of specialized tools in general packages for non-standard (e.g., sensor-based) data, methods to help generalize and match local models (e.g., automated agents), and integration of tools and techniques. It was also agreed that there is a need to change or improve data analysis presentation tools. In this connection, better design of hierarchical visual display and improved techniques for conveying or displaying variability and bias in models were suggested. It was also agreed that a more broad-based education will be required for statisticians, one that includes better links between statistics and computer science.

Research community and paradigm issues include a need to identify success stories regarding the use and analysis of massive data sets; increase the visibility of concerns about massive data sets in professional and educational settings; and explore relevant literature in computer science, statistical mechanics, and other areas.

Discussions during the workshop pointed to the need for a wider variety of statistical models, beyond the traditional linear ones that work well when data is essentially "clean" and possesses nice properties. A dilemma is that analysis of massive, complex data generates involved and complicated answers, yet there is a perceived need to keep things simple.

The culminating activity of the workshop was a panel discussion on fundamental issues and grand challenges during which participants exchanged views on basic concerns and research issues raised to varying extents in the workshop's three group discussion sessions. To facilitate the discussion the panel moderator posed four questions selected from among those generated by panel members prior to the session. The proceedings reflect attempts by workshop participants to address these and related questions. There were significant differences of opinion, but some agreement on items for ongoing exploration and attention—summarized above and listed in Part IV—was reached.

In addition to these proceedings, an edited videotape of the workshop will be available on the World Wide Web in December 1996 at URL: http://www.nas.edu/.

CONTENTS

OPENING REMARKS

Jon Kettenring
Bellcore

Good morning everybody, and welcome! It is nice to see so many people here to talk about opportunities for statistics in dealing with massive data sets. I hope that this workshop is as exciting for you as I think it is going to be. My colleague, Daryl Pregibon, and I are here on behalf of the Committee on Applied and Theoretical Statistics, the sponsor of this workshop. CATS is a committee of the Board on Mathematical Sciences, which is part of the National Research Council. We try to spotlight critical issues that involve the field of statistics, topics that seem to be timely for a push by the statistical community. The topic of massive data sets is a perfect example.

Looking at the names of the 50 or more people on our attendance list, I see that it is quite an interesting group from many perspectives. First, I am happy that we have a number of graduate students here with us. If we are able to make progress in this area, it is most likely the graduate students who are going to lead the way. So I would like to welcome them particularly.

We also have a number of seasoned statisticians who have been grappling with some of these issues now for a number of years. We are hoping that we can draw on their talents to help us understand some of the fundamental issues in scaling statistical methods to massive data sets.

We are particularly happy to have so many people here who have had genuine, real-world experience thinking about how to deal with massive data sets. To the extent that this workshop is successful, it is going to be because of the stimulation that they provide us, based on what they are actually doing in their areas of interest, rather than just talking about it.

We also have a nice cross section of people from business, government, and academia. I think it is also the case that nobody in the room knows more than just a small fraction of the other people. So one of the challenges for us is to get to know each other better.

Let us turn now to the agenda for the workshop. First, the only formal talks scheduled—and indeed, we have kept them as brief as possible—are all applications-oriented talks. There are 10 of these. Our purpose is to let you hear first-hand from people actually working in the various corners of applications space about the problems they have been dealing with and what the challenges are. The hope is that these talks will stimulate more in-depth discussion in the small group sessions. I hope that we can keep ourselves grounded in these problems as we think about some of the more fundamental issues.

We have organized the small group discussions according to particular themes. The first small group discussion will deal with data preparation and the initial unstructured exploration of a massive data set. The second theme will be data modeling and structured learning. The final one will be confirmatory analysis and presentation of results. In each small group session, we hope to focus on four questions: What existing ideas, methods, and tools can be useful in addressing massive problems? Are there new general ones that work? Are there special-purpose ones that work in some situations? Where are the gaps?

The closing session of the workshop will offer a chance to grapple with some of the fundamental issues that underlie all of these challenges posed by massive data. We have a very

interesting cross-sectional panel of people with a wide range of experience in thinking about fundamental issues.

In addition to having the proceedings of the workshop published by the National Academy Press, we hope that various segments will be available on videotape and on the World Wide Web so that more people will be able to benefit from the results of the work that we are going to do together in the next couple of days.

I wonder about the expectations you may have brought to this workshop. For myself, I am looking for insights from real experiences with data, e.g., which methods have worked and which have not. I would like to get a deeper understanding of some of the fundamental issues and the priorities for research. I am hoping—and this is something that CATS generally is particularly interested in—for momentum that might serve as a catalyst for future research in this area. Finally, I am hoping that one result will be a network of people who know each other a little bit better and can communicate about going forward. Indeed, that is one of our not-so-hidden agendas in gathering such a disparate group of people here.

PART I

PARTICIPANTS' EXPECTATIONS FOR THE WORKSHOP

Session Chair: Daryl Pregibon
AT&T Laboratories

Daryl Pregibon. We worked hard to get a very diverse cross section of people at this workshop. Thus some of the participants (including myself!) are familiar with only a small number of the other participants, either personally or professionally. Yet we are going to be working together closely for the remainder of the workshop. We should therefore introduce ourselves to the group, thereby allowing us to put a face to each name that we have been seeing on the e-mail circulation list. The key information we are looking for is your name, affiliation, and what you want to get out of the workshop.

Jon Kettenring (Bellcore). I have a personal interest in applied multivariate analysis. We have many practical techniques and tools that work reasonably well on moderate-sized problems. I fear that these methods, such as principal components analysis for dimensionality reduction and cluster analysis for segmenting data, are not going to scale very well. Yet these methods seem to me to be the sort of tools that I hear a crying need for as I read through the various applications areas. Given a hundred or a thousand variables, should we do a principal components analysis to reduce the number of variables to something manageable? I am not sure that is what I really need to do. I think one of the big challenges in the massive data set area is going from the global to the local, and being able to carve out segments of the space to work in. Again, the techniques of statistical cluster analysis could be very helpful for localizing the space. But I also am well aware of the numerous deficiencies with these methods, and their frequent ineffectiveness even in moderate-sized data sets. So I have a feeling that if we're going to try to solve the same problems in massive data sets, we're going to need algorithms and methods quite different from those we have now.

Pregibon. One reason that I am here is because of my membership in CATS. There are mounds of opportunities for the statistical community—so much opportunity and so little effort being expended. CATS is trying to build the interest and the relationships to involve the statistical community in these problems. The second reason I am here concerns the problems that I see initially in manufacturing and now more in the service side of the corporation. We are dealing with transactions on our network consisting of about 200 million messages a day. We need to harness the information in these data for a variety of applications, including network planning, service innovation, marketing, and fraud detection.

William Eddy (Carnegie Mellon University). I am a former chair of CATS, and so I have a longstanding interest in CATS activities. I have always been interested in large data sets. In the last year and a half, my notion of what was large has changed substantially. About a year ago, I had 1 gigabyte of disk storage on my work station, and I now have 12, and I have been adding at the rate of 2 gigabytes a month because the data sets that we are collecting are growing explosively.

Lyle Ungar (University of Pennsylvania). I spent many years looking at modeling for chemical process control, using neural networks combined with prior knowledge in the form of mass energy balances, comparing these approaches with things like MARS and projection pursuit regression. I am now looking at information retrieval, with the natural language people at the University of Pennsylvania, trying to see what techniques can be pulled from things like PC8, and how well they scale when one gets much bigger data sets. I am interested in seeing what techniques people have for deciding, for example, which variables out of the space of 100,000 are relevant, and using those for applications.

Brad Kummer (Lucent Technologies). I am also not a statistician. I guess I am a physicist mined engineer or engineering manager. I manage a group of folks who are supporting our optical fiber manufacturing facility in Atlanta. We have huge amounts of data on the fiber as it progresses through different stages of manufacture. One of the problems is mapping the corresponding centimeters of glass through the different stages and combining these data sets.

Ralph Kahn (Jet Propulsion Laboratory). I study the climate on Earth and Mars.

Noel Cressie (Iowa State University). I am very interested in the analysis of spatial and temporal data. Within that, I have interests in image analysis, remote sensing, and geographic information systems. The applications areas in which I am interested are mainly in the environmental sciences.

John Schmitz (Information Resources, Inc.). My training is in statistics and economics, but I have worked as a computer programmer all of my life. I have two reasons for wanting to be here. One is that I hardly ever see other people who are thinking in terms of statistics and data analysis. The other concerns the enormous databases that we have. We are trying to figure out how to do subsampling, or something else, to get quick responses to casual questions. I hope to get ideas in that area.

Fritz Scheuren (George Washington University). I have been very heavily involved in the statistical use of administrative records. Until recently, social security records and tax records and other things were thought to be large; I'm not sure anymore, but they are very useful starting points. Currently, my interests are in star, sties and administrative records (federal, state, and local) and how they fit together. I was worrying about being a sampling statistician, but I guess maybe I shouldn't anymore. I am also a member of CATS.

David Lewis (AT&T Bell Laboratories). I work in information retrieval. Over the past 6 or 7 years, I have made the transition from someone who has spent a lot of time thinking about linguistics and natural language processing to someone who spends a lot of time thinking about statistics, mostly classification methods for doing things like sorting documents into categories, or deciding whether documents are relevant. My strongest interest recently has been in what the computer science community calls active learning and what the statistics community calls experimental design. If you have huge amounts of data and you want to get human labeling of small amounts of it for training purposes, how do you choose which small amounts to use? We have been able to reduce the amount of data people have to look at by up to 500-fold. The theory says you can reduce it exponentially, and so there is a long way to go.

Susan Dumais (Bellcore). I have a formal background in cognitive psychology. I suspect I am probably the only psychologist here. I work for a company that is interested in a number of problems in the area of human and computer interaction. One of the problems we have worked on a lot is that of information retrieval. I am going to talk later this morning about some dimension reduction ideas that we have applied to rather large information retrieval problems.

Jerome Sacks (National Institute of Statistical Sciences). The NISS was formed a few years ago to do and stimulate research in statistics that had cross-disciplinary content and impact, especially on

large problems. After this weekend, I trust we will now be working on massive problems, or at least hope to. In fact, some of our projects currently are in transportation, education, and the environment, but they do not involve the massive data questions that people have been discussing and will report on these next two days. We see the locomotive coming down the tracks very rapidly at us, threatening to overwhelm almost anything else that we may be doing with the kind of data sets that we do confront. Another aspect of my own life in connection with this particular project is that I am on one of the governing boards of the CATS committee, namely, the Commission on Physical Sciences, Mathematics, and Applications. I see this particular workshop as leading to stronger policy statements in connection with the future of science research, and the impact of statistical research on science.

James Maar (National Security Agency). The National Security Agency is a co-sponsor of this workshop. We have done four of these projects with CATS, the best known of which is represented by the report *Combining Information* [National Academy Press, Washington, D.C., 1992], which I commend to you. My interest in large data sets started 17 years ago, when we posed some academic problem statements. We have millions of vectors and hundreds of dimensions. We cannot afford to process them with a lot of computer time. We want a quick index, like a matrix entropy measure.

Albert Anderson (Public Data Queries, Inc.). I have spent most of my life helping demographers and graduate students try to squeeze more information out of more data than the computers would usually let us get. We have made some progress in recent years. Five years ago, we targeted data sets such as the Public Use Micro Data Sample—PUMS (of the 5 percent census sampling in 1990), as the kind of data that we would like to handle more expediently. We have got this down to the point that we can do it in seconds now instead of the hours, and even months, that were required in the past. My interest in being here is largely to have some of the people here look over my shoulder and the shoulders of colleagues in the social sciences and say, "Ah ha, why don't you try this?"

Peter Huber (Universität Beyreuth, Germany). My interests have been, for about 20 years now, primarily in the methodology of data analysis and in working on interesting problems, whatever they are. At one time, I felt that data analysis was the big white spot in statistics; now I guess that large data sets are becoming the big white spot of data analysis.

Lixin Zeng (University of Washington). I am an atmospheric scientist. I am working on satellite remote sensing data, whose volume is getting bigger and bigger, and I believe it is going to be massive eventually. My main concern is the current numerical weather prediction model. I am not sure that atmospheric scientists are making full use of the huge amount of satellite data. I believe my horizons will be much broader as a result of this workshop.

Marshall DeBerry (Bureau of Justice Statistics, U.S. Department of Justice). Our agency is responsible for collecting the crime statistics for the United States that you read about in the papers. One of the major programs we have worked on is the National Crime Victimization Survey, which has been ongoing since 1973. It used to be about the second largest statistical survey conducted by the federal government. The other area that we are starting to move into is the NIBER system,

which is going to be a replacement for the uniform crime reports, the information that is put out by the FBI. That has the potential for becoming a rather large data set, with gigabytes of data coming on a yearly basis from local jurisdictions. We are interested in trying to learn some new ways we can look at some of this data, particularly the National Crime Victimization Survey data.

Fred Bannon (National Security Agency). I think it is fair to say that I have lived with this massive data problem for about 5 years now, in the sense that I have been responsible for the analysis of gigabytes of data monthly. There are all sons of aspects of this problem that I am interested in. I am interested in visualization techniques, resolution of data involving mixed populations, all involved together to form the data stream, any sort of clustering techniques, and so on. Anything that can be done in near-real time I am interested in as well.

John Tucker (Board on Mathematical Sciences, National Research Council). My background is in pure mathematics. I am an applied mathematician by experience, having spent 4 to 5 years with a consulting firm. My background in statistics is having studied some as a graduate student as well as having been the program officer for CATS for 4 years, prior to assuming directorship of the Board. I am interested in this problem because I see it as the most important cross-cutting problem for the mathematical sciences in practical problem-solving for the next decade.

Keith Crank (Division of Mathematical Sciences, National Science Foundation). I was formerly program director in statistics and probability and have been involved in the liaison with CATS ever since I came to the Foundation. We are interested in knowing what the important problems are in statistics, so that we can help in terms of providing funding for them.

Ed George (University of Texas). I am involved primarily in methodological development. In terms of *N* and *P*, I am probably a large *P* guy. I have worked extensively in shrinkage estimation, hierarchical modeling, and variable selection. These methods do work on moderate-to small-sized data sets. On the huge staff, they just fall apart. I have had some experience with trying to apply these methods to marketing scanner data, and instead of borrowing strength, it seems that it is just all weakness. I really want to make some progress, and so I am interested in finding out what everybody knows here.

Ken Cantwell (National Security Agency). In addition to the other NSA problems, our recent experiences have been with large document image databases, which have both an image processing and an information retrieval problem.

Peter Olsen (National Security Agency). I came to statistics and mathematics late in my professional career. I spent my first 15 years flying helicopters and doing data analysis for the Coast Guard. I now do signal processing for the National Security Agency. I routinely deal with the problem of grappling with 500 megabytes of data per second. There are 84,000 seconds in the day, and so I want to be able to figure out some way to handle that a little more expeditiously. Sometimes my history does rise up to bite me from the Coast Guard. I am the guy who built the mathematical model that was used to manage the Exxon Valdez oil cleanup.

Luke Tierney (University of Minnesota). My research interests have been twofold. One is developing methods for computation supporting Bayesian analysis, asymptotic, Monte Carlo, and things of that sort. My second interest is in computing environments to support the use and development of statistical methods, especially graphical and dynamic graphical methods. Both of those are highly affected by large data sets. My experience is mostly in small ones. I am very interested to see what I can learn.

Mark Fitzgerald (Carnegie Mellon University). I am a graduate student working on functional magnetic resonance imaging. I have to admit that when I started on this project, I didn't realize I was getting into massive data sets. We had 3 megabytes of pretty pictures, and we soon found out that there were a lot of interesting problems, and now we are up to many gigabytes of data.

Tom Ball (McKinsey & Company, Inc.). I am probably one of the few MBAs in the room, but I also have 8 years of applied statistical experience wrestling with many of the methodological questions that have been raised this morning.

Rob St. Amant (University of Massachusetts, Amherst). I am a graduate student in the computer science department. I am interested in exploratory data analysis. I am building a system to get the user thinking about guiding the process rather than executing primitive operations. I am interested in how artificial intelligence techniques developed in planning and expert systems can help with the massive data problem.

Daniel Carr (George Mason University). I have a long-time interest in large data sets. I started on a project in about 1979 for the Department of Energy, and so I have some experience, though not currently with the massive data sets of today. I have a long-time interest in graphics for large data sets and data analysis management, and in how to keep track of what is done with these data sets. I am very interested in software, and am trying to follow what is going on at the Jet Propulsion Laboratory and so on with the EOSDIS [Earth Observing System Data and Information System] and things like that. One issue is that much of the statistics that we use just is not set up for massive data sets. Some of it is not even set up for moderate-sized data sets.

Ed Russell (Advanced Micro Devices). I am a senior program manager. My involvement in large data sets started in the seismic industry; I then moved into computer simulation models, and now I am working with the electronics industry. All the industries in which I have worked are very data rich. I have seen the large *N*, the large *P*, and the large *N* and *P* problems. I have come up with several techniques, and I would like to validate some of them here, and find out what works and what does not work. I did try to start a study while I was at Sematech, to compare analytical methods. I would like to encourage CATS to push on that, so that we can start comparing methods to find out what their real strengths and weaknesses are with respect to extremely large data sets, with either large *N*, large *P*, or both.

Usama Fayyad (Jet Propulsion Laboratory, California Institute of Technology). I have a machine learning systems group at JPL. We do statistical pattern recognition applied to identifying objects in large image databases, in astronomy and planetary science. NASA sponsors us to basically develop systems that scientists can use to help them deal with massive databases. I am interested in both

supervised learning and unsupervised clustering on very large numbers of observations, say, hundreds of millions to potentially billions. These would be sky objects in astronomy. One more announcement that is relevant to this group is that I am co-chair of KDD-95, the first international conference on knowledge, discovery, and data mining.

David Scott (Rice University). I started out working in a medical school and doing some contract work with NASA. It has been fun to work with all these exploratory tools, which started out on the back of the envelope. I have a sneaking suspicion that my own research in density estimation may be key to expanding the topic at hand. I have a lot of interest in visualization. I hope that we see examples of that today. On the other hand, I am sure that a lot of us do editorial work. I am co-editor of *Computational Statistics*, and I am an editor on the Board of Statistical Sciences. I am very eager to make sure that any keen ideas put forth here see the light of day that way, if possible.

Bill Szewczyk (National Security Agency). I am interested in trying to scale up techniques. The problem of scale concerns me, because many techniques exist that are useful, like MCMC [Markov chain Monte Carlo], and that work for small data sets. If you are going to real time, they need to work for large data sets. But we are in a very time critical arena. I am sure we are not the only ones. We have to get through information for our data quickly, and we do not have the time to let these things run for a couple of days. We need it 5 minutes from now. So I am interested in seeing how some of these new techniques could be applied to real-time processing.

Gad Levy (Oregon State University and University of Washington). I am an atmospheric scientist interested in the application and use of massive data sets, mostly from satellites in the atmospheric sense. I have been working for some years in satellite data, one data set at a time, and recently started to think about how to look at them together. I have been collaborating with colleagues in computer science, who are trying to handle the data management and utilization aspects at the Oregon Graduate Institute and with statisticians at the University of Washington.

Wendy Poston (Naval Surface Warfare Center, Dalton, Virginia). I am interested in signal processing of one-dimensional and two-dimensional signals, where the size of the data sets, as well as the dimensionality, is extremely large.

Carey Priebe (Johns Hopkins University). I am most interested in the real-time implementation of pattern recognition techniques and visualization methods. My interest in the subject of this workshop has developed over probably 10 years of working on Navy and related remote sensing problems. Contrary to the accepted definition, I define massive data sets as data sets with more data than we can currently process, so that we are not using whatever data is there. There is a lot of data like that in other areas—satellite data, image data. There just are not enough people to look at the data. So my statistical interests are in what might be termed preprocessing techniques. If you already have clean data and you know that what you are interested in is there, then that is very nice. That is what I am trying to get to. When you have more data than you can look at with either the available human power or with computationally intensive computer statistical techniques, the first thing you have to do is take all the data and do some sort of clustering. This would be one way to look at it, to get down to saving the data that appears to be usable, so that you can look at it with more extensive processing, and save pointers into the data that tell you what might be useful and

what is not useful. In preprocessing, I am not trying to solve the problem, I am not trying to find the hurricane or find the tumor in digital mammography. I am trying to reduce the load, find places where it might be valuable to use the more expensive processes.

Dan Relies (Rand Corporation). I must say, if this turns into a competition of who has the biggest data set, I am going to lose! Like all statisticians, I am interested in error, but not the kind that you learn about in graduate school, random error or temporal error or those kinds of things. I am interested in human error, the fact that as *N* grows, the probability of our making mistakes grows. It is already pretty high in megabytes, and gigabytes scare me a lot. I used to be idealistic enough to think that I could prevent errors, but now I believe the main problem is to figure out how to deal with them when they come along. If any of you watched the O.J. Simpson trial a couple of weeks ago, you would perhaps appreciate that. What I have tried to do over the years is write software and develop ideas on how to organize empirical work, so that I and the people around me at Rand can be effective.

Art Dempster (Harvard University). I'm afraid I'm one of the old-timers, having been essentially in the same job for 37 years. I am interested not so much in multidisciplinary problems per se, although I think that "multidisciplinary" is getting to be a good term, but rather in more complex systems. I think that highly structured stochastic systems is the buzzword that some people use for an attempt to combine modeling and Bayesian inference and things of that sort. I do not necessarily agree that MCMC is not applicable to large complex systems. For the past year, I have been getting involved in climate studies, through the new statistics project at the National Center for Atmospheric Research in Boulder. That brings me into space-time and certainly multidisciplinary considerations almost instantly. That is, I think, one of the themes I am interested in here. I am also interested in combining information from different sources, which has been mentioned earlier as a possible theme. I am working to some degree in other fields, medical studies, a couple of things I am thinking about there. I have worked in the past on government statistics and census, labor statistics, and so on. Those interests are currently dormant, but I think there are other people here who are interested in that kind of thing, too.

Colin Goodall (Health Process Management, Pennsylvania State University (previously at QuadraMed Corp. and Healthcare Design Systems)). HDS is a company of about 250 who do information processing software development and consulting for the health care, particularly the hospital, industry. These health care data sets are certainly very large. There are 1.4 million hospital discharges in New Jersey each year. Multiply that by 50 states. There are countless many more outpatient visits to add to that. The health care data sets that I will be talking about this afternoon are special, in that not only are they very large or even massive, but the human input that goes into them is very massive also. Every patient record has been input by a nurse or someone else. There has been a lot of human interaction with these data, and a lot of interest in individual patient records. The data we are concerned with are driven by federal and state mandates on data collection, which is collection for uniform billing data for about 230 fields per patient. In the future we might include image data, although this is some way off.

Steven Scott (Harvard University). I am here because one of the database marketing position papers caught my eye. I have had limited contact with marketing people in the past, and I have

noticed that massive data sets seem to be very common—the business people out there seem to have a lot more data than they know what to do with. So I thought I would come and pick your brains while I had the chance.

Michael Cohen (Committee on National Statistics). I am a statistician interested in large data sets.

Allen McIntosh (Bellcore). I deal with very large data sets of the sort that John Schmitz was talking about data on local telephone calls. Up until now, I have been fairly comfortable with the tools I have had to analyze data. But recently I have been getting data sets that are much larger than I am used to dealing with, and that is making me uncomfortable. I am here to talk a little bit about it and to try to learn new techniques that I can apply to the data sets that I analyze.

Stephen Eick (Bell Laboratories). I want to make pictures of large data sets. We work hard on how to make pictures of big networks, and how to come up with ways to visualize software. I think the challenge now, at least for AT&T, is to learn how we can make a picture to visualize our 100 million customers.

Jim Hodges (University of Minnesota). There have been two streams in my work. One is an integral involvement in a sequence of applied problems to the point that I felt I actually knew something about the subject area, originally at the Rand Corporation in the areas of combat analysis and military logistics, and now at the University of Minnesota in clinical trials related to AIDS. The other is an interest in the foundations of statistics, particularly the disconnect between the theory that we learn in school and read about in the journals, and what we really do in practice. I did not know I was interested in massive data sets until Daryl Pregibon invited me to this conference and I started reading the position papers. Prior to this, the biggest data set I ever worked on had a paltry 40,000 cases and a trifling hundred variables per case, and so I thought the position papers were extremely interesting, and I have a lot to learn.

PART II

APPLICATIONS PAPERS

Session Chair: Daryl Pregibon
AT&T Laboratories

Earth Observation Systems: What Shall We Do with the Data We Are Expecting in 1998?

Ralph Kahn
Jet propulsion Laboratory and California Institute of Technology

ABSTRACT

The community of researchers studying global climate change is preparing for the launch of the first Earth Observing System (EOS) satellite. It will generate huge amounts of new data, filling gaps in the information available to address critical questions about the climate of Earth. But many data handling and data analysis problems must be solved if we are to make best use of the new measurements. In key areas, the experience and expertise of the statistics community could be of great help.

1 INTRODUCTION

The first EOS platform is scheduled for launch into polar orbit in June, 1998. It will carry five remote sensing instruments designed to study the surface and atmosphere of Earth. In a broad sense, the purpose of these observations is to find indications of how Earth's climate is changing, and to discover clues to the mechanisms that are responsible for these changes. A 5 to 15 year program of global monitoring is planned, covering many wavelengths, with spatial resolutions as small as 0.25 km and temporal coverage as frequent as daily. Higher resolution data on regional scales will also be acquired.

The surface area of Earth is about 5 × 108 km2. At 0.25 km resolution, a single instrument acquiring 36 channels of data, such as the Multi-angle Imaging SpectroRadiometer (MISR) or the Moderate Resolution Imaging Spectrometer (MODIS) on the EOS platform, will generate upwards of 80 Gbyte/day, or 30 Tbyte/year of basic data. The geophysical quantities are generally retrieved at lower spatial resolution, but must include quality flags and other ancillary information, resulting in a geophysical data set that will be no smaller than 3 Tbyte/year for the MISR instrument alone.

The sheer volume of data creates unprecedented challenges for accomplishing basic data handling operations, such as throughput and storage. But there are deeper issues regarding the scientific use of this huge amount of data. The EOS community has adopted a partial framework, and some terminology, for discussing the questions we must face. However, in many areas the development of an approach to the underlying issues is in its infancy. This paper begins with a brief review of the data classification scheme we use to organize our

thinking about data handling and analysis. This is followed by discussions of some issues relating to specific classes of data, and a summary of areas to which the statistics community may be well equipped to contribute.

2 DATA CLASSIFICATION SCHEME

The Committee on Data Management And Computing define five general classes of spacecraft data, based on the degree of processing involved (CODMAC, 1982, and subsequent refinements):

- Level 0 -The raw data stream from the spacecraft, as received at Earth
- Level 1 -Measured radiances, geometrically and radiometrically calibrated
- Level 2 -Geophysical parameters, at the highest resolution available
- Level 3 -Averaged data, providing spatially and temporally "uniform" coverage
- Level 4 -Data produced by a theoretical model, possibly with measurements as inputs

This paper focuses on Level 2 and Level 3 data, which are the main concerns of most global change research scientists working on EOS instrument teams. Level 2 products are reported on an orbit-by-orbit basis. For a polar-orbiting satellite such as EOS, the Level 2 sampling of Earth is highly non-uniform in space and time, with coverage at high latitudes much more frequent than near the equator. Level 2 data are needed when accuracy at high spatial resolution is more important than uniformity of coverage. These situations arise routinely for validation studies of the satellite observations, in the analysis of field campaign data, and when addressing other local-and regional-scale problems with satellite data.

The spatially and temporally uniform Level 3 data are needed for global-scale budget calculations, and for any problem that involves deriving new quantities from two or more measurements which have different sampling characteristics. To derive a Level 3 product from Level 2 data, spatial and temporal scales must be chosen. It is to this issue that we turn next.

3 GRINNING AND BIDDING TO CREATE LEVEL 3 DATA

The creation of Level 3 data has traditionally involved the selection of a global, 2- or 3-dimensional spatial grid, possibly a time interval as well, and "binning" the Level 2 data into the grid cells. The binning process for large data sets usually entails taking the arithmetic mean and standard deviation of all Level 2 data points failing into a grid cell, with possible trimming of outliers or of measurements flagged as "low quality" for other reasons. Typically, all points included in a grid cell average are given equal weight. Occasionally a median value will be used in place of the mean.

The leading contender for the standard EOS Level 3 grid is a rectangular-based scheme similar to one that has been used by the Earth Radiation Budget Experiment (ERBE) (Green and Wielicki, 1995a). In the proposed implementation for EOS, the Earth is divided zonally into 1.25 degree strips (about 140 km in width). Each strip is then divided into an integral number of quadrilaterals, each approximately 140 km in length, with the origin at the Greenwich meridian. This produces a nearly equal-area grid.

A number of issues arise in using a grid of this sort for the Level 3 data. Anisotropy presents an obstacle for calculating gradients, fluxes, and other quantities based on finite differences. Some neighboring cells share an edge whereas others share only a point, and there is no general rule as to how the contributions of each should be weighted. Only zonal gradients can be calculated in a consistent way on a global scale. Even in the meridional direction, the north-south cell boundaries are aligned only along the prime meridian. Inhomogeneity presents a second set of problems, since the distribution of grid cells varies with latitude, and there are singularities at the poles.

A third set of issues arises from the nesting properties of these grids. Nested grids can be used to relate data sets taken at different spatial resolutions, such as data from ground-based, aircraft, balloon, and satellite instruments. It is often necessary to compare these types of data (particularly for validation work), and to use data from multiple sources to calculate new quantities. To form sub-grids at length scales below 140 km, decisions must be made as to whether the subdivisions will be equi-angular, which are unique and relatively easy to define, or equal area, which has more desirable sampling properties, but requires more complex cell boundaries that increase anisotropy. Performing analysis on data sets from non-nested grids introduces errors that may be significant on a global scale (Green and Wielicki, 1995b), and can be arbitrarily large in regions where the quantities of interest have significant gradients (Kahn et al., 1991).

There are alternative grids, based on triangle or hexagon subdivisions of the spherical surface or a projection thereof, that may alleviate some of these issues (D. Cart and P. Huber, personal communication, MDS Workshop, 1995). A considerable body of work exists that explores the characteristics of nested systems of such grids (White et al., 1992, and references therein).

An effort is being organized to develop such grid schemes into systems that EOS scientists can use (Kiester, Kimerling, Knighton, Olsen, Sahr, and White, personal communication, 1995). A specific choice of grid system is being made, and its geometric properties characterized. Schemes will be needed to address and store data at different levels within the grid system. If the performance of a triangle or hexagon-based grid is promising, efficient translators to and from commonly used addressing systems, such as latitude-longitude, and conversions to popular map projections would need to be derived and implemented in data processing and GIS software packages widely used by the EOS community.

One would like to embed each data set into a grid within a nested system that is appropriate to its resolution and sampling structure. This raises the related issues of how to select a "native" grid size for a given data set, and how best to calculate the value and associated statistics to be assigned to each grid cell from the Level 2 data for both continuous-and discrete-valued quantities. Once this is done, methods may be developed to aggregate and dis-aggregate grids at various spatial resolutions, calculating the associated error characteristics along with the data (N. Cressie, personal communication, MDS Workshop, 1995).

Such a system would revolutionize the way the global climate research community works with data.

4 GENERATING LEVEL 2 DATA

The generation of Level 2 geophysical quantities from calibrated radiances introduces a far more diverse set of issues, since the retrieval algorithms vary greatly with the type of measurement made and the retrieval strategy adopted. For specificity, I use the MISR aerosol retrieval process (Diner et al., 1994) as the basis for the discussion in this section.

Two MISR-related issues similar to ones that arise elsewhere are: how to determine the sensitivity of the instrument to differences in atmospheric aerosol properties, and how to develop climatologies for the retrieved geophysical quantities based on existing constraints.

4.1 Sensitivity Studies

From the point of view of retrieving aerosol properties from MISR observations, the distinctions worth reporting are determined by the sensitivity of the instrument. We use a theoretical model to simulate the measurements at the 4 wavelengths and 9 viewing angles covered by the MISR instrument. We run simulations for a wide range of aerosol size distributions, compositions, and amounts. The full parameter space that must be explored includes mixes of particle size distributions and compositions, atmospheric relative humidity, and surface type.

We designate the one set of simulated reflectances as the "measured" case, and step through "comparison" models covering a range of alternative size distributions, for example. We use simple X^2 statistics to make the comparisons, such as:

$$\chi^2_{\mathrm{abs}} = \frac{1}{N\langle m_k \rangle} \sum_{l=1}^{4} \sum_{k=1}^{9} \frac{m_k [L_{mes}(l,k) - L_{cmp}(l,k)]^2}{\sigma^2_{abs}(l,k)}$$

where L_{mes}, is the simulated "measured" reflectance, L_{cmp} is the simulated reflectance for the "comparison" model, l and k are the indices for wavelength and viewing angle, N is the number of measurements included in the calculation, and σ_{abs} is the absolute measurement error in the reflectance. m_{k} is the weight for terms related to viewing angle k, and $<m_{\mathrm{k}}>$ is the average of the weights for all the viewing angles included in the sum.

Comparisons made in this way reduce the information content of as many as 36 individual measurements (4 wavelengths × 9 angles) to a single number. There is more information in the data. Two partly independent ways to compare cases are the maximum deviation of all the measurements used, and a X^2 statistic weighted by the measurements at the nadir angle:

$$\chi^2_{\mathrm{geom}} = \frac{1}{N\langle m_k \rangle} \sum_{l=1}^{4} \sum_{k=1}^{9} \frac{m_k \left[\frac{L_{mes}(l,k)}{L_{mes}(l,nadir)} - \frac{L_{cmp}(l,k)}{L_{cmp}(l,nadir)} \right]^2}{\sigma^2_{rel}(l,k)}$$

where σ_{rel} is the relative measurement error. We are experimenting with combinations of these metrics as the criteria to be used for evaluating the comparison cases, both in the sensitivity studies, and in the retrieval algorithm.

Our approach to covering the parameter space is also simple. We are planning first to vary particle size distribution and amount for fixed composition, establishing the minimum number of sizes needed to represent the range of expected values within the instrument sensitivity. The discrete sizes will be used to determine sensitivity to composition, which

is represented by the particle index of refraction. The sensitivity to mixtures will then be tested by a similar process.

These procedures are well-defined and systematic. But they are empirical, and it is impractical to capture every possible combination of conditions with them. In the absence of new ideas, we will live with these limitations.

4.2 Climatologies

The Level 2 retrieval algorithms for EOS must run in an automatic mode, rapidly processing huge amounts of data at computing facilities far from the purview of the instrument teams. As a first step in understanding the results, we plan to automatically compare them with" the expectations" — a climatology initially based on the best data available prior to launch.

Consider the aerosol climatology. The quantities of interest are the aerosol column amount and the aerosol "type", which summarizes particle composition, size distribution, and shape. There exist global satellite estimates of aerosol amount at 1 km resolution, over oceans only, on a weekly basis for almost seven years. For these observations, particle type is assumed. There are global models of four of the main particle types, at spatial resolutions ranging from about 100 km to about 1000 kin, at monthly or seasonal intervals. Numerous in situ measurements have also been made, with every conceivable spatial and temporal sampling. Some report aerosol amount, others provide information about aerosol type, and a few include both.

How do we merge all these data into a "climatology?" Our current approach is to ingest monthly cases of the global satellite data set into our geographic information system (GIS) as the primary constraint on aerosol amount. We will then use the global models to assign aerosol type, on a region-by-region basis (Figure 1). It is undecided as yet how the mix of particle types will be determined from the models, or how the uncertainty in the results will be obtained. We plan to use in situ measurements where available, to improve the constraints placed by the global data sets. Again we are undecided as to how to weight the information from different data sources, and how to assign uncertainties. Lastly, we must develop the algorithm that compares the aerosol properties derived from the satellite data with the climatology, and assigns a measure of "likelihood" to the result.

We will develop pragmatic approaches to each of these problems, but a formal procedure for constructing a climatology of this sort is beyond our current capability.

5 SUMMARY OF ISSUES

This paper concentrates on matters of potential interest to the statistics community that relate to the generation of Level 2 and Level 3 data from EOS instruments (Table 1). For Level 3 data, the main issues are: defining an effective system of nested grids, deriving procedures for ingesting Level 2 data into the system, and developing algorithms for aggregating and translating data that is in the system. Level 2 data presents a more diverse set of issues; we focused on performing sensitivity studies and developing climatologies.

The EOS community is preparing to derive geophysical quantities from measurements that will begin appearing in June 1998. All being well, we will soon face the challenges

of actually studying the data, summarizing the trends, identifying and characterizing the exceptions, and exploring the implications of the results for further data acquisition, and for global climate change... more than enough to keep several large and active communities of researchers very busy.

ACKNOWLEDGEMENTS

I thank my colleagues on the EOS MISR Team for providing the context for this work. I also thank the participants in Massive Data Sets Workshop for their interest in our data-handling issues, and their patience with our naive approach to profound problems in statistics. This work is performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration, through the EOS office of the Mission to Planet Earth.

References

[1] Paul R. Cohen, Michael L. Greenberg, David M. Hart, and Adele E. Howe. Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine*, 10(3):32-48, Fall 1989.

[2] John D. Emerson and Michal A. Stoto. Transforming data. In David C. Hoaglin, Frederick Mosteller, and John W. Tukey, editors, *Understanding robust and exploratory data analysis*. Wiley, 1983.

[3] Usama Fayyad, Nicholas Weir, and S. Djorgovski. Skicat: A machine learning system for automated cataloging of large scale sky surveys. In *Proceedings of the Tenth International Conference on Machine Learning* , pages 112-119. Morgan Kaufmann, 1993.

[4] Michael P. Georgeff and Amy L. Lansky. Procedural knowledge. *Proceedings of the IEEE Special Issue on Knowledge Representation* , 74(10):1383-1398, 1986.

[5] 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.

[6] Amy L. Lansky and Andrew G. Philpot. AI-based planning for data analysis tasks. *IEEE Expert*, Winter 1993.

[7] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.

[8] Robert St. Amant and Paul R. Cohen. Toward the integration of exploration and modeling in a planning framework. In *Proceedings of the AAAI-94 Workshop in Knowledge Discovery in Databases*, 1994.

[9] Robert St. Amant and Paul R. Cohen. A case study in planning for exploratory data analysis. In *Advances in Intelligent Data Analysis* , pages 1-5, 1995.

[10] 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*. Springer, 1995. To appear.

Information Retrieval: Finding Needles in Massive Haystacks

Susan T. Dumais
Bellcore

1.0 INFORMATION RETRIEVAL: THE PROMISE AND PROBLEMS

This paper describes some statistical challenges we encountered in designing computer systems to help people retrieve information from online textual databases. I will describe in detail the use of a particular high-dimensional vector representation for this task. Lewis (this volume) describes more general statistical issues that arise in a variety of information retrieval and filtering applications. To get a feel for the size of information retrieval and filtering problems, consider the following example. In 1989, the Associated Press News-wire transmitted 266 megabytes of ascii text representing 84,930 articles containing 197,608 unique words. A term-by-document matrix describing this collection has 17.7 billion cells. Luckily the matrix is sparse, but we still have large p (197,000 variables) and large n (85,000 observations). And, this is just one year's worth of short articles from one source

The *promise* of the information age is that we will have tremendous amounts of information readily available at our fingertips. Indeed the World Wide Web (W has made terabytes of information available at the click of a mouse. The *reality* is that it is surprisingly difficult to find what you want when you want it! Librarians have long been aware of this problem. End users of online catalogs or the WWW, like all of us, are rediscovering this with alarming regularity.

Why is it so difficult to find information online? A large part of the problem is that information retrieval tools provide access to textual data whose meaning is difficult to model. There is no simple relational database model for textual information. Text objects are typically represented by the words they contain or the words that have been assigned to them and there are hundreds of thousands such terms. Most text retrieval systems are *word based*. That is, they depend on matching words in users' queries with words in database objects. Word matching methods are quite efficient from a computer science point of view, but not very effective from the end users' perspective because of the common *vocabulary mismatch* or *verbal disagreement* problem (Bates, 1986; Furnas et al., 1987).

One aspect of this problem (that we all know too well) is that most queries **retrieve irrelevant information**. It is not unusual to find that 50% of the information retrieved in response to a query is irrelevant. Because a single word often has more than one meaning (polysemy), irrelevant materials will be retrieved. A query about "chip", for example, will

return articles about semiconductors, food of various kinds, small pieces of wood or stone, golf and tennis shots, poker games, people named Chip, etc.

The other side of the problem is that we miss relevant information (and this is much harder to know about!). In controlled experimental tests, searches routinely miss 50-80% of the known relevant materials. There is tremendous diversity in the words that people use to describe the same idea or concept (synonymy). We have found that the probability that two people assign the same main content descriptor to an object is 10-20%, depending some on the task (Furnas et al., 1987). If an author uses one word to describe an idea and a searcher another word to describe the same idea, relevant material will be missed. Even a simple concrete object like a "viewgraph" is also called a "transparency", "overhead", "slide", "foil", and so on.

Another way to think about these retrieval problems is that word-matching methods treat words as if they are uncorrelated or independent. A query about "automobiles" is no more likely to retrieve an article about "cars" than one "elephants" if neither article contains precisely the word automobile. This property is clearly untrue of human memory and seems undesirable in online information retrieval systems (see also Caid et al., 1995). A concrete example will help illustrate the problem.

2.0 AN SMALL EXAMPLE

A textual database can be represented by means of a term-by-document matrix. The database in this example consists of the titles of 9 Bellcore Technical Memoranda. There are two classes of documents -5 about human-computer interaction and 4 about graph theory.

Title Database:

c1: *Human* machine *interface* for Lab ABC *computer* applications

c2: A *survey* of *user* opinion of *computer system response time*

c3: The *EPS user interface* management *system*

c4: *System* and *human system* engineering testing of *EPS*

c5: Relation of *user*-perceived *response time* to error measurement

m1: The generation of random, binary, unordered *trees*

m2: The intersection *graph* of paths in *trees*

m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering

m4: *Graph minors: A survey*

The term-by-document matrix corresponding to this database is shown in Table 1 for terms occurring in more than one document. The individual cell entries represent the frequency with which a term occurs in a document. In many information retrieval applications these frequencies are transformed to reflect the ability of words to discriminate among documents. Terms that are very discriminating are given high weights and undiscriminating terms are given low weights. Note also the large number of 0 entries in the matrix-most words do not occur in most documents, and most documents do not contain most words

Table 1 Sample Term-by-Document Matrix (12 terms × 9 documents)

	C1	C2	C3	C4	C5	M1	M2	M3	M4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

Consider a user query about "human computer interaction", Using the oldest and still most common Boolean retrieval method, users specify the relationships among query terms using the logical operators AND, OR and NOT, and documents matching the request are returned. More flexible matching methods which allow for graded measures of similarity between queries and documents are becoming more popular. Vector retrieval, for example, works by creating a query vector and computing its cosine or dot product similarity to the document vectors (Salton and McGill, 1983; van Rijsbergen, 1979). The query vector for the query "*human computer* interaction" is shown in the table below.

Table 2. Query vector for "human computer interaction", and matching documents

Query		C1	C2	C3	C4	C5	M1	M2	M3	M4
1	human	1	0	0	1	0	0	0	0	0
0	interface	1	0	1	0	0	0	0	0	0
1	computer	1	1	0	0	0	0	0	0	0
0	user	0	1	1	0	1	0	0	0	0
0	system	0	1	1	2	0	0	0	0	0
0	response	0	1	0	0	1	0	0	0	0
0	time	0	1	0	0	1	0	0	0	0
0	EPS	0	0	1	1	0	0	0	0	0
0	survey	0	1	0	0	0	0	0	0	1
0	trees	0	0	0	0	0	1	1	1	0
0	graph	0	0	0	0	0	0	1	1	1
0	minors	0	0	0	0	0	0	0	1	1

This query retrieves three documents about human-computer interaction (C1, C2 and C4) which could be ranked by similarity score. But, it also misses two other relevant documents (C3 and C5) because the authors wrote about users and systems rather than humans and computers. Even the more flexible vector methods are still word-based and plagued by the problem of verbal disagreement.

A number of methods have been proposed to overcome this kind of retrieval failure including: restricted indexing vocabularies, enhancing user queries using thesauri, and various AI knowledge representations. These methods are not generally effective and can be time-consuming. The remainder of the paper will focus on a powerful and automatic statistical method, Latent Semantic Indexing, that we have used to uncover useful relationships among terms and documents and to improve retrieval.

3.0 LATENT SEMANTIC INDEXING (LSI)

Details of the of the LSI method are presented in Deerwester et al. (1990) and will only be summarized here. We begin by viewing the *observed* term-by-document matrix as an *unreliable estimate* of the words that could have been associated with each document. We assume that there is some underlying or latent structure in the matrix that is partially obscured by variability in word usage. There will be structure in this matrix in so far as rows (terms) or columns (documents) are not independent. It is quite clear by looking at the matrix that the non-zero entries cluster in the upper left and lower fight comers of the matrix. Unlike word-matching methods which assume that terms are independent, LSI capitalizes on the fact that they are not.

We then use a *reduced or truncated Singular Value Decomposition (SVD*) to model the structure in the matrix (Stewart, 1973). SVD is closely related to Eigen Decomposition, Factor Analysis, Principle Components Analysis, and Linear Neural Nets. We use the truncated SVD to approximate the term-by-document matrix using a smaller number of statistically derived orthogonal indexing dimensions. Roughly speaking, these dimensions can be thought of as artificial concepts representing the extracted common meaning components of many different terms and documents. We use this reduced representation rather than surface level word overlap for retrieval. Queries are represented as vectors in the reduced space and compared to document vectors.

An important consequence of the dimension reduction is that words can no longer be independent; words which are used in many of the same contexts will have similar coordinates in the reduced space. It is then possible for user queries to retrieve relevant documents even when they share no words in common. In the example from Section 2, a two-dimensional representation nicely separates the human-computer interaction documents from the graph theory documents. The test query now retrieves all five relevant documents and none of the graph theory documents.

In several tests, LSI provided 30% improvements in retrieval effectiveness compared with the comparable word matching methods (Deerwester et al., 1990; Dumais, 1991). In most applications, we keep k~100-400 dimensions in the reduced representation. This is a large number of dimensions compared with most factor analytic applications! However, there are many fewer dimensions than unique words (often by several orders of magnitude) thus providing the desired retrieval benefits. Unlike many factor analytic applications, we make no attempt to rotate or interpret the underlying dimensions. For information retrieval we simply want to represent terms, documents, and queries in a way that avoids the unreliability, ambiguity and redundancy of individual terms as descriptors.

A graphical representation of the SVD is shown below in Figure 1. The rectangular term-by-document matrix, X, is decomposed into the product of three matrices-$X = T_0 S_0 D_0'$, such that T_0 and D_0 have orthonormal columns, S_0 is diagonal, and r is the rank of X. This is the singular value decomposition of X. T_0 and D_0 are the matrices of left and right singular vectors and S_0 is the diagonal matrix of singular values which by convention are ordered by decreasing magnitude.

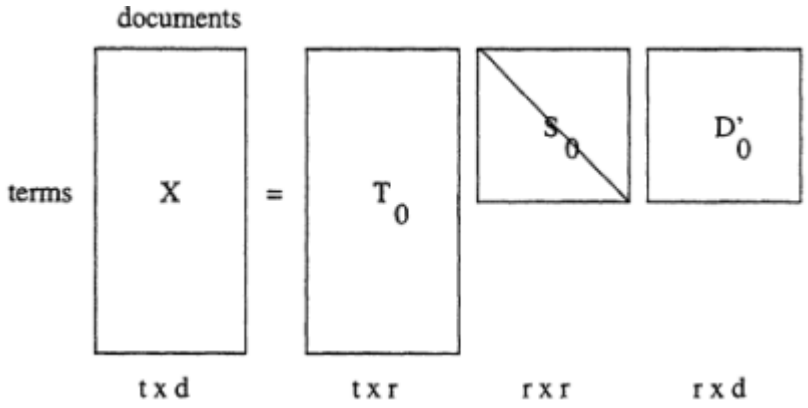

Figure 1. Graphical representation of the SVD of a term-by-document matrix.

Recall that we do not want to reconstruct the term-by-document matrix exactly. Rather, we want an approximation that captures the major associational structure but at the same time ignores surface level variability in word choice. The SVD allows a simple strategy for an optimal approximate fit. If the singular values of S_0 are ordered by size, the first k largest may be kept and the remainder set to zero. The product of the resulting matrices is a matrix $\hat{X}$ which is only approximately equal to X, and is of rank k (Figure 2).

The matrix $\hat{X}$ is the best rank-k approximation to X in the least squares sense. It is this reduced model that we use to approximate the data in the term-by-document matrix.

We can think of LSI retrieval as word matching using an improved estimate of the term-document associations using $\hat{X}$, or as exploring similarity neighborhoods in the reduced k-dimensional space. It is the latter representation that we work with-each term and each document is represented as a vector in k-space. To process a query, we first place a query vector in k-space and then look for nearby documents (or terms).

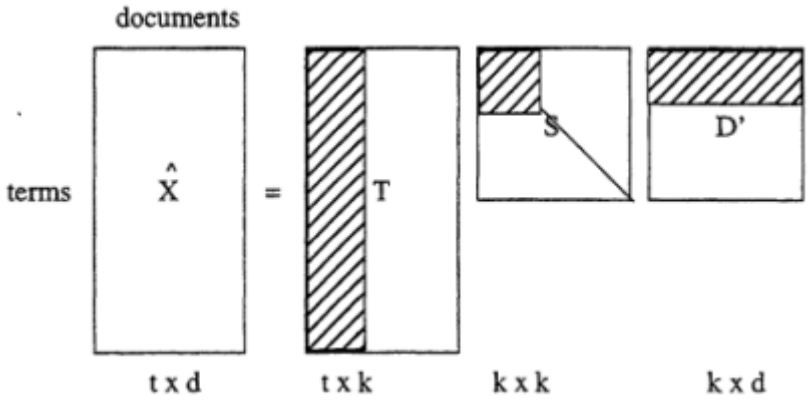

Figure 2. Graphical representation of the reduced or truncated SVD

4.0 USING LSI FOR INFORMATION RETRIEVAL

We have used LSI on many text collections (Deerwester, 1990; Dumais, 1991; Dumais, 1995). Table 3 summarizes the characteristics of some of these collections.

Table 3. Example Information Retrieval data sets

database	ndocs	nterms (>1 doc)	non-zeros	density	cpu for svd; k=100
MED	1033	5831	52012	.86%	2 mins
TM	6535	16637	327244	.30%	10 mins
ENCY	30473	75714	3071994	.13%	60 mins
TREC-sample	68559	87457	13962041	.23%	2 hrs
TREC	742331	512251	81901331	.02%	——

Consider the MED collection, for example. This collection contains 1033 abstracts of medical articles and is a popular test collection in the information retrieval research community. Each abstract is automatically analyzed into words resulting in 5831 terms which occur in more than one document. This generates a 1033 × 5831 matrix. Note that the matrix is very sparse-fewer than 1% of the cells contain non-zero values. The cell entries are typically transformed using a term weighting scheme. Word-matching methods would use this matrix. For LSI, we then compute the truncated SVD of the matrix keeping the k largest singular values and the corresponding left and fight singular vectors. For a set of 30 test queries, LSI (with k=100) is 30% better than the comparable word-matching method (i.e., using the raw matrix with no dimension reduction) in retrieving relevant documents and omitting irrelevant ones.

The most time consuming operation in the LSI analysis is the computation of the truncated SVD. However, this is a one time cost that is incurred when the collection is indexed and

not for every user query. Using sparse-iterative Lanczos code (Berry, 1992) we can compute the SVD for k=100 in the MED example in 2 seconds on a standard Sun Sparc 10 workstation.

The computational complexity of the SVD increases rapidly as the number of terms and documents increases, as can be seen from Table 3. Complexity also increases as the number of dimensions in the truncated representation increases. Increasing k from 100 to 300 increases the CPU times by a factor of 9-10 compared with the values shown in Table 3. We find that we need this many dimensions for large heterogeneous collections. So, for a database of 68k articles with 14 million non-zero matrix entries, the initial SVD takes about 20 hours for k=300. We are quite pleased that we can compute these SVDs with no numerical or convergence problems on standard workstations. However, we would still like to analyze larger problems more quickly.

The largest SVD we can currently compute is about 100,000 documents. For larger problems we run into memory limits and usually compute the SVD for a sample of documents. This is represented in the last two rows of Table 3. The TREC data sets are being developed as part of a NIST/ARPA Workshop on information retrieval evaluation using larger databases than had previously been available for such purposes (see Harman, 1995). The last row (TREC) describes the collection used for the adhoc retrieval task. This collection of 750k documents contains about 3 gigabytes of ascii text from diverse sources like the APNews wire, Wall Street Journal, Ziff-Davis Computer Select, Federal Register, etc. We cannot compute the SVD for this matrix and have had to subsample (the next to last row, TREC-sample). Retrieval performance is quite good even though the reduced LSI space is based on a sample of less than 10% of the database (Dumais, 1995). We would like to evaluate how much we loose by doing so but cannot given current methods on standard hardware. While these collections are large enough to provide viable test suites for novel indexing and retrieval methods, they are still far smaller than those handled by commercial information providers like Dialog, Mead or Westlaw.

5.0 SOME OPEN STATISTICAL ISSUES

Choosing the number of dimensions. In choosing the number of dimensions to keep in the truncated SVD, we have to date been guided by how reasonable the matches look. Keeping too few dimensions fails to capture important distinctions among objects; keeping more dimensions than needed introduces the noise of surface level variability in word choice. For information retrieval applications, the singular values decrease slowly and we have never seen a sharp elbow in the curve to suggest a likely stopping value. Luckily, there is a range of values for which retrieval performance is quite reasonable. For some test collections we have examined retrieval performance as a function of number of dimensions. Retrieval performance increases rapidly as we move from only a few factors up to a peak and then decreases slowly as the number of factors approaches the number of terms at which point we are back at word-matching performance. There is a reasonable range of values around the peak for which retrieval performance is well above word matching levels.

Size and speed of SVD. As noted above, we would like to be able to compute large analyses faster. Since the algorithm we use is iterative the time depends some on the structure of the matrix. In practice, the complexity appears to be O(4*z + 3.5*k), where z is the number of non-zeros in the matrix and k is the number of dimensions in the truncated representation. In the previous section, we described how we analyze large collections by computing the SVD of only a small random sample of items. The remaining items are "folded in" to the existing space. This is quite efficient computationally, but in doing so we eventually loose representational accuracy, especially for rapidly changing collections.

Updating the SVD. An alternative to folding in (and to recomputing the SVD) is to update the existing SVD as new documents or terms are added. We have made some progress on methods for updating the SVD (Berry et al., 1995), but there is still a good deal of work to be done in this area. This is particularly difficult when a new term or document influences the values in other rows or columns-e.g., when global term weights are computed or when lengths are normalized.

Finding near neighbors in high dimensional spaces. Responding to a query involves finding the document vectors which are nearest the query vector. We have no efficient methods for doing so and typically resort to brute force, matching the query to all documents and sorting them in decreasing order of similarity to the query. Methods like kd-trees do not work well in several hundred dimensions. Unlike the SVD which is computed once, these query processing costs are seen on every query. On the positive side, it is trivial to parallelize the matching of the query to the document vectors by putting subsets of the documents on different processors.

Other models of associative structure. We chose a dimensional model as a compromise between representational richness and computational tractability. Other models like nonlinear neural nets or overlapping clusters may better capture the underlying semantic structure (although it is not at all clear what the appropriate model is from a psychological or linguistic point of view) but were computationally intractable. Clustering time, for example, is often quadratic in the number of documents and thus prohibitively slow for large collections. Few researchers even consider overlapping clustering methods because of their computational complexity. For many information retrieval applications (especially those that involve substantial end user interaction), approximate solutions with much better time constants might be quite useful (e.g., Cutting et al., 1992).

6.0 CONCLUSIONS

Information retrieval and filtering applications involve tremendous amounts of data that are difficult to model using formal logics such as relational databases. Simple statistical approaches have been widely applied to these problems for moderate-sized databases with promising results. The statistical approaches range from parameter estimation to unsupervised analysis of structure (of the kind described in this paper) to supervised learning for filtering applications. (See also Lewis, this volume.) Methods for handling more complex models and for extending the simple models to massive data sets are needed for a wide variety of real world information access and management applications.

7.0 References

Bates, M.J. Subject access in online catalogs: A design model. *Journal of the American Society for Information Science*, 1986, *37(6)*, 357-376.

Berry, M. W. Large scale singular value computations. *International Journal of Supercomputer Applications*, 1992, 6, 13-49.

Berry, M. W. and Dumais, S. T. Using linear algebra for intelligent information retrieval. *SIAM: Review*, 1995.

Berry, M. W., Dumais, S. T. and O'Brien, G. W. The computational complexity of alternative updating approaches for an SVD-encoded indexing scheme. In *Proceedings of the Seventh SIAM Conference on Parallel Processing for Scientific Computing*, 1995.

Caid, W. R, Dumais, S. T. and Gallant, S. I. Learned vector space models for information retrieval. *Information Processing and Management* , 1995, *31(3)*, 419-429.

Cutting, D. R., Karger, D. R., Pederson, J. O. and Tukey, J. W. Scatter/Gather: A cluster-based approach to browsing large document collections. In *Proceedings of ACM: SIGIR'92*, 318-329.

Deerwester, S., Dumais, S. T., Landauer, T. K., Furnas, G. W. and Harshman, R. A. Indexing by latent semantic analysis. *Journal of the Society for Information Science*, 1990, *41(6)*, 391-407.

Dumais, S. T. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments and Computers*, 1991, *23(2)*, 229-236.

Dumais, S. T. Using LSI for information filtering: TREC-3 experiments. In: D. Harman (Ed.), *Overview of the Third Text REtrieval Conference (TREC3)*. National Institute of Standards and Technology Special Publication 500-225, 1995, pp.219-230.

Furnas, G. W., Landauer, T. K., Gomez, L. M. and Dumais, S.T. The vocabulary problem in human-system communication. *Communications of the A CM*, 1987, *30(11)*, 964-971.

Lewis, D. Information retrieval and the statistics of large data sets , [this volume].

D. Harman (Ed.), *Overview of the Third Text REtrieval Conference (TREC3)*. National Institute of Standards and Technology Special Publication 500-225, 1995.

Salton, G. and McGill, M.J. *Introduction to Modem Information Retrieval* . McGraw-Hill, 1983.

Stewart, G. W. *Introduction to Matrix Computations*. Academic Press, 1973

van Rijsbergen, C.J. *Information retrieval*. Buttersworth, London, 1979

Statistics and Massive Data Sets: One View from the Social Sciences

Albert F. Anderson
Public Data Queries, Inc.

ABSTRACT

Generating a description of a massive dataset involves searching through an enormous space of possibilities. Artificial Intelligence (AI) may help to alleviate the problem. AI researchers are familiar with large search problems and have developed a variety of techniques to handle them. One area in particular. AI planning, offers some useful guidance about how the exploration of massive datasets might be approached. We describe a planning system we have implemented for exploring small datasets, and discuss its potential application to massive datasets.

1 INTRODUCTION

The computing resources available within a university have become determining forces in defining the academic and research horizons for faculty, students, and researchers. Over the past half century, mechanical calculators have been replaced by batch-oriented mainframe processors which evolved into interactive hosts serving first hardcopy terminals, then intelligent display terminals, and finally PCs. Today, high performance workstations integrated within local and external networks have brought unprecedented computing power to the desktop along with interactive graphics, mass storage capabilities, and fingertip access to a world of computing and data resources. Each transition has opened new opportunities for those who work with demographic, social, economic, behavioral, and health data to manage larger quantities of data and to apply more sophisticated techniques to the analysis of those data.

Social scientists, however, have not realized the potential of this revolution to the extent that it has been realized by their colleagues in the physical, natural, engineering, and biomedical sciences. Distributed computing environments encompassing local and external

networks could provide high speed access to a wide range of mass data using low-cost, high speed, on-line mass-storage coupled to high performance processors feeding graphics workstations. Social scientists, scholars, planners, students. and even the public at large could access, manage, analyze, and visualize larger data sets more easily using a broader range of conventional and graphics tools than heretofore possible. But most of those who use these data today work with the data in ways remarkably similar to methods used by their predecessors three decades ago. Large data sets comprising records for thousands and even millions of individuals or other units of analysis have been, and continue to be, costly to use in terms of the dollars, time, computing facilities, and technical expertise required to handle them. These barriers can now be removed.

This paper looks briefly at the nature of the problem, the opportunities offered by computing and information system technology, and at one effort that has been made to realize the potential of these opportunities to revolutionize the manner in which massive census and survey data sets are handled. As this one example illustrates, realization of these opportunities has the potential for more than just a change of degree in terms of numbers of users, ease of use, and speed of response. Users of demographic, social, economic, behavioral, health, and environmental data can experience a qualitative change in how they work, interacting with data and tools in ways never before possible.

2 THE PROBLEM

Demographers, social scientists, and others who work with census and survey data are often faced with the necessity of working with data sets of such magnitude and complexity that the human and technological capabilities required to make effective use of the data are stretched to their limits-and often beyond. Even today, researchers may find it necessary to coordinate three or more layers of support personnel to assist them with their efforts to retrieve information from data sets ranging to gigabytes (GB) in size. Yet, these data are among the most valuable resources available for gaining insight into the social processes that are changing our world. These challenges are compounded today by the recognition that many of our pressing local, national, and international problems require multi-disciplinary approaches if the problems are to be understood and resolved. The success of these multi-disciplinary endeavors will depend in part upon how readily and how effectively researchers and analysts from the social sciences, environment, public policy, and public health can bring data from their disciplines, along with geographic and topological data, to bear on these problems.

Consequently, public data such as the Public Use Microdata Samples (PUMS). Current Population Surveys (CPS). American Housing Surveys (AHS), Census Summary Tape Files (STF), and National Center for Health Statistics mortality files are of greater potential value to a broader range of researchers, scholars, students, and planners than ever before. Yet, these are data that, because of the cost and difficulty in working with them, have historically been underutilized relative to their potential to lend insight into social, economic, political, historical, health, and educational issues. These are data that are relevant at levels ranging from personal to global concerns.

Unlocking information requires more than just access to data. Getting answers to even

simple questions provides significant challenges to users of multi-gigabyte data sets. The challenges become much greater when more complex questions are asked-questions that require the construction of indices within or across records, matching and merging information across data sets, and displaying data graphically or geographically. For example, to gain insight into college enrollment rates, one might wish to create an index for each child in the PUMS representing the total years of siblings college attendance that would overlap with the given child college years, assuming that each child were to attend college for four years starting at the same age. Such an index could reflect the economic pressure placed on a family by having multiple children in college at the same time. Availability of such an index could immediately suggest a host of questions and possible avenues of inquiry to be pursued-for example, establishing from other data sources the validity of the index as a predictor of college attendance or examining how the index varies over household and family characteristics such as the race, education, occupation, and age cohort of the head and spouse or the family structure within the PUMS data. One might also wish to generate thematic maps of the distribution of relationships within local, regional, or national contexts. To investigate such questions today would require access to costly computing and data resources as well as significant technical expertise. The task would challenge an accomplished scholar working in a well endowed research center.

Challenges exist on the technology side, also. The optimal application of mass storage, high performance processors, and high speed networks to the task of providing faster, cheaper, and easier access to mass data requires that strategies for using parallel and multiple processing be developed, data compression techniques be evaluated, overall latencies within the system be minimized, advantage be taken of the Internet for sharing resources, etc. In the case of latencies, for example, ten second startup and communication latencies are of little consequence to a five hour task, but a severe bottleneck for a one second task.

3 THE OPPORTUNITY

Creative application of currently available computing and information technology can remove the obstacles to the use of massive demographic, social, economic, environmental, and health data sets while also providing innovative methods for working with the data. Tasks related to accessing, extracting, transforming, analyzing, evaluating, displaying, and communicating information can be done in seconds and minutes rather than the hours, days, and even weeks that users have faced in the past. Networks can allow resources too costly to be justified for a small number of users within a local context to be shared regionally, nationally, or even internationally. The models for sharing access to specialized instruments that have worked well in the physical, natural, and medical sciences can be applied equally well to the social sciences.

Dedicated parallel and multiple processing systems have the potential to essentially eliminate the I/O and processing bottlenecks typically associated with handling files containing millions of data records. Serving systems based on closely coupled high performance processors can, in a fraction of a second, reduce massive data to tabulations, variance-covariance matrices, or other summary formats which can then be sent to desktop clients capable of merging, analyzing, and displaying data from multiple sources. Bootstrap and jackknife

procedures can be built into the systems to provide estimates of statistical parameters. Distributing the task between remote servers and desktop processors can minimize the quantity of information that must be moved over networks.

The rapidly falling cost of high performance systems relative to their performance is significantly reducing the hardware costs associated with creating dedicated facilities optimized for the handling of massive census and survey data.

4 ONE PATH TO AN ANSWER

A collaborative effort involving researchers at the Population Studies Center (PSC) at the University of Michigan and the Consortium for International Earth Science Information Network (CIESIN) at Saginaw. Michigan, led to a demonstration in 1993 of the prototype data access system providing interactive access via the Internet to the 1980 and 1990 land person records per file. The prototype, named xplore, as subsequently stimulated the development of the Ulysses system at CIESIN and a commercial system, DQ-Explore, y Public Data Queries. Inc. Users of the CIESIN facilities, who currently number more than 1,000 researchers, scholars, analysts, planners, news reporters, and students around the world, can readily generate tables from these data sets in seconds. The prototype ran on a loosely clustered parallel system of eight HP 735 workstations. The use of more efficient algorithms in the new designs is allowing better performance to be achieved using fewer processors. Other data sets are being added to the system. The prototype system has also been demonstrated on larger parallel processing systems. IBM SP1/SP2s, to provide interactive access to the 1990 5 represent a realization of the promise of high performance information and computing technology to minimize, if not eliminate, the cost in terms of dollars, time, and technical expertise required to work with the PUMS and similar large, complex data sets.

The Explore prototype was designed by Albert F. Anderson and Paul H. Anderson to perform relatively simple operations on data, but to do so very quickly, very easily, and through the collaboration with CIESIN, at very low cost for users. Multi-dimensional tabulations may be readily generated as well as summary statistics on one item within the cross-categories of others. Some statistical packages provide modes of operation that are interactive or that allow the user to chain processes in ways that, in effect, can give interactive access to data, but not to data sets the size of the PUMS and not with the speed and ease possible with Explore. Thirty years ago, interacting with data meant passing boxes of cards through a card sorter again-and again, and again and... More recently, users often interacted with hundreds, even thousands, of pages of printed tabular output, having produced as much information in one run as possible to minimize overall computing time and costs. The Explore approach to managing and analyzing data sets allows users to truly interact with massive data. Threads of interest can be pursued iteratively. Users can afford to make mistakes. Access to the prototype and to Ulysses on the CIESIN facilities has clearly succeeded in letting users access the 1990 PUMS data within an environment that reduces their costs in using such data to such an extent that they are free to work and interact with the data in ways never before possible.

The current development effort at Public Data Queries, Inc., is funded by small business research and development grants from the National Institutes of Health (NIH)-specifically

the National Institute for Child Health and Human Development (NICHD) The PDQ-Explore system combines high speed data compression/uncompression techniques with efficient use of single level store file I/O to make optimum use of the available disk, memory, and hardware architecture on the host hardware. More simply, the active data are stored in RAM while they are in use and key portions of the program code are designed to be held in the on-chip instruction caches of the processors throughout the computing intensive portions of system execution. As a consequence, execution speeds can in effect be increased more than one thousand fold over conventional approaches to data management and analysis. Because the task is by nature highly parallel, the system scales well to larger numbers of higher performance processors. Public Data Queries. Inc., is currently investigating the applicability of symmetric multiprocessing (SMP) technology to the task with the expectation that more complex recoding, transformation, matching/merging, and analytic procedures can be accommodated while improving performance beyond present levels.

Current implementations on HP, IBM, and Intel Pentium systems allow records to be processed at rates on the order of 300,000-500,000 per second per processor for data held in RAM. Processing times are expected to be reduced by at least a factor of two, and probably more, through more efficient coding of the server routines. System latencies and other overhead are expected to reduce to milliseconds. Expectations are that within one year, the system could be running on servers capable of delivering tabulations in a fraction of a second from the 5 data, more than 18 million person and housing records.

For information on the CIESIN Ulysses system, contact: info@ciesin.org

The Challenge of Functional Magnetic Resonance Imaging

William F. Eddy[*]
Mark Fitzgerald[**]
Christopher Genovese[***]
Carnegie Mellon University
Audris Mockus[****]
Bell Laboratories
(A Division of Lucent Technologies)

1 INTRODUCTION

Functional Magnetic Resonance Imaging (fMRI) is an extremely promising and rapidly developing technique used by cognitive neuropsychologists to obtain images of the *active* human brain. Images are obtained while the subject is engaged in a set of cognitive tasks designed to isolate specific brain functions, and the psychologists attempt to use the observed patterns of neural activation to understand and localize these functions. In contrast to traditional methods for mapping brain function, fMRI is non-invasive and allows the study of high-level cognitive processes such as language, visual attention, and problem solving. Since fMRI involves no known toxicity, each subject can be imaged many times, which improves precision and facilitates more sophisticated analyses. As such. fMRI promises to play a vital role in discerning the functional organization of the brain.

2 FUNCTIONAL MAGNETIC RESONANCE IMAGING (FMRI)

Recent developments in magnetic resonance imaging (MRI) have greatly increased the speed with which images of the human brain can be formed and this makes it suitable

[*] Professor of Statistics. Partially supported by ONR Contract N00014-91-J-1024, and NSF Grants IBN-9418982 and DMS-9505007.
[**] Graduate Student. Partially supported by NIH Grant MH15758.
[***] Assistant Professor. Partially supported by NSF Grant DMS-9505007.
[****] Member of Technical Staff, Bell Laboratories. Partially supported by the Center for the Neural Basis of Cognition.

for studying brain functions. An MR scanner is a multi-million dollar device which, by subjecting its contents to carefully modulated magnetic fields and recording the resulting radio signal, produces the Fourier transform of the magnetic field spin density for a particular atomic isotope. Then computing the inverse Fourier transform of the digitized signal reveals an image of the (magnetic field spin density of the) contents of the scanner.

Without going into the detailed physics and neurobiology that relate the magnetic field to brain activity, suffice it to say that increased neuronal activity induces an increase in blood flow to the region of activity (to deliver glucose to the neurons). This increased flow results in an increase of oxygenated blood in the small veins that drain the active region because the increased activity does not require much extra oxygen. The more oxygen carried by the hemoglobin in the blood the smaller the magnetic field generated by the iron in the hemoglobin (the oxygen acts as a magnetic shield) and consequently the less interference with the local magnetic field generated by, e.g., hydrogen nuclei (protons). By mid-1991 researchers had demonstrated that MRI can detect the changes in blood oxygenation caused by brain function and consequently the technique is known as fMRI. Among the first studies to use MRI to assess functional neural activity in humans are [1,2, 3]. The latter two introduced the now common Blood Oxygenation Level Dependent (BOLD) technique just described for characterizing activation in the brain.

There are several important features of fMRI compared to other imaging techniques. First, the signal comes directly from functionally induced changes. Second, it provides both functional and anatomical information. Third the spatial resolution is on the order of 1 or 2 millimeters. Fourth, there is little known risk from fMRI. Finally, the change in signal due to brain activity is quite small (on the order of 1%) and, in particular, smaller than the noise (on the order of 2%). This last feature means that, utilizing current technology, it is necessary to average a large number of images in order to detect the regions of activation.

3 A TYPICAL EXPERIMENT

The simplest fMRI experiment entails the performance of two cognitive tasks which differ in some specific detail. A number of images are gathered during each task and averaged within task. The difference between the average images for the two tasks provides information about the location in the brain of the cognitive function represented by the difference of the two tasks.

An actual experiment might proceed as follows. A subject lies in the MRI magnet with a head restraint intended to minimize movement. A set of preliminary anatomical images are studied to determine the location within the brain where the repeated functional images will be taken. The subject practices each of the two tasks for about a minute each, responding to the task, for example, by pushing a button with the right thumb. The subject performs one of the tasks repeatedly while images are recorded and then switches to the other task. In some of our smaller experiments we are recording 100 images for each task. In order to eliminate left-right effects the entire experiment is repeated with the subject using the left thumb to respond. Thus there are a total of 400 images in this simple experiment. It takes the scanner less than 20 minutes to acquire this amount of data.

The acquired images are multi-slice images with, typically, seven slices: each slice is

128×128 voxels with voxel dimensions roughly 2mm × 2mm × 7mm. Because most of the work we have done to date has been on the two-dimensional slices of these images we will henceforth think in terms of the 7×400=2800 individual slices.

4 DATA PROCESSING

The processing of the 2800 slices from this small experiment in order to detect the regions of activation is a massive task. The raw data is 128×128×2800 32-bit words which occupies 256MB of disk storage. Simply moving this amount of data around is a time-consuming task. Currently, the available bandwidth between the MR scanner and the workstation where we perform the processing is under 200K bytes per second; thus it requires nearly 30 minutes to simply move the data for this small experiment to our workstation. There are plans in place to substantially increase the bandwidth by the end of this calendar year.

Currently, the actual data processing is roughly as follows. We begin with an adjustment to account for inhomogeneity in the main static magnetic field. (We expect, in the future, to implement a further adjustment to account for non-linearity in the secondary dynamic magnetic field.) Then, we perform a "baseline" adjustment to correct for miscalibration of the analog-to-digital converter. (We expect, in the future, to implement a further "jitter" adjustment to account for very small errors in the timing of the data acquisition.) Then we perform a "mean" adjustment to correct for uncontrolled drift in the signal strength. (We expect, in the future, to implement a further pixelwise detrending to account for local drift within the image.) Then we perform an "outlier" adjustment to correct for shot noise. (We expect, in the future, to implement more sophisticated methods for addressing the fact that the data do not follow a Gaussian distribution.) We refer to the data at this point in the processing as the *corrected* data.

Unfortunately, because of the length (in time) of an experiment, the subject will almost certainly move. We address that problem both through the use of a head clamp and through a motion-correction procedure. We calculate the inverse Fourier transform to produce an image for the purposes of estimating the motion. Our motion correction procedure is complicated: first, by a non-linear optimization technique we estimate the amount of movement required to align each image and, second, we adjust the corrected data to account for this movement. We then calculate the inverse Fourier transform of the corrected and motion-corrected data to produce the actual image. At this point we are ready to begin what is called the "statistical" analysis of the data. The average is computed within slices within tasks and then the difference between tasks within slice is calculated. Finally, a statistical test is performed on each of the resulting differences to determine the regions of activation. Depending on the computing power of the workstation performing the calculations and depending on the precise details of the calculations, this can take anywhere from several days down to about twelve hours of processing time.

Our data processing software is designed as a processing pipeline of separate programs, This has the great advantage that modules can be easily interchanged if this will benefit the results. Also, new intermediate steps can be easily inserted at any stage. There is some disadvantage in that the act of storing intermediate results can consume considerable time. Nonetheless, we feel quite strongly that keeping the processing highly modularized is very

beneficial because of the flexibility it provides.

5 STATISTICAL CHALLENGES

The statistical challenges in the analysis of fMRI data are difficult and manifold. They all revolve around our understanding the nature of the noise and its effect on successfully detecting regions of activation. There are two general approaches to dealing with the noise in fMRI experiments. The first is to try to remove the source of the noise: we pursue this approach aggressively. The second is to model the noise through statistical methods: we also pursue this approach aggressively. We believe that both approaches are absolutely necessary.

Noise arises from a variety of sources. A fundamental source of noise is the vibration of the atomic nuclei in the imaged material. This cannot be reduced except by lowering the temperature toward absolute zero. Unfortunately, this noise is not spatially or temporally homogeneous but depends on both the anatomical structure and the function we are trying to detect. Inhomogeneity of the magnetic field, mechanical vibration, temperature instability of the electronics, etc., are all machine-based sources of noise. The machine-maintenance technicians work to limit these sources. The details of how the magnetic field is modulated to produce an image (known as a pulse sequence) effect the noise; we are engaged in studies to assess the relationship.

Physiological processes of the body such as respiration, heartbeat, and peristalsis effect the signal in ways that, in principle, can be modeled. We have begun planning experiments to gather data which might allow us to successfully model the cardiac and respiratory cycles because our more experienced colleagues believe that this is one of the primary sources of noise. Such an experiment is going to require synchronized recording of many images and the associated cardiac and respiratory information. This will be followed by a modelling effort which will view the sequence of images as the dependent variable and the cardiac and respiratory variables as predictors. Unfortunately, there is an interaction between the pulse sequence and the noise caused by physiological processes. This effort will thus require a family of models for each pulse sequence.

Movement of the subject between images is another source of noise. The standard algorithm for image registration in functional neuroimaging, called AIR [4]. works on reconstructed images. It is extremely computationally intensive: registration of images obtained from a single experiment can take as much as 24 hours of computer time. Subject movement appeared to us to be the simplest of the sources to understand and address. We have developed an alternative algorithm [6] for registering the images which operates in the Fourier domain. This method has proven to be more accurate than AIR, less prone to artifacts, and an order of magnitude more efficient. By differentially weighting regions in the Fourier domain, the method can also be made less sensitive to spurious signals that have a strong influence on image domain techniques. It is also readily generalizable to three-dimensional image registration, although we have not yet completed that work.

Finally, there is subject to subject variation. We have not yet focused on this question simply because the experimenters focus their experiments on individual subjects.

All of these sources affect our ability to detect regions of activation. When we began this work, active voxels were being detected by performing an independent t-test on each of the

16384 voxels in an image. We were approached with the question: How should we correct for the "multiple comparisons?" Bonferroni corrections do not result in any "significant" voxels. Ultimately we will have to build a complex spatial-temporal model of the images which allows us to answer the real question: Where are the active regions?

We have developed another approach [5] for identifying active regions, which is called the contiguity threshold method. The idea is to increase the reliability of identification by using the fact that real activation tends to be more clustered than artifactual activation caused by noise. Empirical evidence strongly suggests that this method provides a significant improvement in sensitivity. Of course, although it is more robust than voxel-wise tests, this method, too, depends on simplistic assumptions; we intend it as a stop-gap measure, to be eventually supplanted by more sophisticated analyses.

6 COMPUTATIONAL CHALLENGES

There are three important aspects of the computation. First, the amount of data from a large experiment approaches 1 GB. Any computations on a data set of this size require considerable time on a workstation. Second, there are no sensible ways to reduce the data during the earlier processing steps to speed up the processing. Third, because most of the computations are done on an image-by-image basis (or even on a slice-by-slice basis), there is a tremendous opportunity to speed things up with parallel or distributed methods.

Currently, our standard processing does not take advantage of the inherent parallelism. However, we have just begun experimenting (on our local network of workstations) with Parallel Virtual Machine (PVM) implementations of some of the most time-consuming steps in the processing. Simultaneously, we have begun plans to move the computations to a Cray T3D with 512 processors. In addition to just wanting to get things done faster, another reason for this plan is that we would like to perform the computations while the subject of the experiment is in the scanner and use the results as a guide for further experimentation during the same scanning session.

7 DISCUSSION

We have begun a serious effort to study and improve the statistical methodology of fMRI. and we have made some important preliminary steps.

One of the most fundamental questions about fMRI experiments is the question of reproducibility. If we run the experiment a second time immediately following the first with no intervening time, how similar will the results be? If we wait for a period of time? If we remove the subject from the scanner? If we repeat the experiment next month? We have begun to address this question; our preliminary results are reported in [8].

The analysis of functional Magnetic Resonance Imaging data can in many ways be viewed a prototype for a class of statistical problems that are arising more and more frequently in applications: namely, large data sets derived from a complex process with both spatial and temporal extent. There is a wealth of opportunities for the development of new statistical methodologies, and many of these ideas will apply to a variety of problems beyond neuroimaging.

8 ACKNOWLEDGEMENTS

We are indebted to a large number of colleagues at various local institutions. We list them here alphabetically both to thank them for their contributions to our work and to indicate the absolutely immense size of an undertaking of this kind. All of these people have made genuine contributions to our work either through teaching us, collecting data for us, computer programming, providing problems to solve, or assisting us in our work in other ways. The list includes psychiatrists, psychologists, physicists, electrical engineers, computer scientists, statisticians, computer programmers, and technicians. The organizational codes are given in this footnote[1].

Marlene Behrman, Ph.D., Psychology, CMU; Carlos Betancourt, B.S., MRRC, UPMC: Fernando Boada, Ph.D., MRRC, UPMC; Todd Braver. GS, Psychology, CMU: Patricia Carpenter, Ph.D., Psychology, CMU: Betty Jean Casey, Ph.D., Psychiatry, UPMC: Sam Chang, GS, MRRC, UPMC; Jonathan Cohen, M.D., Ph.D., Psychology, CMU; WPIC, UPMC; Denise Davis, B.S., MRRC, UPMC; Michael DeCavalcante, Undergraduate. CMU; Steven Forman, M.D., Ph.D., VAMC; WPIC, UPMC; Joseph Gillen, B.S., MRRC, UPMC; Nigel Goddard. Ph.D., PSC; Mark Hahn, B.S., LRDC. PITT; Murali Haran, Undergraduate, CMU; Marcel Just, Ph.D., Psychology, CMU; Caroline Kanet, Undergraduate, CMU: Timothy Keller, Ph.D., Psychology, CMU; Paul Kinahan, Ph.D., PETRC, UPMC; Benjamin McCurtain, B.S., WPIC, UPMC; Robert Moore, M.D., Ph.D., WPIC, UPMC; Thomas Nichols, B.S., GS, Statistics, CMU; PETRC, UPMC; Douglas Noll, Ph.D., MRRC, UPMC; Leigh Nystrom, Ph.D., Psychology, CMU; Jennifer O'Brien, B.S., GS, MRRC, UPMC; Brock Organ, B.S., Psychology, CMU; Robert Orr, B.S., Psychology, CMU; Julie Price, Ph.D., PETRC, UPMC; David Rosenberg, Ph.D., WPIC, UPMC; Waiter Schneider, Ph.D., LRDC, PITT; David Servan-Schreiber, Ph.D., WPIC, UPMC; Steven Small, Ph.D., WPIC, UPMC; John Sweeney, Ph.D., WPIC, UPMC; Talin Tasciyan, Ph.D., MRRC, UPMC; Keith Thulborn, M.D., Ph.D., MRRC, UPMC; David Townsend, Ph.D., PETRC, UPMC; James Voyvodic, Ph.D., MRRC, UPMC: Richard Zemel, Ph.D., Psychology, CMU.

[1] CMU = Carnegie Mellon University; GS = Graduate Student; LRDC = Learning Research and Development Center: MRRC = Magnetic Resonance Research Center; PETRC = Positron Emission Tomography Research Center; PITT = University of Pittsburgh; PSC = Pittsburgh Supercomputer Center; UPMC = University of Pittsburgh Medical Center; VAMC = Veterans Administration Medical Center: WPIC = Western Psychiatric Institute and Clinic.

References

[1] Belliveau, J.W., Kennedy, D.N., McKinstry, R.C., Buchbinder, B.R., Weisskoff. R.M., Cohen. M.S., Vevea, J.M., Brady, T.J., and Rosen, B.R. (1991). "Functional mapping of the human visual cortex by magnetic resonance imaging." *Science*. **254,** 716-719.

[2] Kwong. K.K., Belliveau, J.W., Chesler, D.A., Goldberg, I.E., Weisskoff, R.M., Poncelet, B.P., Kennedy. D.N., Hoppel, B.E., Cohen, M.S., Turner, R., Cheng, H., Brady, T.J., and Rosen, B.R. (1992). "Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation," *Proc. Natl. Acad. Sci. U.S.A*., **89,** .567.5.

[3] Ogawa, S., Tank, D.W., Menon, D.W., Ellermann, J.M., Kim, S., Merkle, H., and Ugurbil, K. (1992). "Intrinsic signal changes accompanying sensory stimulation: Functional brain mapping using MRI.'" *Proc. Natl. Acad. Sci. U.S.A*., **89,** 5951-5955.

[4] Woods. R., Cherry, S. and Mazziotta, J. (1992). "Rapid automated algorithm for aligning and reslicing PET images." *Journal of Computer Assisted Tomography*. **16,** 620-633.

[5] Forman, S.D., Cohen, J.D., Fitzgerald, M., Eddy, W.F., Mintun, M.A., and Noll D.C. (1995). "Improved assessment of significant change in functional magnetic resonance imaging (fMRI): Use of a cluster size threshold," *Magnetic Resonance in Medicine*. **33,** 636-647.

[6] Eddy, W.F., Fitzgerald, M., and Noll D.C: (1995). "Fourier domain registration of MR images." Submitted.

[7] Bandettini, P.A., Jesmanowicz, A., Wong, E.C.; and Hyde, J.S. (1993). "Processing strategies for time-course data sets in functional MRI of the human brain." *Magnetic Resonance in Medicine*. **30,** 161.

[8] Eddy, W.F., Behrmann, M., Carpenter, P.A., Chang, S.Y., Gillen, J.S., Just, M.A., Keller, T.A., Mockus, A., Tasciyan, T.A., and Thulborn, K.R. (1995). "Test-Retest reproducibility during fMRI studies: Primary visual and cognitive paradigms. *Proceedings of the Society for Magnetic Resonance, Third Scientific Meeting*, 843.

Marketing

John Schmitz
Information Resources, Inc.

Information Resources is a for-profit corporation that was founded in 1979. The software part of it was an acquired company, originally founded in 1969. Last year we sold about $350 million worth of data and software, with 3,500 employees. We collect consumer package goods data, the stuff that gets sold in grocery stores and drugstores and places like K-Mart and Walmart. We sell the data back to the manufacturers of those products so that they can track and assess their marketing programs. We also sell it to the retailers, so that they can see how they are doing.

We use this information for sales support (that is, how do I sell more product?), for marketing (that is, what products should I build and how should I support them and price them?), and also for logistical planning (that is, pushing stuff through the pipeline, and so forth.)

Our main database is called Infoscan. It was founded in 1986. Our expanded version of Infoscan, called the RI Census, was an expansion of our sample from 3,000 to 15,000 stores. That took place at the beginning of last year.

What we are in the business of doing is answering sales and marketing questions. Most of our products are developed and designed for interactive on-line use by end users who are basically sales people and planners. We do that because our margins are higher on that than on our consulting work. On consulting work, we basically do not make any money. On interactive on-line syndicated products where we have to ship out tapes or diskettes, we make fairly good margins.

We are basically in the business of helping people sell groceries. For example—we went through this with powdered laundry detergent—we helped Procter and Gamble sell more Tide. That is the only business that we are in. We do that by answering a group of questions for them. In a slightly simplified form, but not very simplified, I can classify them into four groups.

The first is tracking how I am doing? What are my sales like? What are the trends like? How am I doing in terms of pricing? How much trade support am I getting? How many of my products axe being sold with displays and advertising support?

The other three questions are aimed at some causal analysis underneath. The first is what we generally call variety analysis, that is, what products should I put in what stores? What flavors, what sizes? How much variety? Does variety pay off or is it just a waste of space? The second is, what price should I try to charge? I say "try to charge" because the manufacturers do not set the prices; the retailers do. The manufacturers have some influence, but they do not dictate it. The final area is what we call merchandising how much effort should I put into trying to get a grocery store to put my stuff at the end of the aisle in a great big heap so that you trip over it and some of the items accidentally fall in your basket?

Anecdotally, displays are tremendously effective. For a grocery product we typically see that sales in a week when there is an end-of-aisle display will be four or five or six times what they are normally.

The main data that we collect is scanner data from grocery stores, drugstores, and mass merchandisers. Our underlying database is basically very simple. It has three keys that indicate

what store a product was sold in, what UPC code was on the product, and what week k was sold. A lot of our data is now coming in dally rather than weekly.

We have only a few direct measures: how many units of a product did they sell and how many pennies' worth of the product did they sell, and some flags as to whether it was being displayed or whether it was in a feature advertisement, and some other kinds of odd technical flags.

We then augment that with a few derived measures. We calculate a baseline of sales by a fairly simple exponential weighted moving average with some correction for seasonality to indicate what the deviations are from the baseline. We calculate a baseline price also, so we can see whether a product was being sold at a price reduction. We calculate lift factors: if I sold my product and it was on display that week, how much of a rise above normal or expected sales did I get because of the display. We impute that. We do it in a very simple way by calculating the ratio of baseline sales to actual sales in weeks with displays. So you can imagine that this data is extraordinarily volatile.

The data is reasonably clean. We spend an enormous amount of effort on quality assurance and we do have to clean up a lot of the data. Five to 15 percent of it is missing in any one week. We infer data for stores that simply do not get their data tapes to us in time.

From this raw data we aggregate the data. We aggregate to calculate expected sales in Boston, expected sales for Giant Food Stores in Washington, D.C., and so on, using stratified sampling weights. We also calculate aggregate products. We take all of the different sales of Tide 40-ounce boxes and calculate a total for Tide 40 ounce, then calculate total Tide, total Procter and Gamble, how they did on their detergent sales, and total category.

This is an issue that comes back to haunt us. There is a big trade-off. If we do this precalculation at run time, at analysis time, it biases the analysis, because all of these totals are precalculated, and it is very expensive to get totals other than the ones that we precalculate.

We also cross-license to get data on the demographics of stores and facts about the stores. The demographics of stores is simply census data added up for some defined trading area around the store. These data are pretty good. Store facts—we cross-license these—include the type of store (regular store or a warehouse store). The data is not very good; we are not happy with that data.

Our main database currently has 20 billion records in it with about 9 years' worth of collected data, of which 2 years' worth is really interesting. Nobody looks at data more than about 2 years old. It is growing at the rate of about 50 percent a year, because our sample is growing and we are expanding internationally. We currently add a quarter of a billion records a week to the data set.

The records are 30, 40, 50 bytes each, and so we have roughly a terabyte of raw data, and probably three times that much derived data, aggregated data. We have 14,000 grocery stores right now, a few thousand nongrocery stores, generating data primarily weekly, but about 20 percent we are getting on a daily basis. Our product dictionary currently has 7 million products in it, of which 2 million to 4 million are active. There are discontinued items, items that have disappeared from the shelf, and so forth.

Remember, we are a commercial company; we are trying to make money. Our first problem is that our audience is people who want to sell Tide. They are not interested in statistics. They are not even interested in data analysis, and they are not interested in using computers. They want to push a button that tells them how to sell more Tide today. So in our case, a study means that a sales manager says, "I have to go to store X tomorrow, and I need to come up with a story for them. The story is, I want them to cut the price on the shelf, so I want to push a button that gives me evidence for charging a lower price for Tide." They are also impatient; their standard for a response time on

computers is Excel, which is adding up three numbers. They are not statistically or numerically trained—they are sales people. They used to have support staff. There used to be sales support staff and market researchers in these companies, but they are not there anymore.

Analysis is their sideline to selling products, and so we have tried to build expert systems for them, with some success early on. But when we try to get beyond the very basic stuff, the expert systems are hard to do.

There are underlying statistical issues that in particular, I need to look for. On price changes, we think that there is a downward-sloping demand curve. That is, if I charge more, I should sell less, but the data does not always say that, and so we have to do either some Baysian calculations or impose some constraints.

The databases are very large. Something I alluded to earlier we are doing all these precalculations, so we are projecting to calculate sales in Washington through a projection array. We are aggregating up an aggregation tree to get some totals for category and so forth. We do this because it saves a whole lot of time at run times, so we can get response times that are acceptable to people, and it saves a lot of space in the data, because we don't have to put in all of the detail. But it forces me to examine the data in the way we have allowed based on the precalculations. So we have a big trade-off here. The relevant subtotal is, what is the total market for powdered laundry detergent?

Those are all the nominal problems. What are the real problems? The real problem is that I have only two programmers who work for me. The tools that we have at our disposal are pretty good, at least as a starting point for front ends. But on the back end, just using SQL Query against Oracle or something simple is not fast enough. I do not have enough programmers to spend a lot of time on programming special-purpose retrievers over and over again. I have competition for my staff from operational projects for relatively simple things that we know are going to pay off. So time to spend on these interesting projects is being competed for by other projects.

Response time, particularly, is always a problem because of the questions that people ask, such as, What is the effect of a price change going to be in Hispanic stores if I increase the price of Tide by 10 percent? They are guessing at what to ask, and so they are not willing to invest a great deal in these questions.

The database setup time and cost are a problem. The setup time on these databases is mostly a "people" cost; it is not so much the computing time. It is getting people to put in all of the auxiliary data that we need around the raw data. So I have difficulty with getting enough auxiliary information in there to structure the analyses.

DISCUSSION

Carolyn Carroll: When you say auxiliary data, what are you talking about?

John Schmitz: A lot of what we are doing is looking at competitive effects, for example. So when I am doing an analysis on Tide, I need to know who Tide's competitors are. To a very limited extent you can do that by looking at the data. To a large extent you have to have somebody go in and enter the list of competitive brands. That is one basic thing.

Another is figuring out reasonable thresholds for defining exception points. A lot of that is manual. We start with automated systems, but a lot of it has to be examined manually.

When I mention a lack of staff, it is not so much a lack of straight programmers, but people who know the programming technology, and people who also understand the subject matter well enough to not need a whole lot of guidance or extremely explicit specifications.

Stephen Eick: So with your data, what are the privacy issues? I have noticed the few times I go to the store that you now have store cards, and so the stores know everything I have bought; they know who I am; they know my historical buying pattern. I am sure they have squirreled all this data away in a database. I am expecting soon to show up at the store and be pitched with coupons as I walk in.

Schmitz: That has not happened to you yet: I am not being facetious; we do not currently run any programs like that, but there are programs of that nature.

Participant: I think since they know everything I have bought, they are going to start targeting me with coupons. I personally resist, because I refuse to have a card. But others use every little coupon they can get.

Schmitz: There are two privacy issues. The privacy issue with our store audit database involves a contract that we have with the grocery chains that we will not release data identified with specific individual stores. We will not release sales data. So when we put out reports and so forth, we have to make sure that we have aggregated to the extent that we do not identify individual stores and say how much of a product they have sold.

The second privacy issue concerns individuals. We do have a sample of 100,000 U.S. households that identified themselves, and from whom we have a longitudinal sample that goes back anywhere from 3 to 10 years. We release that data, but it is masked as to the individuals. We have demographics on the individuals, but we do not identify them.

Eick: The other aspect of privacy involves the security cameras—at some point they are going to start tracking where I go in the store and what I look at. Then when I buy it, they are going to know it was me. So they are going to know not only what I bought, but also what I thought about buying.

Schmitz: Security cameras are used not so much to track people through the stores as to indicate when people are unhappy or happy about things—at hotels and so forth. We are not doing any of that yet.

Lyle Ungar: Are all your computations done off-line, or do you do on-line calculations, and how complex are they? Do you do factor analysis? Do you run correlations with demographics?

Schmitz: We do factor analysis off-line in order to reduce the dimensionality—or principal components—rather than reduce the dimensionality on our demographic data. We do that off-line and keep just component weights. About the most complicated things we do on-line are some fairly simple regressions and correlations with a little bit but not a whole lot of attention to robustization.

Massive Data Sets: Guidelines And Practical Experience From Health Care

Colin R. Goodall
Health Process Management Pennsylvania State University

Note: The author was previously affiliated with QuadraMed Corp. and Healthcare Design Systems.

1 INTRODUCTION

From the moment of birth to the signing of the death certificate, medical records are maintained on almost every individual in the United States (and many other countries). Increasing quantities of data are abstracted from written records, or entered directly at a workstation, and submitted by providers of healthcare to payer and regulatory organizations. Providers include physician offices, clinics, and hospitals, payers include managed care corporations and insurance companies, and regulatory organizations include state and federal government. Trends are towards making the flow of data easier, more comprehensive, and multi-faceted: through EDI (electronic data interchange), CHINs (Community Health Information Networks), and a seemingly ever more intrusive, detailed, and specific involvement by payors in the handling of care and compensation by and for providers.

These so-called clinical and financial administrative health care data are routinely massive. The Health Care Financing Administration's annual MEDPAR data base contains around 14 million discharge abstracts of every medicare-funded acute-care hospital stay. Individual state's administrative data of hospital discharges may include several million records annually. Data are collected in certain standard formats, including uniform billing (UB82, now UB92) for administrative data on hospital stays, and HEDIS (1.0, 2.0, 3.0) on patients in managed care. The more detailed data is often proprietary: for example HEDIS data is often proprietary to the specific payer organization, and includes data only on the organization's enrollees. More ambitious data collecting is underway in selected locations, through the systematic abstraction of supplementary clinical measures of patient health from medical records, through recording of additional patient characteristics, or through recording of more detailed financial information.

In principal the entire medical record is available. A written version might be available online in digital form as an image. Record linkage, for example between members of a family (mother and child), or through the use of unique patient identifiers across multiple episodes of treatment, or from administrative data to the registry of vital statistics (death certificates) and to cancer registries, provides important additional information. However, the availability of such information is, again, restricted.

A traditional uses of health data is in public health assessment and the evaluation of clinical efficacy of particular treatments and interventions.

These data are typically used differently: to analyze the system of health care delivery, seen as an aggregate of public and private, corporate and individual (physician) entities. These data are observational and comprehensive, i.e. a census; thus questions of accuracy and specificity — the appropriateness of the data to address particular clinical and financial concerns — predominate over statistical issues of sampling, estimation, and even censoring of data. Statistical tools may be relatively unsophisticated — with some exceptions (Silber, Rosenbaum and Ross (1995), and Harrell, Lee, and Mark (1995) and related papers) — but can be applied in massive fashion. Following a few further remarks to help set the scene, this paper is concerned with characterizing the interplay of statistical and computational challenges in analysis of massive data in healthcare.

In a real sense, the pressures of the competitive market place are pushing through healthcare reform in a manner which the political process has, in recent years, been poorly equipped to mandate. An oft-repeated criticism of the US health care delivery system has been that its high quality is associated with very high costs. A frequent rejoinder has been that some form of rationing of healthcare would be necessary in order to reduce costs. And indeed, the criticism now appears muted, and accounts of high costs appear to have given way, in the media, to accounts of the pitfalls of fairly apportioning care in a managed environment. Legislation, in New York and New Jersey and other states, has turned to mandating minimum levels of hospital care, notably for mothers delivering their babies.

The analysis of health care data, indeed, the massive analysis of massive health care data sets, has a central role in setting health care policy and in steering healthcare reform, through its influence on the actual delivery of healthcare, one hospital and one health maintenance organization at a time. In a nutshell, the twin objectives are for small consumption of resources, measured primarily in terms of cost, but also in terms of hospital length-of-stay, and for high quality. In practice the effective break-even point, between sacrificing quality and flexibility for resource savings, is obscured in corporate and institutional decision making and policy. Massive amounts of data analysis serves first of all to identify where savings might be made, or where quality might be inadequate. Most often, other hospitals or other physicians provide benchmarks in studying performance. Even relatively low cost and high quality hospitals can find room for improvement in the practice patterns of particular physicians or in treating particular types of patients.

No comparison is adequate without some effort at standardization of patient populations, or risk adjustment that accounts for individual patient characteristics. Adequate risk adjustment requires massive data sets, so as to provide an appropriately matched sample for any patient. Issues of model uncertainty, and the presence of multiple models, appear obviously and often. Orthogonal to patients are the number of possible measures, from overall measures of resource consumption and quality, to more detailed measures such as costs by individual cost center, the use of particular drugs and medications, and outcomes by type of complication. The next section gives a few details of COREPLUS and SAFs, two systems for outcomes analysis and resource modeling, including risk adjustment, developed in part by the author at Healthcare Design Systems.

The very large number of different questions in health care, and the specificity of those questions to individual providers, payers, regulators, and to patients, are compelling reasons to do massive analysis of the data. John Tukey has advised that, as CPU cycles are now cheaper than FTE's, the computers should be constantly running. The challenge is to devise a series of meaningful analyses that will use these resources (that are effectively free at the margin). In the following sections some graphical/tabular (Section 3), statistical (Sections 2 and 4), and computational (Section 5) aspects of implementation are addressed.

Massive analysis of massive health care data finds consumers at all levels, from federal government to state government, from payers to health systems, from hospitals to clinics, to physicians and to patients. The needs may differ in detail, but the overall strategy is dear: To provide multi-faceted insight into the delivery of health care. Consumption here includes both more passive and more active roles in preparing questions to be addressed through data analysis. Thus clinical professionals, their accountants and administrators, and patients, may assimilate already prepared reports and report cards, *and* they may seek immediate answers to questions generated on the spur of the moment, following a line of enquiry.

2 COREPLUS AND SAFS: CASE STUDIES IN MDA

The following is a brief account of the evolution of two project in massive data sets undertaken by Healthcare Design Systems (HDS) and its collaborators. With its parent company, latterly Kaden Arnone and currently QuadraMed, HDS has provided consulting services and software services to the hospitals and managed care companies. COREPLUS, for Clinical Outcomes Resource Evaluation Plus, is a system for analyzing outcomes of hospital care. SAFs, for Severity Adjustment Factor computation, is a system for modeling resource consumption, including cost and length of stay. Both systems have been developed through a collaborative effort between clinical experts, healthcare information processing staff, statisticians, management, and marketing. Clinical expertise is obtained through the New Jersey Hospital Association, as well as in house and by way of hospitals and their physicians.

COREPLUS includes 101 clinical outcomes in six major outcome categories (Vaul and Goodall, 1995). These are *obstetrics*, including Cesarcan section and post-delivery complication rates, *mortality*, including overall and inpatient mortality, pediatric mortality, postoperative mortality (within 48 hours), stroke mortality, and mortality by major diagnostic category (MDC), subdivided into medical, surgery, oncology, and non-oncology patients, *neonatal*, including newborn mortality by birthweight category and newborn trauma, *surgery*, including post-operative infections and various complication rates, *general*, including laparoscopic percent of cholecystectomies, diabetes percent of medical cases, and *cardiac*, including C ABG surgery mortality, cardiac valve mortality, myocardial infarction, and male and female cardiac mortality.

The sample sizes for the data included in each of these clinical outcomes ranges from several thousand to half a million for a single state (New Jersey, with around 1.3 million annual hospital discharges), and proportionately more for national data. The definition of the clinical outcomes is determined by clinical experts. Due to inevitable limitations in the data, some careful choices must be made in these definitions. Hospitals are compared in a *comparative chart* (Figure 1), using a variety of peer groups for comparisons.

Severity adjustment factors are computed in each of approximately 600

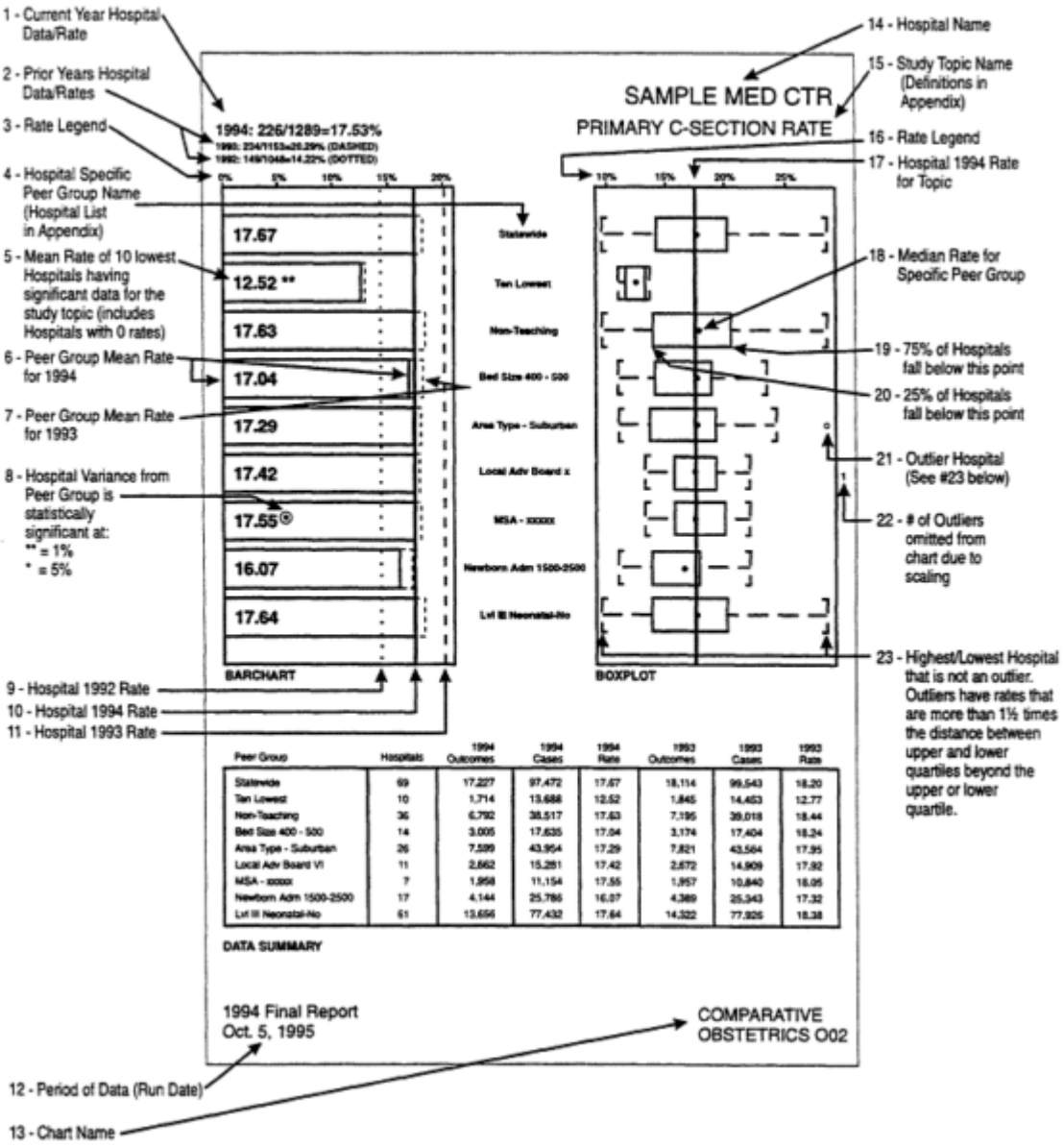

Figure 1 Interpreting COREPLUS™ output—comparative chart.

DRGs. We model length of stay, and costed data.

Prediction models in COREPLUS, and somewhat similarly for SAFs, are built using several categories of variables, including *general variables* , such as a function of age, sex, race group, payer group, and surgical use variables, *general diagnosis and procedure variables*, based on collections on one or more diagnosis and procedure codes in areas such as decubiti, sepsis, diabetes, renal failure, vents, and separately by principal and secondary diagnosis, *specific diagnosis and procedure variables*, specific to the outcome, and *specific additional variables* , for example birthweight. Hospitals are then compared in a *predictive chart* (Figure 2). The point here is not to provide a detailed explanation and justification of the system, but rather to illustrate the size of the problems that are encountered.

Variable selection is hindered by a "large *p*" problem: There are over 10,000 diagnosis and procedure codes to use singly, in aggregates (any one of the codes) or combinations (two or more codes simultaneously), as indicator variables in regression modeling. There is be a tight loop between statistical and clinical collaborators, within which data presentation is designed to convey information about potential variables and about the role of variables in fitting a statistical model to help elucidate clinical judgements.

Model building goes through several clinical and data-analytic steps, including: (1) provisional definition of an outcome in terms of elements of the data, (2) validation of the definition through running it against a data base, observing frequencies and dumps of individual patient detail, (3) marginal analysis of the association between potential predictor variables and the response, (4) determination of a set of candidate predictor variables based on clinical expertise supported by the data analysis, (5) predictive model building by a combination of hierarchical and variable selection methods, (6) review of results of model building for reasonableness of coefficients, (7) goodness of fit analysis overall and for subsets of patients, including those defined by the potential predictor variabels at step (3).

Beyond these steps, model validation is continual and on-going. A typical application of the systems is for quality assurance or utilization review staff at a hospital to undertake analyses of specific categories of patients, using COREPLUS and SAFs as guides towards problem areas. These might be patients whose outcome is contraindicated by the prediction. The patient medical record is consulted for further details, and that can expose factors that might be better accommodated in the definition of the clinical outcome,

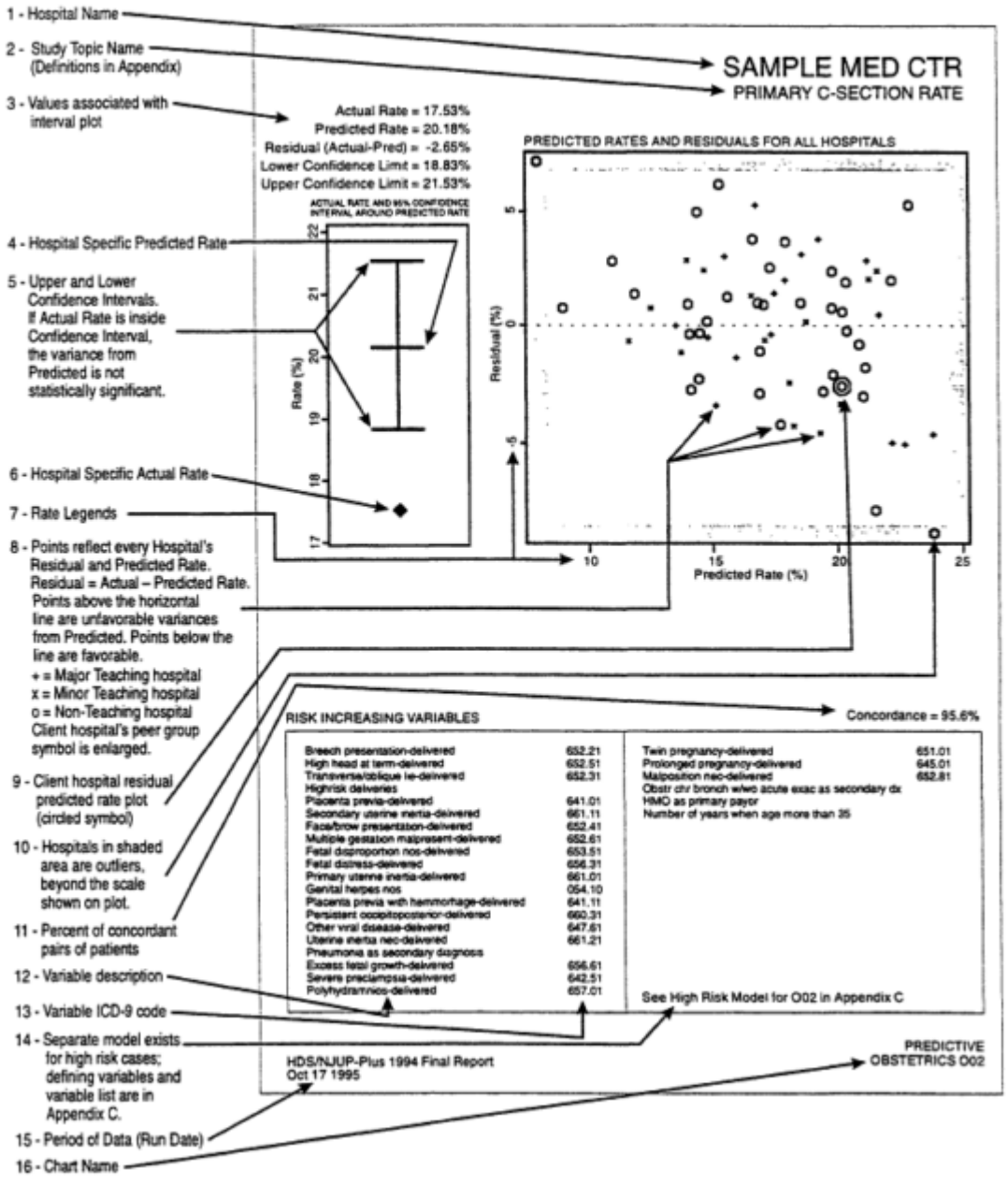

Figure 2 Interpreting COREPLUS™ output—residual plot.

or in the predictive model.

To summarize, the COREPLUS and SAF applications are massive along several dimensions.

- The data are massive. The number of observations is in the millions or tens of millions, the number of fields in the hundreds.
- The number of variables that can be meaningfully derived from the fields in the data is in the tens of thousands, including aggregation and combinations of indicators.
- The clinical resource is massive, from the corpus of medical literature to live clinical expertise.
- The audience is massive, as every person is involved in a professional or in a patient capacity in healthcare.
- The varieties of questions that can be addressed are massive.

3 PRESENTATION OF DATA ANALYSIS

Massive analysis of massive health care data finds consumers at all levels, from federal government to state government, from payers to health systems, from hospitals to clinics, to physicians and to patients. The needs may differ in detail, but the overall strategy is clear: To provide multi-faceted insight into the delivery of health care. Consumption here includes both more passive and more active roles in preparing questions to be addressed through data analysis. Thus clinical professionals, their accountants and administrators, and patients, may assimilate already prepared reports and report cards, *and* they may seek immediate answers to questions generated on the spur of the moment, following a line of enquiry.

Massive analysis of massive data may be presented in two ways. One, as a carefully formatted, detailed report, that presents in tabular, graphical and textual form a balanced and informative account of one or more aspects of the data. Such a report is likely to be condensed into one or a few pages (Figure 1), but many such reports can be combined together to give insight into many aspects of the data. Some further specifics are given in Section 3. Conceptually, this is akin to designing a new subway map for London or New

York, with its goal of clear but dense information content. In the subway map, each component, a subway station or a stretch of track, has specific and individual meaning to many, if not all, users of the system; likewise each component of the graph must be labelled — the conventional scatter plot, comprising a scatter of points with perhaps a few outliers labelled, is not a high performer in this regard. Tools to develop such graphics are now available, eg using Splus (MathSoft, Inc.) and, increasingly, off-the-shelf PC spreadsheet and graphics programs.

The second presentation of massive analysis of massive data in health care is through user friendly, immensely flexible, software for data analysis. Such software is a front end for analysis of data bases abstracted from the massive parent data, and is tailored to the particular needs of the health care investigator. The data themselves are at a patient level, so that patient level detail can be used to help understand patterns among larger groups. However, it is reasonable to save summary statistics only for larger groups, which leads to a hierarchy of different data types within this software, and reports tailored to this structure.

There is no absolute dividing line between reports generated directly from the massive parent data, and reports generated from software at distributed locations. What is clear, however, is that the analysis of massive data is a specialized undertaking, and that exceptional computational resources, in terms of hardware, software, and personnel, as well as clinical and statistical expertise, must accumulate at those central locations. The corresponding computational resources that are available in distributed fashion to the consumers of these data are increasingly powerful, flexible, and user-friendly, but there must always be a significant gap between them and the centralized resources. Even without the difference in sheer computational horsepower, the expertise at a central location allows flexibility in report generation that is beyond the scope of user-friendly, and thus to an extent sanitized, software.

The internet provides additional dimensions that further facilitate massive analyses of data to suit many diverse objectives. A straightforward use of the internet is for data collection and transmittal, in an extensive and multi-layered network. At its highest level this network involves the transmission of massive data sets among state, federal, academic, industrial, and commercial repositories. The rapid evolution of paradigms on the internet, from ftp, to viewing world wide web (WWW) pages, to downloading software on an as-needed basis via WWW, will find echos in the handling of massive

data sets in health care. For example, a massive data set need not reside at a single location; instead, application daemons might visit a succession of sites, collecting and even summarizing information to compile together into a comprehensive analysis.

A first step in integrating the internet into the central-distributed model might include transmitting data on a daily basis from providers (physicians and hospitals) to a central information processing center and its return to the provider with added value, for example, including some standards and predictions for benchmarking. Second, in generating reports at hospitals, queries for additional information might be sent directly to the information processing (IP) center, perhaps complementing the summary statistics already in the provider data base with fully uptodate data, or summary data for a different category of patients. Third, in generating a comparative report for a provider with some additional fields beyond the standard set, the IP center might access comparable data using its priviledged access to other providers databases.

4 STATISTICAL CHALLENGES

In analyzing massive data in healthcare, several concerns are paramount

1. Each patient is an individual. Patients are not exchangeable.
2. No single set of variables can capture all pertinent information on a collection of individual.
3. Even for only moderately large sets of variables, not even the most comprehensive efforts at modeling are enough.

If the cost of my hospitalization appears high, then that appearance is because of some internal benchmarking or expectations. I might ask the physicians who direct my treatment program for an explanation. I may also refine my benchmarks by looking at the costs of hospitalizations for other patients, perhaps initially for a broad group of patients, but increasingly for patients with similar environment and medical history to my own, a process that culminates in the use of one, but, better, many statistical models that provide patient-specific predictions (benchmarks). Of course, even then, the variables that really matter might not be in the patient records.

Each patient has very many alternative peer groups in a massive data set. In the setting of healthcare, massive data does not provide a 'law of large numbers' as protection for standard statistical analyses, but instead leads to a power law increase with sample size in the computational requirements.

These considerations throw the burden of analysis towards exploratory techniques. A starting place is simple 'odds ratio' statements of the form: Of 131,323 patients at risk for the outcome, 17,076 had the outcome, a rate of 13.00%. These data comprise patients with myocardial infarction during 1993 or 1994 in 10 states, and the outcome is death. Of the patients at risk, 24,986 patients had an initial acute myocardial infarction of the interior wall, with 4,065 deaths, a mortality rate of 16.27%. This particular AMI, ICD9 code 410.11, is just one of around 40 codes for AMI, which can be aggregated in diverse ways, or looked at in combination with other codes. There is thus a massive number of statistical statements, and insufficient resources for immensely detailed analyses of small data sets, such as the stack loss data. The need is for what might be called total analysis.

Beyond exploratory analysis, some statistical areas that appear particularly relevant are (i) advances in regression modeling, (ii) missing data methods, (iii) theory for observational data, and (iv) hierarchical models and Bayesian statistics (where reasonable computationally with large data sets).

Three further aspects are important. One is statistical computation and computational performance. A second is data, and the need for intense effort in understanding where the data come from and what are the alternatives. The third is organizational. In healthcare analysis, a divide and conquer strategy is natural given the pervasive subject matter knowledge: patients are naturally divided by major diagnostic category (MDC), for example, those with respiratory problems (MDC 4) and those with cardio-vascular problems (MDC 5). Some organizational/computational considerations of divide and conquer are considered in Section 5.

5 ORGANIZATION OF COMPUTATIONS

The problems of massive data analysis in healthcare involves as much organization as statistical analysis. A divide and conquer strategy becomes all-consuming: total data analysis in which all resources are devoted to the

maintenance, organization, and enhancement of the data.

Several statistical packages provide an "environment" for statistical analysis and graphics, notably SAS (SAS Institute, Cary, NC) and S (Becker, Chambers and Wilks, 1988). Although these systems provide a consistent interface to statistical functions, a programming language, graphics, an interface to operating system tools, and even a programmable graphical user interface, each has limitations. S has the more powerful and flexible environment, but SAS programming expertise is easier to find, SAS jobs are more likely to plug away until complete — inevitably massive data analyses are left to run "in batch" overnight and over the weekend, and it is better to pay a performance penalty than to risk non-completion. SAS programs are a little easier to read, less encumbered by parentheses. Neither environment fully supports imaging, graphics, and the World-Wide Web.

Thus there is a strong use for integration tools, that allow the best and brightest software to work together in a project-specific, possibly jury-rigged, system. Shell programming is important, but Perl stands out, and other approaches (tcl/tk) are promising. Another useful tool is DBMS Copy for transfering data between packages. Modern operating systems, for example IRIX version 5 or later, allow iconographic maintenance and operations. However, a graphical user interface is not so useful without scripting, as an adhoc analysis may be repeated many-fold.

Organizationally, the multi-dimensional arrays found in the the storage of data, where each element of the array is a single number, is echoed at a higher level in the organization of the components of massive data sets. In health care data, the dimensions of this meta-array might be year × state × outcome measure, where each element of the array is itself an array indexed by patient × measurement variable × hospital or episode. A summary of such data might be a table of C-section rate by state by year, including both mean and variation.

In principle these data might be organized into a six-dimensional array, either explicitly into an Oracle or Sybase database (say), or into a SAS or S dataset. Indeed, the present version of S, and even more so a future version, would support programming of a virtual array, in which the meta-array of arrays appears transparently as a six dimensional structure. Massive data sets in health care are constantly changing and evolving, and it is vital not to impose too rigid a framework. There might be a new hospital's data today, of a different sort, or a new type of data (images of patient records), or a

new measurement variable. It is most natural to maintain such data in the file system, not in a statistical package.

Thus the functions of operating system and statistical package meet one another. Simple standalone filters would be invoked from the UNIX shell, for example, to perform simple frequency analyses, merging, and tabulation directly from the command line. That is easily implemented using perl programs, possibly as wrappers for analyses using a statistical package.

Massive data are too large to be hidden in a. Data/ or sasdata/ directory. Instead, the meta-array is found in the file system. The hierarchical structure of the UNIX or DOS file system is limiting. I might wish that my data and their analyses are organized by outcome measure within state within year on one occasion, but, on another occasion, that they are organized by year within state separately for each outcome measure. Symbolic links can provide a clumsy implementation of alternative hierarchies in the file system, a better implementation would be an explicit UNIX shell for massive data sets.

A practical application of perl as an integration tool is to job scheduling. A specific example follows.

5.1 Example: Job Scheduling

An environment for analysis of massive data sets will more often contain multiple workstations (with different architectures), rather than be dominated by a single supercomputer. The analysis of massive data sets requires that these distributed computational resources be utilized in an effective and efficient — in terms of operator time — manner. This is a problem in the general area of job scheduling, but a specialized approach does not appear necessary.

As an example of implementation of the divide and conquer strategy, consider the script in Figure 3. The setup is of 101 outcome measures, each requiring a separate analysis of the data, to be performed on multiple machines. This simple perl script can be hugely effective.

The basic idea is to create an ascii 'state file' containing current state information to each outcome measure, or task, that can be read and written by different job schedulers. When a job scheduler is active on a task, the file is locked. Jobs, or subtasks, executed by the scheduler advance the state of the task; at all times the file contains the current state. When a job scheduler has completed all the subtasks it can with a particular task; the file

```
#!/usr/bin/perl

$m=shift;

$d='KeepTrack';
$max_attempts=10;
$warn_attempts=5;

%cmdname=(
        'a','command-1',   'b','command-2',
        'c','command-3',   'd','command-1'
                );

%cmd=(
        'command-1',
            'sas cmd1 -sysparm OUT
                -log OUT.log -print OUT.lst',
        'command-2',
            'Splus < OUT.q >& OUT.o',
        'command-3','mail anuser <  OUT.o'
            );

%nextstate=(  'a','b',  'b','c',  'c','d',  'd','e'  );

$start='a';
$end='e';

%size=(  'a',1,  'b',3,  'c',4,  'd',1   );

sub write {
        local ($f,$str)=@_;
        open(OUT,">$f");
        print OUT"$str\n";close(OUT);
}

while(@ARGV){$o=shift; $o=~s/.*\//;
        push(@a,$o);
}
```

```
TASK:
while(@a && $tb<$max_attempts){
      print("pausing ..\n"),
            sleep 7 if $tb>=$warn_attempts;
      push(@a,$o=shift(@a));
      $f="$d/$o";
      system "echo $start > $f" if ! -f $f;
      open(IN,"$f"); $s=<IN>; chop $s; close(IN);
      print "$o state $s";
      print(" .. locked\n"),
            next TASK if $s=~/.locked$/;
      &write($f,"$s.locked");
      STATE:
      while(1){
          print(" .. tasks complete\n"),
              pop @a, last STATE if $s eq $end;
          $tb++;
          print(" .. previous error\n"),
              last STATE if $errors{$o,$s};
          print(" .. size $sz > $m\n"),
              last STATE if ($sz=$size{$s})>$m;
          $cn=$cmdname{$s};
          $c=$cmd{$cn}=~s/OUT/$o/g;
          print " .. executing $cn: $c\n";
          system "$c";
          print(" .. error \n"), $errors{$o,$s}++,
              last STATE if $?;
          $tb=0;
          $s=$nextstate{$s};
          &write($f,"$s.locked");
          print "$o state $s";
      }
          &write($f,$s);
      }
```

Figure 3 A perl script for job scheduling in a distributed computational environment.

is unlocked. Conflicts are avoided by keeping state file operations small and fast, compared to the sizes of the subtasks themselves. With relatively long subtask times careful optimization of state file operations is unimportant.

There are several sophisticated features of (UNIX) operating systems that can be used, including shared memory, message passing (streams), file locking using flock, and shared data base technology. However, in a heterogeneous environment, including both UNIX workstations and personal computers say, scheduling must be extremely robust. It is the only dynamic component that must work across platforms. The tasks themselves may utilize ASCII files or various conversion utilities (UNIX-to-DOS, or data base copy of proprietary system's data bases across platforms using, eg, DBMS Copy). A perl script can run with little change on many different types of platform; communication using ASCII files containing state information separately for each task is highly robust.

The script shown in Figure 3 was written quickly, in part to exemplify the effectiveness of wide use of integration tools. Perl itself is eclectic, so no attempt was made for programming elegance. The script is run with several arguments, as in j s 4 A B C, where 4 is the power of the machine, and A, B, and C denote the tasks. State information is saved in files named A, B, C, automatically created if not present at task invocation, in the directory KeepTrack/. Each task comprises a sequence of states, tailored using a set of three associative arrays included in the perl script. Different types of task are accommodated using different initial states. A fourth associate array gives the size of the subtask from each state to the next.

Any number of j s jobs can be started at the same or different times, without explicit reference to other js jobs that are already running. Each job first places a lock on a free task by appending the string '.locked' to the state specified in the task's state file in KeepTrack/. The job proceeds through the states of the task, updating the task's state file, until either the task is complete, or the current subtask is too large for the size of the machine, or the system call returns an error status. The job then removes the lock and moves to the next task. Completed tasks are removed from the array of tasks; the script exits when that array is empty, or when the count of unsuccessful attempts exceeds a preset threshold (when this number exceeds a smaller threshold, a interval is set between attempts).

Environments such as SAS and S provide very powerful tools for statistical analysis. They may use parallel architecture, but they do not offer this kind

of simple support for distributed computing. The perl script given here, which could be implemented in SAS or in S, is an example of a software device that allows computational resources to be fully used with almost no additional effort. In the divide and conquer strategy employed in the analysis of health policy data, such tools are critical.

The script is easily modified and tuned to special situations. For example, in an environment where one machine is more powerful than the others, it may be useful to have job schedulers on that machine allocate effort to the most intensive subtasks when there is one to be done. Or, each task may have different overall size, which might be read from an ancillary file. Or, an upper limit might be placed on the number of active subtasks (because of storage bandwidth say), and a count may be maintained using a further file. Or, a breadth first scheduler may be required, instead of the depth first algorithm given. Or, in a primitive neural-net like approach, a script might learn which tasks and subtasks are accomplished most efficiently. Or, one task may depend on the completion of several other tasks, which can be implemented using an initial state and associated command that checks the states of the respective tasks in the files, possibly coupled with command line arguments that incorporate sublists of such tasks.

References

Harrell, F.E, Lee, K.L., and Mark, D.B. (1995). "Multivariate prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine* 14 to appear.

Iezzoni, L.I. (Ed.) (1994). *Risk Adjustment for Measuring Health Care Outcomes* . Ann Arbor, MI: Health Administration Press.

Silber, J.H., Rosenbaum, P.R., and Ross, R.N. (1995). "Comparing the contributions of groups of predictors: Which outcomes vary with hospital rather than patient characteristics?" *Journal of the American Statistical Association* 907-18.

Vaul, J.H. and Goodall, C.R. (1995). *The Guide to Benchmarking Hospital Value*. St. Anthonys Publishing.

Massive Data Sets in Semiconductor Manufacturing

Edmund L. Russell
Advanced Micro Devices

1 INTRODUCTION

Increasingly in industry and recently in semiconductor manufacturing, partly due to the introduction of sensor-based data collection, network connectivity, and the availability of cheap data storage devices, we are seeing the automation of data collection becoming so common that one of the questions most frequently asked to statisticians is: "How do we turn all this data into information?" The consulting statistician in this industry is beginning to be faced with massive data sets.

2 BACKGROUND

The semiconductor manufacturing environment is a high volume manufacturing environment. In a typical processing sequence, there may be over 100 process operations performed before the raw material, crystalline silicon wafers up to 8 inches in diameter, is converted to wafers carrying up to several thousand unpackaged electronic circuits, called die, on them. There are perhaps a few dozen additional manufacturing operations related to packaging, in which the die are encapsulated in plastic or ceramic, that occur before the product is ready for sale.

In the semiconductor manufacturing industry, the competition is generally very aggressive. Prices for products generally fall throughout any given product's life-cycle. And that product life-cycle can be very short; it is often measured in months. Even in this environment of short life-cycles and failing prices, the demands for product quality and reliability are extreme. Customers are beginning demanding that product be delivered with less than 10 parts per million defective and with reliability such that there are less than 30 failures expected per billion device hours of operation.

These characteristics of falling prices, increasing competition, complex state of the art processes and short life-cycles combined with the high capital cost of bringing a new manufacturing facility on-line are creating a great need to collect an analyze ever greater amounts of data.

3 DATA OVERVIEW

Data taken during manufacturing is available from a variety of sources. Some of the data are collected as a normal part of the statistical process control effort. Some of the data are collected as a normal part of the electrical screening tests that ensure product quality. At AMD we currently collect and summarize approximately 2 gigabytes of such data per day. In order to discover better ways of controlling key process steps, a number of companies are now automatically collecting some data using process state sensors.

Even in the development stage, the data volume from these sensors is huge. It is now possible to collect over 1 megabyte of sensor data per wafer in a plasma etch step alone. Given that there are typically 10 or more such steps in a manufacturing process, when one considers that an average wafer fabrication

site produces several thousand wafers per week, the potential data volume for analysis is huge.

Some of the reasons we wish to collect manufacturing data and perform the analyses include: process and product characterization process optimization yield optimization process control design for manufacturing

The question might be raised as to how these needs are different from the same needs in a more typical manufacturing environment? The first and foremost reason is that data are available from a large number of process operations — and much of that data can be collected automatically. The second reason is that the manufacturing process involves a large number of steps, some of which are essentially single wafer steps and others of which are batch processing steps of various batch sizes.

In addition, much of the summary data collected at this time are highly correlated due to the nature of the underlying physics and chemistry of the processing operations. In addition there is an established practice of taking multiple measures of the same electrical characteristics using test cells of varying sizes and properties. So, many of the apparently "independent" observations

aren't actually independent.

There are other sources of data that are less related to direct manufacturing that may be used with the manufacturing data. These sources of data involve the output of process simulators and die design simulators. It is becoming more standard throughout the semiconductor industry to link these simulators together in chains to get a better picture of the expected performance characteristics of processes and semiconductor devices. these expectations may then be compared to actual manufacturing experience.

Manufacturing process data are typically collected in 4 different stages, each of which provides a characteristic type of data for analysis. These data types are: die fabrication data wafer electrical data sort electrical data final test data

4 DIE FABRICATION DATA

Die fabrication data are typically in-process SPC data at this time. Although SPC data and its uses in the manufacturing environment are fairly well understood, there has been some interest expressed both within AMD and in other companies about further leveraging the

SPC data. Given that there are often about 100 process operations providing SPC data, in a typical manufacturing process there would appear to be a good opportunity for mining the SPC data for process characterization and optimization.

However, even when several of lots of wafers have experienced some similar sort of problem earlier in the process, it can be quite difficult to determine when all the lots ran through a given piece of equipment in a given span of time. this is because the wafer lot processing is not generally serial by lot. This is due to the mix of products in a manufacturing facility and the differences in the process flows among the products.

We are also beginning to get process-state data on individual process operations, such as plasma etch, from process-state sensors that are being added to the equipment. These sensors can provide over 10,000 measurements per wafer. This type of data is being examined by a number of companies for the potential to provide model-based process control for run-to-run, or wafer-to-wafer, modification of the processing parameters.

Because many, if not most, of the process steps during semiconductor manufacture are under control of process controllers and are also capable of being fitted with sensors, the future data potential is enormous. For instance, for the etch step in a typical wafer fabrication site, it would not be unreasonable to expect approximately 35 GB of data per week to be collected at some point in the future.

This process-state sensor data is typically autocorrelated within the process step itself and there is frequently some correlation from wafer to wafer, both in single-wafer steps and in batch-process steps. It is obvious that the observed autocorrelation patterns in the data change over time within the processing of a single wafer — and the autocorrelation patterns may even change in response to changes in the processing environment.

Both SPC and sensor based die fabrication data are now being explored in a variety of ways to assist with the optimization of individual process steps with respect to product quality, reliability, yield and throughput. A complication in this is that it is not known at this time what a "signal" is for a good process or even if the present process-state sensors are the correct ones to use.

For instance, one of the hidden complications of sensor based data is that the sensors have response functions. That is, the signal that is observed can be distorted or even completely disguised by the response of the sensor itself and how it interacts with its measurement environment. There are few statistical tools available today, and precious little training for statisticians, on identifying and dealing with sensor response functions.

Another hidden complication is that most of these process operations are under the active control of a process controller, often a PID controller. So we must also deal with the presence in the data of the apparent signal induced by the process controller itself. Here again, the statisticians are often not very conversant with process controllers and may not even be aware that at times the "signal" from a process may be mostly due to the controller trying to "settle."

5 WAFER ELECTRICAL DATA

This type of data is typically taken towards the end of the manufacturing of the semiconductor circuits, but before the die on the wafer are individually separated and packaged. The

data represents parametric measurements taken from special test structures located near the individual die on the wafers. Generally these test structures are neither a test die nor a part of the circuit itself. They are intended to provide information to the manufacturing site's engineers about the basic health of the manufacturing process.

The typical set of wafer electrical tests is comprised of about 100 to 200 or more electrical tests on fundamental components or building blocks of the electronic circuits. Most of the values retained in the data bases are processed data and not the raw data taken directly from the test structures. The choices of the reported values are typically made so as to be most informative about particular fabrication process operations. This leads to many of the test values being highly correlated with each other.

In addition, for any given test, consistent patterns across the individual silicon wafers or across wafers within the same wafer lot may be evident to the analyst. These patterns are often due to the underlying physics or chemistry of the process steps and so are expected to some extent.

So with wafer electrical test data, we have data which can be both spatially and temporarily correlated by test site as well as highly autocorrelated within test site. This type of data represents some of the potentially most useful data gathered during the manufacturing of the wafer.

It is generally believed by the engineers and managers that there are strong relationships between the wafer electrical tests and the individual processing operations. Indeed, there have been numerous occasions in the industry where misprocessed product was identifiable at wafer electrical test. If this is the case in the more general setting, then it would be highly desirable to produce models relating in-process data to wafer electrical data for "normal" product so that corrections can be made to the process to achieve optimum performance for the process.

6 SORT ELECTRICAL TEST DATA

This data is generated by an electrical pre-screen, usually 100 pre-screen, of each of the individual die on all wafers in a wafer lot. These tests are often functional tests, however some tests my be performed on critical electrical parameters, such as product speed. As in wafer electrical test data, there is a potential for hundreds of data values collected per individual die on a wafer. And as in wafer electrical tests, there are often patterns discernible in the test results across the wafers and from wafer to wafer.

Sort electrical test represents the first time the product itself is actually electrically tested. The primary use of this data is to pre-screen the product so that resources are not wasted in the remainder of the manufacturing process by embedding non-functional die in plastic or ceramic packages.

Another possible use of this data, however one that is not frequently pursued, is to relate the sort results to the wafer electrical test data and thus the wafer fabrication process itself. It is worth noting that in such a case the analyst would have hundreds of dependent variables and independent variables simultaneously.

7 FINAL ELECTRICAL TEST DATA

This delta is electrical test data taken on the finished product. There can be literally thousands of highly correlated tests performed each packaged die. Many of these tests are parametric in nature however only pass/fail data may be available for analysis unless special data collection routines are used. Most of the tests are related directly to product specifications and so are tests of the product's fitness for use. Since there can be considerable interaction between the package and the silicon die, some electrical tests are possible for the first time at this stage of manufacture.

As with the sort electrical test data, we would like to relate these test results to the die fabrication process. In addition we would also like to relate them to the package assembly process. Possible additional uses of this data include the identification of defective subpopulations and marginal product, product characterization and process characterization.

At this stage of the manufacturing process, there can literally be thousands of highly correlated measurements taken on literally millions of packaged die. Even with relatively sparse sampling, the size of the data sets that can accumulate for characterization and analysis can be enormous. Because of the sizes of the data bases and the large number of measurements taken on each die, the available statistical methods for analyzing tend to be highly inadequate. Even seemingly trivial tasks such as validating and reformatting the data become major problems!

8 STATISTICAL ISSUES

The types of data sets that we are beginning to see in the semiconductor manufacturing industry present major challenges to the applied statistician. In this industry there are simultaneously, high "n," high "p," and high "n and p" problems to be dealt with.

Because of the turn around time required for these types of analyses, and the potential for rewarding results, many managers and engineers are trying to use other methods, such as neural nets and CART like methods, to analyze these types of data sets. Most are unaware that these methods are also statistical in nature and have particular strengths and weaknesses.

By and large the best manner to proceed in attacking these types of problems is not really known. We often have great difficulty just dealing with the database issues, much less the analytical issues. Data volume issues even with simple analyses are a significant stumbling block for many software packages. Sematech has investigated a limited number of statistical packages and evaluated their performance in doing simple analyses with merely "large" data sets and found that most gave up or failed to work with reasonable performance.

Once one begins analyzing a "small" problem in say 70 main effects with multiple responses and moderate correlation of some explanatory variables, and you are told that interactions are expected, the analyst quickly discovers the limitations of the common statistical methods at his disposal. Common concepts of even such routine ideas as goodness of fit are less intuitive. Even describing the domain of the explanatory variables can be a challenge!

9 CONCLUSIONS

How do we cope? The short answer is: "Not well." Statistical resources are generally somewhat scarce in many industries — and those resources are often directed towards supporting ongoing SPC and simple design of experiments efforts. In addition, many applied statisticians are at best, somewhat hobbled by a number of other factors: a statistical education that was too focused on the simpler core techniques, a lack of capable and efficient analytical tools, and a lack of ability to rapidly validate and reformat data. An applied industrial statistician that is lacking in any of these three areas on any given project may be in serious trouble.

The more adventuresome analysts explore the use of non-standard (or non-standard uses of) methodologies such as AID, CART, neural nets, genetic

algorithms, kriging, PLS, PCR, ridge regression, factor analysis, cluster analysis, projection pursuit, Kalman filtering, and Latin hypercube sampling to mention only a few. Less adventuresome analysts utilize more commonly taught procedures such as stepwise regression, logistic modeling, fractional factorial and response surface experiments, and ARIMA modeling. However

far too many applied statisticians are unable to apply more than a couple of these techniques. Virtually all the meaningful training is "on the job."

Even very highly experienced analysts fall prey to the special problems found with sensor based data and processes that use automated controllers. These problems include, but are definitely not limited to modeling: processes that have more than one type of regime (such as turbulent flow and laminar flow), non-stationary time series that change order suddenly for brief periods of time, sensor response functions rather than process signal, data from differing stoichiometries in the same experiments, PID controller settling time as noise, the responses of PID controllers rather than the process itself.

Failure to deal appropriately with these sorts of problems in the design of the collection of the data can jeopardize the results of any analysis, no matter how numerically sophisticated the basic techniques.

10 RECOMMENDATIONS

To adequately address these problems, we need a permanent forum focusing specifically on the issues inherent in massive industrial data sets, possibly including an annual conference and a journal. Issues that I would like to see addressed include:

improving the education of applied statisticians. The ability to use a standard statistics package, perform simple tests of hypothesis, design and analyze standard fractional factorial experiments, and set up SPC programs does not represent the needs for the future.

developing an understanding of which types of advanced modeling techniques provide leverage against which types of data for different analytical goals. Developing an understanding of the problems related to sensors in particular is critical. It is no longer sufficient to fit simple linear models and simple time series to data.

developing more graphical aids and more knowledge on how to use existing graphics to assist in analysis. Graphical aids can assist with analysis and can help explain findings to

engineers and management.

developing data handling and reformatting/parsing techniques so that throughput can be increased. A large portion of the time spent in a typical analysis is often in data handling and parsing.

making new algorithms more readily available and providing a source for training in the usage of the new algorithms.

encouraging software vendors to provide analytical tools that can handle large quantities of data, either in the number of observations or in the number of variables, for identified high leverage techniques.

This research is supported by ARPA/Rome Laboratory under contract #F30602-93-0100, and by the Dept. of the Army, Army Research Office under contract #DAAH04-95-1-0466. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes not withstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of the Advanced Research Projects Agency, Rome Laboratory, or the U.S. Government.

References

[1] Paul R. Cohen, Michael L. Greenberg, David M. Hart, and Adele E. Howe. Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine*, 10(3):32-48, Fall 1989.

[2] John D. Emerson and Michal A. Stoto. Transforming data. In David C. Hoaglin, Frederick Mosteller, and John W. Tukey, editors, *Understanding robust and exploratory data analysis*. Wiley, 1983.

[3] Usama Fayyad, Nicholas Weir, and S. Djorgovski. Skicat: A machine learning system for automated cataloging of large scale sky surveys. In *Proceedings of the Tenth International Conference on Machine Learning* , pages 112-119. Morgan Kaufmann, 1993.

[4] Michael P. Georgeff and Amy L. Lansky. Procedural knowledge. *Proceedings of the IEEE Special Issue on Knowledge Representation* , 74(10):1383-1398, 1986.

[5] 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.

[6] Amy L. Lansky and Andrew G. Philpot. AI-based planning for data analysis tasks . *IEEE Expert*, Winter 1993.

[7] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.

[8] Robert St. Amant and Paul R. Cohen. Toward the integration of exploration and modeling in a planning framework. In *Proceedings of the AAAI-94 Workshop in Knowledge Discovery in Databases*, 1994.

[9] Robert St. Amant and Paul R. Cohen. A case study in planning for exploratory data analysis. In *Advances in Intelligent Data Analysis* , pages 1-5, 1995.

[10] 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*. Springer, 1995. To appear.

Management Issues In The Analysis Of Large-Scale Crime Data Sets

Charles R. Kindermann
Marshall M. DeBerry, Jr.
Bureau of Justice Statistics U.S. Department of Justice

1 THE INFORMATION GLUT

The Bureau of Justice Statistics (BJS), a component agency in the Department of Justice. has the responsibility for collecting, analyzing, publishing and disseminating information on crime. criminal offenders, victims of crime. and the operation of justice systems at all levels of government. Two very large data sets— the National Incident-Based Reporting System (NIBRS) and the National Crime Victimization Survey (NCVS)-are part of the analytic activities of the Bureau. A brief overview of the two programs is presented below.

2 NIBRS

NIBRS. which will eventually replace the traditional Uniform Crime Reporting (UCR)[1] Program as the source of official FBI counts of crimes reported to law enforcement agencies. is designed to go far beyond the summary-based U CR in terms of information about crime. This summary-based reporting program counts incidents and arrests. with some expanded data on incidents of murder and nonnegligent manslaughter.

In incidents where more than one offense occurs, the traditional UCR counts only the most serious of the offenses. NIBRS includes information about each of the different offenses (up to a maximum of ten) that may occur within a single incident. As a result. the NIBRS data can be used to study how often and under what circumstances certain offenses. such as burglary and rape, occur together.

The ability to link information about many aspects of a crime to the crime incident marks the most important difference between NIBRS and the traditional UCR. These various aspects of the crime incident are represented in NIBRS by a series of more than fifty data elements. The NIBRS data elements are categorized into six segments: administrative. offenses, property, victim, offender. and arrestee. NIBRS enables analysts to study how these data elements relate to each other for each type of offense.

[1]The Uniform Crime Reporting (UCR) Program is a nationwide, cooperative statistical effort of approximately 16.000 city. county, and state law enforcement agencies voluntarily reporting data on crimes brought to their attention. The Federal Bureau of Investigation (FBI) has administered this program since 1930.

3 NCVS

The Bureau of Justice Statistics also sponsors and analyzes the National Crime Victimization Survey (NCVS). and ongoing national household survey that was begun in 1972 to collect data on personal and household victimization experiences. All persons 12 years of age and older are interviewed in approximately 50.000 households every six months throughout the Nation. There are approximately 650 variables on the NCVS data file. ranging from the type of crime committed, the time and place of occurrence. and whether or not the crime was reported to law enforcement authorities. The average size of the data file for all crimes reported for a particular calendar year is 120 megabytes.

The NCVS utilizes a hierarchical file structure for its data records. In the NCVS there are four types of records: a household link record, followed by the household, personal. and incident records. The household record contains information about the household as reported by the respondent and characteristics of the surrounding area as computed by the Bureau of the Census. The person record contains information about each household member 12 years of age and older as reported by that person or proxy. with one record for each qualifying individual. Finally, the incident record contains information drawn from the incident report, completed for each household or person incident mentioned during the interview. The NCVS is a somewhat smaller data set than NIBRS. but may be considered analytically more complex because 1) there is more information available for each incident and 2) it is a panel design, i.e., the persons in each housing unit are interviewed every six months for a period of three years, thereby allowing for some degree of limited longitudinal comparison of households over time.

4 DATA UTILIZATION

An example of how those interested in the study of crime can tap the potentially rich source of new information represented by NIBRS is seen in the current Supplementary Homicide Reports data published annually by the FBI in its Crime in

the United States series. Crosstabulations of various incident-based data elements are presented, including the age, sex, and race of victims and offenders, the types of weapon(s) used, the relationship of the victim to the offender, and the circumstances surrounding the incident (for example, whether the murder resulted from a robbery, rape, or argument). The NIBRS data will offer a variable set similar in scope.

Currently, portions of eight states are reporting NIBRS data to the FBI. In 1991. three small states reported 500,000 crime incidents that required approximately one gigabyte of storage. If current NIBRS storage demands were extrapolated to full nationwide participation. 40 gigabytes of storage would be needed each year.

Although full nationwide participation in NIBRS is not a realistic short-term expectation. it is realistic to expect that a fourth of the U.S. could be represented in NIBRS within the next several years. The corresponding volume of data, 10 gigabytes each year. could still be problematic for storage and analysis.

Certain strategies may be chosen to reduce the size of the corresponding NIBRS data files. For example, most users of NIBRS data may not need or desire a data file that contains

all twenty-two types of GROUP A offences. which contains crimes such as sports tampering. impersonation. and gambling equipment violations. If a user is interested in much smaller file, only the more common offenses, such as aggravated assault, motor vehicle theft, burglary, or larceny/theft, could be included in the data set. Another area in which data reduction can be achieved is in the actual NIBRS record layout. Although the multiple-record format may aid law enforcement agencies in the inputting of the data, it can create difficulties in analyzing the files. For example, in the current NIBRS format, each record, regardless of type, begins with 300 bytes reserved for originating agency identifier (ORI) information. Currently, nearly a third of each ORI header is filler space reserved for future use. Moreover, the records for the different incident types have been padded with filler so as to be stored as fixed length records instead of variable length records. This wasted space occupied by multiple ORI headers and filler can be eliminated by restructuring and reorganizing the current file structure into a more suitable format that current statistical software packages can utilize.

Even with the restructuring of the current record formats, the annual collection of NIBRS data will still result in a large volume of data to be organized, stored, and analyzed. One strategy BJS is considering is to sample the NIBRS data in order to better manage the volume of data expected. Since the NIBRS program can be viewed as a potentially complete enumeration of incidents obtained by law enforcement agencies, simple random sampling could be employed, thereby avoiding the complications of developing a complex sample design strategy and facilitating the use of" off the shelf" statistical software packages.

Using the sample design of the NCVS, BJS has produced a 100 megabyte longitudinal file of household units that covers a period of four and one half years. This file contains information on both interviewed and noninterviewed households in a selected portion of the sample over the seven interviews. The NCVS longitudinal file can facilitate the examination of patterns of victimization over time, the response of the police to victimizations. the effect of life events on the likelihood of victimization, and the long term effects of criminal victimization on victims and the criminal justice system. However, current analysis of this particular data file has been hampered by issues relating to the sample design and utilizing popular statistical software packages. Since the NCVS utilizes a complex sample design. standard statistical techniques that assume a simple random sample cannot be utilized, Although there are software packages that can deal with complex sample designs, the NCVS data are collected by the Bureau of the Census under Title 13 of the U.S. code. As a result. selected information that would identify primary sampling units and clusters is suppressed to preserve confidentiality. Researchers, therefore, cannot compute variances and standard errors for their analyses on this particular data sets. BJS is currently working with the Bureau of the Census to facilitate the computation of modified sample information to be included on future public use tapes that will facilitate the computation of the appropriate sample variances.

Most of the current statistical software packages are geared to processing data on a case by case basis. The NCVS longitudinal file is structured in a nested hierarchical manner. When trying to examine events over a selected time period, it becomes difficult to rearrange the data in a way that will facilitate understanding the time or longitudinal aspects of the data. For example, the concept of what constitutes a "case record" depends on the perspective of the current question. Is a case all households that remain in sample over all seven interviews,

or is it those households that are replaced at every interview period? Moving the appropriate incident data from the lower levels of the nested file to the upper level of the household can complicate obtaining a "true" count of the number of households experiencing a victimization event, since many statistical software packages duplicate information at the upper level of the file structure down to the lower level.

5 FUTURE ISSUES

Local law enforcement agencies will be participating on a voluntary basis. NIBRS data collection and aggregation at the agency-level will be far more labor and resource-intensive than the current UCR system. What are the implications for coverage and data accuracy?

Criminal justice data have a short shelf life, because detection of current trends is important for planning and interdiction effectiveness. Can new methods be found to process massive data files and produce information in a time frame that is useful to the criminal justice community? Numerous offenses such as sports tampering are not of great national interest. A subset of the full NIBRS file based on scientific sampling procedures could facilitate many types of analyses.

How easy is it to integrate change into such a data system, as evaluations of NIB RS identify new information needs that it will be required to address? Does the sheer volume of data and reporting agencies make this need any more difficult than for smaller on-going data collections? As data storage technology continues to evolve, it is important to weigh both cost and future compatibility needs, particularly in regards to distributing the data to law enforcement agencies and the public. BJS will continue to monitor these technological changes so that we will be able to utilize such advances in order to enhance our analytic capabilities with these large scale datasets.

Analyzing Telephone Network Data

Allen A. McIntosh*
Bellcore

ABSTRACT

As our reliance on computer networks grows, it becomes increasingly important that we understand the behavior of the traffic that they carry. Because of the speeds and sizes involved, working with traffic collected from a computer network usually means working with large datasets. In this paper, I will describe our experience with traffic collected from the Common Channel Signaling (CCS) Network. I will briefly describe the data and how they are collected and discuss similarities with and differences from other large datasets. Next, I will describe our analysis tools and outline the reasons that they have been found useful. Finally, I will discuss the challenges facing us as we strive for a better understanding of more data from faster networks. While my emphasis in this paper is on the CCS Network, it has been my experience that both the problems and the solutions generalize to data from other networks.

As our reliance on computer networks grows, it becomes increasingly important that we understand the behavior of all aspects of the traffic that they carry. One such network, used by millions of people every day, is the Common Channel Signaling (CCS) Network. The CCS Network is a packet-switched network that carries signaling information for the telephone network. It carries messages for a number of applications, including messages to start and stop calls, to determine the routing of toll free (area code 800) calls and to verify telephone credit card usage. Network failures, while rare, can affect hundreds of thousands of telephone subscribers. As a result, telephone companies are very interested in the health of their portion of the CCS Network. At Bellcore, we help network providers keep their part of the CCS Network running smoothly by testing vendor equipment and by monitoring live networks. To test equipment, we build small test networks and subject them to extreme stress and catastrophic failure, such as might be caused by a fire at a switch or a backhoe cutting a cable during a mass call-in. We usually collect all the

* Senior Research Scientist, Statistics and Data Analysis Research Group, Bellcore, 445 South Street, Morristown, NJ 07960.

traffic on the test network, though selective collection may be used if a test involves subjecting the network to a heavy load. To monitor live networks, we collect all the traffic from a small subnetwork. Collecting all the data from a network provider's live network would require monitoring several thousand communication links, and is beyond the capacity of our monitoring equipment.

In North America, a protocol called Signaling System Number 7 (SS7, ANSI (1987)) governs the format of data carried by the CCS Network. [1] The major components of the CCS network are telephone switches (SSP's), database servers (SCP's), and SS7 packet switches (STP's). STP's are responsible for routing traffic, while SSP's and SCP's can only send and receive. STP's are deployed in pairs for redundancy. Each SSP and SCP is connected to at least one pair of STP's. The (digital) communications links between nodes run at a maximum speed of 56000 bits per second. When extra communications capacity is required, parallel links are used. 56000 bits per second is relatively slow, but there are many links in a CCS network, and many seconds in a day. Anyone trying to collect and analyze SS7 data from a CCS network soon must deal with large datasets.

To date, our main data-collection tool for both live and test networks has been a Network Services Test System (NSTS). The NSTS is about the size of a household washing machine. Each NSTS can monitor 16 bi-directional communication links, and store a maximum of 128 megabytes of data. This represents two to four million SS7 messages, which can be anywhere from one hour of SS7 traffic to and from a large access tandem to approximately four days of traffic to and from a small end office. We usually collect data by placing one NSTS at each STP of a pair. Thus, our datasets from Live networks are usually 256 megabytes in size. Datasets from test networks tend to be smaller, depending on the length of the test. Along with every message that it saves, the NSTS saves a header containing a millisecond timestamp, the number of the link that carried the message, and some other status information. The timestamps are synchronized with a central time source, and so are comparable between monitoring sites.

Our SS7 datasets have many of the "standard" features of the large datasets discussed in this volume. Inhomogeneity and non-stationarity in time are the rule. For example, Figure 1 shows the call arrival process on a communication link to a small switch from 22:45 Wednesday to 14:15 Sunday. There are 31,500 points in this plot, joined by line segments. Each point represents the number of calls received during a ten second interval, expressed in units of calls per second. Evidently there is a time-of-day effect, and the effect is different on the weekend. This dataset is

Note: This paper is reprinted by permission from Bellcore. Copyright 1996 by Bellcore.

[1] Telecommunications practice varies between countries. The CCS network considered in this paper is the North American network.

discussed in more detail in Duffy et al. (1993). They show that the call arrival process in Figure 1 is well described by a time-inhomogeneous Poisson process. The overall packet arrival process shows signs of long range dependence, and is not well modeled by a time-inhomogeneous Poisson process. Plots such as this also suggest that it is virtually impossible to find a stationary busy hour, the typical engineering period in traditional telephone networks.

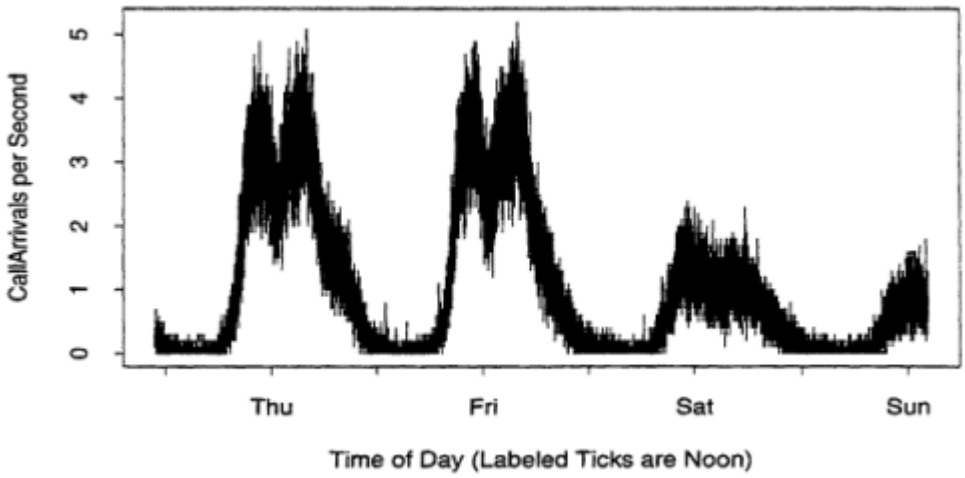

Figure 1: Call Arrival Process

The inhomogeneity can be subtle. For example, Figure 2 illustrates the number of calls in progress for the same switch as a function of time. Since most telephone calls are short, one would expect Figures 1 and 2 to be similar. However, Figure 2 contains an extra peak on Thursday evening which is not present in Figure 1. This suggests that evening calls (from residential customers) tend to be longer than daytime calls (from business customers).[2] Similar signs are visible Friday and Saturday evenings, but they are not as pronounced. Figure 3 is a plot of the number of calls in progress over a one week period, taken from a different dataset. Now we can see that Monday through Thursday evenings are similar, and the remaining evenings are very different. This abundance of features is typical of our data.

Some other features of our SS7 data make analysis more difficult. First, there are many different kinds of message, and most of them have variable length. This makes it difficult to use conventional database tools, which usually insist on fixed length records. Second, SS7 messages are binary, and several fields may be packed

[2] Duffy et al. (1994) show that the distribution of call homing times of nighttime calls has very heavy tails.

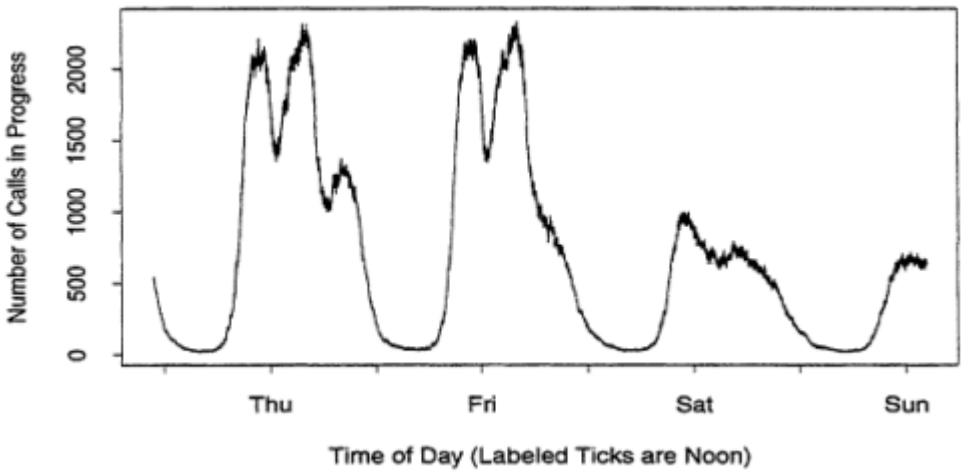

Figure 2: Calls In Progress

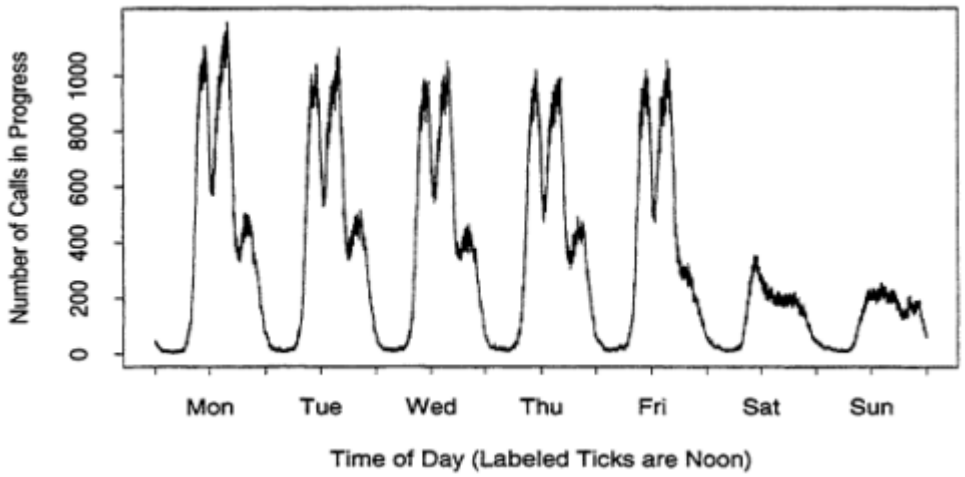

Figure 3: Calls in Progress

into a single byte to save space. This renders them unsuitable for processing by most statistical packages and other commonly used tools (UNIX utilities such as awk and perl, in our case) which are ASCII-based. In a pilot study, we tried converting our data to an ASCII representation and processing it in this form. This resulted in a tenfold increase in the size of our data. The ASCII tools that we used for processing, being very general, were not very efficient to begin with, and the data bloat exacerbated the problem.

In the end, we decided to keep our data in the original binary file format written by NSTS. We wrote a suite of tools (described below) to process the data in this format. We still needed tools to produce ASCII output, but these were usually used in the final steps of an analysis.

In designing our data processing tools, we followed an old and well tested paradigm. Kernighan and Plauger (1976) wrote:

> A surprising number of programs have one input, one output, and perform a useful transformation on data as it passes through. ... A careful selection of filters that work together can be a tool-kit for quite complicated processing.

Because our network datasets are sequential over time, it is natural to view them as sequences of messages. Our tools for analyzing binary SS7 data are thus message filters. Conceptually, they read one or more binary files one message at a time, perform some operation on each message, and write some or all of the messages out again. Some filters translate to ASCII instead of writing in binary. Our filters perform a number of functions, including

- subsetting. Huber (1996) mentions the importance of building subsets, an observation we share. Most of the subsetting operations on our SS7 data are performed by one filter program. This filter appears (usually multiple times) in any analysis I do. Its user interface is easily the most complex of any of our message filters. There are more than two dozen different fields that can be referenced. We have built a GUI (graphical user interface) to make this filter easier to use. Fortunately, the complexity of the external interface is not matched by either internal complexity or long running times.
- sampling.
- content and protocol checking. We can check for malformed SS7 messages, and can do some checking of the protocol for managing calls.
- sort/merge. The raw data files produced by NSTS need to be sorted into time order, and sorted files from multiple NSTS's need to be merged.

- call assembly. It takes several messages to set up and tear down a call. They appear at different times and may appear on different links. [The best analogy I can think of is to imagine point-of-sale data for a store where keystrokes for odd numbered keys are recorded on one tape and keystrokes for even numbered keys are recorded on another tape.] Significant processing is required to gather these together if one wants to analyze calls rather than packets.
- checking for wiring errors. We learned the hard way that this was necessary.
- drawing plots of link load in PostScript[3]
- counting, by message type, priority, destination, and so on.
- translating to ASCII, with varying levels of detail

It is instructive to examine how Figures 2 and 3 were produced. The data gathered by NSTS were stored on disk in time sorted order. To produce the plots, the data were run through the following filters, in sequence:

1. A filter to trim off end effects that arise because the two NSTS's do not start and stop simultaneously.
2. A filter to select only messages related to call setup and termination.
3. A filter to organize messages into groups where each group contains the messages from a single telephone call.
4. A filter to print out (in ASCII) the timestamps of the start and end of every call, along with a +1 (start) and a -1 (end).
5. An ASCII sort filter, since the output of the previous step is no longer in time sorted order.
6. A filter to do a cumulative sum of the result of the previous step and print out the value of the sum at a reasonable number of equally spaced times. The output from this step was handed to plotting software.

The filters in steps one through four are message filters, and the filters in steps five and six are ASCII filters.

Aside from the soundness of the underlying concept, there are a number of reasons why this approach succeeded. First, the message filters we have written are

[3] PostScript is a registered trademark of Adobe Systems Incorporated

efficient. A filter that copies every message individually takes 77 seconds elapsed time to read a 256 Megabyte file on my workstation. A standard system I/O utility takes 73 seconds for the same task. The bottleneck in both tasks is the I/O bandwidth. Most simple filters are I/O bound like this. An exploratory analysis can thus proceed by saving only a few key intermediate results (perhaps from steps requiring more extensive processing), and repeating other intermediate operations when required.

Second, we have made message filters easy to write. To build a simple filter, it is only necessary to write a C ++ function that examines a message and returns a value indicating whether it is to be output or discarded. Argument processing, a "Standard I/O" library, and a support library are provided for the more adventurous. I was both amused and disturbed that in some ways this puts us where we were over 20 years ago with the analysis of small and medium-sized datasets, namely writing one-of-a-kind programs with the help of a large subroutine library. Fortunately, programming tools have improved over the last 20 years, and the one-of-a-kind programs can be connected together.

Third, message filters are easily connected to perform complex processing tasks. This is easily done by combining filter programs, as in the example above. It is also easy to manipulate filters as objects in C++ code when considerations of convenience or efficiency dictate. Filters are implemented as C++ classes derived from a single base class. The interface inherited from the base class is small, and defaults are provided for all functionality. Support is also provided for lists of filters, and for simple Boolean operations like inversion and disjunction.

The analyses we perform on our data are intended for a wide audience whose members have different needs and very different views of the network. They include:

1. Network planners, who are interested in traffic trends over months or years. The network nodes themselves provide gross traffic data. Network monitoring can provide more detail, such as traffic counts for source-destination pairs.
2. Network operations personnel, who are interested in anomalies such as might be found in a set of daily source-destination pair counts. (See Duffy et al. (1993) Section 3.3)
3. Traffic engineers, who are interested in building models for traffic. The intent here is to evaluate buffer management policies and other issues affected by the randomness of traffic. Data can help validate models, though there are some pitfalls, as we shall see.
4. Analysts interested in anomalies in subscriber calling patterns.

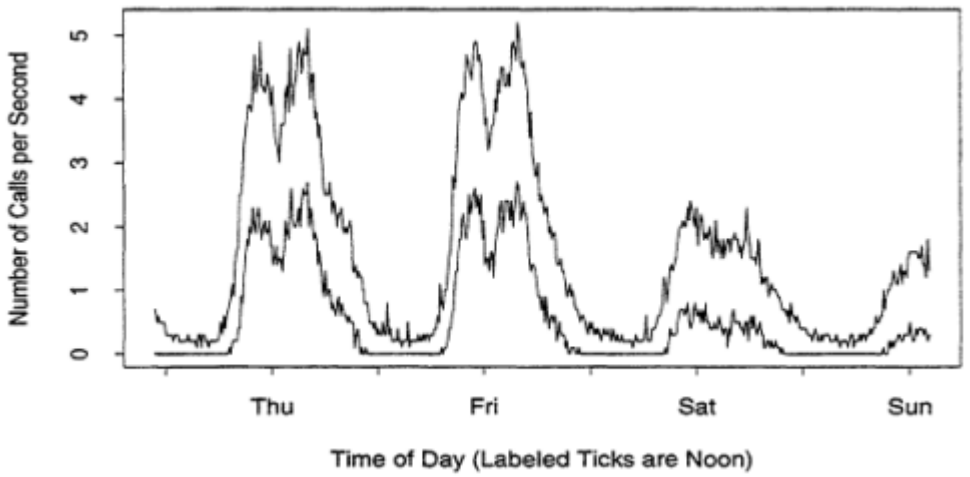

Figure 4: Figure 1 With Data Reduction

5. Testers, who are interested in protocol errors, and do not care about the data from the 99.99% of the time that the test network worked correctly.

To satisfy these needs, we must provide tools to explore both trends and exceptions in our data at differing levels of detail. Our filter building blocks allow us to design software tools to fit each application. Because the applications share a common infrastructure, the programming effort required is not large.

A 256 megabyte dataset contains the details of more telephone calls than there are pixels on my workstation screen, and nearly as many messages as there are dots on an 8.5 by 11 piece of paper at 300 dots per inch. Good visualization tools must therefore be able to avoid burying important details in a blob of ink. Even drawing the blob can be prohibitively expensive. For example, Figure I contains 31,500 line segments, and seems to take forever to draw on an X terminal running over a modem. Figure 4 plots the same data as Figure 1, but reduces it by plotting the minimum and maximum 10 second load in each 10 minute period. Peaks and valleys have been preserved, thus preserving the overall appearance of the plot, but the number of points plotted has been reduced by a factor of 30.

On the other hand, smoothing to reduce data is not a good idea, as Figure 5 illustrates. The upper and lower jagged lines are respectively the maximum and minimum 5-second loads in each one minute period. The smooth line in the middle misses the main feature of the plot completely.

To date, we have not had much opportunity for in-depth modeling of our data. The efforts described in Duffy et al. (1993) and Duffy et al. (1994) barely scratch the

surface. The complexity of Figure 3 suggests that building a comprehensive model is a daunting task. Traditional techniques can be inappropriate for even simple models. For example, a 256 megabyte file contains roughly six million messages. In a certain context, one can view these as Bernoulli trials with $p = 0.5$. Since adherence to the Bernoulli model is not perfect, a sample this size is more than sufficient to reject the hypothesis that $p = 0.5$ in favor of $p = 0.5 + \epsilon$ (and indeed in favor of something that is not Bernoulli at all). Unfortunately, the hypothesis tests don't answer the real question: does it matter?

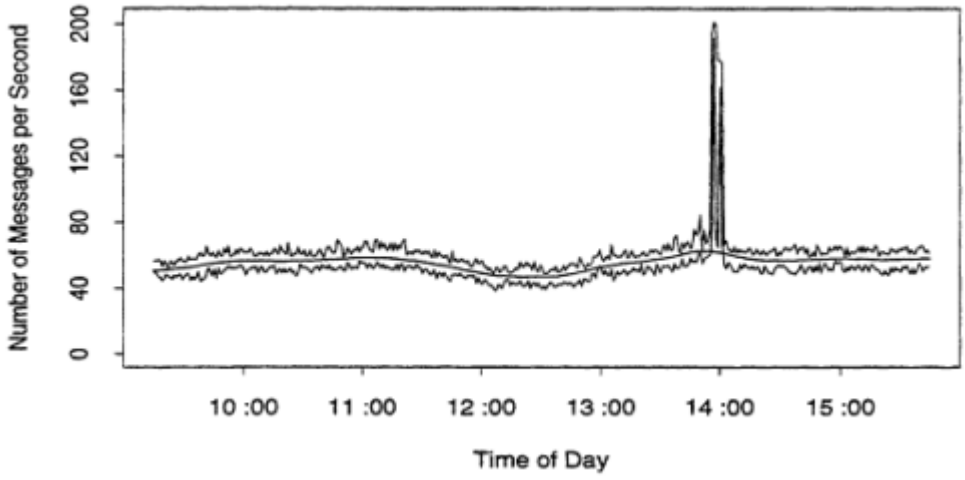

Figure 5: The Dangers of Oversmoothing

There are some classes of analysis, such as resampling methods, that are more practical with large datasets. An adequate model for a large dataset may be so complex that one would be better off resampling. Are there adequate tools for doing this? One potential pitfall that comes to mind immediately: a 32 bit random number generator may not provide enough bits to sample individual records.

The strategy described here has worked well for our 256 megabyte datasets. We have a proven set of tools that enable us to do flexible, efficient analyses. However, there are monsters in our future! We are testing a second generation of SS7 data collection equipment that can store roughly 55 gigabytes of data from 96 (bidirectional) communication links at a single STP site without manual intervention. This is roughly a week of data from a medium-sized Local Access and Transport Area (LATA). If someone is available to change tapes periodically, the amount of data that can be collected is unlimited.

We are still trying to come to grips with datasets of this size. Just reading 55

gigabytes of data off tape and putting it onto a disk takes two and a half days. Storing the data on disk is problematic, because my workstation doesn't allow files bigger than two gigabytes. It is likely that some sort of parallelism will be required for both storage and processing. Splitting the data across the right dimension is a difficult problem. The wrong choice can force an analysis to be serial rather than parallel, and it is not clear that the same choice cam work for all questions of interest.

A parallel effort at Bellcore has involved the collection and analysis of Ethernet traffic (Leland et al. (1994), Willinger et al. (1995), Leland et al. (1995)). Datasets here have been larger, in part because of the higher transmission speed involved, but there has been significantly less detail available because only the IP headers were saved.

We are now turning our attention to traffic on emerging networks, such as ATM and Frame Relay. These networks are used to carry data from other protocols (IP, SS7). As a consequence, we expect to encounter many of the challenges, and the opportunities, that were present in our SS7 and Ethernet datasets.

ACKNOWLEDGEMENTS

Diane Duffy, Kevin Fowler, Deborah Swayne, Walter Willinger and many others contributed to the work described here. The opinions axe mine.

REFERENCES

ANSI (1987). *American National Standard for Telecommunications-Signalling System Number* 7. American National Standards Institute, Inc., New York. Standards T1.110-114.

Duffy, D. E., McIntosh, A. A., Rosenstein, M., and Willinger, W. (1993). Analyzing Telecommunications Traffic Data from Working Common Channel Signaling Subnetworks. In Tarter, M. E. and Lock, M. D., editors, *Computing Science and Statistics: Proceedings of the 25th Symposium on the Interface*, pages 156-165. Interface Foundation of North America.

Duffy, D. E., McIntosh, A. A., Rosenstein, M., and Willinger, W. (1994). Statistical Analysis of CCSN/SS7 Traffic Data from Working CCS Subnetworks. *IEEE Journal on Selected Areas in Communications*, 12(3):544-551.

Huber, P. J. (1996). Need to get the title of this article. In Kettenring, J. and Pregibon, D., editors, *Massive Data Sets*, pages 111-999.

Kernighan, B. W. and Plauger, P. L. (1976). *Software Tools*. Addison-Wesley, Reading, Mass.

Leland, W. E., Taqqu, M. S., Willinger, W., and Wilson, D. V. (1994). On the Self-Similar Nature of Ethernet Traffic (Extended version). *IEEE/ACM Transactions on Networking*, 2:1-15.

Leland, W. E., Taqqu, M. S., Willinger, W., and Wilson, D. V. (1995). Self-Similarity in High-Speed Packet Traffic: Analysis and Modeling of Ethernet Traffic Measurements. *Statistical Science*, 10:67-85.

Willinger, W., Taqqu, M. S., Sherman, R., and Wilson, D. V. (1995). Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level. *Computer Communications Review*, 25:100-113. Proceedings of the ACM/ SIGCOMM'95, Boston, August 1995.

Massive Data Assimilation/Fusion in Atmospheric Models and Analysis: Statistical, Physical, and Computational Challenges

Gad Levy
Oregon State University
Carlton Pu
Oregon Graduate Institute of Science and Technology
Paul D. Sampson
University of Washington

ABSTRACT

The goals and procedures of the most data intensive operations in atmospheric sciences-data assimilation and fusion-are introduced. We explore specific problems which result due to the expansion in observing systems from conventional to satellite borne and the corresponding transition from small, medium, and large data sets to massive data sets. The satellite data, their volumes, heterogeneity, and structure are described in two specific examples. We illustrated that the atmospheric data assimilation procedure and the satellite data pose unique problems that do not exist in other applications and are not easily addressed by existing methods and tools. Existing solutions are presented and their performance with massive data sets is critically evaluated. We conclude that since the problems are interdisciplinary, a comprehensive solution must be interdisciplinary as well. We note that components of such a solution already exist in statistics, atmospheric, and computational sciences, but that in isolation they often fail to scale up to the massive data challenge. The prospects of synthesizing an interdisciplinary solution which will scale up to the massive data challenge are thus promising.

1 INTRODUCTION

The purpose of data assimilation is to combine atmospheric measurements and observations with our knowledge of atmospheric behavior in physical atmospheric models, thus producing a best estimate of the current state of the atmosphere. The similar but distinct purpose of data fusion is to extract the best information from a multitude of heterogeneous data sources, thus devising an optimal exploitation of the synergy of these data. The resulting analyses (a.k.a. 'initialization fields') have great diagnostic value, and are the basis for

model prediction. This procedure of analysis and model initialization has seen an exponential growth in the volume of observational geophysical data. The main purpose of this paper is to (1) critically evaluate how existing methods and tools scale up to the massive data challenge: and (2) explore new ideas/methods/tools appropriate for massive data sets problems in atmospheric science. Our interest and focus is in the joint exploration of the different facets of what we consider some of the weakest components of current data assimilation/fusion schemes in atmospheric and climate models as they attempt to process massive data sets. We firmly believe that since the problems are interdisciplinary, a comprehensive solution must bring together statisticians, atmospheric and computational scientists to explore general methodology towards the design of an efficient, truly open (i.e., standard interface), widely available system to answer this challenge. Recognizing that the greatest proliferation in data volume is due to satellite data, we discuss two specific problems that arise in the analysis of such data.

In a perfect assimilation scheme, the processing must allow merging of satellite and conventional data, interpolated in time and space, and for model validation, error estimation and error update. Even if the input and output data formats are compatible, and the physical model is reasonably well understood, the integration is hampered by several factors. These roadblocks include: the different assumptions made by each of the model components about the important physics and dynamics, error margins and covariance structure, uncertainty, inconsistent and missing data, different observing patterns of different sensors, and aliasing (Zeng and Levy, 1995).

The Earth Observing System and other satellites are expected to down-load massive amounts of data, and the proliferation of climate and General Circulation Models (GCM) will also make the integrated models more complex (e.g., review by Levy, 1992). Inconsistency and error limits in both the data and the modeling should be carefully studied. Since much of the data are new, methods must be developed which deal with the roadblocks just noted, and the transformation of the (mostly indirect) measured signal into a geophysical parameter.

The problems mentioned above are exacerbated by the fact that very little interdisciplinary communication between experts in the relevant complementary fields takes place. As a consequence, solutions developed in one field may not be applied to problems encountered in a different discipline, efforts are duplicated, wheels re-invented, and data are inefficiently processed. The Sequoia 2000 project (Stonebraker et al., 1993) is an example of successful collaboration between global change researchers and computer scientists working on databases. Their team includes computer scientists at UC Berkeley, atmospheric scientists at UCLA, and oceanographers at UC Santa Barbara. Data collected and processed include effects of ozone depletion on ocean organisms and Landsat Thematic Mapper data. However, much of the data management and statistical methodology in meteorology are still being developed 'in house' and are carried out by atmospheric scientists rather than in collaborative efforts. Meanwhile, many statisticians do not use available and powerful physical constraints and models and are thus faced with the formidable task of fitting to data statistical models of perhaps unmanageable dimensionality.

As a preamble, we describe in the next section the satellite data: volumes, heterogeneity, and structure, along with some special problems such data pose. We then describe some existing methods and tools in section three and critically evaluate their performance with massive data sets. We conclude with some thoughts and ideas of how methods can be

improved and developed to scale up to the massive data sets challenge.

2 THE SATELLITE DATA

In atmospheric studies, as in many other fields of science, researchers are increasingly relying upon on the availability of global data sets to support their investigations. Problems arise when the data volumes are huge and the data are imprecise or undersampled in space and time as is often the case with satellite sampling. Numerical weather prediction models have traditionally accepted observations at given time intervals (synoptic times) from a network of reporting stations, rawindsondes, island stations, buoys, weather ships, ships of opportunity, aircrafts and airports, treating the state variables in a gridded fashion. This has set the norm for the acceptable data format in these studies, dictating the need for most observations to be eventually brought to 'level 3' (gridded) form. It has also contributed to the development of the statistical and meteorological field known as objective analysis. However, volumes and sampling patterns of satellite data often lead to bottlenecks and to the failure of traditional objective analysis schemes in properly processing asynoptic satellite data to level 3 geophysical records as the examples in the next paragraphs demonstrate.

Figure 1 presents a small (storm-size) scale schematic illustration of data structure, volume, and heterogeneity. In it, data from three different satellite instruments (two wind speed products from different channels of the Special Sensor Microwave Imager (SSM/I) on the Defense Meteorological Satellite Program (DMSP) space craft, and one wind vector product from the Active Microwave Instrument (AMI) on board the first European Remote Sensing (ERS1) satellite) are combined with the European Centre for Medium-Range Weather Forecasts (ECMWF) model thermodynamic output to create composite analysis fields (upper panels). Each of the data products has been sampled at different times and locations and has already undergone some quality control and data reduction procedures. The compositing procedure of eighteen looks at a single storm such as the one illustrated in fig. 1 required the reduction of approximately 3 Gb of input data to 290 Mb in the final product. Operational weather prediction centers need to process similar products four times daily, globally, at 20 vertical levels, and with additional conventional and satellite data. The imperative of having fast algorithms and fast data flow is clear in this example.

The monthly mean (climate scale level 3 product) stratospheric water vapor from the Microwave Limb Sounder (MLS) on board the Upper Atmosphere Research Satellite (UARS) for January 1992 is shown in Figure 2. Spatial structure which is related to the satellite orbital tracks is apparent in this figure. Careful inspection of the maps provided by Halpern et al. (1994) reveals similar structures in the ERS1/AMI monthly and annual mean for 1992, as well as in the Pathfinder SSM/I monthly mean wind speeds maps. The corresponding monthly means created from the ECMWF daily synoptic maps (also presented in Halpern et al., 1994) do not show this structure. These observations strongly imply that the structure is caused by the sampling rather than by an instrument error. Zeng and Levy (1995, hereafter, ZL95) designed an experiment to confirm that the structure is indeed a result of the satellite sampling. The ECMWF surface wind analyses were sampled with the ERS1 temporal-spatial sampling pattern to form a simulated data set which exhibited the same structure as in figure 2. In their experiment, the application of a bicubic spline filter to the monthly

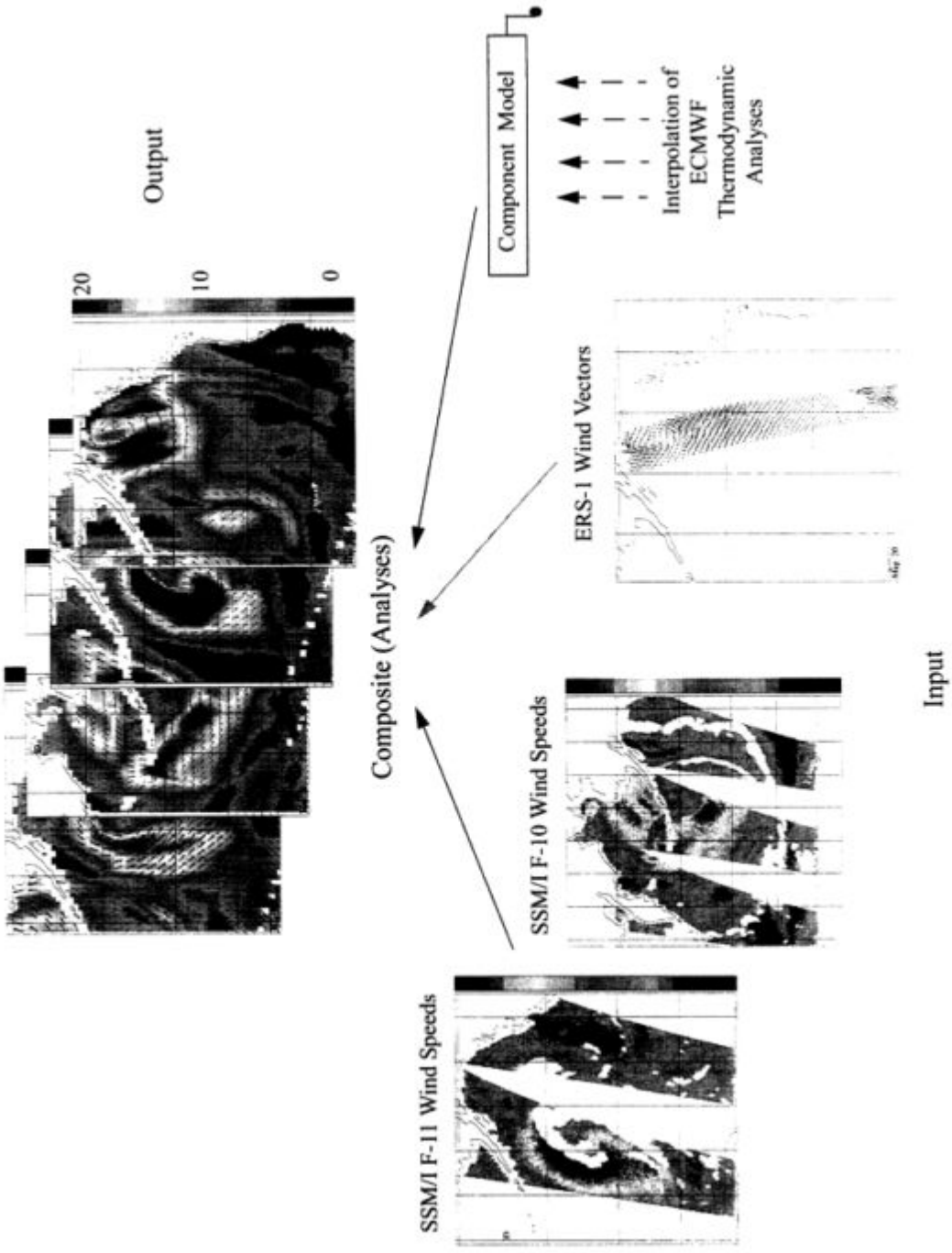

Figure 1: Schematic of Compositing Process

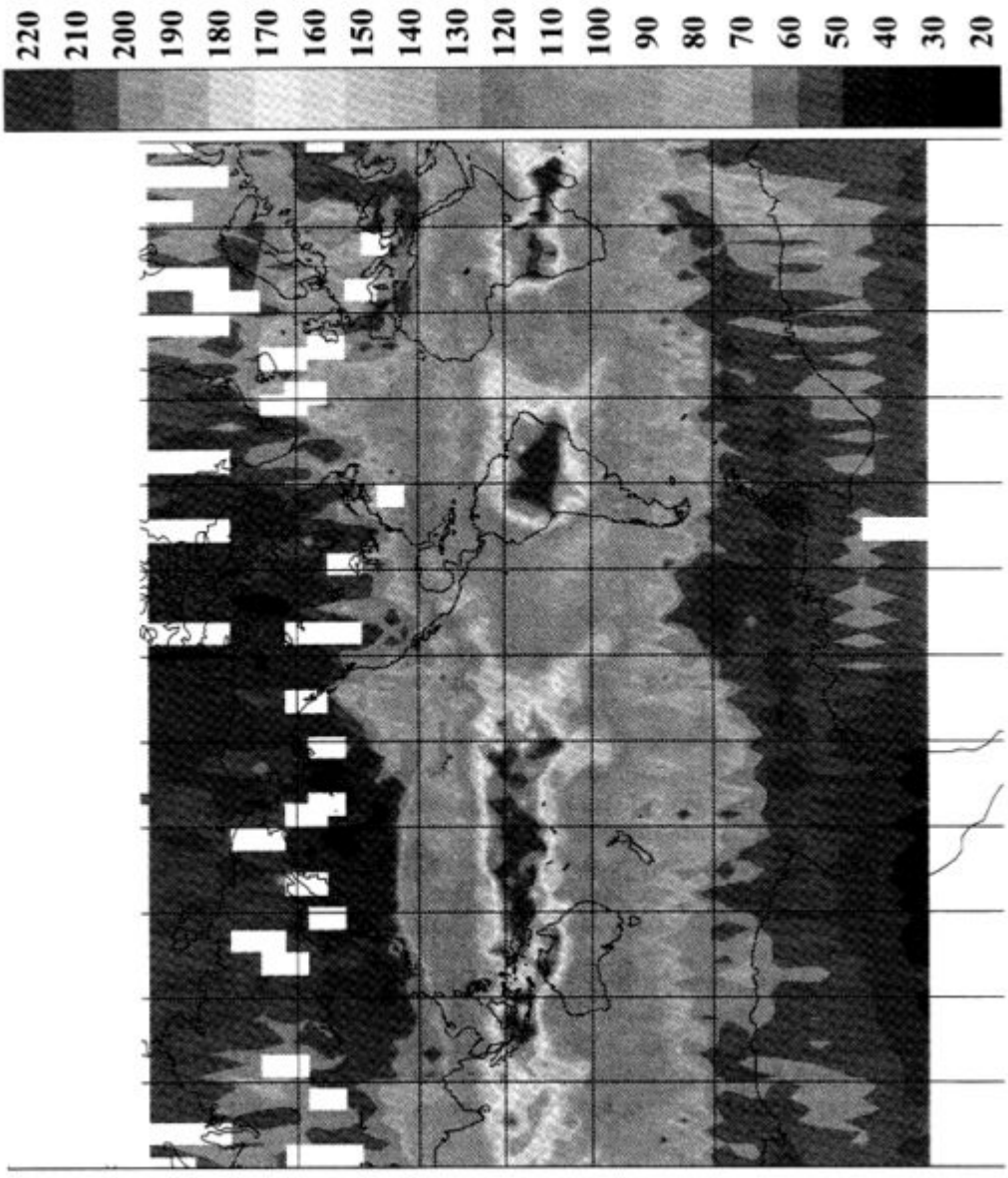

Figure 2: UARS MLS 215hPa H2O (ppmv) - January 1992

mean simulated data resulted in a field that was usually free of the structure. However, large biases of up to 3 m s-1 remain in several areas, and there was no significant reduction in the variance added to the control set by the aliasing (spatial structure) caused by the satellite sampling in these areas. The scatter plot of the smoothed field versus the control set presented by ZL95 demonstrates that the smoothed field has less variance but is still seriously biased from the control set (0.5 m s-1 globally). Problems with spectral analysis of both the simulated and simulated-smoothed fields were also reported by ZL95.

The examples above underscore two serious problems with the analysis and assimilation of satellite data in atmospheric studies. The first example demonstrates the special need for the construction of efficient, inexpensive, maintainable, and modular software tools for application in synoptic scale atmospheric models. The second example and the analysis in ZL95, clearly show how irregular sampling and undersampling at higher frequencies by polar-orbiting satellite instruments can lead to aliasing at scales which are of interest to climate studies requiring monthly means. It was pointed out by Halpern (1988) and Ramage (1984) that an error of 0.5 m s-1 in surface wind in the tropics may lead to an uncertainty of about 12 W m-2 in surface heat flux. This amount of uncertainty was associated with a 4K increase in global sea surface temperature in model sensitivity tests reported by Randall et al. (1992).

3 EXISTING METHODS AND TOOLS: PROSPECTS AND LIMITATIONS

In this section we critically evaluate some disciplinary and interdisciplinary methods that we have experimented with while trying to address the specific needs described in section two. We also discuss potential improvement and available products that may make these methods scale up to massive data.

Part of the special needs demonstrated by the first example in the previous section is for efficient data storage and retrieval tools. The main goal of database researchers in the Sequoia 2000 project is to provide an efficient storage and retrieval mechanism for the scientific data being collected, characterized by massive size (100 Terabytes in four sites), complex data types as mentioned above, and sophisticated searching for scientific research. The project has been quite successful in the creation of software for the management of tertiary storage (tapes and cartridges), since the amount of data still exceeds the current economic capacity of magnetic disks.

However, the Sequoia 2000 benchmark, created as a model for testing and evaluating databases for Earth Science users, does not address the critical issues mentioned previously in this paper. Concretely, the Sequoia 2000 benchmark consists primarily of four kinds of data: raster data, point data, polygon data, and directed graph data. While these benchmarks are very useful, they do not touch data assimilation problems that are the real bottleneck in the system.

In general, the lack of attention to data assimilation has been the situation with computer vendors as well. Many vendors currently offer Hierarchical Storage Managers capable of storing and handling petabytes of data, typically using a combination of magnetic disks, optical juke boxes, and tapes/cartridges managed by robots. The system software knows

how to handle and process bits, possibly arranged in columns as in relational database management systems, but the handling of missing data and irregularly taken samples is beyond the state of art of current systems software.

More recently, new object-relational database management systems such as Illustra have created support for new access methods particularly suitable for special applications such as Earth Sciences. However, these new access methods are still concerned primarily with bit movement and comparisons. Most commercial databases and software, including Illustra, simply assume that data are consistent and clean. In fact, discussions on data assimilation are absent from almost all of Sequoia 2000 papers on data management, including Stonebraker et al. (1993). This is another manifestation of the difficulties of cross-fertilization among disciplines.

As noted above, most database management systems provide support only for precise data, though in the physical world data are often imprecise. In these cases, the scientist is left with the unpleasant decision of whether to ignore the imprecise information altogether and store some approximation to the precise value, or to forgo the use of a standard database management system and manage the data directly. The latter is the decision most commonly taken. However, with ever increasing volumes of data and with real time processing demands (e.g., first example in section 2), such a decision can no longer be afforded. In many of the situations where precise data are not available, information much more useful than "value unknown" or "predicate is possibly true" is available, even though imprecise. One of the most common and useful forms of information available to a scientist is the ability to bound the amount of imprecision and estimate the error associated with the data. We think that it could prove valuable if the database management system were able to represent and store values with bounded error, along with error covariance information, thus supporting the direct manipulation of imprecise data in a consistent and useful manner according to physical models. In this context, the most immediate problems we need to address are: (1) can we represent imprecise and error information, (2) can we develop a data model for imprecise information, and (3) is it feasible to manipulate imprecise data in a consistent manner? The value representation will be designed explicitly to deal with imprecise data with known error bounds. Our preliminary (disciplinary) work includes an algebra for interval relations that uses methods from interval arithmetic as operators (Barga and Pu, 1993). Algorithms designed for Epsilon Serializability (Pu and Leff, 1991) provide us with the means for bounding the amount of the imprecision introduced into the data.

In atmospheric modeling, generally a forecast is combined with data in a manner that takes account of correlation structure between the various sources of data. Certain kinds of errors or imprecision have a complicated correlation structure. Putting an interval about such numbers and propagating these intervals by interval methods do not capture common kinds of error structure that occur in meteorological observational data bases. Similarly, the often non-linear physical model operations require different methods of propagating imprecision. Although interval methods are applicable in other sciences, more sophisticated error handling must be included to be useful in atmospheric modeling. We are interested in utilizing error covariance information and testing the feasibility of building software tools that facilitate such integration. A toolkit approach is necessary to lower the cost of incorporating these error handling techniques into atmospheric models, so the model prediction produced can achieve a quality inaccessible to naive models.

It is clear from the previous section that the spatial-temporal sampling requires new methods for the interpolation and extrapolation of the gridded data in a manner that will provide accurate estimates in gap areas and times. Most applications of spatial estimation (i.e., objective analysis) have used models for the spatial-temporal covariance structure that are (1) separable in time and space—i.e., that factor into separate models for the spatial and temporal correlation structure, and (2) stationary in time and space. Some do accommodate nonstationarity in the mean, but do not accurately reflect nonstationarity (heterogeneity) in the spatial covariance structure. There is little reason to expect spatial covariance structures to be homogeneous over the spatial scales of interest in our application. In our attempts to perform analyses based primarily on scatterometer data (e.g., Levy and Brown, 1991: Levy, 1994) we have experimented with common methods of interpolation and extrapolation of the original data. Most (e.g., the B-spline interpolation and smoothing in section 2) are incapable of handling the unique satellite nonsynoptic sampling pattern even on the much larger (monthly) scale (e.g., the second example in section 2).

ZL95 have designed a three-dimensional spatial-temporal interpolator. It makes use of both the temporal and spatial sampling pattern of the satellite, substituting temporal information where spatial information is missing, and vice versa. Since there are usually non-missing values around a missing value when both the time and space dimensions are considered, a missing value at a point in space and time can be estimated as a linear (weighted) combination of the N non-missing values found within a prescribed space and time 'neighborhood'.

There are several shortcomings to the ZL interpolator which need to be addressed if it is to be generalized. Since the interpolator does not use any spatial or temporal correlation structure information the weights it employs may be sub-optimal. Establishing a systematic approach to determine the optimal weight function for specific satellite instruments or study goals would make the interpolator more robust. Unfortunately, since only simulated data were used in ZL95, there was no control set to verify the results in an absolute sense or to test whether the interpolator weights are optimal for the true field. The ECMWF field (control set) in ZL95 does not contain high frequency oscillations with temporal scale shorter than the 6-hour ECMWF analysis interval or spatial scale smaller than the grid spacing, which may appear in real satellite data. The rest of this section outlines ideas for methods that may address these shortcomings and scale up to the massive satellite data.

Sampson and Guttorp, 1992 (hereafter SG92) developed a modeling and estimation procedure for heterogeneous spatial correlation that utilizes the fact that much environmental monitoring data are taken over time at fixed monitoring sites, and thus provide a sequence of replicates from which to compute spatial covariances. Their technique uses multidimensional scaling to transform the geographic coordinates into a space where the correlation structure is isotropic and homogeneous so standard correlation estimation techniques apply. When these methods are applied to polar orbiting satellite data rather than to reporting stations one is faced again with the unique problem related to the sampling frequency: waiting for the satellite to return to the same spatial location may result in an irrelevantly long temporal lag. Additionally, with massive data and increased resolution, the dimensionality of the problem gets to be unmanageably large. Therefore the adaptation of the SG92 model to the massive satellite data sets requires different data processing and estimation procedures. We propose to start by simultaneously reducing the data into representative summaries and

modeling the space-time correlation error structure. One can then directly carry out the space-time estimation. Relying on good physical models ensures that the statistical model is now needed merely to describe an error field which is usually much smaller in magnitude than the observed or predicted field. This approach derives from an analysis and modeling of temporal autocorrelation functions and space-time cross-correlation functions bound by strong physical constraints. It incorporates statistical error information into the ZL95 idea of proper data reduction and substituting temporal information for missing spatial information, while relaxing some of the assumptions implicit in the ZL method (e.g., stationarity, isotropy).

4 SUMMARY AND CONCLUDING REMARKS

The greatest proliferation in data volumes in atmospheric studies is due to satellite data. The importance of these data for monitoring the earth atmosphere and climate cannot be underestimated. However, the unique perspective from space that polar orbiting satellites have is accompanied by massiveness of data and a unique sampling pattern which pose special problems to the traditional data assimilation and management procedures. For most applications these data need to be transformed into 'level 3' (gridded) form. We have presented two specific examples of two different processes to illustrate the problems and special needs involved in properly generating the level 3 data. Without devising proper tools for error assessment and correction, many of the level 3 global data sets may lead to serious misinterpretation of the observations which can be further propagated into atmospheric models.

We have identified some disciplinary partial solutions, along with their limitations. New object-relational database management systems offer some needed support for new access methods and for the management and storage of massive data sets, but do not handle imprecise data. Interval methods attempt to handle imprecise data but do not propagate observational error information properly. The ZL95 method properly interpolates and reduces data on some scales, but may be sub-optimal for some sensors. It is scale dependent, and does not incorporate statistical error information. The SG92 estimation technique handles heterogeneous spatial correlation for small and medium data sets, but does not scale up to massive data sets as its dimensionality increases unmanageably with increased data volume and resolution.

A comprehensive solution is possible by a synergistic combination of the partial disciplinary solutions. We have outlined an idea for a general methodology to incorporate the statistical error structure information with the physical and dynamical constraints and with proper data reduction into representative summaries. A better statistical, physical, and numerical understanding of the error structure and growth may then lead to software solutions that will properly propagate imprecision.

ACKNOWLEDGMENTS

The authors are grateful to Suzanne Dickinson for generating the figures and commenting on the manuscript. This work was jointly supported by the Divisions of Mathematical Sciences and Atmospheric Sciences at the National Science Foundation under grant DMS-9418904.

References

[1] Barga R., and C. Pu. *Accessing Imprecise Data: An Interval Approach* . IEEE Data Engineering Bulletin, 16, 12-15, 1993.

[2] Halpern, D., *On the accuracy of monthly mean wind speeds over the equatorial Pacific* J. Atmos. Oceanic Technol., 5, 362-367, 1988.

[3] Halpern, D., O. Brown, M. Freilich, and F. Wentz, *An atlas of monthly mean distributions of SSMI surface wind speed, ARGOS buoy drift. AVHRR/2 sea surface temperature. AMI surface wind components, and ECMWF surface wind components during 1992*. JPL Publi, 94-4, 143 pp., 1994.

[4] Levy, G., *Southern hemisphere low level wind circulation statistics from the Seasat scatterometer*. Ann. Geophys., 12, 65-79, 1994.

[5] Levy, G., *Trends in satellite remote sensing of the Planetary Boundary Layer, 1993. (Review chapter), in* Trends in Atmospheric Sci., 1 (1992), 337-347. Research Trends Pub.

[6] Levy, G., and R. A. Brown, *Southern hemisphere synoptic weather from a satellite scatterometer*. Mon. Weather Rev., 119, 2803-2813, 1991.

[7] Pu C., and A. Leff, *Replica control in distributed systems: An asynchronous approach*. In Proceedings of the 1991 ACM SIGMOD International Conference on Management of Data, 377-386, Denver, May 1991.

[8] Ramage, C.S., *Can shipboard measurements reveal secular changes in tropical air-sea heat flux?* J. Clim. Appl. Meteorol., 23, 187-193, 1984.

[9] Randall, D.A., et al., *Intercomparison and interpretation of surface energy fluxes in atmospheric general circulation models*. J. Geophys. Res., 97, 3711-3724, 1992

[10] Sampson P.D., and P. Guttorp, *Nonparametric estimation of nonstationary spatial covariance structure*. Journal of the American Statistical Association 87, 108-119, 1992.

[11] Stonebraker, M., Frew, J., Gardels, K., and J. Meredith, *The Sequoia 2000 Storage Benchmark, in* Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., 1993.

[12] Zeng L. and G. Levy, *Space and time aliasing structure in mean polar-orbiting satellite data*. Journal of Geophysical Research, Atmospheres, 100, D3, pp 5133-5142, 1995.

PART III

ADDITIONAL INVITED PAPERS

Massive Data Sets and Artificial Intelligence Planning

Robert St. Amant and Paul R. Cohen
University of Massachusetts

ABSTRACT

Generating a description of a massive dataset involves searching through an enormous space of possibilities. Artificial Intelligence (AI) may help to alleviate the problem. AI researchers are familiar with large search problems and have developed a variety of techniques to handle them. One area in particular, AI planning, offers some useful guidance about how the exploration of massive datasets might be approached. We describe a planning system we have implemented for exploring small datasets, and discuss its potential application to massive datasets.

1 THE INFORMATION GLUT

It has been said that the Sunday New York Times contains more unique assertions than an average person encountered during a lifetime in the Renaissance. After a week of email and telephone calls, reading papers, listening to the news, skimming direct-mail catalogs and channel-hopping on TV, we are given little respite by the Times. Researchers in Artificial Intelligence (AI) have developed methods to both manage and exploit this information glut, so that we might all enjoy a more contemplative life. We provide a quick tour of AI applications for processing large amounts of data, and then focus on what AI can contribute directly to data analysis.

A variety of AI applications have been developed to take advantage of the the availability of huge quantities of information in electronic form. Examples include personal assistants that filter news stories, schedule meetings, and "crawl the web" in search of information at remote Internet sites. Many of these applications are easy to build and require little intelligence; the effort of a few hours is enough, for example, to write a filtering program to find calls for papers in the welter of Internet traffic. More intelligence is required if an assistant is to take account of human goals and preferences. One's personal email assistant should learn that broadcast messages about cars with lights on in the parking lot are irrelevant unless the car might be one's own. Similarly, an information retrieval assistant should

understand language—no mean feat—including indirect speech acts. If a student were to tell a hypothetical librarian assistant that she's looking for "a book by Tukey," she means that she doesn't know which book she wants, she'd like to see some titles, and then she'd like the assistant to send her one.

Thomas, the front-end to Congress's information system, relies on sophisticated information retrieval algorithms in responding to user requests [ref]. Natural language processing systems exploit online databases of newswire stories to learn rules that resolve word-sense ambiguity [ref]. Machine learning techniques can autonomously generate classifications of stellar objects in large scale sky surveys [3]. Planning systems can help organize access to enormous quantities of satellite image data [6]. Researchers in information retrieval, machine learning, knowledge discovery in databases, and natural language understanding are finding that the information glut is actually helpful, providing an inexhaustible supply of training data, test cases, and opportunities to develop new techniques. Their results are case studies in how massive datasets in different domains can be handled.

Our own research interest lies in exploratory data analysis (EDA). We have developed an assistant for intelligent data exploration, called AIDE, to help users with the task. AIDE is a knowledge-based planning system that incrementally explores a dataset, guided by user directives and its own evaluation of indications in the data [10]. The system is mixed-initiative: the user can let AIDE run unattended, searching for its own conception of interesting structure, or can use AIDE as an interactive statistics package, or can combine the two processes, letting AIDE perform some of the tasks of exploration, but under human guidance. While AIDE is not a system for exploration of massive datasets, we believe that the issues addressed by AIDE are relevant to any system that would explore data semi-autonomously.

2 AN PLANNING PERSPECTIVE ON DATA ANALYSIS

EDA is a search problem, not unlike medical diagnosis or chess. At every juncture, you can take one of several actions—transform a variable, order a CAT scan, sacrifice a bishop. After each action, your state of knowledge changes. Because actions are generally not free, you look into the future to evaluate potential outcomes, so the action you eventually select is by some local criterion "best." If local criteria guaranteed global success, chess wouldn't be a challenge. Nor would diagnosis or EDA. But even if local criteria are heuristically good, the search space in the vicinity of a decision can be enormous. The average branching factor of a search is the average number of actions feasible at each decision. Chess has an average branching factor of 40, or thereabouts, and because games last roughly 50 moves, the search space for chess contains 40^{50} positions. EDA has a much larger branching factor. Any statistics package permits dozens of operations on dozens or hundreds of subsets of variables. An analyst might run any of hundreds of operations at each juncture, from simple transformations, to descriptive statistics, to model specification and analyses of residuals, and so on. The search space for EDA, even for a small dataset, is effectively infinite.

AI tames intractable search spaces two ways, by reformulating a problem to have a smaller search space, and by using expert knowledge to focus attention on small pieces of the space. Our approach to data exploration relies on both of these techniques. In order to explain further, we need to take a short detour for some background information about AI planning.

Planning is a form of search, in which the task is to construct a sequence of steps that lead from a set of initial conditions to some desired conclusion. Planners formulate the search problem in terms of states, goals, and sequences of actions to achieve goals. States are partial descriptions of the world, in the internal representation of the system. In a data exploration system, the state includes the data under consideration, models of the behavior of the data, observations of unusual characteristics, and so forth. Goals specify desired states. A goal might be, for example, to find a set of predictors for some specific variable. Actions take the system from one state to another. Examples of EDA actions include generating a linear fit for a relationship, power transforming a variable, and other such operations. Actions are defined by the preconditions or requirements for their application and the effects of their application.

Planners rely on task decomposition to solve complex problems, using knowledge about states, actions, and goals to structure the search at different levels of abstraction [7]. The simplest planners work by backward-chaining. Given a goal state, a planner begins by examining those actions that achieve the goal state. By treating the preconditions of these actions as goals to be satisfied in turn, and taking note of potential interactions between actions, the planner recursively generates a sequence of appropriate actions.

Traditional AI planners construct plans from primitive actions, stating from scratch for each new problem. Partial hierarchical planning takes a different approach. If a planning system repeatedly encounters similar problems, or subproblems, it becomes wasteful to generate similar solutions from scratch each time. Instead, a partial hierarchical planner can rely on a library of partial plans that apply to common cases. Rather than choosing an appropriate action at some point during the planning process, the planner can choose a partial plan in which many or all of the actions have already been decided on. If the library has sufficient coverage, planning largely reduces to a matter of choosing which plan to apply at what time.

Problem reformulation and expert knowledge apply in the AIDE planning framework as follows. Problem reformulation corresponds to capitalizing on knowledge of abstraction. It's clear that statisticians do not think of EDA as sequences of independent operations selected from menus of statistics packages. The fact that menus have names (e.g., Transform Data) suggests that elementary operations can be grouped into equivalence classes. Data analysis plans can be formulated in terms of these classes of operations, not the individual operations. Reformulating EDA in terms of abstract operations (such as transform, decompose, fit, etc.) reduces the branching factor of the search because there are fewer abstract actions to consider at each juncture. A combinatorially intractable search is replaced by two simpler searches: one to generate an abstract plan, the other for operations to instantiate the plan. For example, a useful abstract plan involves partitioning one or more variables, performing an operation on each partition, then presenting the results in an organized way. Histograms do this for one variable, contingency tables for two or more variables; tables of means in analysis of variance do it, too.

AIDE takes advantage of expert knowledge by storing abstract plans, instead of trying to generate them de novo. We have developed a simple but powerful language in which users can express plans for EDA. Plans are triggered by indications (which AIDE discovers automatically). For example, if a variable has "gaps" in its range, abstract plans are triggered to to search for another variable to explain the gaps and produce a contingency table, to

examine the separate partitions more closely, and so on. The number of options at any juncture is still large because AIDE generally finds many indications and can apply many plans with different parameterizations, but it is better structured and thus more manageable.

3 AN PLANNING SYSTEM FOR ANALYSIS OF SMALL DATASETS

To make the discussion above concrete, we present AIDE's planning representation. While the data handling aspects of the language are novel and powerful, we make no claim that they will scale to massive datasets. The plan language, on the other hand, implements strategic aspects of the exploration, which we believe are comparable for datasets of all sizes.

3.1 Representation

AIDE's processing is built around a planning representation. In a functional language, procedures call one another directly. A statistical-summary procedure, for example, might call a mean procedure during its processing. In a planning language, control moves between plans indirectly, by the establishment and satisfaction of goals. Rather than specifying that a mean should be computed, a plan might establish a goal (compute-location ?x). This goal could be satisfied by a mean, or median, or trimmed-mean procedure, depending on the characteristics of ?x. The advantage gained is flexibility. Instead of relying on functions that must be heavily parameterized to produce appropriate behavior, the planning language distinguishes between two separate control issues: *what* should be done, and *how* it should be done.

An example of a plan specification is given below. A plan has a name, a goal that the plan can potentially satisfy, constraints on its bindings, and a body. The body of a plan is a control schema of subgoal specifications, subgoals which must be satisfied for the plan to complete successfully. An action specification is similar to a plan specification, except that its body contains arbitrary code, rather than a control schema.

```
(define-plan histogram
:goal (generate-description :histogram-type ?batch ?histogram)
:body (:SEQUENCE
(:SUBGOAL (decompose (function unique-values) ?batch ?bins))
(:MAP/TRANSFORM (?bin ?bins ?histogram)
(:SUBGOAL (reduce (function count) ?bin)))))
:goal (generate-description :histogram-type ?batch ? histogram)
:body (:SEQUENCE
(:SUBGOAL (decompose(function unique-values) ?batch ?bins))
(:MAP/TRANSFORM (?bin ?bins ?histogram)
(:SUBGOAL (reduce (function count) ?bin)))))
```

This plan implements a simple procedure for generating a histogram of a discrete-valued variable. In words: divide the range of the variable into its unique values; break the variable into subsets, one subset per value; count the number of elements in each subset. The resulting counts are the bar heights of the histogram. In other words, we decompose the variable, which generates a new relation with a single attribute that contains the subsets. We then apply a transformation, with an embedded reduction, which maps each subset relation to a single value, the "count" statistic of the subset.

Let's now consider a contingency table generated for a relationship between categorical variables <*x,y*>. The procedure is as follows. We divide the relationship into subsets, one

corresponding to each unique combination of *x* and *y* values. Each subset is structurally identical to the original relationship. We now record the number of observations in each subset. The resulting values are the cell counts for the contingency table. A contingency table can thus be viewed as a two-dimensional analog of a histogram. Not surprisingly, the two procedures are identical in form:

```
(define-plan contingency-table
:goal (generate-description :contingency-table ?xy-relationship ?table)
:body (:SEQUENCE
(:SUBGOAL (decompose (function unique-values) ?xy-relation ?cells))
(:MAP/TRANSFORM (?cell ?cells ?table)
(:SUBGOAL (reduce (function count) ?cell)))))
```

In fact, we might use a single plan for both procedures, relying on subordinate plans to specialize appropriately on the type of the input data structure. These combinations are simple, hardly different from the way they would appear in a functional representation. The planning representation offers more flexibility than shown here. In addition to sequencing and functional composition of primitives, the body of a plan can specify that subgoals should be satisfied or actions executed in parallel, iteratively, conditionally, or in other ways. Further, we can control how we evaluate whether a plan has completed successfully or not; we might wish to iterate over all possible ways to satisfy a subgoal, or simply accept the first plan that succeeds.

A histogram/contingency table plan is a simple example of how procedures may generalize. A more complex example is shown below. This plan was initially designed to iteratively fit a resistant line to a relationship [2]. The resistant line generated by some methods is initially only an approximation, in the sense that executing the same procedure on the residuals may give a line with non-zero slope. When this happens, the fit is reapplied to the residuals and the line parameters updated appropriately. When the magnitude of the incremental changes falls below some heuristic threshold, the iteration stops. This plan captures the processing involved in iterating over the residuals. (The subgoals have been elided for the sake of presentation.)

```
(define-plan iterative-fit-plan
:goal (describe :iterative-fit ?operation ?relationship ?description)
:body (:SEQUENCE
(:SUBGOAL generate-fit)
(:ACTION generate-iteration-record)
(:WHILE (not (null ?continue-iteration-p))
(:SEQUENCE
(:SUBGOAL extract-residual-relationship-subgoal)
(:SUBGOAL residual-fit-subgoal)
(:ACTION update-iteration-record)))
(:SUBGOAL evaluate-fit-subgoal)))
```

This kind of heuristic improvement is also part of other procedures, lowess being the most familiar. The same plan can be used for both procedures. The fit subgoal is satisfied by different subordinate plans, their selection depending on context. This example is again

typical of the way complex procedures may be generated dynamically through plan combination. Most plans do not specify behavior down to the level of primitive operations, but let the selection of appropriate subordinate plans depend on the context of intermediate results generated.

Finally, we have an example of a simple incremental modeling plan, instantiated in the exploration of a dataset. It generates an initial model of the dataset of an appropriate type. It then establishes the goal of elaborating the model. With the plans in the AIDE library, elaboration will involve adding relationships, one at a time, to the model. One of the plans that matches the elaborate-model subgoal recursively establishes an identical subgoal, with ?model bound to the incrementally extended model.

```
(define-plan explore-by-incremental-modeling ()
:goal (explore-by :modeling ?model-type ?description ?structure ?model)
:constraint ((?structure ((:dataset-type dataset))))
:body (:SEQUENCE
(:WHEN (null ?model)
(:SUBGOAL initialize-subgoal
(generate-initial-model ?description ?structure
?model-type ?model)))
(:SUBGOAL elaborate-model
(elaborate-model ?model-type ?activity
?description ?structure ?model))))
```

AIDE's plan library currently contains over 100 plans. Fewer than a dozen plans are usually applicable at any point during exploration, and each plan constrains the set of subordinate plans applicable at later points. Many of the plans are simple, on the order of univariate transformations and residual generation. Others are more complex descriptive and model-building procedures. Some, like the last plan presented, implement control strategies applicable in a variety of different situations. With the plan representation we have found little difficulty in implementing most familiar exploratory procedures. In addition to descriptive plans for resistant lines, various box plot procedures, and smoothing procedures, we have implemented forward selection algorithms for cluster analysis and regression analysis [9], a causal modeling algorithm [8], and a set of weaker opportunistic modeling procedures.

3.2 Controlling Plan Execution

Unless we can ensure that only a single procedure is applicable at any time, and that we know the exact subset of the data we should concentrate on—an improbable scenario—we must address a set of questions concerning control:

- Which data structure (variable, relationship, model, etc.) should be considered?
- Which plan (or plan type, in the case of higher level decisions) is appropriate?
- Given a set of comparable but different results, produced by similar procedures (e.g., a least-squares or a resistant line; four clusters or five), which result is best?

In AIDE, rules are used to make these decisions. The relevance of a rule is determined by structure and context constraints. By structure constraints, we mean that the feature and indication values specified in a rule must be present in the structure under consideration. Rules can thus constrain plans to act, for example, only on variables with indications of clustering, or on relationships between continuous variables. Context constraints apply to the history of plan and subgoal activations that have led up to the current decision point. Using context constraints a rule can, for example, prevent clusters detected in the residuals of a linear fit from being explored if the clustering plan has not yet been applied at the level of the original relationship. These constraints can prune and order related decisions. These rules control the ordering and selection of data to explore, the plans to explore them, and the results they produce.

AIDE can thus run without assistance from the user. Its rules let it evaluate candidate data structures to explore; its library provides the plans that perform the exploration. Not surprisingly, however, AIDE's lack of contextual knowledge will take the exploration in directions an informed human analyst would not need or want to go. If a user is interested in gender issues, for example, partitions of data by sex will form a significant part of the exploration. AIDE cannot know that one user is interested in gender, another studies time series of climate change, and so on. For this reason, AIDE is an analyst's assistant, not an autonomous package. It hunts for indications and suggests plans to explore them, but the analyst guides the whole process.

The process is mixed-initiative in the following sense. AIDE explores a dataset by elaborating and executing a hierarchy of plans. Decision points in the hierarchy are made available for the user's inspection. The user guides exploration by changing the ordering of plans, by selecting new plans, or by shifting to different areas in the data. This guidance includes selecting appropriate dataset variables and subsets to be explored as well as applying any available statistical manipulations to the selected data. The arrangement gives the user most of the flexibility of the usual statistical interface, while still remaining within the planning framework.

Each decision point gives the user the opportunity to view the choices—and the data— currently under consideration. The user can step through the process, letting the planner execute only until another decision is reached. The user can let planner execution continue until some descriptive result has been generated, at which point another decision can be made using the new information. At any time the user can override AIDE's decisions about the operation to execute or data to examine. AIDE keeps the user constantly informed with descriptions of the data being explored, justifications for candidate operations at a decision point, ordering constraints on choices, and descriptions of the plans under consideration.

4 DISCUSSION

AIDE represents a compromise between two eventually unworkable approaches to data analysis. On one hand we have statistics packages, which are driven by users, and treat actions as independent. The search space for EDA is intractable when elementary actions are independent selections from statistics package menus. On the other hand, we have autonomous "black boxes" from machine learning and statistics, algorithms like stepwise multiple regres

sion, CART, and C4. These require nothing but data from the user, but nor does the user have much opportunity to control how they work, and the results are never explained. Your choice is to relinquish all control, or control every detail, of analyses. AIDE offers an alternative, involving the user in the strategic aspect of data analysis, but reducing the tactical burden.

To manage the process AIDE relies on the techniques of AI planning. We have encountered some strong similarities between planning problems and the types of problems found in EDA. These similarities yield some some potentially useful implications for the processing of massive datasets, implications that we have only begun to study.

Peter Huber observes that data analysis differs from other computer-supported tasks, such as programming and text preparation. In the latter tasks the final version is everything; there is no need to know the path by which it was reached. In data analysis the correctness of the end product cannot be checked without inspecting the path leading to it [5]. Huber argues that existing paradigms are thus inappropriate for data analysis tasks. If a massive dataset is to be analyzed without the direct supervision of a human user, then a representation of the process carried out is a necessary component of the result. This is one of the key distinctions between planning and other forms of search—the aim is to generate a sequence (or more complex combination) of operations, not simply the end result.

One of the best-known observations to arise in the planning literature is that problem decomposition is most effective when the subproblems that are generated are largely independent of one another. A recent AI textbook contains an example in which a search problem, if solved directly, requires a search of 10^{30} states. In contrast, a hierarchical decomposition brings the search space down to 600 states [7]. By imposing structure on the search space we can sometimes transform an intractable problem into a relatively simple one. Using an analogy developed during the Massive Datasets Workshop, we observe that the effectiveness of exploration of a massive dataset will depend on our knowledge of its geography: we need to know either where to look for patterns or what kinds of patterns to look for. In the absence of this knowledge we have no way to carve up the search space into smaller, more manageable blocks. A hierarchical approach to exploring a massive dataset can be effective to the extent that the decomposition follows the geography of the dataset.

A planner can derive great benefit from compiled knowledge in the form of a plan library. Much of the knowledge we bring to bear in solving problems takes the form of procedures or sequences of actions for achieving particular goals [4]. Planning problems in some complex environments are too difficult to solve without the user of sophisticated plan libraries [1]. Similarly, the exploration of massive datasets poses a problem that may only be solved with both knowledge of individual data analysis operations and knowledge about how they can be combined.

Our long-term objectives for this research include fully automated model-building and discovery mechanisms driven by an opportunistic control strategy. We expect to develop the automated strategies from our experience with the system as a manual analysis decision aid, letting human analysts provide much of the initial reasoning control strategy. Although we are developing this system for our own use in modeling complex program behavior, we believe it to be potentially useful for any scientific modeling problem, particularly those in which the process of data collection is itself automated and the resulting volume of data overwhelming for human analysts (e.g., astronomical surveys or remote sensing of terrestrial

features).

In sum, AI researchers are responding to the challenges and opportunities afforded by massive amounts of electronic data. Capitalizing on the opportunities involves statistical reasoning, although not all AI researchers recognize it as such. Statistics provides a foundation for analysis and design of AI programs. Reciprocally, some AI programs facilitate the work of applied statisticians, not only in EDA, but also in model induction and testing.

ACKNOWLEDGMENTS

This research is supported by ARPA/Rome Laboratory under contract #F30602-93-0100, and by the Dept. of the Army, Army Research Office under contract #DAAH04-95-1-0466. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes not withstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of the Advanced Research Projects Agency, Rome Laboratory, or the U.S. Government.

References

[1] Paul R. Cohen, Michael L. Greenberg, David M. Hart, and Adele E. Howe. Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine*, 10(3):32-48, Fall 1989.

[2] John D. Emerson and Michal A. Stoto. Transforming data. In David C. Hoaglin, Frederick Mosteller, and John W. Tukey, editors, *Understanding robust and exploratory data analysis*. Wiley, 1983.

[3] Usama Fayyad, Nicholas Weir, and S. Djorgovski. Skicat: A machine learning system for automated cataloging of large scale sky surveys . In *Proceedings of the Tenth International Conference on Machine Learning* , pages 112-119. Morgan Kaufmann, 1993.

[4] Michael P. Georgeff and Amy L. Lansky. Procedural knowledge. *Proceedings of the IEEE Special Issue on Knowledge Representation* , 74(10):1383-1398, 1986.

[5] 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.

[6] Amy L. Lansky and Andrew G. Philpot. AI-based planning for data analysis tasks. *IEEE Expert*, Winter 1993.

[7] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modem Approach*. Prentice Hall, 1995.

[8] Robert St. Amant and Paul R. Cohen. Toward the integration of exploration and modeling in a planning framework. In *Proceedings of the AAAI-94 Workshop in Knowledge Discovery in Databases*, 1994.

[9] Robert St. Amant and Paul R. Cohen. A case study in planning for exploratory data analysis. In *Advances in Intelligent Data Analysis* , pages 1-5, 1995.

[10] 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*. Springer, 1995. To appear.

Massive Data Sets: Problems and Possibilities, with Application to Environmental Monitoring

Noel Cressie
Iowa State University
Anthony Olsen
U.S. Environmental Protection Agency
Dianne Cook
Iowa State University

ABSTRACT

Massive data sets are not unlike small to large data sets in at least one respect, namely it is essential to know their context before one starts an analysis. That context will almost certainly dictate the types of analyses attempted. However, the sheer size of a massive data set may challenge and, ultimately, defeat a statistical methodology that was designed for smaller data sets. This paper discusses the resulting problems and possibilities generally and, more specifically, considers applications to environmental monitoring data.

INTRODUCTION

Massive data sets are measured in gigabytes (10^9 bytes) and terabytes (10^{12} bytes). We can think about them, talk about them, access them, and analyze them because data storage capabilities have evolved over the last 10,000 years (especially so over the last 20 years) from human memory, to stone, to wood, bark, and paper, and to various technologies associated with digital computers. With so much data coming on line and improvements in query languages for data base management, data analysis capabilities are struggling to keep up.

The principal goal of data analysis is to turn data into information. It should be the statistician's domain but the technology will not wait for a community whose evolutionary time scales are of the order of 5-10 years. As a consequence, scientists working with massive data sets will commission analyses by people with good computer training but often minimal statistics training. This scenario is not new but it is exacerbated by massive-data riches (e.g., in environmental investigations, an area familiar to us).

We would argue that statisticians do a better job of data analysis because they are trained to understand the nature of variability and its various sources. Non statisticians often think of statistics as relevant only for dealing with measurement error, which may be the least important of the sources of variability.

TYPES OF MASSIVE DATA SETS

Although "massive data sets" is the theme of this workshop, it would be a mistake to think, necessarily, that we are all talking about the same thing. A typology of the origin of massive data sets is relevant to the understanding of their analyses. Amongst others, consider: observational records and surveys (health care, census, environment); process studies (manufacturing control); science experiments (particle physics). Another "factor" to consider might be the questions asked of the data and whether the questions posed were explicitly part of the reason the data were acquired.

STATISTICAL DATA ANALYSIS

As a preamble to the following remarks, we would like to state our belief that good data analysis, even in its most exploratory mode, is based on some more or less vague statistical model. It is curious, but we have observed that as data sets go from "small" to "medium," the statistical analysis and models used tend to become more complicated, but in going from "medium" to "large," the level of complication may even decrease! That would seem to suggest that as a data set becomes "massive," the statistical methodology might once again be very simple (e.g., look for a central tendency, a measure of variability, measures of pairwise association between a number of variables). There are two reasons for this. First, it is often the simpler tools (and the models that imply them) that continue to work. Second, there is less temptation with large and massive data sets to "chase noise." Think of a study of forest health, where there are 2×10^6 observations (say) in Vermont: a statistician could keep him(her)self and a few graduate students busy for quite some time, looking for complex structure in the data. Instead, suppose the study has a national perspective and that the 2×10^6 observations are part of a much larger data base of 5×10^8 (say) observations. One now wishes to make statements about forest health at both the national and regional level but for *all* regions. But the resources to carry out the bigger study are *not* 250 times more. The data analyst no longer has the luxury of looking for various nuances in the data and so declares them to be noise. Thus, what could be signal in a study involving Vermont only, becomes noise in a study involving Vermont and all other forested regions in the country.

The massiveness of the data can be overwhelming and may reduce the non statistician to asking over-simplified questions. But the statistician will almost certainly think of stratifying (subsetting), allowing for a between-strata component of variance. Within strata, the analysis may proceed along classical lines that looks for replication in errors. Or, using spatio-temporal analysis, one may invoke the principle that nearby (in space and time) data or objects tend to be more alike than those that are far apart, implying redundancies in the data.

Another important consideration is dimension reduction when the number of variables is large. Dimension reduction is more than linear projections to lower dimensions such as with principal components. Non-linear dimension reduction techniques are needed that can extract lower-dimensional structure present in a massive data set. These new dimension-reduction techniques in turn imply new methods of clustering. Where space and/or time co-ordinates are available, these data should be included with the original observations.

At the very least they could simply be concatenated together to make a slightly higher dimensional (massive) data set. However, the special nature of the spatial and temporal co-ordinates is apparent in problems where the goal is space-time clustering for dimension reduction.

An issue that arises naturally from the discussion above is how one might compare two "useful" models. We believe that statistical model evaluation is a very important topic for consideration and that models should be judged both on their predictive ability and on their simplicity.

One has to realize that interactive data analysis on all but small subsets of the data may be impossible. It would seem sensible then that "intelligent" programs, that seek structure and pattern in the data, might be let loose on the data base at times when processing units might otherwise be idle (e.g., Carr, 1991; Openshaw, 1992). The results might be displayed graphically and animation could help compress long time sequences of investigation. Graphics, with its ability to show several dimensions of a study on a single screen, should be incorporated whenever possible. For example, McDonald (1992) has used a combination of real-time and animation to do rotations on as many as a million data points.

It may not be necessary to access the whole data set to obtain a measure of central tendency (say). Statistical theory might be used to provide sampling schemes to estimate the desired value; finite population sampling theory is tailor-made for this task. Sampling (e.g., systematic, adaptive) is particularly appropriate when there are redundancies present of the type described above for spatio-temporal data.

It is not suggested that the unsampled data should be discarded but, rather, that it should be held for future analyses where further questions are asked and answered. In the case of long-term monitoring of the environment, think of an analogy to medicine where all sorts of data on a patient are recorded but often never used in an analysis. Some are, of course, but those that are not are always available for retrospective studies or for providing a baseline from which unusual future departures are measured. In environmental studies, the tendency has been to put a lot of resources into data collection (i.e., when in doubt, collect more data).

Sampling offers a way to analyze massive data sets with some statistical tools we currently have. Data that exhibit statistical dependence do

not need to be looked at in their entirety for many purposes because there is much redundancy.

APPLICATION TO THE ENVIRONMENTAL SCIENCES

Environmental studies, whether they are involved in long-term monitoring or short-term waste-site characterization and restoration, are beginning to face the problems of dealing with massive data sets. Most of the studies are observational rather than designed and so scientists are scarcely able to establish much more than association between independent variables (e.g., presence/absence of pollutant) and response (e.g., degradation of an ecosystem). National long-term monitoring, such as is carried out in the U.S. Environmental Protection Agency (EPA)'s Environmental Monitoring and Assessment Program (EMAP), attempts to deal with this by establishing baseline measures of mean and variance from which future departures

might be judged (e.g., Messer, Linthurst, and Overton, 1991). National or large regional programs will typically deal with massive data sets. For example, sample information (e.g., obtained from field studies) from a limited number of sites (1,000 to 80,000) will be linked with concomitant information in order to improve national or regional estimation of the state of forests (say). Thematic mapper (10^{10} observations), AVHRR (8×10^6 observations), digital elevation (8×10^6 observations), soils, and so forth, coverage information is relatively easy and cheap to obtain. Where do we stop? Can statistical design play a role here to limit the "factors"? Moreover, once the variables are chosen, does one lose anything by aggregating the variables spatially (and temporally)? The scientist always asks for the highest resolution possible (so increasing the massiveness of the data set) because of the well known ecological fallacy that shows a relationship between two variables at an aggregated level may be due simply to the aggregation rather than to any real link. (Simpson's paradox is a similar phenomenon that is more familiar to the statistics community.) One vexing problem here is that the various spatial coverages referred to above are rarely acquired with the same resolution. Statistical analyses must accommodate our inability to match spatially all cases in the coverages.

Environmental studies often need more specialized statistical analyses that incorporate the spatio-temporal component. These four extra dimensions suggest powerful and obvious ways to subset (massive) environmental data sets. Also, biological processes that exhibit spatio-temporal smoothness can be described and predicted with parsimonious statistical models, even though we may not understand the etiologies of the phenomena. (The atmospheric sciences have taken this "prediction" approach even further, now giving "data" at every latitude-longitude node throughout the globe. Clearly these predicted data have vastly different statistical properties than the original monitoring data.)

Common georeferencing of data bases makes all this possible. Recent advances in computing technology have led to Geographic Information Systems (GISs), a collection of hardware and software tools that facilitate, through georeferencing, the integration of spatial, non-spatial, qualitative, and quantitative data into a data base that can be managed under one system environment. Much of the research in GIS has been in computer science and associated mathematical areas; only recently have GISs begun to incorporate model-based spatial statistical analysis into their information-processing subsystems. An important addition is to incorporate exploratory data analysis tools including graphics into GISs, if necessary by linking a GIS with existing statistical software (Majure, Cook, Cressie, Kaiser, Lahiri, and Symanzik, 1995).

CONCLUSIONS

Statisticians have something to contribute to the analysis of massive data sets. Their involvement is overdue. We expect that new statistical tools will arise as a consequence of their involvement but, equally, we believe in the importance of adapting existing tools (e.g., hierarchical models, components of variance, clustering, sampling). Environmental and spatio-temporal data sets can be massive and represent important areas of application.

ACKNOWLEDGMENT

This research was supported by the Environmental Protection Agency under Assistance Agreement No. CR822919-01-0.

References

Carr, D. B. (1991). Looking at large data sets using binned data plots, in A. Buja and P. A. Tukey, eds. *Computing and Graphics in Statistics*, Springer Verlag, New York, 7-39.

McDonald, J. A. (1992). Personal demonstration of software produced at University of Washington, Seattle, WA.

Majure, J. J., Cook, D., Cressie, N., Kaiser, M., Lahiri, S., and Symanzik, J. (1995). Spatial CDF estimation and visualization with applications to forest health monitoring. *Computing Science and Statistics* , forthcoming.

Messer, J. J., Linthurst, R. A., and Overton, W. S. (1991). An EPA program for monitoring ecological status and trends . *Environmental Monitoring and Assessment*, **17**, 67-78.

Openshaw, S. (1992). Some suggestions concerning the development of AI tools for spatial analysis and modeling in GIS. *Annals of Regional Science*, **26**, 35-51.

Visualizing Large Data Sets

Stephen G. Eick*
Bell Laboratories (A Division of Lucent Technologies)

ABSTRACT

Visualization is a key technology for understanding large datasets. It is useful throughout the analysis process, for exploratory descriptive analysis, to aid in model building; and for presenting the analysis results. Our approach to visualizing abstract, non-geometric data involves domain-specific representations, multiple linked views, color, and a highly-interactive user interface using filtering and focusing to reduce visual clutter. We have developed a software infrastructure embodying our design principles for producing novel, high-quality visualizations of corporate datasets.

1 INTRODUCTION

Just as spreadsheets revolutionized our ability to understand small amounts of data, visualization will revolutionize the way we understand large datasets. Our research focuses on extracting the information latent in large databases using visual techniques. The difficulty in extracting this information lies in understanding the complexity of the databases. To aid in this task, we have created many novel, highly interactive visualizations of large datasets. This involved developing the techniques, software tools, and infrastructure to mine knowledge from corporate databases so that it can be put to competitive and commercial advantage.

* AT&T Bell Laboratories-Rm 1G-351, 1000 East Warrenville Road, Naperville, IL 60566, email: eick@research.att.com

2 DOMAIN-SPECIFIC REPRESENTATION

A key component of an effective visualization involves the visual representation of the data. The representation determines how the items in the dataset are rendered on the computer display. The best representations are often domain-specific: scatterplots for statistical data, maps for spatial data, and node and link diagrams for network data, for example. Inventing a representation for a new domain is a difficult, creative, and iterative process.[1] The representation should take full advantage of perceptual cues such as size. positions. color, depth, and may even use motion and sound.

3 HIGH INFORMATION DENSITY

Our representations are often compact, color-coded glyphs positioned spatially. By using compact glyphs that overplot gracefully we can pack a lot of information into an image and thereby display a large dataset. A high-resolution 1280×1024 workstation monitor has over 1,300,000 pixels. Our goal is to use every pixel to display data, thereby maximizing the information content. in the image.

In some cases is is possible to display an entire dataset on a single screen, thereby eliminating the difficult navigation problems associated with panning and zooming interfaces that focus on small portions of the database.

4 INTERACTIVE FILTERS

Often information-dense displays become overly cluttered with too much detail. One approach to solving the display clutter problem involves interactive filters that reduce the amount of information shown on the display. Humans have sophisticated pattern recognition capabilities, perhaps due to our evolution, and are very efficient at manipulating interactive controls to reduce visual clutter. We exploit this to effortlessly solve the complex computational problems involved with determining when a display is too busy for an easy interpretation. Our approach is to leverage people's natural abilities by designing user interface controls that parameterize the display complexity.

[1] See the Figures for examples from some domains that we have considered.

5 MULTIPLE LINKED VIEWS

The power of our representations is magnified through the use of interaction and linked views. Each view, whether custom or standard (color keys, bar charts, box plots, histograms, scatter plots, etc.), functions both as a display and a control panel. Selecting and filtering data in one view instantly propagates to the other views, thereby providing additional insights. Linking multiple views interactively provides an integrated visualization far more powerful than the sum of the individual views.

6 SYSTEMS

Our systems have been used to successfully analyze and present software version control information. file system sizes. budgets, network traffic patterns, consumer shopping patterns, relational database integrity constraints, resource usage on a compute server, etc. The amount of information that our systems present on a single screen is between 10,000 and 1,000,000 records. Some of the more interesting systems we have built include:

1. SeeSoft™-lines of text in files [Eic94] (Figure 1)
2. SeeSlice-program slices and code coverage [BE94] (Figure 2)
3. SeeLog-time-stamped log reports [EL95] (Figure 3)
4. SeeData-relational data [AEP95] (Figure 4)
5. SeeNet-geographic networks data [BEW95] (Figures 5 and 6)
6. NicheWorks™-abstract networks [EW93] (Figure 7)
7. SeeDiff™-file system differences
8. SeeLib-bibliographic databases [EJW94] (Figure 9)
9. SeeSys-hierarchical software modules [BE95] (Figure 10)
10. SeeSales™-retail sales inventory and forecasts (Figure 11)
11. SeeTree-hierarchical data

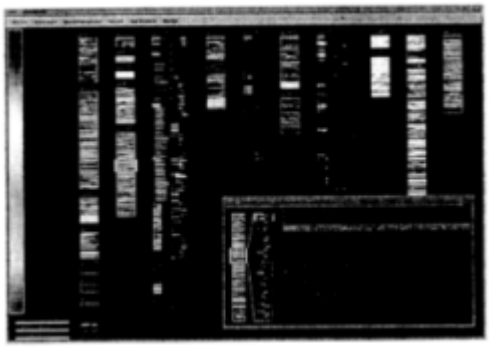

Figure 1. Lines of Code Colored by Age

Figure 2. Forward Program Slice

Figure 3. Log File View

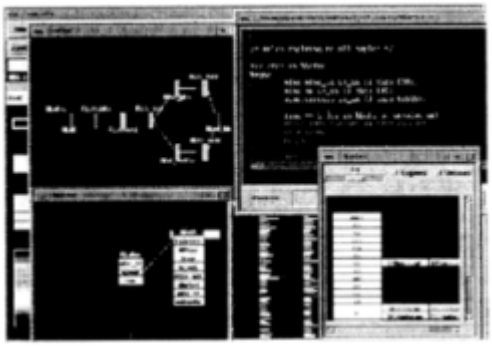

Figure 4. Relational Database View

Figure 5. Christmas Morning Long-Distance Traffic

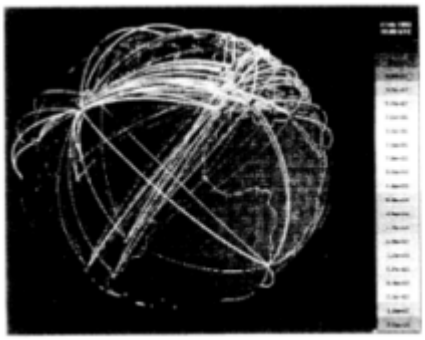

Figure 6. World Wide Internet Traffic

Figure 7. Market Basket Analysis

Figure 8. Demographic Information

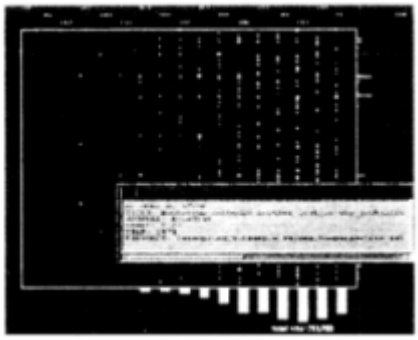

Figure 9. Document Retrieval

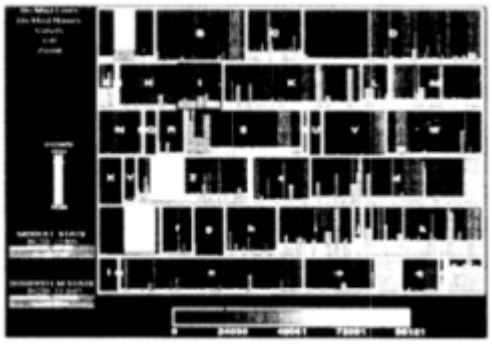

Figure 10. Hierarchical system view

Figure 11. Sales by Week and Event

Figure 12. Organization Productivity by Week

12. SeeFraud-network calling fraud.

Since the needs of each user are unique, the best visualizations are task-oriented. The most successful visualizations help frame interesting questions as well as answer them. Our visualizations:

- Make use of existing data. In many cases large databases of vital importance to an organization already exist. Our visualizations extract meaningful information from this data.
- Are directed toward real problems with targeted users. Our efforts are motivated by business needs and address real problems.
- Focus on understanding and insight. Results are more important than any particular technique.
- Are used throughout the analysis process including the initial data exploration, intermediate model formulation, and final result presentation.

7 SOFTWARE AND TECHNOLOGY

Underlying all of our visualizations is a common infrastructure embodied in a C++ library that handles interaction, graphics, and view linking. This C++ Visualization Library helps us to:

- Minimize our development time,
- Encapsulate expertise and design principles,
- Build cross-platform systems (UNIX/X11, Open GL, and PC/Windows), and
- Keep visualization application code small.

8 CONCLUSION

Visualization is a key technology that can help users understand the complexity in industrial-sized systems. We have exploited this technology to investigate a variety of large and complex data sets. Interactive data visualization is complementary to other analytic, model-based approaches and will become a widely used tool for extracting the information contained in large complex datasets.

ACKNOWLEDGMENTS

The research presented here represents the joint efforts of Jackie Antis. Dave Atkins, Tom Ball, Brian Johnson, Ken Cox, Nate Dean, Paul Lucas, John Pyrce, and Graham Wills.

References

[AEP95] Jacqueline M. Antis, Stephen G. Eick, and John D. Pyrce. Visualizing the structure of relational databases. *IEEE Software*, Accepted for publication 1995.

[BE94] Thomas Ball and Stephen G. Eick. Visualizing program slices. In *1994 IEEE Symposium on Visual Languages*, pages 288-295, St. Louis, Missouri, 4 October 1994.

[BE95] Marla J. Baker and Stephen G. Eick. Space-filling software displays. *Journal of Visual Languages and Computing*, 6(2), June 1995.

[BEW95] Richard A. Becker, Stephen G. Eick, and Allan R. Wilks. Visualizing network data. *IEEE Transactions on Visualization and Graphics*, 1(1):16-28, March 1995.

[Eic94] Stephen G. Eick. Graphically displaying text. *Journal of Computational and Graphical Statistics*, 3(2):127-142. June 1994.

[EJW94] Stephen G. Eick, Eric E. Sumner Jr., and Graham J. Wills. Visualizing bibliographic databases. In John P. Lee and Georges G. Grinstein, editors. *Database Issues for Data Visualization*, pages 186-193. Springer-Verlag, October 1994. Lecture Notes in Computer Science.

[EL95] Stephen G. Eick and Paul J. Lucas. Displaying trace files. *Software Practice and Experienced*, Accepted for publication 1995.

[EW93] Stephen G. Eick and Graham J. Wills. Navigating large networks with hierarchies. In *Visualization '93 Conference Proceedings*, pages 204-210. San Jose, California, 25-29 October 1993.

From Massive Data Sets to Science Catalogs: Applications and Challenges

Usama Fayyad
Microsoft Research
Padhraic Smyth
University of California, Irvine

ABSTRACT

With hardware advances in scientific instruments and data gathering techniques comes the inevitable flood of data that can render traditional approaches to science data analysis severely inadequate. The traditional approach of manual and exhaustive analysis of a data set is no longer feasible for many tasks ranging from remote sensing, astronomy, and atmospherics to medicine, molecular biology, and biochemistry. In this paper we present our views as practitioners engaged in building computational systems to help scientists deal with large data sets. We focus on what we view as challenges and shortcomings of the current state-of-the-art in data analysis in view of the massive data sets that are still awaiting analysis. The presentation is grounded in applications in astronomy, planetary sciences, solar physics, and atmospherics that are currently driving much of our work at JPL.

keywords: science data analysis, limitations of current methods, challenges for massive data sets, classification learning, clustering.

Note: Both authors are affiliated with Machine Learning Systems Group, Jet Propulsion Laboratory, M/S 525-3660, California Institute of Technology, Pasadena, CA 91109, http://www-aig.jpl.nasa.gov/mls/.

1 INTRODUCTION

The traditional approach of a scientist manually examining a data set, and exhaustively cataloging and characterizing all objects of interest, is often no longer feasible for many tasks in fields such as geology, astronomy, ecology, atmospheric and ocean sciences, medicine, molecular biology, and biochemistry.

The problem of dealing with huge volumes of data accumulated from a variety of sources is now largely recognized across many scientific disciplines. Database sizes are already being measured in terabytes (10^{12} bytes), and this size problem will only become more acute with the advent of new sensors and instruments [9, 34]. There exists a critical need for information processing technologies and methodologies to manage the data avalanche. The future of scientific information processing hinges upon the development of algorithms and software that enable scientists to interact effectively with large scientific data sets.

1.1 Background

This paper reviews several ongoing automated science cataloging projects at the Jet Propulsion Laboratory (sponsored by NASA) and discusses some general implications for analysis of massive data sets in this context. Since much of NASA's data is remotely-sensed image data the cataloging projects have focused mainly on spatial databases, which are essentially large collections of spatially-gridded sensor measurements of the sky, planetary surfaces and Earth where the sensors are operating within some particular frequency band (optical, infra-red, microwave, etc). It is important to keep in mind that the scientific investigator is primarily interested in using the image data to investigate hypotheses about the physical properties of the target being imaged and he or she is not directly interested in the image data per se. Hence, the image data merely serve as an intermediate representation that facilitates the scientific process of inferring a conclusion from the available evidence.

In particular, scientists often wish to work with derived image products, such as catalogs of objects of interest. For example, in planetary geology, the scientific process involves examination of images (and other data) from planetary bodies such as Venus and Mars, the conversion of these images into catalogs of geologic objects of interest (such as craters, volcanoes, etc.), and the use of these catalogs to support, refute, or originate theories about the geologic evolution and current state of the planet. Typically these catalogs contain information about the location, size, shape, and general context of the object of interest and are published and made generally available to the planetary science community [21]. There is currently a significant shift to computer-aided visualization of planetary data, a shift which is driven by the public availability of many planetary data sets in digital form on CD-ROMS [25].

In the past, both in planetary science and astronomy, images were painstakingly analyzed by hand and much investigative work was carried out using hardcopy photographs or photographic plates. However, the image data sets that are currently being acquired are so large that simple manual cataloging is no longer practical, especially if any significant fraction of the available data is to be utilized. This paper briefly discusses NASA-related projects where automated cataloging is essential, including the Second Palomar Observatory Sky Survey (POSS-II) and the Magellan-SAR (Synthetic Aperture Radar) imagery of Venus returned by the Magellan spacecraft. Both of these image databases are too large for manual visual analysis and provide excellent examples of the need for automated analysis tools.

The POSS-II application demonstrates the benefits of using a trainable classification approach in a context where the transformation from pixel space to feature space is well-understood. Scientists

often find it easier to define features of objects of interest than to produce recognition models for these objects. POSS-II illustrates the effective use of prior knowledge for feature definition: for this application, the primary technical challenges were in developing a classification model in the resulting (relatively) high-dimensional feature space.

In the Magellan-SAR data set the basic image processing is not well-understood and the domain experts are unable to provide much information beyond labeling objects in noisy images. In this case, the significant challenges in developing a cataloging system lie in the feature extraction stage: moving from a pixel representation to a relatively invariant feature representation.

1.2 Developing Science Catalogs from Data

In a typical science data cataloging problem, there are several important steps:

1. Decide what phenomena are to be studied, or what hypotheses are to be evaluated.
2. Collect the observations. If the data are already in existence then decide which subsets of the data are of interest and what transformations or preprocessing are necessary.
3. Find all events of interest in the data and create a catalog of these with relevant measurements of properties.
4. Use the catalog to evaluate current hypotheses or formulate new hypotheses of underlying phenomena.

It is typically the case that most of the work, especially in the context of *massive data sets* is in step 3, the *cataloging* task. It is this task that is most tedious and typically prohibitive since it requires whoever is doing the searching, be it a person or a machine, to sift throughout the entire data set. Note that the most significant "creative" analysis work is typically carried out in the other steps (particularly 1 and 4).

In a typical cataloging operation, the recognition task can in principle be carried out by a human, i.e., a trained scientist can recognize the target when they come across it in the data (modulo fatigue, boredom, and other human factors). However, if the scientist were asked to write a procedure, or computer program, to perform the recognition, this would typically be very difficult to do. Translating human recognition and decision-making procedures into algorithmic constraints that operate on raw data is in many cases impractical. One possible solution is the pattern recognition or "training-by-example" approach: a user trains the system by identifying objects of interest and the system automatically builds a recognition model rather than having the user directly specifying the model. In a sense, this training-by-example approach is a type of exploratory data analysis (EDA) where the scientist knows what to look for, but does not know how to specify the search procedure in an algorithmic manner. The key issue is often the effective and appropriate use of prior knowledge. For pixel-level recognition tasks, prior information about spatial constraints, invariance information, sensor noise models, and so forth can be invaluable.

2 SCIENCE CATALOGING APPLICATIONS AT JPL

2.1 The SKICAT Project

The Sky Image Cataloging and Analysis Tool (SKICAT, pronounced "sky-cat") has been developed for use on the images resulting from the POSS-II conducted by Caltech. The photographic plates

are digitized via high-resolution scanners resulting in about 3,000 digital images of 23,040 × 23, 040 pixels each, 16 bits/pixel, totaling over three terabytes of data. When complete, the survey will cover the entire northern sky in three colors, detecting virtually every sky object down to a *B* magnitude of 22 (a normalized measure of object brightness). This is at least one magnitude fainter than previous comparable photographic surveys. It is estimated that at least 5×10^7 galaxies and 2×10^9 stellar objects (including over 10^5 quasars) will be detected. This data set will be the most comprehensive large-scale imaging survey produced to date and will not be surpassed in scope until the completion of a fully digital all-sky survey.

The purpose of SKICAT is to facilitate the extraction of meaningful information from such a large data set in an efficient and timely manner. The first step in analyzing the results of a sky survey is to identify, measure, and catalog the detected objects in the image into their respective classes. Once the objects have been classified, further scientific analysis can proceed. For example, the resulting catalog may be used to test models of the formation of large-scale structure in the universe, probe Galactic structure from star counts, perform automatic identifications of radio or infrared sources, and so forth [32, 33, 8]. Reducing the images to catalog entries is an overwhelming manual task. SKICAT automates this process, providing a consistent and uniform methodology for reducing the data sets.

2.1.1 Classifying Sky Objects

Each of the 3,000 digital images is subdivided into a set of partially overlapping frames. Low-level image processing and object separation is performed by a modified version of the FOCAS image processing software [20]. Features are then measured based on this segmentation. The total number of features measured for each object by SKICAT is 40, including magnitudes, areas, sky brightness, peak values, and intensity weighted and unweighted pixel moments. Some of these features are generic in the sense that they are typically used in analyses of astronomical image data [31]: other features such as normalized and non-linear combinations are derived from the generic set.

Once all the features are measured for each object, final classification is performed on the catalog. The goal is to classify objects into four categories, following the original scheme in FOCAS: *star, star with fuzz, galaxy*, and *artifact* (an artifact represents anything that is not a sky object, e.g. satellite or airplane trace, film aberrations, and so forth).

2.1.2 Classifying Faint Sky Objects

In addition to the scanned photographic plates, we have access to CCD images that span several small regions in some of the plates. The main advantage of a CCD image is higher spatial resolution and higher signal-to-noise ratio. Hence, many of the objects that are too faint to be classified by inspection on a photographic plate are easily classifiable in a CCD image. In addition to using these images for photometric calibration of the photographic plates, the CCD images are used for two purposes during training of the classification algorithm: (i) they enable manual identification of class labels for training on faint objects in the original (lower resolution) photographic plates, and (ii) they provide a basis for accurate assessment of human and algorithmic classification performance on the lower resolution plates. Hence, if one can successfully' build a model that can classify faint objects based on training data from the plates that overlap with the limited high-resolution CCD images, then that model could in principle classify objects too faint for visual classification by astronomers or traditional computational methods used in astronomy. Faint objects constitute the majority of objects in any image.

The classification learning algorithms used are decision tree based, as in [3, 24]. The particular algorithms used in SKICAT are covered in [12, 14, 11]. The basic idea is to use greedy tree growing algorithms to find a classifier in the high dimensional feature space. A system called RULER [12] is then used to optimize rules from a multitude of decision trees trained using random sampling and cross validation. RULER applies pruning techniques to rules rather than trees as in CART [3, 24]. A rule is a single path from a decision tree's root to one leaf.

2.1.3 SKICAT Classification Results

Stable test classification error rates of about 94% were obtained using RULER, compared to the original trees which had an accuracy of about 90%. Note that such high classification accuracy results could only be obtained after expending significant effort on defining more robust features that captured sufficient invariances between various plates. When the same experiments were conducted using only the generic features measured by the standard schemes, the results were significantly worse. The SKICAT classifier correctly classified the majority of faint objects (using only the original lower resolution plates) which even the astronomers cannot classify without looking at the special CCD plates: these objects are at least one magnitude fainter than objects cataloged in previous surveys. This results in a 200% increase in the number of classified sky objects available for scientific analysis in the resulting sky survey catalog database.

A consequence of the SKICAT work is a fundamental change in the notion of a sky catalog from the classical static entity "in print," to a dynamic on-line database. The catalog generated by SKICAT will eventually contain about a billion entries representing hundreds of millions of sky objects. SKICAT is 'part of the development of a new generation of intelligent scientific analysis tools [33, 8]. Without the availability of these tools for the first survey (POSS-I) conducted over four decades ago, no objective and comprehensive analysis of the data was possible. Consequently only a small fraction of the POSS-I data was ever analyzed.

2.1.4 Why was SKICAT Successful?

It is important to point out why a decision-tree based approach was effective in solving a problem that was very difficult for astronomers to solve. Indeed there were numerous attempts by astronomers to hand-code a classifier that would separate stars from galaxies at the faintest levels, without much success. This lack of success was likely due to the dimensionality of the feature space and the non-linearity of the underlying decision boundaries. Historically, efforts involving principal component analysis, or "manual" classifier construction, by projecting the data down to 2 or 3 dimensions and then searching for decision boundaries, did not lead to good results.

Based on the data it appears that accurate classification of the faint objects requires at least 8 dimensions. Projections to 2-4 dimensions lose critical information. On the other hand, human visualization and design skills cannot go beyond 3-5 dimensions. This classification problem is an excellent example of a problem where experts knew what features to measure, but not how to use them for classification. From the 40-dimensional feature-space, the decision tree and rule algorithms were able to extract the relevant discriminative information (the typical set of rules derived by RULER by optimizing over many decision trees references only 8 attributes). One can conclude that the combination of scientist-supplied features (encoding prior knowledge), and automated identification of relevant features for discriminative rules were both critical factors in the success of the SKICAT project.

2.2 Cataloging Volcanoes in Magellan-SAR Images

2.2.1 Background

On May 4th 1989 the Magellan spacecraft was launched from Earth on a mapping mission to Venus. Magellan entered an elliptical orbit around Venus in August 1990 and subsequently transmitted back to Earth more data than that from all past planetary missions combined [26]. In particular, a set of approximately 30,000, 1024 × 1024 pixel, synthetic aperture radar (SAR), 75m/pixel resolution images of the planer's surface were transmitted resulting in a high resolution map of 97% of the surface of Venus. The total combined volume of pre-Magellan Venus image data available from various past US and USSR spacecraft and ground-based observations represents only a tiny fraction of the Magellan data set. Thus, the Magellan mission has provided planetary scientists with an unprecedented data set for Venus science analysis. It is anticipated that the study of the Magellan data set will continue well into the next century [21,27, 5].

The study of volcanic processes is essential to an understanding of the geologic evolution of the planet [26], and volcanoes are by far the single most visible geologic feature in the Magellan data set. In fact, there are estimated to be on the order of 10^6 visible volcanoes scattered throughout the 30,000 images [1]. Central to any volcanic study is a catalog identifying the location, size, and characteristics of each volcano. Such a catalog would enable scientists to use the data to support various scientific theories and analyses. For example, the volcanic spatial clustering patterns could be correlated with other known and mapped geologic features such as mean planetary radius to provide evidence for (or against) particular theories of planetary history. However, it has been estimated that manually producing such a catalog of volcanoes would require 10 man-years of a planetary geologist's time. Thus, geologists are manually cataloging small portions of the data set and inferring what they can from these data [10].

2.2.2 Automated Detection of Volcanoes

At JPL we have developed a pattern recognition system, called the JPL Adaptive Recognition Tool (JARtool), for volcano classification based on matched filtering, principal component analysis, and quadratic discriminants. Over certain regions of the planet the system is roughly as accurate as geologists in terms of classification accuracy [4]. On a more global scale, the system is not currently competitive with human classification performance due to the wide variability in the visual appearance of the volcanoes and the relatively low signal-to-noise ratio of the images.

For this problem the technical challenges lie in the detection and feature extraction parts of the problem. Unlike the stars and galaxies in the SKICAT data, volcanoes are surrounded by a large amount of background clutter (such as linear and small non-volcano circular features) which renders the detection problem quite difficult. Locating candidate local pixel regions and then extracting descriptive features from these regions is non-trivial to do in an effective manner. Particular challenges include the fact that in a complex multi-stage detection system, it is difficult to jointly optimize the parameters of each individual component algorithm. A further source of difficulty has been the subjective interpretation problem: scientists are not completely consistent among themselves in terms of manual volcano detection and so there is no absolute ground truth: this adds an extra level of complexity to model training and performance evaluation. Thus, in the general scheme of science cataloging applications at JPL, the volcano project has turned out to be one of the more difficult.

2.3 Other Science Cataloging Projects at JPL

There are several other ongoing automated cataloging projects currently underway at JPL—given the data rates for current and planned JPL and NASA observation missions (including the recently-launched SOHO satellite) there will continue to be many such applications. For example, there is currently a project underway to catalog plages (bright objects in the ultraviolet Sun; somewhat analogous to sunspots) from full-disk solar images taken daily from terrestrial observatories. The data in one spectral band from one observatory is a sequence of 10^4, roughly $2K \times 2K$ pixel images taken since the mid-1960s. Of interest here is the fact that there is considerable prior knowledge (going back to the time of Galileo) about the spatial and temporal evolution of features on the surface of the Sun: how to incorporate this prior information effectively in an automated cataloging system is a non-trivial technical issue.

Another ongoing project involves the detection of atmospheric patterns (such as cyclones) in *simulated* global climate model data sets [29]. The models generate simulations of the Earth's climate at different spatio-temporal resolutions and can produce up to 30 terabytes of output per run. The vast majority of the simulated data set is not interesting to the scientist: of interest are specific anomalous patterns such as cyclones. Data summarization (description) and outlier detection techniques (for spatio-temporal patterns) are the critical technical aspects of this project.

3 GENERAL IMPLICATIONS FOR THE ANALYSIS OF MASSIVE DATA SETS

3.1 Complexity Issues for Classification Problems

Due to their size, massive data sets can quickly impose limitations on the algorithmic complexity of data analysis. Let N be the total available number of data points in the data set. For large N, linear or sub-linear complexity in N is highly desirable. Algorithms with complexity as low as $O(N^2)$ can be impractical. This would seem to rule out the use of many data analysis algorithms; for example, many types of clustering.

However, in reality, one does not necessarily need to use all of the data in one pass of an algorithm. In developing automated cataloging systems as described above, two typical cases seem to arise:

Type S-M Problems for which the statistical models can be built from a very small subset of the data, and then the models are used to segment the massive larger population (Small work set-Massive application, hence S-M)

Type M-M Problems for which one *must* have access to the entire data set for a meaningful model to be constructed. See Section 3.1.2 for an example of this class of problem.

3.1.1 Supervised Classification can be Tractable

A supervised classification problem is typically of type S-M. Since the training data needs to be manually labeled by humans, the size of this labeled portion of the data set is usually a vanishingly small fraction of the overall data set. For example, in SKICAT, only a few thousand examples were used as a training set while the classifiers are applied to up to a billion records in the catalog database. For JARtool, on the order of 100 images have been labeled for volcano content (with considerable time and effort), or about 0.3% of the overall image set. Thus, relative to the overall size of the data set, the data available for model construction can be quite small, and hence complexity (within the bounds of reason) may not to be a significant issue in model construction.

Once the model is constructed, however, prediction (classification) is typically performed on the entire massive data set: for example, on the other 99.7% of unlabelled Magellan-SAR images. This is typically not a problem since prediction is linear in the number of data-points to be predicted, assuming that the classifier operates in a spatially local manner (certainly true for detection of small, spatially bounded objects such as small volcanoes or stars and galaxies). Even algorithms based on nearest neighbor prediction, which require the training set be kept on-line can be practical provided the training set size n is small enough.

3.1.2 Unsupervised Classification can be Intractable

A clustering problem (unsupervised learning) on the other hand, can easily be a Type M-M problem. A straightforward solution would seem to be to randomly sample the data set and build models from these random samples. This would only work if random sampling is acceptable. In many cases, however, a stratified sample is required. In the SKICAT application, for example, uniform random sampling would simply defeat the entire purpose of clustering. Current work on SKICAT focuses on exploring the utility of clustering techniques to aid in scientific discovery. The basic idea is to search for clusters in the large data sets (millions to billions of entries in the sky survey catalog database). A new class of sky objects could potentially show up as a strong cluster that differs from known objects: stars and galaxies. The astronomers would then follow up with high resolution observations to see whether indeed the objects in the suspect cluster constitute a new class of what one hopes are previously unknown objects. The basic idea is that the clustering algorithms serve to focus the attention of astronomers on potential new discoveries.

The problem, however, is that new classes are likely to have very low prior probability of occurrence in the data. For example, we have been involved in searching for new high-redshift quasars in the universe. These occur with a frequency of about 100 per 10^7 objects. Using mostly classification, we have been able to help discover 10 new quasars in the universe with an order of magnitude less observation time as compared to efforts by other teams [23].

However, when one is searching for new classes, it is clear that random sampling is *exactly* what should be avoided. Clearly, members of a minority class could completely disappear from any small (or not so small) sample. One approach that can be adopted here is an iterative sampling scheme[1] which exploits the fact that using a constructed model to classify the data scales linearly with the number of data points to be classified. The procedure goes as follows:

1. generate a random sample S from the data set D.
2. construct a model M_s based on S (based on probabilistic clustering or density estimation).
3. apply the model to the entire set D, classifying items in D in the clusters with probabilities assigned by the model M_s.
4. accumulate all the residual data points (members of D that do not fit in any of the clusters of M_s with high probability). Remove all data points that fit in M_s with high probability.
5. if a sample of residuals of acceptable size and properties is collected, go to step 7, else go to 6.
6. Let S be the set of residuals from step 4 mixed with a small sample (uniform) from D, return to step 2.

[1] Initially suggested by P. Cheeseman of NASA-AMES in a discussion with U. Fayyad on the complexity of Bayesian clustering with the AutoClass system, May 1995.

7. perform clustering on the accumulated set of residuals, look for tight clusters as candidate new discoveries of minority classes in the data.

Other schemes for iteratively constructing a useful small sample via multiple efficient passes on the data are also possible [22]. The main idea is that sampling is not a straightforward matter.

3.2 Human Factors: The Interactive Process of Data Analysis

There is a strong tendency in the artificial intelligence and pattern recognition communities to build *automated* data analysis systems. In reality, fitting models to data tends to be an interactive, iterative, human-centered process for most large-scale problems of interest [2]. Certainly in the POSS-II and Magellan-SAR projects a large fraction of time was spent on understanding the problem domains, finding clever ways to pre-process the data, and interpreting the scientific significance of the results. Relatively little time was spent on developing and applying the actual algorithms which carry out the model-fitting. In a recent paper, Hand [17] discusses this issue at length: traditional statistical methodology focuses on solving precise mathematical questions, whereas the art of data analysis in practical situations demands considerable skill in the *formulation* of the appropriate questions in the first place. This issue of *statistical strategy* is even more relevant for massive data sets where the number of data points and the potentially high-dimensional representation of the data, offer huge numbers of possibilities in terms of statistical strategies.

Useful statistical solutions (algorithms and procedures) can not be developed in complete isolation from their intended use. Methods which offer parsimony and insight will tend to be preferred over more complex methods which offer slight performance gains but at a substantial loss of interpretability.

3.3 Subjective Human Annotation of Data Sets for Classification Purposes

For scientific data, performance evaluation is often subjective in nature since there is frequently no "gold standard." As an example consider the volcano detection problem: there is no way at present to independently verify if any of the objects which appear to look like volcanoes in the Magellan-SAR imagery truly represent volcanic edifices on the surface of the planet. The best one can do is harness the collective opinion of the expert planetary geologists on subsets of the data. One of the more surprising aspects of this project was the realization that image interpretation (for this problem at least) is highly subjective. This fundamentally limits the amount of information one can extract. This degree of subjectivity is not unique to volcano-counting: as part of the previously mentioned project involving automated analysis of sunspots in daily images of the Sun, there appears also to be a high level of subjectivity and variation between scientists in terms of their agreement. While some statistical methodologies exist for handling subjective opinions of multiple experts [30][28]: there appears to be room for much more work in this area.

3.4 Effective Use of Prior Knowledge

A popular (and currently resurgent) approach to handling prior information in statistics is the Bayesian inference philosophy: provided one can express one's knowledge in the form of suitable prior densities, and given a likelihood model, one then can proceed directly to obtain the posterior (whether by analytic or approximate means). However, in practice, the Bayesian approach can be difficult to implement effectively particularly in complex problems. In particular, the approach is difficult for non-specialists in Bayesian statistics. For example, while there is a wealth of knowledge available concerning the expected size, shape, appearance, etc., of Venusian volcanoes, it is quite

difficult to translate this high-level information into precise quantitative models for at the pixel-level. In effect there is a gap between the language used by the scientist (which concerns the morphology of volcanoes) and the pixel-level representation of the data. There is certainly a need for interactive, "interviewing" tools which could elicit prior information from the user and automatically construct "translators" between the user's language and the data representation. This is clearly related to the earlier point on modelling statistical strategy as a whole, rather than focusing only on algorithmic details. Some promising approaches for building Bayesian (graphical) models from data are beginning to appear (see [19] for a survey).

3.5 Dealing with High Dimensionality

Massiveness has two aspects to it: the number of data points and their dimensionality. Most traditional approaches in statistics and pattern recognition do not deal well with high dimensionality. From a classification viewpoint the key is effective feature extraction and dimensionality reduction. The SKICAT application is an example of manual feature extraction followed by greedy automated feature selection. The Venus application relies entirely on a reduction from high-dimensional pixel space to a low dimensional principal component-based feature space. However, finding useful low-dimensional representations of high-dimensional data is still something of an art, since any particular dimension reduction algorithm inevitably performs well on certain data sets and poorly on others. A related problem is that frequently the goals of a dimension reduction step are not aligned with the overall goals of the analysis, e.g., principal components analysis is a descriptive technique but it does not necessarily help with classification or cluster identification.

3.6 How Does the Data Grow?

One of the recurring notions during the workshop[2] is whether massive data sets were more complex at some fundamental level than familiar "smaller" datasets. With some massive data sets, it seems that as the size of the data set increases, so do the sizes of the models required to accurately model it. This can be due to many factors. For example, inappropriateness of the assumed model class would result in failure to capture the underlying phenomena generating the data. Another cause could be the fact that the underlying phenomena are changing over time, and the "mix" gets intractable as one collects more data without properly segmenting it into the different regimes. The problem could be related to dimensionality, which is typically larger for massive data sets.

We would like to point out that this "problematic growth" phenomenon is not true in many important cases, especially in science data analysis. In case of surveys, for example, one is careful about the design of data collection and the basic data processing. Hence, many of the important data sets are by design intended to be of type S-M. This is certainly true of the two applications presented earlier. Hence growth of data set size is not always necessarily problematic.

4 CONCLUSION

The proliferation of large scientific data sets within NASA has accelerated the need for more sophisticated data analysis procedures for science applications. In particular, this paper has briefly discussed several recent projects at JPL involving automated cataloging of large image data sets. The issues of complexity, statistical strategy, subjective annotation, prior knowledge, and high dimensionality were discussed in the general context of data analysis for massive data sets. The

[2] Massive Data Sets Workshop (July, 1995, NRC), raised by P. Huber and other participants.

subjective human role in the overall data-analysis process is seen to be absolutely critical. Thus, the development of interactive, process-oriented, interpretable statistical procedures for fitting models to massive data sets appears a worthwhile direction for future research.

In our view, serious challenges exist to current approaches to statistical data analysis. Addressing these challenges and limitations, even partially, could go a long way in getting more tools in the hands of users and designers of computational systems for data analysis. Issues to be addressed include:

- Developing techniques that deal with structured and complex data (i.e., attributes that have hierarchical structure, functional relations between variables, data beyond the flat feature-vector that may contain multi-modal data including pixels, time-series signals, etc.)
- Developing summary statistics beyond means, covariance matrixes, and boxplots to help humans better visualize high-dimensional data content.
- Developing measures of data complexity to help decide which modelling techniques are appropriate to apply in which situations item Addressing new regimes for assessing overfit since massive data sets will by definition admit much more complex models.
- Developing statistical techniques to deal with high dimensional problems.

In conclusion, we point out that although our focus has been on science-related applications, massive data sets are rapidly becoming commonplace in a wide spectrum of activities including healthcare, marketing, finance, banking, engineering and diagnostics, retail, and many others. A new area of research, bringing together techniques and people from a variety of fields including statistics, machine learning, pattern recognition, and databases, is emerging under the name: Knowledge Discovery in Databases (KDD) [16, 15]. How to scale statistical inference and evaluation techniques up to very large databases is one of the core problems in KDD.

ACKNOWLEDGEMENTS

The SKICAT work is a collaboration between Fayyad (JPL), N. Weir and S. Djorgovski (Caltech Astronomy). The work on Magellan-SAR is a collaboration between Fayyad and Smyth (JPL), M.C. Burl and P. Perona (Caltech E.E.) and the domain scientists: J. Aubele and L. Crumpler, Department of Geological Sciences, Brown University. Major funding for both projects has been provided by NASA's Office of Space Access and Technology (Code X). The work described in this paper was carried out in part by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] Aubele, J. C. and Slyuta, E. N. 1990. Small domes on Venus: characteristics and origins. *Earth, Moon and Planets*, 50/51, 493-532.

[2] Brachman, R. and Anand, T. 1996. The Process of Knowledge Discovery in Databases: A Human Centered Approach, pp. 37-58, *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), Boston: MIT Press.

[3] Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.J. 1984. *Classification and Regression Trees*. Monterey, CA: Wadsworth & Brooks.

[4] Burl, M. C., Fayyad, U. M., Perona, P., Smyth, P. and Burl, M. P. 1994. Automating the hunt for volcanoes on Venus. In *Proceedings of the 1994 Computer Vision and Pattern Recognition Conference, CVPR-94* , Los Alamitos, CA: IEEE Computer Society Press, pp.302-309.

[5] Cattermole, P. 1994. *Venus: The Geological Story*, Baltimore, MD: Johns Hopkins University Press.

[6] Cheeseman, P. and Stutz, J. 1996. Bayesian Classification (AutoClass): Theory and Results. In *Advances in Knowledge Discovery and Data Mining* , U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), Boston: MIT Press, pp.153-180.

[7] Dasarathy, B.V. 1991. *Nearest Neighbor Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, Los Alamitos, CA.

[8] Djorgovski, S.G., Weir, N., and Fayyad, U. M. 1994. Processing and Analysis of the Palomar-STScI Digital Sky Survey Using a Novel Software Technology. In D. Crabtree, R. Hanisch, and J. Barnes (Eds.), *Astronomical Data Analysis Software and Systems III, A.S.P. Conf. Ser*. 61, 195.

[9] Fasman, K. H., Cuticchia, A.J., and Kingsbury, D. T. 1994. The GDB human genome database anno 1994. *Nucl. Acid. Res*., 22(17), 3462-3469.

[10] Guest, J. E. et al. 1992. Small volcanic edifices and volcanism in the plains of Venus. *Journal of Geophysical Research*, vol.97, no.E10, pp.15949-66.

[11] Fayyad, U.M. and Irani, K.B. 1993. Multi-Interval Discretization of Continuous-Valued attributes for Classification Learning. In *Proc. of the Thirteenth Inter. Joint Conf. on Artificial Intelligence*, Chambery, France: IJCAI-11.

[12] Fayyad, U.M., Djorgovski, S.G. and Weir, N. 1996. Automating Analysis and Cataloging of Sky Surveys. In *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), Boston: MIT Press, pp.471-494.

[13] Fayyad, U.M. 1994. Branching on Attribute Values in Decision Tree Generation. In *Proc. of the Twelfth .National Conference on Artificial Intelligence AAAI-94*, pages 601-606, Cambridge, MA, 1994. MIT Press.

[14] Fayyad, U.M. 1995. On Attribute Selection Measures for Greedy Decision Tree Generation. Submitted to *Artificial Intelligence*

[15] Fayyad, U. and Uthurusamy, R. (Eds.) 1995. *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95*). AAAI Press.

[16] Fayyad, U., Piatetsky-Shapiro, G. and Smyth P. 1996. From Data Mining to Knowledge Discovery: An Overview. *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), Boston: MIT Press, pp.1-36.

[17] Hand, D. J. 1994. Deconstructing statistical questions. *J. R. Statist. Soc. A*, 157(3), pp.317-356.

[18] J. W. Head et al. 1991. Venus volcanic centers and their environmental settings: recent data from magellan. *American Geophysical Union Spring meeting abstracts*, EOS 72:175.

[19] Heckerman, D. 1996. Bayesian Networks for Knowledge Discovery, pp. 273-306, *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), Boston: MIT Press.

[20] Jarvis, J., and Tyson, A. 1981. FOCAS: Faint Object Classification and Analysis System. *Astronomical Journal* 86, 476.

[21] *Magellan at Venus: Special Issue of the Journal of Geophysical Research*, American Geophysical Union, 1992.

[22] Kaufman, L. and Rousseeuw, P. J. 1990. Finding Groups in Data: An Introduction to Cluster Analysis, New York: Wiley.
[23] Kennefick. J.D., de Carvalho, R.R., Djorgovski, S.G., Wilber, M.M., Dickson, E.S., Weir, N., Fayyad, U.M. and Roden, J. 1995. The Discovery of Five Quasars at z>4 Using the Second Palomar Sky Survey. *Astronomical Journal* (in press).
[24] Quinlan, J. R. 1986. The induction of decision trees. *Machine Learning*, 1(1).
[25] *NSSDC News*, vol.10, no.1, Spring 1994, available from *request@nssdc.gsfc.nasa.gov*.
[26] Saunders, R. S. et al. 1992. Magellan mission summary. *Journal of Geophysical Research*, vol.97, no. E8, pp.13067-13090.
[27] *Science*, special issue on Magellan data, April 12, 1991.
[28] Smyth, P., 1995. Bounds on the mean classification error rate of multiple experts, *Pattern Recognition Letters*, in press.
[29] Stolorz, P. et al. 1995. Fast spatio-temporal data mining of large geophysical datasets. In *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, pp.300-305, U. M. Fayyad and R. Uthurusamy (eds.), AAAI Press.
[30] Uebersax, J. S., 1993. Statistical modeling of expert ratings on medical treatment appropriateness. *J. Amer. Statist. Assoc*., vol.88, no.422, pp.421-427.
[31] Valdes, F. 1982. The Resolution Classifier. In *Instrumentation in Astronomy IV*, volume 331:465, Bellingham, WA, SPIE.
[32] Weir, N., Fayyad, U.M., and Djorgovski, S.G. 1995. Automated Star/Galaxy Classification for Digitized POSS-II. *The Astronomical Journal*, 109-6:2401-2412.
[33] Weir, N., Djorgovski, S.G., and Fayyad, U.M. 1995. Initial Galaxy Counts From Digitized POSS-II. *Astronomical Journal*, 110-1:1-20.
[34] Wilson, G. S., and Backlund, P. W. 1992. Mission to Planet Earth. *Photo. Eng. Rein. Sens*., 58(8), 1133-1135.

Information Retrieval and the Statistics of Large Data Sets

David D. Lewis
&T Bell Laboratories

ABSTRACT

Providing content-based access to large quantities of text is a difficult task, given our poor understanding of the formal semantics of human language. The most successful approaches to retrieval, routing, and categorization of documents have relied heavily on statistical techniques. We briefly review some of those techniques and point out where better statistical insight could lead to further advances.

1 IR AND STATISTICS TODAY

Information retrieval (IR) is concerned with providing access to data for which we do not have strong semantic models. Text is the most notable example, though voice, images, and video are of interest as well. Examples of IR tasks include *retrieving* documents from a large database in response to immediate user needs, *routing* or *filtering* documents of interest from an ongoing stream over a period of time, and *categorizing* documents according to their content (e.g. assigning Dewey Decimal numbers to abstracts of articles).

Statistical approaches have been widely applied to these systems because of the poor fit of text to data models based on formal logics (e.g. relational databases) [8]. Rather than requiring that users anticipate exactly the words and combinations of words that will appear in documents of interest, statistical IR approaches let users simply list words that are likely to appear in relevant documents. The system then takes into account the frequency of these words in a collection of text, and in individual documents, to determine which words are likely to be the best clues of relevance. A score is computed for each document based on the words it contains, and the highest scoring documents are retrieved, routed, categorized, etc.

There are several variations on this approach [5, 17, 18, 19]. *Vector space models* treat the words suggested by the user as specifying an ideal relevant document in a high dimensional space. The distance of actual documents to this point is used as a measure of relevance. *Probabilistic models* attempt to estimate, for instance, the conditional probability of seeing particular words in relevant and nonrelevant documents. These estimates are combined under independence assumptions and documents are scored for probability of membership in the class of relevant documents. *Inference*

net models are a subclass of probabilistic models which use network representations of the distribution of words. A variety of other formal and ad hoc statistical methods, including ones based on neural nets and fuzzy logic have been tried as well.

In IR systems documents are often represented as vectors of binary or numeric values corresponding directly or indirectly to the words of the document. Several properties of language, such as synonymy, ambiguity, and sheer variety make these representation far from ideal (but also hard to improve on [13]). A variety of unsupervised learning methods have been applied to IR, with the hope of finding structure in large bodies of text that would improve on straightforward representations. These include clustering of words or documents [10, 20], factor analytic decompositions of term by document matrices [1], and various term weighting methods [16].

Similarly, the retrieval query, routing profile, or category description provided by an IR system user is often far from ideal as well. Supervised learning techniques, where user feedback on relevant documents is used to improve the original user input, have been widely used [6, 15]. Both parametric and nonparametric (e.g. neural nets, decision trees, nearest neighbor classifiers) have been used. Supervised learning is particularly effective in routing (where a user can supply ongoing feedback as the system is used) [7] and in text categorization (where a large body of manually indexed text may be available) [12, 14].

2 THE FUTURE

These are exciting times for IR. Statistical IR methods developed over the past 30 years are suddenly being widely applied in everything from shrinkwrapped personal computer software, up to large online databases (Dialog, Lexis/Nexis, and West Publishing all fielded their first statistical IR systems in the past three years) and search tools for the Internet.

Until recently, IR researchers dealt mostly with relatively small and homogeneous collections of short documents (often titles and abstracts). Comparisons of over 30 IR. methods in the recent NIST/ARPA Text Retrieval Conferences (TREC), have resulted in a number of modifications to these methods to deal with large (one million documents or more) collections of diverse full text documents [2, 3, 4]. Much of this tuning has been ad hoc and heavily empirical. Little is known about the relationship between properties of a text base and the best IR methods to use with it. This is an undesirable situation, given the increasing variety of applications IR is applied to, and is perhaps the most important area where better statistical insight would be helpful.

Four observations from the TREC conferences give a sense of the range of problems where better statistical insight is needed:

1. Term weighting in long documents: Several traditional approaches give a document credit for matching a query word proportional to the number of times the word occurs in a document. Performance on TREC is improved if the logarithm of the number of occurrences of the word is used instead. Better models of the distribution of word occurrences in documents might provide less ad hoc approaches to this weighting.
2. Feedback from top ranked documents: Supervised learning methods have worked well in TREC, with some results suggesting that direct user input becomes of relatively little value

when large number of training instances are available. More surprisingly, applying supervised learning methods to the top ranked documents from an initial retrieval run, as if they were known to be relevant, has been found to be somewhat useful. This strategy had failed in all attempts prior to TREC. Is the size of the TREC collection the key to success? Can this idea be better understood and improved on (perhaps using EM methods)?

3. Massive query expansion: Supervised learning approaches to IR often augment the original set of words suggested by a user with words from documents they have judged to be relevant. For probabilistic IR methods, adding only a few words from relevant documents has been found to work best. However, for vector space methods, massive expansion (adding most or all words from known relevant documents) seems to be optimal. Reconciling this with the usually omnipresent curse of dimensionality is an interesting issue.
4. The difficulty of evaluating IR systems: TREC has reminded researchers that we do not have a good understanding of how to decide if one IR system is significantly better than another, much less how to predict in advance the level of effectiveness that an IR system can deliver to a user. Indeed, it is often unclear what reasonable measures of effectiveness are. These issues are of more interest than ever, given the large number of potential new users of IR technology.

TREC reveals just a few of the IR problems where better statistical insight is crucial. Others include dealing with time-varying streams of documents (and time-varying user needs), drawing conclusions from databases that mix text and formatted data, and choosing what information sources to search in the first place. On the tools side, a range of powerful techniques from statistics have seen relatively little application in IR, including cross-validation, model averaging, graphical models, hierarchical models, and many others. Curiously, highly computational methods have seen particularly little use. The author has been particularly interested in methods for actively selecting training data (ala statistical design of experiments) for supervised learning [9, 11]. Since vast quantities of text are now cheap, while human time is expensive, these methods are of considerable interest.

In summary, the opportunities for and need of more statistical work in IR is as vast as the flood of online text engulfing the world!

References

[1] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic indexing. *Journal of the American Society for Information Science*, 41(6):391-407, September 1990.

[2] D. K. Harman, editor. *The First Text REtrieval Conference (TREC-1*), Gaithersburg, MD 20899, 1993. National Institute of Standards and Technology. Special Publication 500-207.

[3] D. K. Harman, editor. *The Second Text REtrieval Conference (TREC-2)* , Gaithersburg, MD 20899, 1994. National Institute of Standards and Technology. Special Publication 500-215.

[4] D. K. Harman, editor. *Overview of the Third Text REtrieval Conference (TREC-3*), Gaithersburg, MD 20899-0001, 1995. National Institute of Standards and Technology. Special Publication 500-225.

[5] Donna Harman. Ranking algorithms. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval: Data Structures and Algorithms*, pages 363-392. Prentice Hall, Englewood Cliffs, NJ, 1992.

[6] Donna Harman. Relevance feedback and other query modification techniques. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval: Data Structures and Algorithms*, pages 241-263. Prentice Hall, Englewood Cliffs, NJ, 1992.

[7] Donna Harman. Overview of the third Text REtrieval Conference (TREC-3). In D. K. Harman, editor, *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 1-27, Gaithersburg, MD, 1995. U. S. Dept. of Commerce, National Institute of Standards and Technology.

[8] David D. Lewis. Learning in intelligent information retrieval. In *Eighth International Workshop on Machine Learning*, pages 235-239, 1991.

[9] David D. Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In William W. Cohen and Haym Hirsh, editors, *Machine Learning: Proceedings of the Eleventh International Conference on Machine Learning*, pages 148-156, San Francisco, CA, 1994. Morgan Kaufmann.

[10] David D. Lewis and W. Bruce Croft. Term clustering of syntactic phrases. In *Thirteenth Annual International A CM SIGIR Conference on Research and Development in Information Retrieval*, pages 385-404, 1990.

[11] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In W. Bruce Croft and C. J. van Rijsbergen, editors, *SIGIR 94: Proceedings of the Seventeenth Annual International A CM-SIGIR Conference on Research and Development in Information Retrieval*, pages 3-12, London, 1994. Springer-Verlag.

[12] David D. Lewis and Philip J. Hayes. Guest editorial. *ACM Transactions on Information Systems*, 12(3):231, July 1994.

[13] David D. Lewis and Karen Sparck Jones. Natural language processing for information retrieval. *Communications of the ACM*, 1996. To appear.

[14] David D. Lewis and Marc Ringuette. A comparison of two learning algorithms for text categorization. In *Third Annual Symposium on Document Analysis and Information Retrieval*, pages 81-93, Las Vegas, NV, April 11-13 1994. ISRI; Univ. of Nevada, Las Vegas.

[15] Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288-297, 1990.

[16] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management* , 24(5):513-523, 1988.
[17] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, New York, 1983.
[18] H. R. Turtle and W. B. Croft. A comparison of text retrieval models. *The Computer Journal*, 35(3):279-290, 1992.
[19] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, second edition, 1979.
[20] Peter Willett. Recent trends in hierarchic document clustering: A critical review. *Information Processing and Management*, 24 (5):577-598, 1988.

Some Ideas About the Exploratory Spatial Analysis Technology Required for Massive Databases

Start Openshaw
Leeds University

ABSTRACT

The paper describes an emerging problem of an explosion in spatially referenced information at a time whilst there is only essentially legacy geographical analysis technology. It offers some ideas about what is required and outlines some of the computationally intensive methods that would seem to offer considerable potential in this area of applications.

1 A GLOBAL SPATIAL DATA EXPLOSION

The Geographic Information Systems revolution of the mid 1980's in conjunction with other historically unprecedented developments in Information Technology are creating an extremely spatial data rich world. Suddenly, most map and map related databases have increasingly fine resolution geography on them. The paper map using industries throughout the world have gone digital. In many countries this ranges from the traditional AM/FM areas of GIS (viz utilities), it includes the large scale mapping agencies (viz in the UK the Ordnance Survey, the Geological Survey, etc), it includes all manner of remotely sensed data and just as exciting it extends to virtually all items of fixed infrastructure and certainly all people related databases (that is any and all postal address data, property, road, and land information systems). At a world level there is increasing interest in global environmental databases (Mounsey and Tomlinson, 1988). There is now a vast wealth of digital backcloth material (viz the digital maps) and related attribute data covering most aspects of human life, ranging from the cradle to the grave and an increasing proportion of behaviour related events in between. Most governmental and commercial sector statistical and client information systems are well on the way to becoming geo-statistical. In fact nearly all computer information can be regarded as being broadly geographical in the sense that there are usually some kind of spatial coordinates on it or implicit in it. This may range from the traditional

geographic map related data to non-traditional sources such as 2 and 3 dimensional images of virtually any kind. Indeed, it is worth observing that several existing GIS's already contain sufficient numeric resolution to work down to nanometre scales! But lets stick with the more traditional geographic data. The technology needed to create, store, and manipulate these land and people related databases exists, it well developed, and is fairly mature. What is almost entirely missing are the geographical analysis and modelling technologies able to deal with the many new potential opportunities that these data rich environments now make possible.

It is not unusual for governments and commercial organisations to spend vast sums of money on building databases relevant to their activities. Many have huge amounts of capital tied up in their databases and concepts such as data warehouses are providing the necessary IT infrastructure. They know that their future depends on building and using these information resources. Most businesses and governmental agencies are becoming information industries but currently there seems to be an almost universal neglect of investment in the analysis tools needed to make the most of the databases.

2 A GLOBAL DATA SWAMP

Maybe the problem is that the data holders and potential users are becoming "data swamped" and can no longer appreciate the opportunities that exist. Often their ideas for analysis on these massive databases seems to mainly relate to an earlier period in history when data was scarce, the numbers of observation were small and the capabilities of the computer hardware limited. As a result there are seemingly increasingly large numbers of important, sometimes life critical and sometimes highly commercial, databases that are not being fully analysed; if indeed they are being spatially analysed at all. Openshaw (1994a) refers to this neglect as a type of spatial analysis crime. For some data there is already an over-whelming public imperative for analysis once the data exist in a suitable form for analysis. Is it not a crime against society if critical databases of considerable contemporary importance to the public good are not being adequately and thoroughly analysed. This applies especially when there might well be a public expectation that such analysis already occurs or when there is a strong moral imperative on the professions involved to use the information for the betterment of people's lives. Examples in the UK would include the non-analysis of much spatially referenced information: examples include most types of cancer data, mortality data, real-time crime event data, pollution data, data needed for real-time local weather forecasting, climatic change information, and personal information about people who are to be targeted for special assistance because they exist in various states of deprivation. There are many examples too involving the spatial non-analysis of major central and local government databases: for example, tax, social security payments, education performance and housing benefits. In the commercial sector also, it not unusual for large financial sector organisations to create massive customer databases, often containing longitudinal profiles of behaviour, increasingly being updated in real-time; and then do virtually nothing clever when it comes to analysis. Yet every single business in the IT Age knows that their long term future viability depends on themselves making good use of their information resources. Some of the opportunities involve spatial analysis; for example, customer targeting strategies, spatial planning and re

organisation of business networks. Economic well being and growth may become increasingly dependent on making more sophisticated use of the available data resources. Of course there are dangers of intrusion into private lives but all too often the confidentiality problems are grossly exaggerated as a convenient excuse for doing nothing (Openshaw, 1994b).

To summarise, there are an increasing number of increasingly large and complex databases containing potentially useful information that for various reasons are not currently being adequately analysed from a spatial statistical geographic perspective. In many countries, if you ask the question Who is responsible for monitoring key indicators of health and well-being for unacceptable or preventable local abnormalities, then the answer will usually be 'non-one' because the technology needed simply does not exist. In the UK the most famous suspected leukaemia cancer cluster was uncovered by journalists in a Pub rather than by any detailed spatial statistical analysis or data base monitoring system. Spatial analysis is a branch of statistics that is concerned with the quantitative analysis (and modelling) of the geographic arrangement, patterns, and relationships found in and amongst map referenced data of one form or another. The GIS revolution has greatly increased the supply of spatially referenced data without any similar provision of new spatial analysis tools able to cope with even the basic analysis needs of users with very large spatial databases. In the last couple of years high parallel computing systems such as the Cray T3D provide memory spaces just about big enough to handle all but the very largest of databases. The need now is for new data exploratory tools that can handle the largest databases and produce suggestions of geographical patterns or relationships; identify, geographically: localised anomalies if any exist: and provide a basis for a subsequent more focused analysis and action. Note also that the need is for analysis but not necessarily because there are expectations of discovering anything. Analysis has a major re-assurance aspect to it. In modern societies it is surely not an unreasonable expectation that benign Big Brother surveillance systems should be continually watching for the unusual and unexpected.

3 NEW TOOLS ARE REQUIRED

In a spatial database context there is an urgent need for exploratory tools that will continually sieve and screen massive amounts of information for evidence of patterns and other potentially interesting events (if any exist) but without being told, in advance and with a high degree of precision, WHERE to look in space, WHEN to look in time. and WHAT to look for in terms of the attributes that are of interest other than in the broadest possible ways. Traditionally, spatial analysts either start with a priori theory which they then attempt to test or else they use interactive graphics procedures, perhaps linked to map displays, to explore spatial data. However, as the level of complexity and the amount of data increases this manual graphics based approach become inadequate. Equally, in the spatial data rich 1990's there are not many genuine and applicable a priori hypotheses that can be tested; usually, we just do not know what to expect or what might be found. Exploration is seen as a means of being creative and insightful in applications where current knowledge is deficient.

The combination of a lack of prior knowledge and the absence spatial analysis tools sufficiently powerful to handle the task of exploratory spatial data analysis especially in

highly multivariate situations with large numbers of observations, has resulted in a situation where very little investigative and applied analysis actually occurs. The technical problems should not be underestimated as spatial data is often characterised by the following features which serve to complicate the analysis tasks:

1. non-normal frequency distributions, 2. non-linearity of relationships, 3. spatially autocorrelated observations, 4. spatially structured variations in degrees of precision, noise, and error levels, 5. large and increasing data volumes, 6. large numbers of potential variables of interest (i.e. highly multivariate), 7. data of varying degrees of accuracy (i.e. can be variable specific), 8. often misplaced confidentiality concerns (i.e. it might identify someone!), 9. non-ideal information (i.e. many surrogates), 10. mixtures of measurement scales, 11. modifiable areal unit or study region effects, 12. patterns and relationships that are localised and not global, and 13. presence of database errors.

Traditionally, people have coped by being highly selective whilst working with very few variables and relatively small numbers of observations. However, this is increasingly hard to achieve. Many spatial epidemiological studies decide in advance on the selection of disease, the coding of continuous time into discrete bands, the recoding of the data, the geographical aggregation to be used, and the form of standardisation to be applied. Then they expect to "let what is left of the data to speak for itself" via exploratory analysis, after having first strangled it by the study protocols imposed on it in order to perform the analysis. This is crazy! Heaven alone knows what damage his may have done to the unseen patterns and structure lurking in the database or indeed, what artificial patterns might have been accidentally created. No wonder exploratory spatial epidemiology has had so few successes. It is useful, therefore, to briefly review some of the developments that appear to be needed to handle the geographical analysis of massive spatial databases.

3.1 Automated map pattern detectors

One strategy is to use a brute force approach and simply look everywhere for evidence of localised patterns. The Geographical Analysis Machines (GAM) of Openshaw et al (1987) is of this type, as is the Geographical Correlates Exploration Machine (GCEM) of Openshaw et al (1990). The search requires a supercomputer but is explicitly parallel and thus well suited for the current parallel supercomputers. The problems here are the dimensionally restricted nature of the search process, being limited to geographic space; and the statistical difficulties caused by testing millions of hypotheses (even if only applied in a descriptive sense). Nevertheless, empirical tests have indicated that the GAM can be an extremely useful spatial pattern detector that will explore the largest available data sets for evidence of localised geographic patterning. The strength of the technology results from its comprehensive search strategy, the lack of any prior knowledge about the scales and nature of the patterns to expect, and its ability to handle uncertain data.

3.2 Autonomous database explorer

Openshaw (1994c, 1995) outlines a further development based on a different search strategy. Borrowing ideas from Artificial Life, an autonomous pattern hunting creature can be used to search for patterns by moving around the spatial database in whatever dimensions are

relevant. It operates in tri-space defined by geographic map coordinates, time coordinates. and also attribute coordinates (a dissimilarity space). In the prototype the creature is a set of hyperspheres that try to capture patterns in the data by enveloping them. The dimensions and locations of the spheres is determined by a Genetic Algorithm and performance is assessed by a sequential Monte Carlo significance test. The characteristics of the hyper-spheres indicate the nature of the patterns being found. This technology is being developed further and broadened to include a search for spatial relationships and also linked to computer animation to make the analysis process visible and thus capable of being more readily understood by end-users (Openshaw and Perrie. 1995).

3.3 Geographic object recognition

Another strategy is that described in Openshaw (1994d) which viewers the problem as being one of pattern recognition. Many spatial databases are now so detailed that it is becoming increasingly difficult to abstract and identify generalisable and recurrent patterns. As the detail and resolution have increased dramatically, geographers have lost the ability to stand back and generalise or even notice recurrent patterns. It is ironic that in geography the discovery of many of the principal spatial patterns and associated theoretical speculations that exist pre-date the computer. In the early computer and data starved era geographers tested many of these pattern hypotheses using statistical methods, and looked forward to better resolution data so that new and more refined spatial patterns might be found. Now that there is such an incredible wealth of spatial detail, it is clear that here too there is no good ideas of what to do with it! Beautiful, multi-coloured maps accurate to an historically unprecedented degree shown so much detail that pattern abstraction by manual means is now virtually impossible and, because this is a new state, the technology needed to aid this process still needs to be developed. Here, as in some other areas, finer resolution and more precision has not helped but hindered.

If you look at a sample of towns or cities you can easily observe broad pattern regularities in the location of good-bad-average areas etc. Each town is unique but the structure tends to repeat especially at an abstract level. Now examine the same towns using the best available data and there is nothing that can be usefully abstracted or generalised, just a mass of data with highly complex patterns of almost infinite variation. Computer vision technology could in principle be used to search for scale and rotationally invariant two or three dimensional geographic pattern objects, with a view to creating libraries of recurrent generalisations. It is thought, that knowledge of the results stored in these libraries might well contribute to the advancement of theory and concepts relating to spatial phenomenon.

3.4 Very large spatial data classification tools

Classification is a very useful data reduction device able to reduce the number of cases/observations from virtually any very large number to something quite manageable such as 50 clusters (or groups) of cases/ observations that share common properties. This is not a new technology, however, there is now some need to be able to efficiently and effectively classify several millions (or more) cases. Scaling up conventional classifiers is not a difficult task; Openshaw et al (1985) reported the results of a multivariate classification of 22 million Italian

households. However, much spatial data is imprecise, non-random, and of variable accuracy. Spatio-neural network methods based on modified Kohonen self-organising nets provide an interesting approach that can better the problems (Openshaw, 1994e). This version uses a modified training method that biases the net towards the most reliable data. Another variant also handles variable specific data uncertainty and has been put into a data parallel form for the Cray T3D.

Another form of classification is provided by spatial data aggregation. It is not unusual for N individual records describing people or small area aggregations of them to be aggregated to M statistical reporting areas, typically, M is much less than N. Traditionally, statistical output area definitions have been fixed, by a mix of tradition, historical, accident, and fossilised by inertia and outmoded thinking. However, GIS removes the tyranny of users having to use fixed areas defined by others that they cannot change. User controlled spatial aggregation of very large databases is potentially very useful because: (1) it reduces data volumes dramatically but in a user controlled or application specific way: (2) it provides a mechanism for designing analytically useful zones that meet confidentiality restrictions and yield highest possible levels of data resolution: and (3) it is becoming necessary purely as a spatial data management tool. However, if users are to be allowed to design or engineer their own zoning systems then they need special computer systems to help them A start has been made but much remains to be done, Openshaw and Rao (1995).

3.5 Neurofuzzy and hybrid spatial modelling systems for very large data bases

Recent developments in AI, in neural networks and fuzzy logic modelling have created very powerful tools that can be used in geographical analysis. The problem is applying them to very large spatial databases. The size and relational complexity of large data bases increasingly precludes simply downloading the data in a fiat file form for conventional workstation processing. Sometimes this will be possible but not always, so how do you deal with a 10 or 40 gigabyte database containing many hundred hierarchically organised table? There are two possible approaches: Method 1 is to copy it all in a decomposed fiat file form onto a highly parallel system with sufficient memory to hold all the data and sufficient speed to do something useful with it. The Cray T3D with 2.56 processors provides a 16 gigabyte memory space albeit distributed. However, there is clearly the beginnings of large highly parallel systems with sufficient power and memory to at least load the data. The question of what then still however needs to be resolved.

Method two is much more subtle. Why not leave the database alone, assuming that it is located on a suitably fast parallel database engine of some kind. The problem is doing something analytical with it using only SQL commands. One approach is to re-write whatever analysis technology is considered useful so that it can be run over a network and communicates with the database via SQL instructions. The analysis machines starts off with a burst of SQL commands. it waits patiently for the answers: when they are complete, it uses the information to generate a fresh burst of SQL instructions: etc. Now it is probably not too difficult to re-write most relevant analysis procedures for this type of approach.

4 DISCOVERY HOW TO EXPLOIT THE GEOCYBERSPACE

It is increasingly urgent that new tools are developed and made available that are able to analyse and model the increasing number of very large spatial databases that exist. It is not data management or manipulation or hardware that is now so critical but analysis. It is not a matter of merely scaling existing methods designed long ago because usually the underlying technology is so fundamentally inappropriate. Nor is it a matter of being more precise about what we want to do: in fact there is often just too much data for that. Instead, there is an increasing imperative to develop the new highly computation and intelligent data analysis technologies that are needed to cope with incredibly data rich environments. In geography the world of the 21st century is perceived to be that of the geocyberspace, the world of computer information (Openshaw. 1994f). It is here where the greatest opportunities lie but it is also here where an extensive legacy of old fashioned, often early computer, philosophically inspired constraints still dominate thinking about what we can and should not do. Maybe it is different elsewhere.

References

[1] Mounsey, H., Tomlinson, R., 1988 *Building databases for global science*. Taylor and Francis, London

[2] Openshaw, S., Sforzi, F., Wymer, C., 1985 *National classifications of individual and area census data: methodology, comparisons, and geographical significance*. Sistemi Urbani 3, 283-312

[3] Openshaw, S., Charlton, M., Wymer, C., Craft, A., 1987 *A Mark I Geographical analysis · machine for the automated analysis of point data sets*. Int J. of GIS 1. 33.5-358

[4] Openshaw, S., Cross, A., Charlton, M., 1990 *Building a prototype geographical correlates exploration machine*. Int J of GIS 3, 297-312

[5] Openshaw, S., 1994a *GIS crime and Spatial Analysis in Proceedings of GIS and Public* Policy Conference. Ulster Business School 22-34

[6] Openshaw, S., 1994b *Social costs and benefits of the Census*. Proceedings of XVth International Conference of the Data Protection and Privacy Commissioners Manchester. 89-97

[7] Openshaw, S., 1994c *Two exploratory space-time attribute pattern analysers relevant to GIS*. in Fotheringham, S., and Rogerson, P., (eds) GIS and Spatial Analysis Taylor and Francis. London 83-104

[8] Openshaw, S., 1994d *A concepts rich approach to spatial analysis. theory generation and scientific discovery in GIS using massively parallel computing*, in Worboys, M.F. (ed) Innovations in GIS Taylor and Francis, London 123-138

[9] Openshaw, S., 1994e *Neuroclassification of spatial data*, in Hewitson, B.C., and Crane, R.G., (eds) Neural Nets: Applications in Geography Kluwer Publishers, Boston 53-70

[10] Openshaw, S., 1994f *Computational human geography; exploring the geocyberspace* Leeds Review 37, 201-220

[11] Openshaw, S., 1995 *Developing automated and smart spatial pattern exploration tools for geographical information systems*. The Statistician 44, 3-16

[12] Openshaw, S., Rao, L., 1995 *Algorithms for re-engineering 1991 census geography*. Environment and Planning A 27. 425-446

Massive Data Sets in Navy Problems

J.L. Solka, W.L. Poston, and D.J. Marchette
Naval Surface Warfare Center
E.J. Wegman
George Mason University

I. ABSTRACT

There are many problems of interest to the U. S. Navy that, by their very nature, fall within the realm of massive data sets. Our working definition of this regime is somewhat fuzzy although there is general agreement among our group of researchers that such data sets are characterized by large cardinality ($>10^6$), high dimensionality (>5), or some combination of moderate levels of these two which leads to an overall large level of complexity. We discuss some of the difficulties that we have encountered working on massive data set problems. These problems help point the way to new needed research initiatives. As will be seen from our discussions one of our applications of interest is characterized by both huge cardinality and large dimensionality while the other is characterized by huge cardinality in conjunction with moderate dimensionality. This application is complicated by the need to ultimately field the system on components of minimal computational capabilities.

II. INTRODUCTION

We chose to limit the focus of this paper to two application areas. The first application area involves the *classification of acoustic signals* arising from military vehicles. Each of these acoustic signals takes the form of (possibly multiple) time series of varying duration. Some of the vehicles whose acoustic signatures we may desire to classify include helicopters, tanks, and armored personal carriers. In Figures la and b we provide an example plot of signals collected from two different rotary wing vehicle sources. These signals are variable in length and can range anywhere from tens of thousands of measurements to billions of measurements depending on the sampling rate and data collection interval. Part of our research with these signals involves the identification of advantageous feature sets and the development of classifiers which utilize them. As will be discussed below this process in itself is fraught with associated massive data set problems. An additional layer of difficulty comes from the fact that we are interested in ultimately

fielding an instrument which will be capable of classifying acoustic signatures under battlefield conditions in as near to a real-time manner as possible. This burden places severe restrictions on the complexity and robustness of the system that is ultimately fielded.

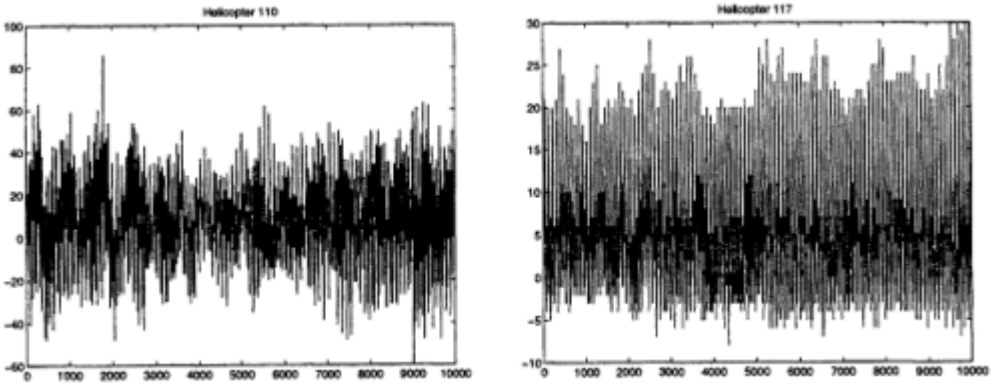

Figure 1 Time series plots for two helicopter models.

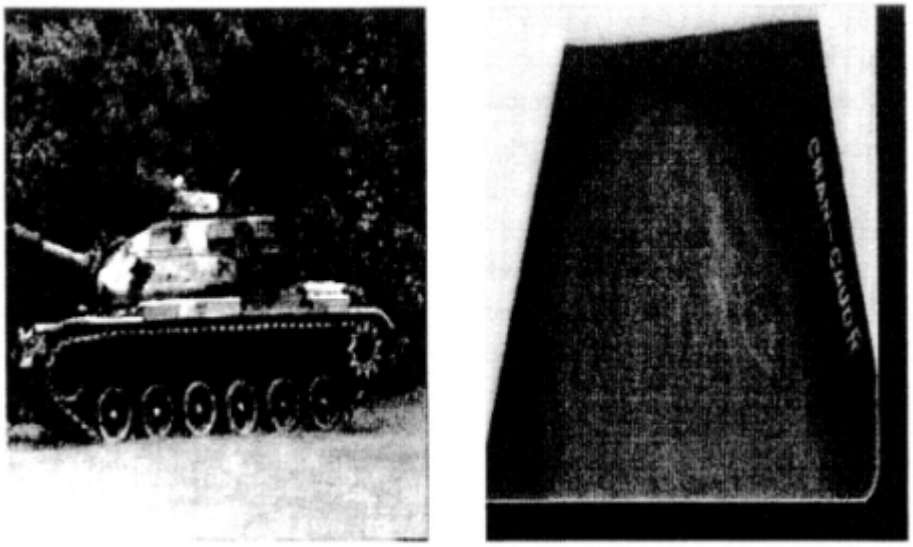

Figure 2 Grayscale images of a tank and mammogram.

Our second application focuses on the *automatic segmentation of images into regions of homogeneous content*, with an ultimate goal of classifying the different regions. Our research has focused on a wide variety of image modalities and desired classification systems. They include the detection of combat vehicles in natural terrain (Solka, Priebe, and Rogers, 1992), the detection of man-made regions in aerial images (Priebe, Solka, and Rogers, 1993 and Hayes et al., 1995), and the characterization of mammographic parenchymal patterns (Priebe et al., 1995). In Figure 2, we provide representative vehicular and mammographic images. The homogeneity criterion in this case is based on functionality of the region. For example, we wish to

have the capability to identify a given region as containing a tank or a bush, normal parenchymal tissue or tumor. This problem is more difficult then the previous one with regard to the size of the data sets, the spatial correlation of the extracted features, and the dimensionality of the extracted features.

In the following sections we discuss at length the specific difficulties that we have encountered analyzing the data sets associated with these two applications. There are associated problems with both the exploratory data analysis of potential features, the development of density estimates of these features, and the ultimate testing and fielding of classification systems based on these features. After presenting these problems, we next suggest a course of potential research initiatives which will help alleviate these difficulties. This is followed by a brief summary of our main points.

III. DISCUSSIONS

We first discuss the simpler of these problems. In the *acoustic signal processing* application, one is initially concerned with identifying features in the signals that will be useful in distinguishing the various classes. Fourier-based features have been useful in our work with rotary wing vehicles because the acoustic signals that we have studied to date have been relatively stationary in their frequency content. Depending on the type of signal which one is interested in classifying, one typically keeps a small number of the strongest frequency components. We currently employ a harmogramic hypothesis testing procedure to decide which and how many of the spectral components are significant (Poston, Holland, and Nichols, 1992). For some of our work this has led to two significant spectral components. In the case of rotary wing aircraft, these components correspond to the main and tail rotor frequencies (see Figure 3). Given a signal for each class of interest one extracts these features from each signal based on overlapping windows. In our application this easily leads to approximately 10^6 points residing in R^2 for each of 5 classes.

However, once we extend the problem to the classification of moving ground vehicles, these simple features are no longer adequate. In fact, it is not surprising that the periodograms from moving ground vehicles exhibit nonstationary characteristics. In Figure 4, we present a periodogram for one of the vehicle types. This problem requires that one must perform exploratory data analysis on the possible tilings of regions of the time-frequency plane in order to discover their utility as classification features. This increases the dimensionality of our problem from 2 to between 16 and 100 depending on our tiling granularity. So we are faced with the task of performing dimensionality reduction using millions of observations in tens of dimensions. Even if one utilizes automated techniques such as projection pursuit for the determination of the needed projection, there can be problems with long projection search times and overplotting in the lower-dimensional projected views.

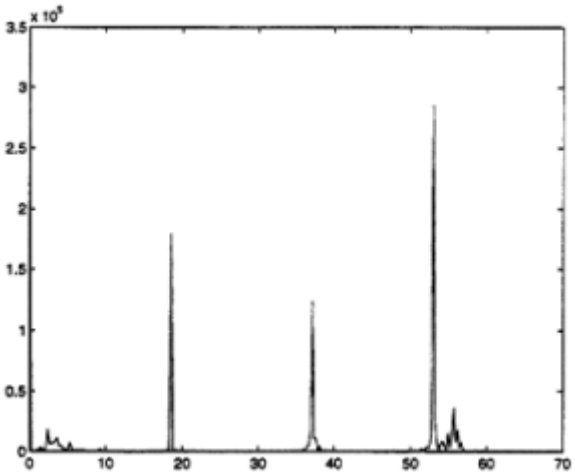

Figure 3 Power spectral density of one of the helicopter signals.

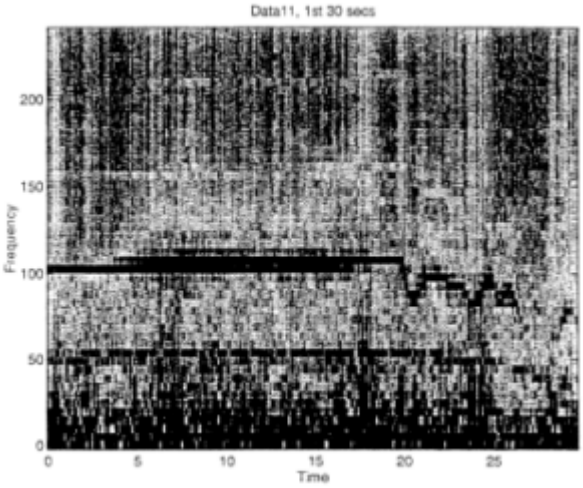

Figure 4 Periodogram of a moving vehicle.

In order to cast the classification problem within a hypothesis testing framework, probability density estimates are built for each of the classes. Performing quality probability density estimation in a moderately low dimensional space (<4) using millions of features can be difficult. Many density estimation procedures are analogous to clustering and as shown by Wegman (1995), clustering procedures are computationally costly. With an ultimate cost of between $O(n^{3/2})$ and $O(n^2)$ depending on the nature of the algorithm, extremely long execution times can be expected for the analysis of these large data sets.

These probability density estimators are ultimately deployed as part of the fielded system. In addition it is desirable for the system to have some limited capability to either refine the existing estimates for a given class or to add in an estimate of a new class's probability density function as part of its field operation. Although we have developed and do utilize schemes for density estimation that are recursive in nature (Priebe and Marchette, 1991 and Priebe, 1994), given a new signal of limited duration one may need to have stored the features for a period of time in order to obtain a good maximum likelihood density estimate for the signal features. In addition the nature of this problem dictates our interest in density estimation and dimensionality reduction procedures in the presence of partial classification information.

The second problem is much more daunting in extent. To begin with it is not uncommon to deal with images that are on the order of 4,000 × 4,000 pixels in size, with anywhere from 1 to 7 spectral bands per pixel. Typical data sets can range from tens to thousands of images. Unlike some industrial applications where there is tight control on the background and possible content of the image, these images can have a very wide range of subjects. A typical example is to detect and identify the ground vehicles within an image. The range of possible backgrounds is enormous: desert, farmlands, forest, jungle, and city streets. Each background will require different kinds of processing to eliminate clutter. Further, one is often forced to determine the background from the image without *a priori* knowledge (for example, for an automated system on a satellite).

In order to solve this problem, features must be computed from the image. These are then used for the classification task. Typically the problem is partitioned into sub-problems. For example, one might first determine the environment (forest, desert, etc.). This would then dictate the types of algorithms to be used for the detection of regions of interest (ROI), the detection of objects, and finally the classification of the objects into the desired categories. Each of these problems will require (potentially) different features to be extracted and processed.

Given the nature of the image classification problem there are many different types of features that can be extracted at each pixel or within each region of the image. Even if one limits their attention to "low level" features (for ROI detection, for example), it is very easy to find 30 features of interest, each of which must be computed for each pixel or small region of pixels. Assuming 20 images are used to identify good combinations of features for classification this still leads to roughly 9.6 billion observations in R^{30}. This may have to be repeated for each different environment or class of potential targets. At this level of complexity even the most rudimentary exploratory data analysis (EDA) tasks are inadequate.

The curse of dimensionality (Scott, 1992) often requires a projection to a lower dimensional space before density estimation or other classification techniques can be performed but this still reduces the complexity of the problem only by an order of magnitude. The added complexity from the selection of appropriate projections more than makes up for this reduction. Rampant problems with overplotting and slow plotting times are the norm for this kind of data, making interactive EDA extremely difficult. Again, Wegman (1995) suggests that direct visualization of data sets over about 106 data points is virtually impossible. In fact, the unwieldy nature of these images makes it difficult for human operators to obtain ground truth on them. In addition, one often produces a number of "interesting" projections or other derived data associated with a given set of features. It becomes a very difficult task, just in the data structures area, to keep track of all of this associated data.

Even if one first segments the image into regions and extracts features based on regions rather than individual pixels, there are still serious problems. One still must deal with the entire image in order to do the segmentation and so the original requirement to deal with approximately 16 million pixels per image still holds. In addition, different features may require different segmentation algorithms and so the apparent reduction in number of feature vectors is easily matched by an increase in computational complexity.

Once a "best" combination of features is found, one must build probability density estimates for each of the classes in the image. In this case the huge data sets that are associated with each class can lead to extremely slow convergence of recursive density estimation procedures. This phenomena has been observed by us in the case of likelihood maximization of over determined mixture models under the Expectation Maximization (EM) algorithm (Solka, 1995). In these cases careful monitoring of the parameter estimation process is essential (Solka, Poston, Wegman 1995). If one attempts to mitigate this problem through the use of iterative procedures one is faced with the overhead associated with the storage of all of the observations.

The problem becomes more pronounced when one realizes that 20 images is by no means sufficient for validation of the algorithms. Many hundreds or thousands of images must be processed in order to get good estimates of performance. These images are then typically used as part of a leave-one-image-out cross validation procedure. Furthermore, these numbers may increase dramatically as the number of target types increases.

A final problem, alluded to above, is that often truth values are known only for a very small subset of the images. Imagine trying to build a system to detect Scud missiles in Iraq. This obviously would require images of these missiles in the terrain in which they were likely to be deployed. However, in order to obtain these images one must either be lucky, spend a lot of time examining images, or already have a method for finding Scud missiles in Iraq. While it is sometimes possible to use images of similar targets, or construct mock targets, there is always the danger of building a system which finds mock targets (as opposed to real ones). As a result of this problem, detection and classification algorithms must be able to handle partially categorized and uncategorized data.

Fielded systems that will be utilized to build classifiers "on the fly" are required to deal with hundreds or thousands of images even in the training phase of the procedure. After an initialization phase these classifiers will be expected to continue to update their density estimates for the existing "target" classes and also build new estimates for any new target classes that might appear. As we discussed with regard to the acoustic signal processing work the use of recursive estimation procedures becomes essential.

IV. PROPOSED NEEDS

Given this daunting list of problems what are some of the possible solutions? We will discuss our "wish list" of solutions in the order into which we decompose the classification problem. First we will discuss the needs of feature extraction, followed by exploratory data analysis, then density estimation, and finally classification. We hope that these observations will help point the way to some fruitful research areas.

First and foremost the problems associated with the design of classification systems must be managed by some sort of database management system. There has been some recent press by such developers as StatSci regarding database utilities which will be included in their new releases. There are also some packages provided by third party developers such as DBMS COPY. These packages are aimed at providing a seamless interface between statistical analysis software and database management systems. That being said, there are yet fundamental research issues associated with *the seamless integration of statistical software and database management systems*. There are numerous types of information that occur routinely in the classification process that these systems could be used to manage. We have focused on situations in which ele

ments of the database may be multispectral images. How, for example, can one index and browse such multispectral image databases? Low resolution, thumbnail sketches, for example, may essentially obscure or eliminate texture which often is the very feature for which we may be browsing. To our knowledge, no database management system has proven to be adequate for the problems addressed above, especially those inherent in managing a large image library with associated features and class models.

More to the point, statisticians as a group tend not to think in terms of massive data files. The implication of this is that flat file structures tend to be adequate for the purposes of most small scale data and statistical analysis. There is no perceived need to understand more complex database structures and how to exploit these structures for analysis purposes. Very few academic statistics programs would even consider a course in data structures and databases as a legitimate course for an advanced degree program. One interesting consequence of this is the newfound interest in the computer science community for the discipline they now call *data mining*. While principally focused on exploratory analysis of massive financial transaction databases, tools that involve not only statistical and graphical methods, but also neural network and artificial intelligence methods are being developed and commercialized with little input from the statistics community. Two URLs which give some insight into the commercial *knowledge discovery* and *data mining* community are **http://info.get.com/~kdd/** and **http://www.ibi.com/**. In short, we believe that a fundamental cultural shift among statistical researchers is needed.

Next we turn our attention to the EDA portion of the task. Some of the needs here include techniques for dimensionality reduction in the presence of partial classification information and recursive dimensionality reduction procedures. Since the dimensionality reduction technique ultimately is part of any fielded system, it seems that recursive techniques for updating these transformations based on newly collected observations are needed. Techniques to deal with the problem of overplotting are also required. Density estimation has been suggested by Scott (1992) as a technique to improve on the overplotting problem. The inclusion of binning procedures such as the hexagonal procedure of Carr et al. (1992) as part of the existing statistical analysis packages would be helpful. Another approach to this problem is the use of sampling techniques to thin the data before plotting. These techniques have the requirement of efficient methods for outlier identification (Poston et. al., 1995 and Poston, 1995). Techniques for the visual assessment of clusters in higher dimensional space are also needed. We have previously developed a system for the visualization of data structures in higher dimensions based on parallel coordinates (Wegman, 1990), but much work remains to be done on this and new approaches to higher dimensional cluster visualization.

There remain many technical issues associated with probability density estimation for massive data sets in moderately high-dimensional spaces. Efficient procedures for estimating these densities and performing cluster analysis on them are needed. These procedures must be highly efficient in order to combat the large cardinality of these data sets. Further study into recur

sive estimation procedures is needed. Recursive or some sort of hybrid iterative/recursive procedure becomes essential when the data sets become too massive for full storage. Continued study into overdetermined mixtures models is needed. In particular additional work into the visualization of these models and the derivation of their theoretical properties is needed.

Last we turn our attention to some needs in the area of discriminant analysis. Image classification procedures based on the use of Markov random field models continues to be a fruitful area of research. In addition research into the use of pseudo-metrics such as Kullback Leibler for comparisons of probability density functions as an alternative to the standard likelihood ratio hypothesis testing deserves continued research. Finally procedures that attempt to combine some of the modem density estimation techniques with the Markov random field approach have merit. On very large images, Markov random field techniques may be inappropriate due to computation time. Fast versions of these techniques need to be developed.

V. CONCLUSIONS

We have examined some of the problems that we have encountered in building systems which perform automatic classification on acoustic signals and images. The design of these systems is a multifaceted problem that spawns a large number of associated massive data set subproblems. We have not only indicated some of the problems that we have encountered but we have also pointed the way to fruitful problems whose solutions are required to produce the tools which researchers in this area need.

A final point is the use of video over single images, which is becoming more important as the technology advances. The use of video not only vastly decreases the time allowed to process each image (or frame) but also greatly increases the number of images. With image frame rates of 30 fps and higher this gives a potential new meaning to the word massive data set. Since these images are highly correlated in time and space, this raises potentially a whole new conceptual framework. We conceive, for example, an animation or movie as a sequence of two dimensional images. Alternatively we could conceive of an animation or movie as a function $f(x, y, t)$ which lives in R^4. This change of perspective raises the need for filtering, processing, and analysis techniques for three dimensional "images" as opposed to the two dimensional images we have been primarily discussing here. Of course, at each "voxel" location it is possible to have a vector-valued function (e.g. a multispectral vector). But this discussion raises many new issues, and is beyond the scope of this paper.

VI. ACKNOWLEDGMENTS

The authors (ILS, WLP and DIM) would like to acknowledge the support of the NSWCDD Independent Research Program through the Office of Naval Research. In addition, the work of EJW was supported by the Army Research Office under contract number DAAH04-94-G-0267, by the

Office of Naval Research under contract number N00014-92-J-1303.

VII. References

Carr, D.B., Olsen, A.R. and White, D. (1992), "Hexagon mosaic maps for display of univariate and bivariate geographical data," ***Cartography and Geographic Information Systems***, 19(4), 228-236 and 271.

Poston, W.L., Holland, O.T., and Nichols, K.R. (1992) "Enhanced Harmogram Analysis Techniques for Extraction of Principal Frequency Components," TR-92/313

Poston, W.L., Wegrnan, E.J., Priebe, C.E., and Solka, J.L. (1995) "A recursive deterministic method for robust estimation of multivariate location and shape," accepted pending revision to the ***Journal of Computational and Graphical Statistics*.**

Poston, W.L. (1995) ***Optimal Subset Selection Methods***, Ph.D. Dissertation, George Mason University: Fairfax, VA.

Priebe, C.E., (1994), "Adaptive Mixtures," ***Journal of the American Statistical Association***, 89, 796-806.

Priebe, C.E., Solka, J.L., Lorey, R.A., Rogers, G.W., Poston, W.L., Kallergi, M., Qian, W., Clarke, L.P., and Clark, R.A. (1994), "The application of fractal analysis to mammographic tissue classification," ***Cancer Letters***, 77, 183-189.

Priebe, C. E., Solka, J. L., and Rogers, G. W. (1993), "Discriminant analysis in aerial images using fractal based features," ***Adaptive and Learning Systems II***, F. A. Sadjadi, Ed., ***Proc. SPIE 1962***, 196-208.

Priebe, C.E. and Marchette, D.J. (1991) "Adaptive mixtures: recursive nonparametric pattern recognition," ***Pattern Recognition***, 24(12), pp. 1196-1209.

Scott, D. W. (1992), ***Multivariate Density Estimation***, John Wiley and Sons: New York.

Solka, J.L., Poston, W.L., and Wegman, E.J. (1995) "A visualization technique for studying the iterative estimation of mixture densities," ***Journal of Computational and Graphical Statistics***, 4(3), 180-198.

Solka, J.L. (1995) ***Matching Model Information Content to Data Information*** , Ph.D. Dissertation, George Mason University: Fairfax, VA.

Solka, J. L., Priebe, C. E., and Rogers, G. W. (1992), "An initial assessment of discriminant sur

face complexity for power law features," ***Simulation***, 58(5), 311-318.

Wegman, E.J. (1990) "Hyperdimensional data analysis using parallel coordinates," ***Journal of the American Statistical Association***, 85, 664-675.

Wegman, E.J. (1995) "Huge data sets and the frontiers of computational feasibility," to appear ***Journal of Computational and Graphical Statistics*** , December, 1995.

Massive Data Sets Workshop: The Morning After

Peter J. Huber
Universität Bayreuth

ABSTRACT.

Some issues crucial for the analysis of massive data sets are identified: computational complexity, data management questions, heterogeneity of data, customized systems, and some suggestions are offered on how to confront the challenges inherent in those issues.

1 INTRODUCTION.

This paper collects some of my observations at, reactions to, and conclusions from, the workshop on Massive Data Sets in Washington-D.C., July 7-8, 1995.

At the workshop, we had not gotten as far as I had hoped. We had discussed long wish-lists, but had not winnowed them down to a list of challenges. While some position papers had discussed specific bottlenecks, or had recounted actual experiences with methods that worked, and things one would have liked to do but couldn't, those examples had not been elaborated upon and inserted into a coherent framework. In particular, the discussions in the Small Groups barely had scratched the implications of the fact that massive sets differ from smaller ones not only by size. Maybe an additional day, providing more time for thinking and for informal contacts and discussions, would have been beneficial.

I shall try to continue the discussion of some of the points we left unfinished and connect some of the open ends.

2 DISCLOSURE: PERSONAL EXPERIENCES.

Clearly, the personal viewpoints of the workshop participants were heavily influenced by the data sets they had worked with. We somehow resembled the proverbial group of blind men confronted with an elephant. This makes it mandatory to disclose the data that have shaped one's views. In my case these were: children's growth data, census data, air traffic radar data, environmental data, hospital data, marketing research data, road quality data, agricultural and meteorological data, with sizes ranging from 3 Mbytes to 2 Gbytes. Most data sets were observational, a few were opportunistic; there were no imaging data. The census data were an outlier in several respects. I shall later cite specific examples for illustrative purposes. Perhaps the most important thing I have learned from these

experiences was: even though the data sources and the analysis goals at first blush seemed disparate, the analyses almost invariably converged or are expected to converge toward a sometimes rudimentary, sometimes elaborate, customized data analysis system adapted to a particular data set. The reason of course is that in the case of large data sets many people will have to work for an extended period of time with the same or similar data.

3 WHAT IS MASSIVE? A CLASSIFICATION OF SIZE.

A thing is massive, if it is too heavy to be moved easily. We may call a data set massive, if its mere size causes aggravation. Of course, any such a characterization is subjective and depends on the task, one's skills, and on the available computing resources.

In my position paper (Huber 1994b), I had proposed a crude objective classification of data by size, from tiny (10^2 bytes), small (10^4), medium (10^6), large (10^8), huge (10^{10}) to monster (10^{12}). The step size 100 is large enough to turn quantitative differences into qualitative ones: specific tasks begin to hurt at well defined steps of the ladder. Whether monster sets should be regarded as legitimate objects of data analysis is debatable (at first, I had deliberately omitted the "monster" category, then Ed Wegman added it under the name "ridiculous"). Ralph Kahn's description of the Earth Observing System however furnishes a good argument in favor of planning for data analysis (rather than mere data processing) of monster sets.

Data analysis goes beyond data processing and ranges from data analysis in the strict sense (non-automated, requiring human judgment based on information contained in the data, and therefore done in interactive mode, if feasible) to mere data processing (automated, not requiring such judgment). The boundary line is blurred, parts of the judgmental analysis may later be turned into unsupervised preparation of the data for analysis, that is, into data processing. For example, most of the tasks described by Bill Eddy in connection with fNMR imaging must be classified as data processing.

- With regard to visualization, one runs into problems just above medium sets.
- With regard to data analysis, a definite frontier of aggravation is located just above large sets, where interactive work breaks down, and where there are too many subsets to step through for exhaustive visualization.
- With regard to mere data processing, the frontiers of aggravation are less well defined, processing times in batch mode are much more elastic than in interactive mode. Some simple standard data base management tasks with computational complexity $O(n)$ or $O(n \log(n))$ remain feasible beyond terabyte monster sets, while others (e.g. clustering) blow up already near large sets.

4 OBSTACLES TO SCALING.

By now, we have very considerable experience with data analysis of small and medium sets. Present-day PCs are excellently matched to the requirements of interactive analysis of medium sets-if one tries to go beyond, one hits several bottlenecks all at once. The problem is to scale our tools up to larger sets. Some hard obstacles to scaling are caused by human limitations, by computational complexity or by technological limits. Others are financial (e.g. memory costs) or lack of software (for massive parallelism).

4.1 Human limitations: visualization.

Direct visualization of a whole data set through scatterplots and scatterplot matrices is feasible without significant information loss up to about medium size. For larger sets one must either step through subsets or show summaries (e.g. density estimates). This limitation has more to do with the resolution of the human visual system than with display hardware. Ed Wegman's position paper elaborates on this.

This limitation holds across the board, also for imaging data: an efficiently coded high-resolution picture comprises at most a few megabytes.

4.2 Human-machine interactions.

Necessary prerequisites for interactivity are: the task is such that a sequence of reasonably straightforward decisions have to be made in relatively quick succession, each of them based on the results of the preceding step. All three prerequisites can be violated for large sets: the decisions may be not straightforward because of data complexity, the response may be too slow (the human side of the feedback loop is broken if response time exceeds the order of human think time, with the latter depending on the task under consideration), and it may be difficult to provide a rational basis for the next decision if one cannot visualize the preceding results.

In interactive work, the timing requirements are stringent: For high-interaction graphics the response time must be a fraction of a second, for most other tasks of interactive data analysis it can be a few seconds, but it may exceed 10-20 seconds only very rarely.

4.3 Storage requirements.

According to experiences with medium to large sets on PCs and workstations, disk size nowadays rarely is a problem, but memory size frequently is a bottleneck preventing full use of fast processors.

Backup storage (disk) must be large enough to hold the raw data plus several derived sets. For comfortable work it ought to provide space for the equivalent of about 10 copies of the raw data set.

On a single-processor machine with fiat high-speed memory, the latter must be large enough to hold 4 copies of the largest array one intends to work with (otherwise one runs into severe swapping problems with two-argument array operations such as matrix multiplications); for comfortable work, it ought to be at least twice as large. Thus, in order to run the processor at full speed, one sometimes may need almost as much free memory as free disk space.

4.4 Computational complexity.

Processor speed does not scale well, since computational complexity tends to increase faster than linearly with data size.

The position papers of Huber (1994b) and Wegman contain extensive discussions of computational complexity issues. so it suffices to sketch the salient points.

At present, batch tasks involving a total of 10^{12} floating point operations (flops) are easy to handle on PCs, and 10^{15} flops are just about feasible on supercomputers (this corresponds to 1 Gflop per second, sustained for two weeks). If performance continues to double every two years as in the past, we may reach 10^{18} operations in 20 years or so. This argument ignores all questions of data flow, whose just-in-time management may become a much worse bottleneck, considering that light moves a mere 0.3 mm in 10^{-12} seconds.

Even so, it follows that we can safely forget about so-called computer-intensive methods. For large sets and beyond, tasks with a computational complexity of $O(n^2)$ are not feasible as batch jobs on any machine now. The practical limit is around $O(n^{3/2})$ operations-up to huge sets now, up to monster sets in 20 years.

A simple order-of-magnitude calculation shows that computer-intensive operations on the whole of a huge sets are infeasible even when the data has a coarse-grained structure and the intensive operations are restricted to one grain at a time, unless those grains are small. For the sake of the argument let us regard a total of $n^{3/2}$ operations as feasible in the huge to monster range. If there are $n^{1-}\alpha$ grains with a size $m = n\alpha$ each, and if a calculation with m^3 operations is to be performed on each grain, we reach the feasible total of $n^{3/2}$ operations with $\alpha = 0.25$. For $n = 10^{10}$ and $n = 10^{12}$, this corresponds to grain sizes of merely 316 and 1000, respectively.

If the data is arranged in the form of a matrix with r rows and c columns, $n = rc$, with $r > \sqrt{n} > c$, then the same kind of argument shows that tasks with complexity $O(nc)$, such as multiple regression, singular value decomposition and multiplication with a c-by-c matrix on the right, all are feasible, while clustering algorithms, with complexity $O(nr)$, are out.

5 ON THE STRUCTURE OF LARGE DATA SETS.

Larger data sets tend to be structurally different from smaller ones. They are not just more of the same, they are larger because they have to be larger. In particular, they are, as a rule, much more heterogeneous.

5.1 Types of data.

Data can be experimental (from a designed experiment), observational (with little or no control of the process generating the data), or opportunistic (the data have been collected for an unrelated purpose). Massive data sets rarely belong to the first category, since by a clever design the data flow often can be reduced already before it is recorded (of course there are exceptions, e.g., computational fluid dynamics). But they often belong to the third category for plain reasons of economy.

Sometimes, data sets are massive because their collection is mandated by law (e.g. census and certain health data), or because they are collected anyway for other purposes (e.g. financial data). Often, however, they have to be massive because smaller sets will not do, and the predominant reason why they will not do is that the data in question are intrinsically heterogeneous. In particular, there may be many observers and many observed objects, both being located in space and time (e.g. aircraft traffic radar data).

I wonder whether the onslaught of massive data sets will finally force us to acknowledge and heed some studiously ignored. but long-standing admonitions going back to Deming (1940), and reiterated by Tukey (1962), to wit: The statistical protession as a whole is paying much too little attention to the need for dealing with heterogeneous data and with data that arise from conditions not in statistical control (randomness).

5.2 How do data sets grow?

If we think in terms of a hierarchical data organization, data sets may grow by acquiring

- more hierarchical layers, or
- more branches, or
- bigger leaves,

or all of the above. For example, some data sets are extremely large because each single leaf is an image comprising several megabytes.

It must be stressed that actual data sets very often either do not possess a tree structure, or else several conflicting ones. Instead of "leaf", the more neutral terms "case" or "grain" might therefore be more appropriate.

5.3 On data organization.

Statistical data bases often have a tree structure imposed on them through sampling or data collection (e.g. census districts-housing blocks-households-persons). But there may be several simultaneous conflicting tree structures (e.g. households and employers). Different priority orderings of categorical variables define different tree structures. For very large sets, a clean tree structure is rather the exception than the rule. In particular, those sets often are composed during the analysis from several, originally unrelated sources (for example health data and environmental data, collected independently for different purposes), that are linked as an afterthought through common external (here: geographical) references. In

our work with some opportunistic data sets, we found that this kind of associative joining of originally unrelated data sets was one of the most important operations. Moreover, the larger the data set is, the more important are subsetting operations, and also these cut across hierarchies or establish new ones.

No single logical structure fits all purposes. In our experience, the flat format traditional in statistics usually turned out to be most expedient: the data are organized as a family of loosely linked matrices, each row corresponding to a "case", with a fixed number of columns, each column or "variable" being of a homogeneous type.

Sometimes, an even simpler linear organization is preferable: a very long unstructured sequence of items, each item consisting of a single observation together with the circumstances under which it was made (who observed whom, which variable, when, where, and the like). From that basis, the interesting parts are extracted as required and restructured into matrix form.

How such a logical organization should be implemented physically is of course an entirely different question. The problem with massive data is to distribute not only the data, but also ancillary materials and retrieval tools over a hierarchy of storage devices, so that the ad hoc retrieval and reorganization tasks to be encountered in the course of a data analysis can be performed efficiently. For example, when should one work with pointers, when with copies?

5.4 Derived data sets.

It has been said that data analysis is a progress through sequences of derived data sets. We can distinguish between at least four levels of derived data sets:

- raw data set: rarely accessed, never modified,
- base data set: frequently accessed, rarely modified,
- low level derived sets: semi-permanent,
- high level derived sets: transient.

The base set is a cleaned and reorganized version of the raw set, streamlined for fast access and easy handling. The base set and low level derived sets ordinarily must be maintained on some mass storage device for reasons of space. Their preparation may involve sophisticated and complex, time-consuming data processing. The highest level derived sets almost by definition must fit into high speed memory for reasons of computational efficiency. The actual sizes and details of organization clearly will be governed by the available hardware and software. To fix the idea: on presently available super-workstations (100 Mflops, 1 Gbyte memory) with almost-present-day software one may just be able to handle a huge raw data set (10 Gbytes). In this case, high level derived sets might comprise about 100 Mbytes of data each for non-graphical tasks, but at most a few Mbytes for tasks involving highly interactive visualization. With massively parallel hardware some of these figures can be pushed higher, but adequate software does not yet exist. Here we have a definite software challenge: to design and implement a pilot system for general purpose data analysis of massive data sets on massively parallel hardware. More comments on this are contained in Section 10.

Derived sets can be formed in various ways. In our experience, low level derived sets mostly are created by application specific preprocessing, or by subsetting (more about this in Sections 8 and 9). Summaries are problematic with large sets-one ought not to group or summarize across heterogeneity-and splitting into homogeneous parts may be an overly expensive clustering problem. Thus, one will be restricted in practice to splitting based on external a priori information or, if data based, to CART-like single-variable methods. Afterwards, summaries of the homogeneous parts then may be recombined into new derived sets.

6 DATA BASE MANAGEMENT AND RELATED ISSUES.

With larger data sets, data base management operations become ever more important.

6.1 Data base management and data analysis systems.

In view of the importance and central role of data base operations, it has been suggested that future data analysis (DA) systems should be built around a data base management (DBM) kernel. But paradoxically, all the usual DBM systems do a very poor job with large statistical data bases. For an explanation why this is so, see French (1995), who confronts the design goals of the ordinary DBM systems with those of decision support systems (DSS). Data analysis needs all facilities of a DSS, but more flexibility, in particular read-write symmetry to assist with the creation and manipulation of derived sets. As a consequence, the designer of a DA system must perforce also design and implement his or her own DBM subsystem.

6.2 Problems and challenges in the data base area.

Data base type operations get both harder and more important with larger sets, and they are used more frequently. With small and medium sets, where everything fits into high speed memory with room to spare and where all tasks are easily handled by a single processor, one does not even realize when one is performing a data base operation on the side. Larger sets may have to be spread over three or four hierarchical storage levels, each level possibly being split into several branches. Parallel processors and distributed memory create additional complications.

With large sets, processing time problems have to do more with storage access than with processor speed. To counteract that, one will have to produce small, possibly distributed, derived sets that selectively contain the required information and can be accessed quickly, rather than to work with pointers to the original, larger sets, even if this increases the total required storage space and creates tricky problems with keeping data integrity (e.g. with carrying back and expanding to a superset some changes one has made in a subset).

To get a good grip on those problems, we must identify, categorize and rank the tasks we actually perform now with moderately sized sets. We then must identify specific tasks that become harder, or more important, or both, with massive data sets, or with distributed processors and memory. In any case, one will need general, efficient subset operations that can operate on potentially very large base sets sitting on relatively slow storage devices.

7 THE STAGES OF AN DATA ANALYSIS.

Most data analysis is done by non-statisticians, and there is much commonality hidden behind a diversity of languages. Rather than to try to squeeze the analysis into a too narrow view of what statistics is all about, statisticians ought to take advantage of the situation, get involved interdisciplinarily, learn from the experience, expand their own mind, and thereby their field, and act as catalysts for the dissemination of insights and methodologies. Moreover, the larger the data sets are, the more important the general science aspects of the analysis seem to become relative to the "statistical" aspects.

I believe that some of the discussions at the workshop have become derailed precisely because they were too much concerned with categories defined in terms of classical statistical concepts. In retrospect, it seems to me that it might have been more profitable to structure the discussions according to stages common to most data analyses and to watch out for problems that become more pronounced with more massive data.

At the risk of belaboring the obvious, I am providing a kind of commented check-list on steps to be watched.

7.1 Planning the data collection.

Very often, the data is already there, and one cannot influence its collection and its documentation any more.

The planning of a large scale data collection runs into problems known from Big Science projects: many different people are involved over several years in a kind of relay race. By the time the data are ready to be analyzed, the original designers of the experiment have left or are no longer interested, and the original goals may have been modified beyond recognition.

The obvious conclusion is that big scale data collection must be planned with am open mind for unforeseen modes of use. Beware of obsolescence. The documentation must be complete and self-sufficient -10 years later, technical specifications of the measuring equipment may be lost, and names of geographical locations and the scope of ZIP code numbers may have changed. Even the equipment to read the original media may be gone.

If one is planning to collect massive data, one should never forget to reserve a certain percentage of the total budget for data analysis and for data presentation.

7.2 Actual collection.

It is not possible to plan and specify correctly all details ahead. In particular, minor but crucial changes in the coding of the data often remain undocumented and must afterwards be reconstructed through painstaking detective work. Whoever is responsible for collecting the data must also be held responsible for documenting changes to the code book and keeping it up-to-date.

Everybody seems to be aware of the need for quality control, in particular with regard to instrument drift and the need for continuous calibration. There is much less awareness that also the quality of hardware, software and firmware of the recording system must be closely watched. I have personally encountered at least two unrelated instances where leading bits were lost due to integer overflow, in one case because the subject matter scientist had underestimated the range of a variable, in the other case because a programmer had overlooked that short integers do not suffice to count the seconds in a day. I also remember a case of unusable data summaries calculated on-line by the recording apparatus (we noticed the programming error only because the maximum occasionally fell below the average).

7.3 Data access.

As data analysts we need tools to read raw data in arbitrary and possibly weird binary formats. An example was given by Allen McIntosh in connection with Telephone Network Data. Not only the actual reading must be efficient, but also the ad hoc programming of data input must be straightforward and easy; we have repeatedly run into similar problems and have found that very often we were hunting a moving target of changing data formats.

In addition, we must be able to write data in similarly weird formats, in order that we can force heterogeneous sets into a homogeneous form.

7.4 Initial data checking.

Usually, the problem is viewed as one of legality and plausibility controls. What is outside of the plausible range is turned into missing values by the checking routine. This is a well-tested, successful recipe for overlooking obvious, unexpected features, such as the ozone hole.

The real problem of data checking has to do with finding systematic errors in the data collection. and this is much harder! For example. how does one find accidental omissions or duplications of entire batches of data? The "linear" data organization mentioned in Section 5.3 facilitates such checks.

7.5 Data analysis proper.

A person analyzing data alternates in no particular sequence between the following five types of activities:

- Inspection,
- Modification,
- Comparison,
- Modelling,
- Interpretation.

With massive data sets, both the inspection and the comparison parts run into problems with visualization. Interpretation is thinking in models. Models are the domain of subject matter specialists, not of statisticians; not all models are stochastic! Therefore, modelling is one of the areas least amenable to a unified treatment and thus poses some special challenges with regard to its integration into general purpose data analysis software through export and import of derived sets.

7.6 The final product: presentation of arguments and conclusions.

With massive data, also the results of an analysis are likely to be massive. Jim Hodges takes the final product of an analysis to be an argument. I like this idea, but regard it a gross oversimplification: in the case of massive data we are dealing not with a single argument, but with a massive plural of arguments. For example with marketing data, a few hundred persons may be interested in specific arguments about their own part of the world, and once they become interested also in comparisons ("How is my product X doing in comparison to product Y of my competitor?"), complexity grows out of hand. However, there is a distinction between potential and actual: from a near infinity of potential arguments, only a much smaller, but unpredictable, selection will ever be actually used.

With massive data, the number of potential arguments is too large for the traditional pre-canned presentation in the form of a report. One rather must prepare a true decision support system, that is a customized, special-purpose data analysis system sitting on top of a suitable derived data set that is able to produce and present those arguments that the end user will need as a basis for his or her conclusions and decisions. If such a system does a significantly better job than, say, a 1 000-page report, everybody will be happy; this is a modest goal.

8 EXAMPLES AND SOME THOUGHTS ON STRATEGY.

By now, we have ample experience with the analysis of medium size data sets (data in the low megabyte range), and we begin to feel reasonably comfortable with large sets (10^8 bytes, or 100 megabytes), even though direct visualization of larger than medium sets in their entirety is an unsolved (and possibly unsolvable) problem. Let us postulate for the sake of the argument-somewhat optimistically-that we know how to deal with large sets.

Assume you are confronted with a huge data set (10^{10} bytes, or 10 gigabytes).

If a meaningful analysis is possible with a 1% random subsample, the problem is solved-we are back to large sets. Except for validation and confirmation, we might not even need the other 99%.

Assume therefore that random samples do not work for the problem under consideration. They may not work for one of several possible reasons: either because the data are very inhomogeneous, or because they are highly structured, or because one is looking for rare events, or any combination of the above. Density estimates or summary statistics then will not work either.

Example: Air traffic radar data. A typical situation is: some 6 radar stations observe several hundred aircraft, producing a 64-byte record per radar per aircraft per antenna turn, approximately 50 megabytes per hour. If one is to investigate a near collision, one extracts a subset, defined a by window in space and time surrounding the critical event. If one is to investigate reliability and accuracy of radars under real-life air traffic conditions, one must differentiate between gross errors and random measurement errors. Outlier detection and interpretation is highly non-trivial to begin with. Essentially, one must first connect thousands of dots to individual flight paths (technically, this amounts to tricky prediction and identification problems). The remaining dots are outliers, which then must be sorted out and identified according to their likely causes (a swarm of birds, a misrecorded range measurement, etc. etc.). In order to assess the measurement accuracy, one must compare individual measurements of single radars to flight paths determined from all radars, interpolated for that particular moment of time. Summary statistics do not enter at all, except at the very end, when the results are summarized for presentation.

I believe this example is typical: the analysis of large sets either begins with task and subject matter specific, complex preprocessing, or by extracting systematic subsets on the basis of a *priori* considerations, or a combination of the two. Summaries enter only later. Often, the subsets will be defined by windows in space and time. Even more often, the selection has two stages: locate remarkable features by searching for exceptional values of certain variables, then extract all data in the immediate neighborhoods of such features. For a non-trivial example of preprocessing, compare Ralph Kahn's description of the Earth Observing System and the construction of several layers of derived data sets. For one beginning with subsetting, see Eric Lander's description of how a geneticist will find the genes responsible for a particular disease: in a first step, the location in the human genome (which is a huge data set, 3×10^9 base pairs) is narrowed down by a factor 1 000 by a technique called genetic mapping (Lander 1995).

After the preparatory steps, one may want to look up additional information in other data bases, possibly from informal external sources:

Example: Environmental data. We found (through EDA of a large environmental data set) that very high radon levels were tightly localized and occurred in houses sitting on the locations of old mine shafts.

In this example, indiscriminate grouping would have hidden the problem and would have made it impossible to investigate causes and necessary remedies. The issue here is one of "data mining", not one of looking, like a traditional statistician, "for a central tendency, a measure of variability, measures of pairwise association between a number of variables". Random samples would have been useless, too: either one would have missed the exceptional values altogether, or one would have thrown them out as outliers.

Data analysis is detective work. The metaphor is trite but accurate. A careful distinction between tasks requiring the acumen of a first rate sleuth and tasks involving mere routine work is required. After perusing some of the literature on data mining, I have begun to wonder: too much emphasis is put on futile attempts to automate non-routine tasks, and not enough effort is spent on facilitating routine work.

In particular, everybody would like to identify noteworthy, but otherwise unspecified features by machine. From my experience with projection pursuit on small to medium sets I think this is a hopeless search for the holy Grail (computational complexity grows too fast with dimensionality). Pattern discovery is intrinsically harder than pattern recognition. A less ambitious, still hard task is the approximate match-up problem: find all structures in data set A that are approximately similar to some structure in data set B, where A and B are large. It is not at all clear whether even such problems can be solved within the desired $O(n^{3/2})$-complexity.

9 VOLUME REDUCTION.

Volume reduction through data compression (with information loss) sometimes is advocated as a kind of panacea against data glut: "keep only what is exceptional, and summarize the rest". At least in the case of observational, as against experimental data, I think this is a daydream, possibly running counter to several of the reasons why the data are being collected in the first place! It is only a slight exaggeration to claim that observational data deserve to be saved either completely or else not at all. For example, a survey of the sky must be preserved completely if one later wants to check the early stages of supernovae.

But prior to *specific* analyses, targeted volume reduction usually can be performed on a data matrix either by reducing the number of rows (cases) or the number of columns (variables), or both. This might be done based on a *priori* interest. Data-driven reduction might be done for example by aggregation (i.e. by combining similar rows) or by summarizing (e.g. by forming means and variances over a homogeneous set of rows). Reducing the number of variables is usually called "dimension reduction" and can be done for example by variable selection (pick one of several highly correlated columns), or by forming (linear or non-linear) combinations of variables.

But it is difficult to put the general notion of dimension reduction on a sound theoretical basis; exploratory projection pursuit comes closest, but as its computational complexity increases exponentially with dimension, it is not well suited to massive data sets.

The next best general approach is dimension reduction through principal component or correspondence analysis (i.e. the truncated singular value decomposition): remember that the k leading singular values yield the best approximation (in the square norm sense) to the data matrix by a matrix of rank k. According to Sue Dumais this was surprisingly successful in the context of information retrieval even with quite large data matrices.

However, the most powerful type of dimension reduction is through fitting local models:

Example: Children's growth data. Several hundred children were observed periodically from birth to adulthood. Among other things. for each child, 36 measurements of body length were available. It was possible to reduce dimension from 36 to 6, by fitting a 6-parameter growth curve to each child. The functional form of that curve had several components and had been found by estimating some 30 global parameters from the total available population of children. Most of the 6 parameters specific to a particular child had an intuitive interpretation (age at puberty, duration and intensity of pubertal growth spurt, etc.); the residual error of the fit was only slightly larger than the intrinsic variability of the measurements.

Local model fitting is computationally expensive, but typically, it seems to stay within the critical $O(n^{3/2})$-limit.

10 SUPERCOMPUTERS AND SOFTWARE CHALLENGES.

On presently available super-workstations (100 Mflops, 1 Gbyte memory) one can certainly handle large sets with almost-present-day software, and one may just barely be able to handle huge sets (10 Gbytes). To push those figures higher, one would have to invest in massively parallel supercomputers and novel software. Is such an investment worthwhile? What are its chances to succeed? I do not have final answers, but would like to offer some food for thought.

10.1 When do we need a Concorde?

I believe the perceived (or claimed) need for supercomputers exceeds the real need.

The problem is that of the Concorde: flying the Concorde is a status symbol, but if too much time is spent on the ground, the fast flight is not worth the money. Response times in fractions of a second are neither needed nor appreciated, if it takes several minutes, and possibly hours, to think up a question and to digest the answer.

We must learn to identify and differentiate between situations where supercomputers are necessary or at least truly advantageous, and situations where they are not.

We have encountered several examples with raw data sets in the 10-900 MByte range, where it had been claimed that a large mainframe or supercomputer plus several months of customized programming were needed in order to accomplish certain data analytic tasks. In all those cases we found that a well endowed PC, plus a a few weeks of customized programming in a high-level interpreted data analysis language (ISP) would be adequate, with comparable or even shorter execution times, at total costs that were orders of magnitude lower.

10.2 General Purpose Data Analysis and Supercomputers.

Can general purpose data analysis take advantage of supercomputers? Dongarra's famous benchmark comparison (I have the version of April 13, 1995 in front of me) highlights the crux of the problem: without special hand-tuning efforts to improve and distribute the data flow, the fastest multi-processor supercomputers beat the fastest single-processor superworkstations merely by a factor 4. which is not worth the money. Tuning may yield another factor 20 or so. We need to discuss strategies for recovering that factor 20. Ad hoc code tweaking on a case by case basis is so labor intensive and error-prone that ordinarily it will be out of the question. That is, we have a very serious software challenge.

Our experiences with ISP suggest how a speed-up could be done in a general purpose system. ISP is a small, general purpose, array oriented, interpretive data analysis language somewhat similar to S. It contains a core of 100-200 building blocks (commands or functions). Explicit, interpreted looping is slow, but it is rarely needed. Efficiency relative to compiled code increases with data size because of array orientedness. I believe that many of the building blocks can be beefed up for parallelism, but there may be snags. For example, reorganization and redistribution of data between subtasks might be more expensive than anticipated.

The proof of the pudding is in the eating, i.e. we need to conduct the experiment suggested by Huber (1994b, at the end of section 8) and build a working pilot system. To prove the point we should aim for a small, relatively unspecific, but universal system. For a basis, I would choose ISP over S precisely because it has these three properties. But we should take the opportunity to build a new, better system from scratch, rather than trying to port an existing system. From past experiences I estimate that it will take three years before the pilot system for such an experiment attains sufficient maturity for beta testing. We better begin soon.

The next section summarizes the issues and experiences I consider important for such an undertaking.

10.3 Languages, Programming Environments and Data-Based Prototyping.

Here is an account of some things we have learned from our ISP experience (see also Huber (1994a)). The traditional programming languages (Fortran, C, Pascal, ...) are too low-level for the purposes of data analysis. We learned that already back in the 1970's: there is much re-programming and re-use under just slightly different circumstances, and for that, these languages are too clumsy and too error-prone (cf. also the comments by Allen McIntosh). Subroutine libraries like LINPACK help, but are not enough. We need a **high-level array-oriented language** on top, with a simple syntax and safe semantics, whose units or building blocks must be very carefully selected for universal use. The language must be user-extensible through combination of those units into new building blocks with the same syntax. The core building blocks must be highly optimized.

In the 1980's we became aware of the need for **programming environments** in the style of Smalltalk and LISP machines. In data analysis, you never hit it right the first, or the second, or even the third time around, and it must be possible to play interactively with modifications, but without having to start everything from scratch. Rather than building a system on top of Smalltalk or LISP, we decided to augment our data analysis language ISP so that it acquired its own programming environment.

Around 1990, we realized that we had to go even further into what we call data-based prototyping: build a customized data analysis system while actually doing production work with the data. The basic problem is that the user (whether it is a customer or we ourselves) never is able to specify the requirements in advance. Our solution is to mock up a rough-and-ready working prototype, and let the user work with it on his or her actual data. Without involving the actual user early and actively in the use and re-design of the system, in particular in issues of presentation of results (what to show and how), it is extremely hard to arrive at a satisfactory solution. Some comments made by Schmitz and Schoenherr in the context of marketing databases nicely illustrate the difficulty. A high-level language and a good programming environment are indispensable prerequisites for data-based prototyping.

None of the existing languages and systems is entirely satisfactory. After seeing Carr's list of preferences in Section 4 of his position paper, it seems to me that our own software (ISP) comes closer to an ideal, universal system than I ever would have suspected. It does a job superior to Matlab in the area of visualization; we use it instead of GIS because the latter systems have difficulties with discontinuous values that are attached to arbitrary points rather than to grid points or political districts; after becoming dissatisfied with all available general data base software, we began to improvise our own approaches in ISP; we prefer it to SAS and S for data analysis, especially with large sets. Carr's comment on the "diminished influence of the statistical community upon my work" is reflected by the fact that in ISP we never felt compelled to go beyond a frugal minimum of statistical functions.

11 SUMMARY OF CONCLUSIONS.

- With the analysis of massive data sets, one has to expect extensive, application-and task-specific preprocessing. We need tools for efficient ad hoc programming.
- It is necessary to provide a high-level data analysis language, a programming environment and facilities for data-based prototyping.
- Subset manipulation and other data base operations, in particular the linking of originally unrelated data sets, are very important. We need a data base management system with characteristics rather different from those of a traditional DBMS.
- The need for summaries arises not at the beginning, but toward the end of the analysis.
- Individual massive data sets require customized data analysis systems tailored specifically toward them, first for the analysis, and then for the presentation of results.
- Pay attention to heterogeneity in the data.
- Pay attention to computational complexity; keep it below $O(n^{3/2})$, or forget about the algorithm.
- The main software challenge: we should build a pilot data analysis system working according to the above principles on massively parallel machines.

12 References.

Deming, W. E. (1940). Discussion of Professor Hotelling's Paper. Ann. Math. Statist. 11 470-471.

French, C. D. (1995). "One Size Fits All" Database Architectures Do Not Work For DSS. SIGMOD RECORD, Vol. 24, June 1995. Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data. ACM Press.

Huber, P. J. (1994a). Languages for Statistics and Data Analysis. In: Computational Statistics, P. Dirschedl and R. Ostermann (Eds.), Physica-Verlag, Heidelberg.

Huber, P. J. (1994b). Huge Data Sets. In: Proceedings of the 1994 COMPSTAT Meeting, R. Dutter and W. Grossmann (Eds.), Physica-Verlag, Heidelberg.

Lander, E. S. (1995). Mapping heredity: Using probabilistic models and algorithms to map genes and genomes, Notices of the AMS, July 1995, 747-753. Adapted from: Calculating the Secrets of Life. National Academy Press, Washington, D.C. 1995.

Tukey, J. W. (1962). The Future of Data Analysis. Ann. Math. Statist. 33 1-67.

PART IV

FUNDAMENTAL ISSUES AND GRAND CHALLENGES

Panel Discussion

Moderator: James Hodges
University of Minnesota

My name is Jim Hodges. First, I would like to introduce our panelists: Art Dempster, Harvard University; Usama Fayyad, Jet Propulsion Laboratory; Dan Carr, George Mason University; Peter Huber, Universität Bayreuth; David Madigan, University of Washington; Mike Jordan, Massachusetts Institute of Technology; and Luke Tierney, University of Minnesota. Last night at dinner we test marketed a group of questions and we had a contention meter at the table. The four questions with the highest scores are the four questions we are going to discuss, so you will see an argument. I am going to start off by giving the four questions.

The first question: What is a theory, and what are theories good for? One of the things that a theory does is to carve the world into categories. And statistical theory, at least as it is taught in schools and as it is propounded in journals, carves the statistical world into categories labeled estimation and hypothesis testing, and, to a lesser extent, model selection and prediction.

This is not a very good characterization of what we do. For one thing, where is model identification? We do it all the time, but we do not really know how to interpret statistical inference in the presence of model identification. Where in this list of categories is predictive validation?

I would also say that this standard litany of hypothesis testing, model selection, and so on is not a helpful characterization of what we need to do in particular problems. It does not characterize problems in terms of the burden of proof in any given problem. I am responsible for the line on group B's slide about how a statistician's product is an argument, and there are qualitatively different arguments which I don't want to get into now. But each comes with its distinctive burdens of proof. I would argue that that would be a start on a more useful categorization of statistical problems. But I will pose this question to the panel: What is a more useful categorization than estimation and hypothesis testing and so on, for statistical problems?

I think we also need a second new categorization, which is categorizing problems in terms of data structures that are peculiar to them. I have this queer faith that there really is only a short list of truly different kinds of data structures out there in the world, five or six, maybe, and if we could figure out what those are, we would be a long way toward developing fairly generic software.

The second question changes direction somewhat. Before Fisher, statistics was very data-based. We looked at tables; we looked at a lot of pictures. There was not a lot of sophisticated machinery. Fisher is probably responsible, more than anyone else, for making statistics a model-based endeavor, perhaps too much so. One might get the impression from the workshop talks that the advent of massive data sets means that we are going to be coming back around again to being a more data-driven discipline. But you might alternatively say that we have just identified the need for richer classes of models, or a more flexible vocabulary of structures. Or as Peter Huber put it last night, How does your data grow?

The third question: A lot of computer-intensive techniques like Monte Carlo Markov chain [MCMC], bootstrap, or jackknife have been developed for medium-sized data sets. Can those methods coexist with massive data sets?

The fourth question: In the usual sort of statistical theory, the idea of sufficiency makes life simple, because it reduces the amount of data that you have to worry about. So in a standard

sort of an i.i.d. normal model, all you have to worry about are a couple of sufficient statistics. But sufficiency is dependent on the model in the usual sort of theory. Or you can ask the question, Can we define a notion or an analog to sufficiency that does not depend on a specific model as a means of compressing data without loss of information?

Those are the four questions. We will discuss question 1 first.

QUESTION 1: WHAT ARE SOME ALTERNATIVES FOR CLASSIFYING STATISTICAL PROBLEMS?

Peter Huber: I think question 1 is not connected with massive data sets. It is much older. As a demonstration of this, I would like to put up a slide from the *Annals of Mathematical Statistics*, 1940. It is about the teaching of statistics, a very short discussion of Hotelling by Deming. Deming essentially says in slightly different words what was said as question 1, namely, that the statisticians coming out of the schools somehow try to fit all the problems into modem statistical techniques, the so-called model series of estimation, and so on. So first of all, this is not exactly new.

James Hodges: But perhaps it is no less compelling.

Huber: I think if you look at it carefully, Deming pointed out the sample defect in the Hotelling scheme for the teaching of statistics.

Usama Fayyad: Here is my view of what you have to do to find models of data, and this is probably not new to any of you. There are three basic things you have to do. First is representation of your models. That is the language, how much complexity you allow, how many degrees of freedom, and so forth. Second is model evaluation or estimation. This is where statistics comes in big-time on only some of the aspects, not all of them. Finally, there is model search, which is where I think statistics has done very little. In other words, statisticians do not really do search in the sense of optimization. They go after closed forms or linear solutions and so forth, which we can achieve, and there is a lot to be said for that from a practical point of view. But that can only buy you so much.

In the middle, in the estimation stage, I see a high value for statistics in measurement of fit and so forth, and parameter selection for your models, once you have fixed the model. There are these notions of novelty or being interesting, accounting for utility. It has not been addressed. In practical situations, that is probably where most of your "bang for the buck" is going to come from, if you can somehow account for those or go after those, and your methods should have models of those dimensions.

Finally, on model representation, I think statistics has stuck to fairly simple, fairly well-analyzed and understood models. With that, they brushed away the problem of search. They do not search over a large set of models to select models to begin with. Computers now, even with massive data sets, allow you to do things like try out many more alternatives than are typically tried. If you have to take one pass at the data, take one pass, but you can evaluate many things in that pass. That is my basic response to how I view question 1, and I am not sure what the answers are.

Arthur Dempster: I agree that the textbooks have the categories wrong. I have a set of categories that I tend to use all the time. I do not have a slide, but imagine that there are five of them. They are a set of interlocking technologies which I think are the basis of applied statistics.

You can go back and forth among them, but you generally start with data structures in some broad sense, meaning the logical way the traits are connected in time and space and so on. I would also include in that the meta data, that is, the tying of the concepts to the science or to the real-world phenomenon. That is the first set of things; you have to learn how to set all of that up, not just for the data, but for the problem. There has to be a problem that is behind it, or a set of problems.

The second category is what is sometimes called survey design, experimental design—data selection and manipulation type issues, what choices have been made there, and things of that sort. If you can control them so much the better.

The third category is exploratory data analysis, summarizing, describing, digging in, and just getting a sense of whatever data you have.

The fourth category is modeling. I am mostly concerned with stochastic modeling, although that includes deterministic models in some special cases.

The fifth category is inference, which I think is primarily the Bayesian inference, at least in my way of thinking, that is to say, reasoning with subjective probabilities for assumptions about what may be going on.

But I think of all of these as forms of reasoning. They are opportunities for reasoning about the specific phenomenon. The idea of the specific phenomenon is the key, because one of the major complaints about statistical theory as formulated and taught in the textbooks is that it is theory about procedures. It is divorced from the specific phenomenon. That is part of the reason that statisticians have so much trouble getting back to the real problems.

There is a second part of the question which asks, Do we need canonical data structures? Yes, of course, and a lot of canonical models, as well. I'll quit there.

Daniel Carr: I actually like that description very much. The only thing I would do is emphasize that we need training to interact with people in other disciplines. There is a big change that has gone on from small science to big science. If we want to be part of big science, that means we must be part of teams, and we need to learn how to work on teams, learn to understand and talk with people from other disciplines.

I have seen lots of models being put out about how big science is done. A model might have a computer scientist, application specialist, and engineer. I do not even see the statistician listed in the models. That concerns me. We have seen some nice examples of teamwork here. I really like Bill Eddy's example, where there is a group of people. That is big-time science, and it is great.

David Madigan: The one comment I would add is, it might also be useful to think of the purposes to which we put our models as being of some relevance.

Sometimes we use models purely as devices to get us from the data to perhaps a predictive distribution or a clustered distribution, or whatever it is we do, in which case the model itself is not of much interest. Sometimes we build models to estimate parameters. We are interested in the size of an effect, and we build models to estimate the size of the effect, with primary interest focused on one parameter.

Then finally, sometimes we build models as models of the process that generated the data. But in my experience, we as statisticians do not do as much of that as we should. In the first case, where the model is purely a means to an end, when dealing with massive data sets we should choose models that are computationally efficient. There is a lot of scope for, in particular, graphic models in that context, because you can do the computations very efficiently.

But at the other end of the scale where you are trying to build models of the process that generated the data, I am not sure. I haven't a clue how we are going to build those kinds of models

using massive data. The causative modeling folks are very excited about the possibilities of inferring causation from data sets, and the bigger they are, the better. But in practice, I do not know if we are going to be able to do it or not.

Michael Jordan: My experience has been in the neural net community. The size of data sets that are usually studied there are what we used to think of as large. You had to have at least 10,000 data points, and 100,000 was quite impressive. That is nothing here, but on the other hand, there has been some success—I do not want to overemphasize it—in that community with dealing with that size data set, and there are a few lessons that can be learned.

One of them was that the neural net people immediately started to rediscover a great deal of statistics and brought it on board very quickly. That has now happened completely. If you go to a neural net conference you will feel quite at home. All the same terminology, all the way up to the most advanced Bayesian terminology, is there.

In fact, it was clear quite early on to many of us who had some statistical training that a lot of these ideas had to be closely related. You end up using logistic functions, for example, in neural nets quite a bit. It has got to have something to do with logistic regression. You have lots of hidden latent variables; it has got to have something to do with latent variable factor analysis and structure, and ridge regression; all of these techniques are there.

In fact, when people started to analyze their networks after they fit them on data, it was interesting, because the tools for analyzing the networks were right out of multivariate statistics books. You would cluster or you would do canonical correlations on variables, and so it is a closely related set of ideas.

This has happened increasingly. Some of the ideas we are working on are to put layered networks into a more statistical framework, or to see them as generalizations of things like factor analysis, layers of factor analysis of various kinds. We also rediscovered mixture models, but in a much broader framework than they are commonly used in statistics. In fact, one of the pieces of work we did a few years ago was to recast the whole idea of fitting a decision tree as a mixture model problem. You can convert a decision tree probabilistically into a mixture model. Therefore, all the usual apparatus like the EM algorithm and so on are applicable directly to decision trees.

As these ideas have started to be promulgated in the literature, the issues have become whether you can compute the probabilistic quantities that you need. In the neural network literature, the problems are always nonlinear. The fitting algorithms always have to be nonlinear. No question of $(X' X)^{-1}$ is ever even conceived in neural nets. One of the standard ideas that has become important is that you fit data incrementally as you go. You get a new data point, you make some changes to your model, but you do not ever have one model; you usually have several models, and you make changes to the models. In fact, you do not make changes to all the models uniformly; you calculate something like a posterior probability of a model, given a data point, and you update those models that are most likely to have generated the data.

If I were approaching one of the data sets that I have heard about in the last couple of days, this is the kind of methodology I would most want to use, to look at data points one after another, and start building sets of models, start verifying them with each new data point that comes in, and start allowing myself to throw away data points that way. It is like smooth bootstrap, which is an example I gave in our discussion group—a lovely technique for allowing yourself to use an important technique, the bootstrap, which you could never use on a gigabyte of data, but which you could use on a smoothed version of a gigabyte of data, conceptually.

In neural nets, this has been by far the most successful technique. In those small problems where you can calculate an $(X'X)^{-1}$ type batch version of a fitting algorithm, it is usually much less successful. You have to do many more iterations in terms of real computer time than you do with a simple gradient or more general on-line incremental schemes. It has to do with the redundancy of the dam sets.

There is a lot to be learned from this literature that works for medium-sized dam sets that I would tend to want to carry over.

Another way to think about this is that general clustering sorts of methodology are not just a simple technique; it is a way of life; it is a philosophy. Models can be thought of as the elements of clusters. They do not have to be simple centroids in some kind of Euclidian space. Think of each new dam point coming in as belonging to one of the models, the posterior probability model giving the dam point, say, as your measure. Then as you are going through dam sets and making models, you are effectively doing a generalized kind of clustering, and you can think of these as mixture models, if you like that framework.

I would like to have a lot of parallel processors, each one containing the developing model fitting the dam sets, and by the time I got through with my massive dam set, I would have obtained a much smaller set of parameterized models that I would then want to go back through again and validate, and so on.

We tend to use a hodgepodge of techniques. Sometimes it may look ad hoc, but the problem is that no one technique ever works on any of the dam sets we look at. So we always find ourselves using bits and pieces of a hierarchical this and a mixture of this and a bootstrap of this. I think that has to carry over to these more complicated situations.

Luke Tierney: I do not have a whole lot more to add, just a few little things. On the second point about canonical dam structure, to reemphasize something I have said, this is an area where I am sure the computer science database community can offer us some help that is worth looking at and that we should be sure not to ignore.

On the other aspect, I cannot add much to what Deming said. It is clear that these standard classifications in mathematical statistics not being directly related to statistical practice is a problem that has been around forever, and it is not connected to the largeness or smallness of the dam. That does not mean that these ideas are irrelevant. If you think of them as useful tools for giving you guidance about, for example, what is worth computing if you have to compute one thing, that is a useful piece of information to have. Not the only piece, but it is useful, even in small dam situations, though there are many things for which we have no such theory that helps us.

With these larger data sets, if we find ourselves faced with situations where there are so many questions that we cannot possibly answer all of them, then we are going to have to start thinking in terms of the output on an analysis of a large dam set that is not going to be necessarily a static report, not one number, not even one picture, maybe not even a smile report, but a system, a software system that allows us to interactively ask questions. How does one decide that one such system is better than another? Mean squared error is not going to help a lot.

Huber: Can I add two remarks? One is, I strongly endorse Luke Tierney's last remark. Essentially, a massive dam set requires massive answers, and massive answers you cannot do except with some software system. The other is on what would be a more useful categorization of statistical problems. I wonder whether one should rephrase it into a classification of statistical activities rather than problems. My five favorite categories are, first, inspection, and then modification, comparison, modeling, and interpretation. The inspection can comprise almost

anything, such as inspection of residuals from a model. Modification may mean modification of the model, or modification of the data by eliminating certain things that do not belong. Comparison is all-important.

Hodges: I would like to take Usama Fayyad's comment and then throw it open to the floor.

Fayyad: I just wanted to reinforce what Michael Jordan said in terms of the clustering being a way of life. One analogy I would like to draw is to this room here. If you wanted to study it in every detail, it is a huge massive data set. Think of what your visual system is doing. You encode this wood panel here as a panel. You cluster it away, and you do not spend your life staring at every fine little detail of variation in it. That is how you move yourself through this huge data set and navigate through the room. So it seems to me that partitions of the data, clusters, are the key to dealing with large data sets.

That brings us head-on with the big problem of how you can cluster if you need to order N^2, where N is very large. But there are clever solutions for that. I don't believe data varies so much or grows in such hideous ways that we cannot deal with it.

Jerome Sacks: I have only one quick comment about all of the discussion here about what is important about statistics. Everybody has left out the idea of statistical design.

Daniel Relies: I had forgotten what Art Dempster said, too. There was an impressive list of five things, which did not include computing. I thought at one point you were trying to get the kinds of techniques that you want to teach, and computing was not one of them.

I remember what the dictionary said about statistics. It says that we are about the collection, manipulation, analysis, and presentation of masses of numerical data. Numerical, I think we are all ready to shed; that is a little too restrictive.

But when I try to tell people what I do, telling them I do inference, or telling them I do modeling, is a little too low-level for them to understand. Those are subcategories of the bigger picture. I frankly find myself doing mostly collection and manipulation and relatively little of the analysis and presentation.

I guess my feeling is that, if you do the collection and manipulation right, then you do not have to do much analysis.

Dempster: It all depends on the problem. As for computing, I am trying to put more emphasis on thinking about what you want to compute, rather than the problems of computing per se, which have dominated this workshop.

Hodges: There are five categories for problems. Five seems to be the magic number today. The first kind of problem is causal problems, either inferential, where you want to determine whether A caused B in the past, or predictive, whether A will cause B.

The second kind of problem is non-causal predictor problems. I thought Susan Dumais' example was perfect. You want to predict the performance of a research procedure and the causality involved, and why it has that performance is of no consequence whatsoever; you just want to know what it is going to do.

The third class of problems I would consider is description, where, for example, you have a massive data set and you just want to summarize that data in a useful way, where there is no sampling or uncertainty involved.

The fourth category is existence arguments. They are very popular in medicine, the area I work in, where basically a case series allows you to say, yes, it is possible to do coronary artery bypass grafts on people with Class IV congestive heart failure without killing three-quarters of them.

The last class of problems I would describe as hypothesis generation, where the thing you produce is an argument that "here is a hypothesis that has not got any uninteresting explanation like a selection bias, and that we should therefore go expend resources on has not."

Where do I get these categorizations? If you look at each of these arguments, the burden of proof in each argument is qualitatively distinct, and the activities that you have to do to make those arguments are qualitatively distinct.

For example, in Susan Dumais' case, the burden of proof is to show that, in fact, you do get the kind of performance from a search tool that you say you will get. That is what the argument has to establish. Now, there is a lot of statistical ingenuity that goes into building a tool like that, but the crux of the argument is to show that, yes, we have applied it to this group of data sets, we have measured its performance in this way, and we have these reasons to believe that it will generalize that way.

I think that if you focus on the problem, then that tells you what kind of activities you have to do to support the argument you need to make in that problem, and that in turn is going to drive things like what kind of data structures you want to have, so you can do those activities.

Peter Olsen: I would like to follow up on what Usama Fayyad said, in two senses. The first thing is, I think we need to be able to take better advantage of the bandwidth we have on our own internal sensors to understand the data that we are going to deal with. I suspect that we can process well in excess of T1 rates, in terms of our senses, but we cannot present that data to our senses in that way. I am aware as I stand here of the sights, the sounds, the temperatures, the feel of the floor, all these things. But when I go to look at data, I look at data with my computer. I have a 13-inch diagonal screen.

The second thing is that we are not born with this ability. If you look at a baby, he or she does not get the concept of points, planes, lines. It takes a while for small children to come to grips with and be able to extract the world around them. We seem to have become very good at being able to recognize very complex data structures. I can already tell the difference between you by sight and associate names with your faces, which is far beyond my ability to compute on a computer. Perhaps we ought to give some more thought to how people do these things, because we seem to be very successful at it.

QUESTION 2: SHOULD STATISTICS BE DATA-BASED OR MODEL-BASED?

Hodges: Does the advent of massive data sets mean that we are becoming more data-driven and less model-driven, or does it just identify the need for richer classes of models? Do the models in fact stay the same as N grows?

Tierney: In the discussion groups that I have been in, less modeling is not what we have been talking about. There has been a lot of talk about different kinds of models, and about hierarchical models in a variety of different senses.

I think to some degree one can say that hierarchical modeling is a divide-and-conquer strategy. One thing that you do with hierarchical modeling, whether it is by partitioning or viewed in the most general sense, is bring a massive data set problem down to a situation where in some local sense, you have maybe a hundred data points per parameter, just as you have many leaves of a tree [model] at which there is one parameter with a hundred data points sitting underneath them.

We have talked a lot about more modeling or more complex models. It looks like that is a way to make progress. Whether that is necessarily the right way, or it is just a matter of trying to bring what we think we understand, which is small data ideas, to bear in the larger thing, is hard to tell.

Fayyad: I voiced my opinion in the group, but I will voice it formally on the record—I am very skeptical of this attitude: "Here is a large data set and it is cracking my back; I can't carry it, and it is very complicated."

I have dealt mostly with science data sets and a little bit with manufacturing data sets, not as massive as the science ones. For those, I can understand why there is some degree of homogeneity in the data, so that your models do not have to explode with the data as it grows.

But in general, I am not convinced that you need to go after a significant jump in sophistication in modeling to grapple with massive data sets. I have not seen evidence to that effect.

Dempster: Climate models are pretty complicated. I do not know how you get away from that.

Fayyad: Is the complication an inherent property of whether the data set is small or not? I agree that some data sets are complex and some data sets are totally random or not compressible. I am not disagreeing with that fact. The question is, about these massive data sets that we collect nowadays, Does their massiveness imply their complexity? That is the part that I disagree with a bit.

Dempster: I think it is the phenomenon that is complex, and it drags the data in with it.

Fayyad: By the way, Peter Huber made an excellent statement in one of the groups, and I filed it away as a quotation to keep. I agree with his point a lot, even though he disagrees with my thinking. If data were homogeneous, one would not need to collect lots of it.

Jordan: The issue that always hits us is not N but always P [the number of parameters]. If P is 50, which is not unusual in a lot of the situations we are talking about, especially if you get somewhat heterogeneous data sets where the P are not commensurate and you have to treat them somewhat separately, then K to the P, where K is a small integer like 2, is much too large, and is going to overwhelm any data set you are possibly going to collect in your whole lifetime.

So these are not large data sets, in some sense. All of these issues that arise in small data sets of models being wrong, models being local, uncertainty, et cetera, are still there. I do not think these are massive data sets in any sense.

Carr: I had some thoughts about modeling. I tried to write down four classes of models. The first class I thought about was differential equations. I think of these as being very high compression models. They are typically simple equations, and I can almost hold them in my memory.

Then we have a slightly more expansive set of models that I call heuristic functional fitting. The number of parameters is a lot more. I do not know exactly how to interpret those. I might have standard errors for them, but I really do not know exactly what they mean, but they fit the data pretty well.

Then we can have a database model, where we have domain variables and predictor variables. If we could use nearest-neighbor methods, it is almost like a table look-up. I latch onto the domain, find something that is close, and then use the predictor variables as my prediction. So the database almost is a model itself

Or more generally, we have tables. That is a model. This is way beyond my human memory, and so I use the computer to actually handle the models.

There are also rule-based methods and probabilistic variance of that, where we try to encapsulate our wisdom, our experience. So we have these different classes. It seems like the computer has made the difference in that our models can in some sense be massive, or way beyond our own memories.

Huber: I should add a remark on the structure and growth of data. I think the situation we have is roughly this: In a small data set, we have one label or two labels, and with larger sets we have more labels, more branches and bigger leaves. A single leaf may be a pixel.

What is even worse, it is often not a tree. We impose the tree structure on it as a matter of convenience. So I have begun to wonder whether the old-fashioned way of representing data as a family of matrices might not be technically more convenient. The logical representation is not necessarily the internal representation.

The other point was in connection with the advent of massive data sets, that the data will again become more prominent relative to the model. I am not so sure whether this is a correct description of the situation. I think it is more like a situation in which we are evolving along a spiral. In the 17th century, statistics was basically tables. Then in the 18th century, it was a little bit more incidence-oriented. Then the 19th century was the century of graphics, population statistics, and so on. Then it mined over again to mathematics in statistics. I think we are now again in the other side of the spiral, where developments are on a larger time scale and they have relatively little to do with massive data sets.

Of course, the computer is the driving force behind it, and one does not forget what happened before. Sometimes I am worried that the proponents of one particular branch tend to forget all the other branches, and that the others are all wrong.

Eddy: Could we get Peter Huber to give us an example of a model that is not a tree?

Huber: It is a question of subsetting. If you have a classical matrix structure, then a zero-one variable corresponds to subsets. Say you have males and females, you have unemployed and employed—of course, it is possible to push the stuff into a tree if you subdivide into four branches, or you can do it by two. You can do it as you want.

I think the facility to rearrange data by subsetting is very important. Once you have cast the whole thing into a tree structure, rearrangement is conceptually complicated. That is one of the things that I think is most important and gets more important with larger sets—the ability to do subsetting in an efficient manner.

Eddy: I would like to agree, but it seems to me you can also think about it as multiple trees.

Huber: Of course.

Madigan: I just have one point in response to that question, that massive data sets for modeling might go away or something. Nobody said that, but it was implied by the question.

I think that inferring models of underlying physical processes is of fundamental importance and a major contribution of statistics. I think that massive data sets offer opportunities to build better models.

Dempster: History has been raised, and I thought I would talk about it briefly. There was a comment before that Fisher statistics were very data-based, and that was echoed in Peter Huber's comment. I do not see that at all, and so I am challenging the notion of oscillation a little bit.

The 19th century was full of very modem-sounding statistical inference. Edgeworth in the 1885 50th anniversary publication of the Royal Statistical Society was doing the kind of data analysis with small data sets that people would have done in the 1950s and 1960s. There are many

examples of Galton and his correlation and regression and of Gauss very concerned with theoretically justifying least squares and linear models and so on.

So there has always been a balance. I would have thought that in the 19th century, the capabilities for doing data analysis empirically were limited, until there was almost none of it being done. So I would have thought the other way around, but I am willing to see that there is a balance.

I do think that the current situation, for technical reasons, because of computing, is a whole different bail game, almost. I think, though, that the null hypothesis should still be that there will be this balance. It may in fact be—maybe I am echoing David Madigan—that you are more likely to get drowned if you do empirical data analysis on the huge thing, unless you are guided by the science somehow.

So the role of models as simplifiers is likely to be stronger, I would think.

Jordan: When you pick up the telephone nowadays and say "collect" or "operator," what is happening is that there are about a hundred HMMs [hidden Markov models] sitting back there analyzing those words, and they are divided into different ones, one for each word.

The way they train these things is very revealing. They know good and well that HMMs are not a good model of the speech production process, but they have also found that they are a good flexible model with a good-fitting algorithm, very efficient. So they can run 200 million words through the thing and adjust the parameters. Therefore, it is a data reduction scheme that gets you quite a ways.

After you fit these models with maximum likelihood, you try to classify with them, and you get pretty poor results. But nonetheless, it is still within a clean statistical framework, and they do a lot of model validation and merging and parameter tests and so on.

They do not use that, though. They go another stop beyond that, which is typically called discriminant training; it is a maximum mutual information type procedure. What that does is move around the boundaries around the models. You do not just train a model on its data; you also move it away from all the other data that correspond to the other models.

At the end of that procedure, you get a bunch of solutions that are by no means maximum likelihood, but they will classify and predict better, often quite a bit better. It is very computer intensive, but this is how they solve the problem nowadays.

I think that is very revealing. It just shows that here is a case, a very flexible model that you can fit, and you do not trust, but nonetheless it can be very useful for prediction purposes, and you use it in these different ways. For model validation, you still stay within maximum likelihood. But for classification prediction, you use another set of techniques.

Edward George: I like Usama Fayyad's metaphor of thinking about the room. It also keeps getting reinforced; I am seeing everything now through hierarchical models.

Another representation for thinking about that is multiresolution analysis. Consider wavelets and what is happening with wavelets right now: in some sense, an arbitrary function is a massive data set. So how do we organize our model of it? We come up with something very smooth at the top level, and then a little less smooth, and we just keep going down like that.

It is probably also the way we think about the room. For that panel, we have our gestalt, and that is way down at the bottom, but we have top things as well. That is probably the fight way to organize hierarchical information for simplicity.

Relies: I think we need to become more data-based. That is where the action is. The complexity of the data imply that model-dam interactions could very well come up to bite you. With a small data set, we would have the luxury of taking the information, spending a day or two or

a week and boning up on it, and then fitting our models and publishing our journal articles. I do not think we can do that. We are going to make mistakes if we try to do that now.

I think that the message has to be that data-oriented activities have to become recognized as a legitimate way for a statistician to spend a career. That gets back to something I said yesterday. I think that the journals are terrible about wanting sophisticated academic numerical techniques in an article in order to publish it. How I fix my data set seems like a boring topic, but it is the foundation on which everything else rests. If we as statisticians do not recognize that and seize it, then our situation is going to be like what happened to the horse when the automobile came along.

All the excitement I have had in my career as a statistician has been derived from the data and the complexity that I have had to deal with there, and not from integrating out the five-dimensional multivariate normal distribution, which is what I learned in school.

David Lewis: I absolutely agree with that. From the standpoint of a consumer of statistics, I am much more interested in reading about what you just described than reading about five-dimensional integration to do some fancy thing. But there is this huge unspoken set of things that applied statisticians do that we urgently need to know about, because a lot of us who are not statisticians are thrown into the same data analysis game right now.

Carr: We are talking about these complex hierarchical models. Yes, maybe that is the only thing that is going to work on some problems, but I would like to think of models as running the range from a scalpel to a club. The differential equations are more like a scalpel, and some of these huge models that are only stored in the computer are like a club. I am not content until I get it down to something that I can understand.

Some problems may be impossible to get to that level, but I would like to seek understanding. Yes, I have all this processing going on in this room when I look at it, but if it does not have meaning to me, I am going to ignore it, block it out. A lot of these complicated models work, but they are hard to grapple with if I don't understand what all these thousands of coefficients mean, or all these tables mean.

Dempster: In case there is misunderstanding, I agree with Dan Relies that it is certainly data-based. The question is, Where is it data driven, meaning very empirical, or are model structures a part of it? The truth is that it is an interaction between the two things. The danger in not paying attention to the interaction is I think a key thing, as Dan pointed out.

Tierney: Usama Fayyad made a comment early on about not wanting too complex models. I am not a hundred percent sure from what he said whether a hierarchical model would be complex or not, in his view.

One of the points of it is that hierarchical models tend to be simple things put together in simple ways to build something larger. So in one sense they are simple, and in other senses they are more complex. But there are different views that you can take of them. They can be a model. You can also think, as many people do, that once you fit one, it is a descriptive statistic; it is a model but it is used for the data, and it is something you need to understand. It is a dimensionality reduction usually, even if it is a binary tree type of thing. You are going down by a factor of two, the next level up by another factor of two. It simplifies. It is still complicated, and it is still something we have to understand maybe, but we can bring new parts of it as data, use data analytic tools to look at the coefficients of these models. We have to understand the small pieces and use our tools to understand the bigger pieces. I do not see the dichotomy necessarily. Things flow together and work well. They need to be made to work better.

Fritz Scheuren: For me there are questions and data and models, and models are in between questions and data, and there is a back and forth. I think the interactive nature of the process needs to be emphasized. I think in this world of fast computing and data that flows over you, you can focus more on the question than on the models, which we used to do in the various versions of history, and I think the questioner is important, and we need to talk about the questioners, with the models as a data reduction tool.

Fayyad: About the model complexity, just a quick response. If your data size is *N*, as *N* grows, is there some constant somewhere, *M*, that is significantly smaller than *N*, after which if you continue growing N your models do not need to go beyond that *M*? Or does your decision tree or whatever it is, hierarchical model, grow with log *N* or whatever it is? That is the distinction. I find it hard to believe. There are so many degrees of freedom, at least in these redundant data sets that we collect in the real world, that that must be the case.

I do not consider those beyond understanding. They imply a partition as a tool that lets you look at very small parts of the data and try to analyze them. It does not give you a global picture, though. But I do not think that humans will be able to understand large data sets. I think we just have to face up to that, that we understand parts of them, and that is life. We may never get the global picture.

QUESTION 3: CAN COMPUTER-INTENSIVE METHODS COEXIST WITH MASSIVE DATA SETS?

Hodges: Now we are going to change gears quite radically, although it is not quite a change of gears if we have decided that hierarchical models are Nirvana. Then we have to come to the issue of how we compute them. In some sense, Monte Carlo Markov chains and hierarchical models go together quite beautifully.

Madigan: Can they coexist with massive data sets? I think that they have to coexist with massive data sets, and that is that. We must build these hierarchical models. The massive data sets will enable us to do it, and we have to build the MCMC schemes to enable us to do it.

Jordan: The way they are built now, they do not do it at all. This is an important research question. The way the MCMC schemes are built now, they are batch-oriented. There is a whole data set for every sample of a stochastic sample, and that is hopeless. So there is a big gulf between use and theory.

Those schemes historically came out of statistical physics, developed in the 1940s for studying gasses. The statistical physicists still use them a great deal and have developed them quite a bit since then. I gave a reference, a lead article on some of the recent work, and I guess Ed George would know a lot more about this, too.

But it rams out that there is also a second class of techniques that the physicists use even more, loosely called renormalization group methods, or mean field type methods. These are just as interesting, in fact, more interesting in many cases. Physicists like them because they get analytical results out of them. But if you look at the equations that come out of these things, you can immediately turn them into interactive kinds of algorithms to fit data. I think that for the research-oriented academicians in the room, this is a very important area. Just taking that one method from physics did not exhaust the set of tools available.

Fayyad: What I don't understand is what the bootstrap would have to do with the massive data sets. What role would it play?

Dempster: I was going to comment on the bootstrap. The sampling theory of bootstrap was invented 20 years ago, when the chemists were eating up all the computer time, and the statisticians wanted something that would do an equivalent kind of thing. It generated a lot of wonderful theory, and still is generating it. But I do not think it is especially relevant with massive data sets. At least, I don't see how.

Hodges: Why not? In high-dimensional spaces, you have still got to figure out what your uncertainty is. It could be very large, and you would like to know that. The massive data set is not large *N*.

Dempster: I was just displaying my Bayesian prejudices. I agree with David Madigan. Whether it is MCMC or some other magic method, those are the ones we need.

Huber: For massive data sets, you rarely operate in any particular instance from the whole data set. The problem is to narrow it down to the path of interest. On that part, you can use anything.

Jordan: Another way of saying it is, if you can do clustering first, then you can do bootstrap. The technical term for that is smooth bootstrap. Clustering is density estimation. Then if you sample from that, you are doing smooth bootstrapping. That is a very useful way of thinking for large data sets.

Tierney: The bootstrap is a technical device. You can do a lot of things with it, but one of the things it gets at is variability of an estimate. I think a real question is, When is variability of an estimate relevant? Sometimes it is. It is not when a dominating problem is model error: no matter how I can compute the standard error of the estimate, it is going to be small compared to the things that really matter. But at times it will be important, and if it is important, then whatever means I need to compute it, whether it is the bootstrap, the jackknife, or anything else, I should go for it.

Another comment is that design of experiments is something I think we have neglected. There has got to be some way of using design ideas to help us with a large data set to decide what parts are worth looking at. I could say more than that, but I think that is something that must be worth pursuing.

Hodges: I know you could say a little more on that. Would you, please?

Tierney: I wish I could. It is something we need to think about. We have had design talked about as one of the things we should do in statistics. We have not talked about it very much in the groups I have been in. There has got to be potential there, to help make Markov chain Monte Carlo work in problems where you cannot loop over every data point. There must be some interesting ideas from design that we can leverage.

Hodges: You mean selecting data points or selecting dimensions in a high-dimensional problem?

Tierney: Those might be possible. I do not claim to have answers.

Dempster: Luke Tierney raised the issue of model error. Other people said there is no true model. So if model error is the difference between the used model and the true model, then there is no model error, either. So could somebody elaborate on that concept?

Hodges: This is partly in response to Luke Tierney. If you hearken back to a little paper in *The American Statistician* by Efron and Gong in 1983, and to some work that David Freedman did in the census undercount controversy, you can use bootstrapping to simulate the outcome of model selection applied to a particular data set. Given the models you select on the different bootstrap samples, you may be able to get some handle on the between-model variability or the between-model predictive uncertainty.

I am not enthusiastic about it myself, but that is one sense in which, to disagree with Luke, something like the bootstrap could be brought to bear.

Madigan: I think the Bayesian solution is a complete solution to the problem that the bootstrap fails utterly to solve. The basic problem is, if you are going to do some sort of heuristic model selection, you are likely to select the same model every time. You will not explore models, and therefore you will not reflect true model uncertainties.

Hodges: For the people who are not that familiar with what he is talking about, the Bayesian idea he is referring to is this: you entertain a large class of models and take the predictive distributions from those models, and combine them using some weighting such as the posterior probabilities of the model giving the data, so that you do not pick any models; you smoothly choose models.

Carr: One simple observation on the bootstrap is that if your sample is biased, your massive data set isn't really adequately representing the phenomenon of interest, and then the bootstrap is not going to tell you that much. You may model the database very well, and still not model the phenomenon of interest.

George: A short comment. Why are you talking about the bootstrap and not cross-validation? This is a highly rich environment when we are talking about massive data sets. We cannot do cross-validation when you have small data sets, but here we can. The bootstrap is in some sense a substitute for that when you do not have enough data.

John Cozzens: Would somebody enlighten me and please give me a definition of a massive data set? A lot of the things I have heard are, as far as I can tell, much ado over nothing. Certainly in many scientific communities, I can give you examples of data sets that would overwhelm or dwarf anything you have described and yet, they can be handled very effectively, and nobody would be very concerned.

So what I would like to know is, particularly when you are talking about these things, where is the problem? I think to understand that, I would like a definition of what we really mean by a massive data set.

Daryl Pregibon: I think that you are asking a valid question. I think that some of the applications we heard about are examples. Maybe we can just draw on Allen McIntosh's talk in the workshop, where we are dealing with packets of information on a network and accumulate gigabytes very quickly. There are different levels of analyses, different levels of aggregation, and there are important questions to be answered.

I do not think a statistician or a run-of-the-mill engineer can grapple with these issues very easily using the current repertoire of tools, whether they be statistical or computational. I do not think we have the vocabulary to describe massiveness and things like that; these are things beyond our reach. Maybe it is complexity, and clearly there is a size component. But I do not know how to deal with the problem that Allen is dealing with. If you know how, I think you should talk to Allen, because he would love to talk to you.

Lewis: One simple answer to that is that the people in physics write their own software to handle their gigantic data sets. The question is, Are we going to require that everybody go out and write their own software if they have a large data set, or are we going to produce software and analytic tools that let people do that without becoming computer scientists and programmers, which would seem like a real waste of time?

William Eddy: I think there is an issue that John Cozzens is missing here about inhomogeneity of the data. We are talking about data sets that are vastly inhomogeneous, and that is why we are worrying about clustering and modeling and hierarchy and all of this staff.

I think that John is saying, "I can process terabytes of data with no problem." Yes, you can process it, but if it consists of large numbers of inhomogeneous subsets, that processing is not going to lead to anything useful.

Cozzens: Now you are beginning to put a definition on this idea of a massive data set.

Dempster: My operational definition is very simple. If I think very hard about what it is I want to compute in terms of modeling and using the data and so on, and then I suddenly find that I have very difficult computing problems and nobody can help me and it is going to take me months, then I am dealing with a massive data set.

Huber: I once tried to categorize data sets according to size, from tiny to small, medium, large, huge, and monster. Huge would be large enough so that its size would create aggravation. I was specifically not concerned with mere data processing, that is, grinding the data sequentially through some meat grinder, but with data analysis.

I think it is fairly clear that with today's equipment, the aggravation starts around 10^8 bytes. It depends on the equipment we have now.

Sacks: I think the definition I prefer for massive data sets is the one the Supreme Court applies to pornography: I don't know what it is, but I know it when I see it.

QUESTION 4: IS THERE AN MODEL-FREE NOTION OF SUFFICIENCY?

Hodges: Can we define a model-free notion or analog for sufficiency? The interest here is as a means of compressing data without loss of information.

Tierney: Sufficiency in some ways has always struck me as a little bit of a peculiar concept, looking for a free lunch, being able to compress with no loss. Most non-statisticians who use the term data reduction do not expect a free lunch. They expect to lose a little bit, but not too much. I think that is a reasonable thing to look at.

I have a gut feeling I need to confirm. If you look at some of the ways people talk about efficiency of estimators, early on, for example in Rao's book, it is not from some of the fancy points of view that we often see, but more from the point of view of wanting to do a data reduction, and losing a little bit, but not losing too much.

Another way of putting this is, Can we think about useful ideas of data compression? Somebody raised an issue yesterday with the motion picture industry being able to produce a scene. Fractal ideas help a lot. You can very simply represent a scene in terms of a fractal model that does not reproduce the original exactly, but in a certain sense gives you the texture of a landscape, or something similar. You have lost information, but maybe not important information. You have to allow for some loss.

Huber: I have pretty definite ideas about it. Sufficiency requires some sort of model. Otherwise, you cannot talk about sufficiency. But maybe you may use approximate sufficiency in the way that Le Cam used it many years ago. The model may mean separating the staff into some structure plus noise.

Dempster: We did have a discussion of this, and my thought was that sufficiency is a way to get rid of things if they are independent of anything you are interested in from an inferential point of view.

Hodges: I can see defining a notion of sufficiency that is determined not by the model but by the question that you are interested in. I think it is impossible to throw away some of the data forever, even by some fractal kind of idea, for example, because for some questions, the data at the most detailed level is exactly what you need. You may be able to throw away higher-level data because they are irrelevant to the issue you are interested in. So it is the question that you are answering that determines sufficiency in that sense. Perhaps we have come to confuse the question with models because from the first 15 minutes of my first statistical class, for example, we were already being given models of parameters, and being told that our job was to answer questions about parameters.

George: Luke Tierney talked about sufficiency and efficiency. I think another major issue for massive data sets is economies of scale. That turns a lot of the trade-offs that we use for small data sets on their head, like computing cost versus statistical efficiency and summarization. For example, if you want to compute the most efficient estimate for a Gaussian random field for a huge data set, it would take you centuries. But you have enough data to be able to blithely set estimates equal to some appropriate fixed values, and you can do it in a second. It is inefficient, but it is the smart thing to do.

Ed Russell: I do not see how you can possibly compress data without having some sort of model to tell you how to compress it.

Huber: Maybe I should ask Ralph Kahn to repeat a remark he made in one of the small sessions. Just think of databases in astronomy, surveys of the sky, which are just sitting there, and when a supernova occurs, you look up at what was sitting in the place of the supernova in previous years.

If you think about data compression, sufficiency, I doubt that you could invent something reasonable that would cover all possible such questions that you can solve only on the basis of a historical database. You do not know what you will use. But you might use any particular part to high accuracy.

Pregibon: I am partly responsible for this question. I am interested in it for two reasons. One is, I think in the theory of modem statistics, we do have a language, we do have some concepts that we teach our students and we have learned ourselves, such as sufficiency and other concepts. So this question was a way to prompt a discussion of whether these things are relevant for the development of modem statistics. Are they relevant for application to massive amounts of data?

When we do significance testing, we know what is going to happen when N grows. All of our models are going to be shot down, because there is enough data to cast each in sufficient doubt. Do we have a language, or can we generalize or somehow extend the notions that we have grown up with to apply to large amounts of data, and maybe have them degrade smoothly rather than roughly?

The other point about sufficiency is, there is always a loss of information. A sufficient statistic is not going to lose any information relative to the parameters that are captured by the sufficient statistic, but you are going to lose the ability to understand what you have assumed, that is, to do goodness-of-fit on the model that the parameters are derived from. So you are willing to sacrifice information on one piece of the analysis, that is, model validation, to get the whole or the relevant information on the parameters that you truly believe in.

Items for Ongoing Consideration

DATA PREPARATION

- Elevation of status of data preparation and data quality stages in professional societies
- Clear articulation of what is meant by a massive data set
- Development of rigorous, theory-based methods for reduction of dimensionality
- Systematic study of how, when, and why methods used with small and medium-sized data sets break down with large size data sets; understanding of how far current methods, both statistical and computational, can be pushed; articulation of the variety of models that might be useful
- Development of methods for integration of tools and techniques
- Development of specialized tools in general "packages" for non-standard (e.g., sensor-based) data
- Establishment of better links between statistics and computer science
- Exploration of the use of "infinite" data sets to stimulate methods for massive data sets
- Creation of richer language for describing structure in data
- Educational opportunities—for nonstatisticians who use some statistical techniques and for statisticians, to broaden the knowledge base and provide better links to computer science

MODELS AND DATA PRESENTATION RESEARCH ISSUES

- Discovery and comparison of homogeneous groups
- Communication and display of variability and bias in models
- Better design of hierarchical visual display
- New modeling metaphors and richer class of presentation approaches
- Methods to help "generalize" and "match" local models (e.g., automated agents)
- Robust or multiple models; sequential and dynamic models

- Alternatives to internal cross-validation for model verification
- Retooling of computing environment for modeling massive data sets
- Simple presentation of "massive" complex data analyses

Closing Remarks

Jon Kettenring
Bellcore

Very briefly, I would like to thank all the people who have made the last two days an interesting time. On behalf of the National Research Council and the Committee on Applied and Theoretical Statistics, I hope that this is a beginning of a lot of excitement in the area of massive data sets. I think we have struggled with some of the very difficult aspects. I hope that we have a little momentum going. I hope that we have a useful network, and that you have had a chance to make some new connections here that will be valuable to you. We will follow up in some ways that we know about now, and perhaps some others that some of you will take as a lead-on to other efforts.

Again, thank you all very much.

APPENDIX

WORKSHOP PARTICIPANTS

Jack Alexander, *National Research Council*
Albert F. Anderson, *Public Data Queries, Inc*.
Thomas E. Ball, *McKinsey & Company, Inc*.
Fred Bannon, *National Security Agency*
Kenneth S. Cantwell, *National Security Agency*
Daniel B. Cart, *George Mason University*
Carolyn Carroll, *Stat Tech*
Michael Cohen, *Committee on National Statistics*
Dianne Cook, *Iowa State University*
John Cozzens, *National Science Foundation*
Keith Crank, *National Science Foundation*
Noel Cressie, *Iowa State University*
Marshall M. DeBerry, Jr., *Bureau of Justice Statistics, U.S. Department of Justice*
Arthur Dempster, *Harvard University*
Susan T. Dumais, *Bellcore*
Cheryl Eavy, *National Science Foundation*
William F. Eddy, *Carnegie Mellon University*
Stephen G. Eick, *Bell Laboratories (A Division of Lucent Technologies)*
Usama Fayyad, *Jet Propulsion Laboratory*
Mark Fitzgerald, *Carnegie Mellon University*
Edward George, *University of Texas*
Colin R. Goodall, *Health Process Management, Pennsylvania State University*
James Hodges, *University of Minnesota*
Peter J. Huber, *Universität Bayreuth*
Michael I. Jordan, *Massachusetts Institute of Technology*
Ralph Kahn, *Jet Propulsion Laboratory*
Jon R. Kettenring, *Bellcore*
Charles R. Kindermann, *Bureau of Justice Statistics, U.S. Department of Justice*
R. Brad Kummer, *Lucent Technologies*
Gad Levy, *Oregon State University and University of Washington*
David D. Lewis, *AT&T Bell Laboratories*
James Maar, *National Security Agency*
David Madigan, *University of Washington*
Fred Mann, *Department of Defense*
D.J. Marchette, *Naval Surface Warfare Center*
Alien A. McIntosh, *Bellcore*
Audris Mockus, *Bell Laboratories (A Division of Lucent Technologies)*
Ruth E. O'Brien, *National Research Council*
Peter Olsen, *National Security Agency*
Stan Openshaw, *Leeds University*
W.L. Poston, *Naval Surface Warfare Center*

Daryl Pregibon, *AT&T Laboratories*
Carey Priebe, *Johns Hopkins University*
Daniel Relies, *Rand Corporation*
Edmund L. Russell, *Advanced Micro Devices*
Jerome Sacks, *National Institute of Statistical Sciences*
Fritz Scheuren, *George Washington University*
John Schmitz, *Information Resources, Inc*.
David W. Scott, *Rice University*
Steven Scott, *Harvard University*
J.L. Solka, *Naval Surface Warfare Center*
Padhraic Smyth, *University of California, Irvine*
Robert St. Amant, *University of Massachusetts, Amherst*
William F. Szewczyk, *National Security Agency*
Luke Tierney, *University of Minnesota*
John R. Tucker, *National Research Council*
Lyle Ungar, *University of Pennsylvania*
E.J. Wegman, *George Mason University*
Lixin Zeng, *University of Washington*