Chris Jermaine — Research Statement

The world is awash in data. In medicine, every visit to a doctor, every medical test performed, every device connected to a patient—all these produce data, and increasingly, those data are archived. In medical research, PubMed contains references to 25 million research articles, many of which can be found online and store nearly the totality of human medical knowledge. Commercial concerns generate huge repositories of data; WalMart sees around 140 billion customer visits per year, all of which are logged. 300 hours of video are uploaded to YouTube per minute. The ubiquity of such electronic information has ushered in the Big Data era.

Big Data has the potential to change the world. Data-driven decision making can radically improve patient care. Scientific hypotheses can be automatically generated from a corpus of research papers, pushing science forward. Companies can use data to better serve customers and increase profits.

Unfortunately, realizing value from Big Data is not easy. Analysts struggle with the *size* of the data, the *variety* and *heterogeneity* of the data, and the *accuracy* of the data.

I study *data analytics*: how to build software to store, retrieve, and analyze huge and heterogeneous data sets. Among those who work on such problems, my skill set is unique in that I combine expertise in the statistics that underlie modern data analytics and machine learning with expertise in building software systems. My work focuses on:

1 The systems-oriented problems that arise when building software to manage large and diverse data sets; and

2 The foundational, mathematical questions that arise when statistical methods are used to analyze such data sets.

Since my arrival at Rice in January 2009, my research projects have evolved in both size and scope, from single-investigator projects to large, multi-institution projects where I play a leadership role. For example, I am one of three Rice co-PIs (with Vivek Sarkar, Swarat Chaudhuri) running the $11 million DARPA Pliny project, spread across four institutions (Rice, UW-Madison, UT Austin, GrammaTech, Inc.). Key projects include:

Systems for Managing Uncertainty (supported by the National Science Foundation and the Department of Energy; in collaboration with IBM Almaden Research Center). Sources of uncertainty in data abound: measurement errors, missing values that must be imputed, uncertain predictions about the future, entity resolution, and so on. It is very natural to characterize uncertainty about data using a probability distribution. This introduces randomness into data analysis, giving rise to the field of *stochastic analytics*. A long-term project has been the design and implementation of the *Monte Carlo Database System* (or MCDB), which allows a user to specify the uncertainty present in his or her data; the system then automatically takes those uncertainties into account when answering questions. For example, after specifying an appropriate statistical model, the user can ask, “What would my profits have been in 2014 had I raised my margins by 5%?” and the system will use the model to compute a *distribution* of possible results, taking into account the uncertainty in the modeling process. Because the system itself computes the model and performs the calculation on a distributed compute cluster, MCDB scales to very large data sets (terabytes of data) and complex models. Our paper on MCDB won the IBM-wide 2009 Pat Goldberg Memorial Best Paper Award in Computer Science, Electrical Engineering and Math.

Systems for Supporting Large-Scale Machine Learning (supported by the National Science Foundation and the Department of Defense; in collaboration with IBM Almaden Research Center). Applying statistical machine learning to the largest data sets is difficult. The process involves three steps: (1) designing a new statistical model or doing the math necessary to apply an existing one to a particular analysis task; (2) building and evaluating a prototype of the model using a mathematical programming tool such as R or MATLAB; and (3) implementing a version of the model that can run on a cluster of many computers, often in the "cloud" (hosted by Amazon or Microsoft). Step (3) in particular is difficult and error-prone, often taking a team of people with specialized skills months of trial and error. We focus on making this step much less painful, by developing systems that take code that looks a lot like what was prototyped in step (2), and *automatically* figuring out how best to run it in the cloud. One such project is SimSQL, a system that uses techniques from parallel database systems to take a high-level specification of a learning algorithm and run it in parallel on hundreds or even thousands of machines. A public release of SimSQL is available (`http://cmj4.web.rice.edu/SimSQL/SimSQL.html`).

Analysis of Large Biomedical Data Sets (supported by the National Science Foundation; in collaboration with researchers from Baylor College of Medicine, MD Anderson Cancer Center, and Texas Children's Hospital). Increasingly, patient data is collected and stored electronically. This includes structured data (demographics, billing codes, procedures performed), vital sign time series, textual notes and emails, images, and even genetic data. Analyzing these data has the potential to improve medical care and reduce costs. I am particularly interested in developing models to predict long-term outcomes. For example, we recently designed a new statistical model that can be used to analyze surgical vital sign data to predict outcomes such as mortality. The model has several uses, including giving anesthesiologists feedback on performance, and in optimizing post-operative care. We are now working on an outcome-prediction model that ingests data collected as a patient makes his/her way through the surgical process (initial admission, pre-operative assessment, surgery, etc.). As time passes, the model's accuracy increases, to such an extent that a few days after surgery, we can predict outcomes with almost total certainty.

Systems for Management of Complex Data (supported by the Department of Defense). Classically, database systems were not designed to store complex data structures: graphs, trees, maps, etc. But increasingly, modern data analytics tasks feature exactly such types of data. For just one example, consider one of the Big Data tasks we are working on at Rice: mining "Big Code"—the hundreds of billions of lines of program source code produced by millions of programmers over the last few decades. Data extracted from these codes are naturally represented as trees (abstract syntax trees) and graphs (dependencies among program modules and data). We are building a system at Rice called the "Pliny Database," or PDB for short, that is meant for storing just such complex data types on large compute clusters.

Chris Jermaine — Teaching Statement

Building prototype systems and writing papers about them is enjoyable, but the way that a University professor really impacts the world is by educating students—particularly PhD students. The single teaching accomplishment that I am most proud of is graduating twelve PhD students since I received my own PhD at the end of 2002 (including four PhD students during my six years at Rice). Currently, I am advising 7.5 PhD students, as well as a postdoc (another postdoc just left to take a faculty position).

Because of the long and close relationship an advisor has with a PhD student, it is possible to have a significant effect on a student's intellectual development. Early on, much of what a PhD student asserts is wrong, or poorly thought out. The advisor becomes accustomed to pointing out where the student is wrong, and suggesting what he or she might do better. But the relationship invariably shifts over time, as the student becomes the expert, and the advisor becomes the student. The most rewarding experiences as an advisor are those moments when you realize that it is not the student whose argument is flawed, but your own. The feeling is similar to that of being a proud parent! Invariably, the roles reverse, and the advisor comes to depend upon the student for intellectual guidance in the research. At that point, the student is ready to graduate.

In the classroom, the recent accomplishment I am most proud of is incorporating cutting-edge data management and mining technology into the Rice undergraduate curriculum. Soon after I arrived at Rice, I developed a new course, COMP 215. Ostensibly, 215 has a simple mandate: teach CS undergraduates how to program. It is the first programming course that Rice CS students take (typically at the sophomore level). However, I didn't want to simply teach programming to students. I wanted to develop a course where students would implement a significant software artifact using cutting-edge techniques from data management and data mining; students would learn to program almost as a side-effect of building a cutting-edge system. In the version of 215 that I developed, students spend the semester implementing a document indexing and retrieval system, that is capable of storing many thousands of text documents, "understanding" their content, and performing smart retrieval. The system relies on the Latent Dirichlet Allocation model (a Bayesian text model) for processing the documents and extracting topics from them. The students also implement an M-Tree, an advanced indexing structure to index the documents based on those extracted topics.

Another accomplishment I'm proud of is simply handling the large volume of students we have seen in recent years. Recently at Rice, I have taught COMP 215 (Introduction to Program Design, the first programming class taken by computer science students at Rice) and COMP 430 (Introduction to Database Systems). Over the last four semesters, I have taught 388 students in COMP 215 and COMP 430, for an average class size of 97 students per semester (the average decreases to 83 students per semester if one assigns 50% credit to my co-instructor for COMP 215 in the Fall of 2014). I suspect that these averages are at the very high end for tenure track faculty in the School of Engineering at Rice.

Chris Jermaine — Service Statement

In computer science, conference publications are the primary method for dissemination of original research results. There are three top-tier conferences in the data management area: ACM SIGMOD, VLDB, and IEEE ICDE. Each of these conferences is attended by between 600 and 1000 researchers, and receives between 400 and 1,000 research submissions per year. All of those submissions are rigorously reviewed. The most important service to the community at large that a data management researcher can perform is to help review for and organize these three conferences.

Along those lines, in 2013, I was one of the three technical program committee co-chairs for the IEEE ICDE conference. This required organizing a committee of 100+ researchers, recruiting 15 area chairs, making sure that around 1,500 reviews are performed in a timely fashion, and then working with the area chairs to make accept-reject decisions. This is a yearlong job, requiring anywhere from a few hours per week to nearly full-time work in the weeks immediately before decisions are returned.

Currently, I am working to organize the ACM SIGMOD conference (as general chair) which will be held in Houston in 2018.

Each year, I serve on the program committees for two or more of these three conferences. This requires reviewing between 8 and 12 research papers for each conference. Each paper is roughly equivalent to a journal submission in another research discipline, and the review process is similar. I've been on the SIGMOD program committee in 2006, 2009, 2010, 2012, 2013, 2015 and 2016, and on the VLDB program committee every year from 2004 through 2015, except 2013 and 2014.

While conference papers are the primary method for dissemination of research results in computer science, journals also play an important role. During my time at Rice, I've served as an associate editor for each of the top journals in the data management area: ACM Transactions on Database Systems, IEEE Transactions on Knowledge and Data Engineering, and the Very Large Database Journal.

Internally within Rice Computer Science, I've served on the computer science graduate studies committee every year since I first arrived in 2009, with the exception of the current year. The main responsibility has been evaluating several dozen MS and PhD applications per year.

Most recently, I chaired the 2015 Tenure Track Rice Computer Science Faculty Search Committee. This required me to personally review more than 200 faculty applications. I had to supervise a group of six committee members (including two from outside of computer science), making sure that all 200+ applications were reviewed by multiple committee members, and manage the process that pared that pool down to eight candidates who actually visited. I then had to manage those eight candidate visits, coordinating each with a faculty host, and then run the process that delivered a file of feedback on each candidate to the department chair.