Automating Open Science for Big Data

By
MERCÈ CROSAS,
GARY KING,
JAMES HONAKER,
and
LATANYA SWEENEY

The vast majority of social science research uses small (megabyte- or gigabyte-scale) datasets. These fixed-scale datasets are commonly downloaded to the researcher's computer where the analysis is performed. The data can be shared, archived, and cited with well-established technologies, such as the Dataverse Project, to support the published results. The trend toward big data—including large-scale streaming data—is starting to transform research and has the potential to impact policymaking as well as our understanding of the social, economic, and political problems that affect human societies. However, big data research poses new challenges to the execution of the analysis, archiving and reuse of the data, and reproduction of the results. Downloading these datasets to a researcher's computer is impractical, leading to analyses taking place in the cloud, and requiring unusual expertise, collaboration, and tool development. The increased amount of information in these large datasets is an advantage, but at the same time it poses an increased risk of revealing personally identifiable sensitive information. In this article, we discuss solutions to these new challenges so that the social sciences can realize the potential of big data.

Keywords: big data; repository; archive; data privacy; big data methods; big data algorithms; differential privacy

As with all science, science derived from big data research must be reproducible and transparent. A growing number of research

Mercè Crosas is the director of the Data Science Program at the Institute for Quantitative Social Science at Harvard University. Her team combines software development, statistics, and data stewardship to build software applications for sharing and analyzing research data. For more information, see datascience.iq.harvard.edu.

Gary King is the Albert J. Weatherhead III University Professor and director of the Institute for Quantitative Social Science at Harvard University. He has been elected as a fellow in six honorary societies and won more than forty "best of" awards for his work. For more information, see GaryKing.org.

DOI: 10.1177/0002716215570847

claims are based on increasingly large datasets—starting with several gigabytes (GBs; 10^9 bytes) to terabytes (TBs; 10^{12} bytes) or even petabytes (PBs; 10^{15} bytes) and exabytes (EBs; 10^{18} bytes)—from a multitude of sources, including sensors, apps, instruments, social media, and news. Decision-making is increasingly driven by evidence derived from such sources. While the potential for positive impact is substantial, large datasets are not easily shared, reused, or referenced with available data publishing software, or easily analyzed with mainstream statistical packages. Big datasets have limited value without applicable analytical tools, and the analytical results have limited value without known provenance. Currently, datasets small enough to be downloaded to the researcher's computer for local analysis can then be shared, cited, and made easily accessible through community data repositories or archives, and reused by others to validate and extend the original work. The scientific community needs to provide these same high standards and conveniences for research based on big data, by building analytical tools that scale and by facilitating reproducibility of the results through citable, reusable data and transparent analysis.

The challenges of the increasing scale in data are not new, but are part of a continual evolution in scientific dissemination. Throughout the history of the social sciences, a battle has raged between the size of computing facilities and the size of available data, with both speedily and continually increasing, but in different ratios. During the mainframe computer era, all computations were done on the same machines owned by corporations or governmental organizations. Then, most data analyses moved to desktop computers, at the hands of researchers. Then networked devices. At each step, new standards for data preservation and distribution have been required to keep pace with the boundaries of research methods. Now come big data, where the data and analyses are on the cloud. This increased computational ability allows access to entirely new modes of data analysis, but these datasets are immense in size and often streaming or continually updated in real time, and may contain masses of private confidential information.

Dynamic datasets larger than a few GBs present new challenges for data sharing, citation, and analysis. One challenge is that the analysis of large data often

---

James Honaker is a senior research scientist in the Data Science Program at the Institute for Quantitative Social Science at Harvard University. His research focuses on computational statistical models for social science problems, such as missing data, measurement error, and privacy preservation, including writing numerous open source software packages for scientific computing.

Latanya Sweeney creates technology to assess and solve societal problems. She is the founder/director of the Data Privacy Lab at Harvard University, an elected fellow of the American College of Medical Informatics, and former chief technologist at the Federal Trade Commission with patents, academic publications, and explicit citations in government regulations.

NOTE: The authors thank Michael Bar-Sinai, Christine Choirat, Vito D'Orazio, David O'Brien, Alexandra Wood, Urs Gasser, Micah Altman, Salil Vadhan, Kobbi Nissim, Or Sheffet, and Adam Smith for insightful and helpful discussions. Portions of the work on Dataverse and privacy tools are funded by the NSF (CNS-1237235), the Alfred P. Sloan Foundation, and Google and Microsoft Research gifts.

requires new optimization procedures or alternative algorithms that are not available in common analytical software packages. Another is that the sheer size of the data makes it impractical and inefficient for researchers to download such datasets to their personal machine. Any large dataset that is not efficiently hosted will have vast swaths of data left unexplored, as it is no longer the case that an individual team can explore every facet of its datasets. Furthermore, there are no standard solutions yet to cite a subset of large, streaming data in a way that others can get back to it—a critical requirement for scientific progress. Finally, there is a challenge in preserving privacy while maximizing the access of big data for research. Privacy concerns are more prominent in large, diverse datasets, which increasingly track nuanced detail of participant behavior, than in small datasets that can be more easily de-identified.

We propose solutions to these challenges in this article by extending two widely used frameworks for data sharing (Dataverse.org)[1] and analysis (ZeligProject.org) that we have developed, and by integrating them with privacy tools to allow all researchers to reuse the data and analysis, even when a dataset contains sensitive information.

An Extensible Framework for Long-Term Access to Big Data

Sharing, citing, and reusing big data

Accessible and reusable data are fundamental to science to continuously validate and build on previous research. Progressive, expansive scientific advance rests on access to data accompanied by sufficient information for reproducible results (King 1995), a scientific ethic to maximize the utility of data to the research community, and a foundational norm that scientific communication is built on attribution. Data repositories, such as the Harvard Dataverse, ODUM Dataverse, ICPSR, and Roper, as well as other general-purpose repositories, such as Dryad and Figshare, have played an important role in making small- and medium-scale research data accessible and reusable. In parallel, journals and funding agencies are now requiring that the research data associated with scientific studies be publicly available. Furthermore, standards and broader use of formal data citations (Altman and King 2007; Altman and Crosas 2013) are helping to establish how data should be referenced and accessed and provide incentives to authors to share their data.

Research with big data should be conducted following the same high standards that apply to all science. A researcher should be able to cite any large-scale dataset used in a research study; and any researcher should be able to find, access, and reuse that dataset, with the appropriate limitations applied to sensitive data.

What would a framework for sharing, citing, and reusing big data look like? At a minimum it must:

—Support extensible storage options and Application Programming Interfaces (APIs) to find and access subsets of the data;
—Allow users to cite subsets of the data with a persistent link and attribution to the data authors; and
—Provide data curation tools, that is, tools to allow adding information about the data (or metadata) so that the data can be easily found and reused.

Extending the Dataverse software for big data

In the last decade, the Data Science team at Harvard's Institute for Quantitative Social Science (IQSS) (King 2014) has developed open-source software infrastructure and tools to facilitate and enhance data sharing, preservation, citation, reusability, and analysis. A primary research software product delivered by this work is the *Dataverse Project* (King 2007, 2014; Crosas 2011, 2013), a repository infrastructure for sharing research data. The Dataverse software enables researchers to share and preserve their own datasets, and find, cite, and reuse datasets from others. In its current form, the software provides a rich set of features for a comprehensive, interoperable data repository for sharing and publishing research data, including:

—Control and branding of your own Dataverse (or individual archive), and widgets to embed your Dataverse in your website;
—Data deposit for any data file (up to a few GBs in size);
—Data citation, with a Digital Object Identifier (DOI), and with attribution to the data authors and the repository;
—Metadata support to describe the datasets in great detail;
—Multiple levels of access: open data, data with terms of use, and restricted data that require the user to be authenticated and authorized;
—Conversion of tabular data files to multiple formats, including a preservation format (that is, a commonly used format that does not depend on a proprietary software package, such as a tab delimited text file);
—Discrete versioning of datasets, with full trace of all previous versions and changes made in each version;
—Workflows to integrate article submission to scientific journals with data submission to the repository;
—Integration with data exploration and analysis (see below); and
—Support for APIs to get metadata and data, perform searches and deposit data.

With these foundations and a flexible architecture, the Dataverse software can be extended to support big data in the following ways.

Storage and API for big data. Any repository software needs to support a way to deposit and transfer large-scale data, and provide storage that can easily manage and provide quick access to these large amounts of data. A traditional HTTP

upload that only supports files up to a few GBs is insufficient, and the storage component cannot be based on only a traditional file system or relational database. Better approaches to managing and storing continuously growing, very large datasets include (1) an abstract file management system such as the Integrated Rule-Oriented Data System (iRODS) (Ward et al. 2011), which can serve as a collaborative platform for working with large amounts of raw data; (2) NoSQL databases, such as the document-based MongoDB (Bonnet et al. 2011) or the Apache Cassandra column-based database (Lakshman and Malik 2010), which use a storage mechanism that makes it faster to retrieve subsets of data; and (3) adaptive indexing and adaptive loading database systems to optimize finding and getting subsets of the data based on the type of data (Idreos, Kersten, and Manegold 2007). Depending on the type of big data, one of these solutions, or a combination of them, is more appropriate. In addition, the software needs a deposit API that allows for transfer of TB-scale data files. This can be accomplished, for example, by leveraging the Globus technology for sharing large data files, which uses a high-performance file transfer protocol called GridFTP (Foster 2011). Finally, the software needs an API for accessing subsets of the entire dataset through queries based on metadata fields (e.g., time ranges, geospatial coordinates). This API is central to enable extensions of the framework to explore, analyze, and visualize the data.

Some of this work is already under way. The ODUM Institute at the University of North Carolina, in collaboration with our team at IQSS and the Renaissance Computing Institute (RENCI), is in the process of integrating Dataverse with iRODS to combine the user-friendly features in a Dataverse repository with an underlying infrastructure for managing and storing large amounts of raw data. The integration of iRODS with Dataverse follows the research workflow of the scientific community. Researchers generate data and deposit them in their local data grid or cloud storage. This event is captured by a component of iRODS and triggers a replication to a Dataverse repository. When the data enter the Dataverse repository, other researchers or data curators can be notified so that they can add additional metadata to describe the data (Xu et al. 2014). The dataset is published in Dataverse with a formal data citation and extensive metadata. Alternatively, instead of replicating the entire dataset to a Dataverse repository, only a selected subset of the data stored in iRODS can be made publicly available through Dataverse, when it is ready to be published.

Citation of a subset of a large, dynamic dataset. Support for citation of large, dynamic datasets presents many problems not encountered in the bibliographic citation of literature or manuscripts (Sanderson and Van de Sompel 2012). Contrary to most written publications, datasets generated by sensors, instruments, or social media are often continuously expanded with time, and in some cases even streaming. Discrete versioning systems cannot handle this type of streaming data. Data citation tools for big data need to allow one to cite a subset of the data based on (1) selected variables and observations for large quantitative data, (2) time-stamp intervals, and (3) spatial dimensions.

The Dataverse software follows the data citation standard proposed by Altman and King (2007). This standard allows one to cite a subset of the data by inserting in the citation format the specific variables that define the subset. For large, dynamic datasets, we propose to extend this standard to insert queries based on a time range (for example, tweet data during a specific month, or sensor data between two dates), or on a region in space, or on any other variable for which a subset can be well defined.

Curation tools. Sole access to a data file or a subset of the file is not sufficient to reuse the data. At the extreme, a file with just numerical values has insufficient information to be of any use. At a minimum, the data values must be accompanied with metadata that describe every column. Preferably, a published dataset must have a web page with sufficient metadata and all the complementary files needed to understand and interpret the data. Curation tools should support ways to automatically, when possible, or otherwise manually, add metadata and files that describe the data. This metadata and additional documentation facilitates data discovery through search tools, and informs other researchers about the format, variables, source, methodology, and analysis applied to the data. The Dataverse software already supports a web page for each dataset (that is, the landing page that the persistent URL in the data citation links to) with metadata and complementary files. Supporting metadata and curation for big data would require additional tools to automate retrieving metadata from a variety of large, dynamic data files (for example, metadata retrieved from Facebook posts, from tweets, or blogs on a website).

Extensible Framework for Analysis of Big Data

New models of old models needed for inference in big data

The fundamental structural problem of massive-scale data occurs when the data are too large to reside at any one processor, and so smaller fragments of the total data, referred to as *shards*, are created and distributed across processors, sometimes called *workers*. Even if sharding is not necessary purely for the limitations of storage, taking advantage of the computational abilities of distributed processors often requires partitioning the data in this fashion, into manageable-sized pieces that allow for computationally light problems for each worker.

Many machine learning algorithms are conducive to operating with minimal communication on smaller problems and then combining for individual answers to form a grand solution. MapReduce (and its popular implementation Hadoop) is a more general technique for defining smaller tasks of a large scale problem, and distributing them across workers (the Map) and then communicating this information and combining the answers (the Reduce) in a fault tolerant fashion if some processes fail. However, many canonical statistical techniques cannot be implemented currently with sharded data. Preprocessing steps such as multiple imputation, which statistically corrects for the bias and inefficiency of incomplete

observations (Schafer 1997; King et al. 2001), and matching algorithms and propensity scores, which achieves balance among covariates to mirror the properties of randomized designs (Stuart 2010, Ho et al. 2007), are crucial steps for valid inference in many statistical models and have no algorithms for distributed settings.[2]

Similarly, many statistical models that are common, or even foundational, in traditional small fixed-scale data, have no analogous method of estimation in distributed settings. Pioneering work exists for solutions that simply run a large number of independent small-scale models, and then combine to an answer: some frequentist statistics can be calculated in this fashion (see review in Zhang, Duchi, and Wainwright 2012); the Bag of Little Bootstraps (Kleiner et al. 2014) uses small bootstraps of the larger data on each processor, upweighted to return to the original sample size; Consensus Monte Carlo (Scott et al. 2013) runs independent Markov Chain Monte Carlo (MCMC) chains on small samples of the data and combines sampled draws.[3] However, many statistical models that are highly independent across different groups or strata of the data can be estimated only by reference to the whole. For example, hierarchical (multilevel) models, small area estimation, and methods for estimating systems of structural equations have a large number of interdependent parameters specific to numerous different partitions of the full data. These models are common in economics, psychology, sociology, demography, and education—all fields where big data promises to unlock understanding of the behavior of individuals in complex social systems—and yet have no simple solution for distributed computation across sharded data. Solutions for these models, and for crucial techniques such as multiple imputation and matching, are urgently required for big data science.

Interoperable tools

The absence of key statistical techniques for big data is notable given the general and growing abundance of published open source utilities for big data analytics. While there is no lack of Big Data tools, most of the tools do not communicate or interoperate with each other. What is needed is a common framework to structure tools on, or a platform on which to share utilities across tools.

This lack of interoperable tools is commonly attributed to the distribution of languages used in big data analytics, and to the wide distribution of backgrounds and skill sets, disciplines, and training. However, the same issues arose in the previous decade with the emergence of the R language as the focal open sourced tool for applied statistics; there the language was common, and the training of the pioneering users much more focused and similar. The R statistical language is a giant open source project that spans all domains of applied statistics, visualization, and data mining. At the time of writing, R contained 5,698 different code libraries, or *packages*, most of which are written by a unique author. Among the advantages of this decentralized, dispersed organization are the speed and depth of coverage across statistical domains with which researchers share software and tools they have developed. A drawback of this massive contribution base is that each contributed R package can often have its own definitions for how data

should be structured, divided, accessed, and how formulas should be expressed and arguments named, meaning every researcher has to learn each package's unique calls and notation, and possibly restructure data, before seeing if that package has any useful application to her or his quantitative project.

The development of R encountered the same problems of interoperability that big data analytics tools now share. These issues strike at the relative advantages and drawbacks of open sharing networks of code. Individual researchers build individual tools focused exactly on the tasks connected to their own research; these tools are expertly constructed for and tailored to the exact task at hand. The shared distribution of these tools allows open access to the best possible tools of experts in each field, but means each tool requires specialized knowledge to learn and to apply outside the initial domain.

The *Zelig: Everyone's Statistical Software* package for R, developed and maintained by our team, brings together an abundance of common statistical models found across packages into a unified interface, and provides a common architecture for estimation and interpretation, as well as bridging functions to absorb more models into the collective library (Imai, King, and Lau 2008; Choirat et al. 2015). Zelig allows each individual package, for each statistical model, to be accessed by a common uniformly structured call and set of arguments. Researchers using Zelig with their data only have to learn one notation to have access to all enveloped models. Moreover, Zelig automates all the surrounding building blocks of a statistical workflow—procedures and algorithms that may be essential to one user's application but which the original package developer perhaps did not use in their own research and thus might not themselves support. These procedures or algorithms include statistical utilities such as bootstrapping, jackknifing, matching, and reweighting of data. In particular, Zelig automatically generates predicted and simulated quantities of interest (such as relative risk ratios, average treatment effects, first differences, and predicted and expected values) to interpret and visualize complex models (King, Tomz, and Wittenberg 2000).

A Zelig model for big data analytics

The vast promise and broad range of big data applications have steadily begun to be tapped by new tools, algorithms, learning techniques, and statistical methods. The proliferation of tools and methods that have been developed for specific tasks and focused solutions are myriad. But largely, these pioneering tools stand in towering isolation of one another. Often initiated as solutions to specific big data applications, the current open source methods available may each expect different data formats and use different call structures or notations, not to mention different languages.

We think the Zelig architecture devised for R can also solve this similar problem for big data science. We propose that a fundamental need in big data science is the proper construction of an *abstraction layer* that allows users to see quantitative problems through their commonality and similar metaphors and attacks, while abstracting away the implementation of any algorithm in any given language on any particular storage device and computational setting. This

framework would create an interoperable architecture for big data statistical and machine learning methods.

We propose that the architecture developed for Zelig for R can be mirrored in a language agnostic fashion for tools in Scala, Java, Python, and other languages that can scale much more efficiently than R, and can be used to bridge together the growing number of statistics and analytics tools that have been written for analysis of big data on distributed systems (such as Apache Mahout, Weka, MALLET). This will provide easier access for applied researchers, and, going forward, writers of new tools will have the ability to make them more generally available. Critically, such a framework must:

—*Allow users to use one call structure, and have access to the range of big data statistical and learning methods written across many different languages*. Rather than any user needing to learn new commands, languages, and data structures every time they try a new exploratory model, users will be able to seamlessly explore the set of big data tools applicable to their problems, increasing exploration, code reuse, and discovery.

—*Allow any developer of a new tool to easily bridge their method into this architecture*.

—*Provide common utilities for learning and statistics in big data analytics that can be easily interoperable and available to every model*. There is a large body of general purpose techniques in statistical models (e.g., bootstrapping, subsampling, weighting, imputation) and machine learning (e.g., k-folding, bagging, boosting) that are of broad applicability to most any model, but may only be available in a particular open source tool if one of the original authors needed that technique in their own research application. It should not be required of every method author to reinvent each of these wheels, nor should users of tools be constrained to only those techniques of use by the original author of their tool, and our architecture will make all these utilities interoperable across packages.

—*Enable interpretation of analytical models in shared and relevant quantities of interest*.

Preserving Privacy of Big Data

While we support open data in all possible forms, the increasing ability of big data, ubiquitous sensors, and social media to record our lives brings new ethical responsibilities to safeguard privacy. We need to find solutions to preserve privacy, while still allowing science the fundamental ability to learn, access, and replicate findings.

Curator models and differential privacy

A *curator model* for privacy preservation supposes a trusted intermediary who has full access to private data, and a system for submitting and replying to queries

from the world at large (Dwork and Smith 2009). The data remain in secure storage and are only available to the curator. In an *interactive* set up, the curator answers all queries, perhaps as simple as the count of the number of individuals who meet some set of restrictions, or as complicated as the parameter values of an estimated statistical model. In a *noninteractive* setting the curator produces a range of statistics initially believed to be of use to other researchers, and then closes the dataset to all future inquiry. With sufficient forethought, the noninteractive set up can extensively mimic the interactive use case; if the curator publishes all the sufficient statistics of a particular class of statistical model, then future users can run any desired model in that class without needing to see the original data. As an example, in the case of linear regression, this means publishing the sample size, means, and covariances of the variables. Any future user could then run any possible regression among the variables. The answers that the curator returns may intentionally contain noise so as to guard against queries that reveal too much private information.

Differential privacy is one conception of privacy preservation that requires that any reported result not reveal information about any one single individual (Dwork et al. 2006, 2009). That is, the distribution of answers or queries one would get from a dataset that does not include myself, would be indistinguishable from the distribution of answers from the same dataset where I had added my own information or observation. Thus nothing informationally is revealed about my personal information. Many differentially private algorithms function by adding some calculated small degree of noise to all reported answers that is sufficient to mask the contribution of any one single individual. *Synthetic data* is another privacy preserving approach that allows access to simulated data that does not contain raw, private data of individuals, but instead is simulated from a statistical model that summarizes (nonprivate) patterns found in the data (Reiter 2010). The advantage of releasing simulated data is that researchers familiar with exploring raw tabular data can use the tools they are most familiar with to analyze that data. However, a chief drawback is that it may be impossible to discover evidence of true phenomena if they were not originally encompassed or nested within the model used to drive the simulations.

In general, future data repositories that hold private data will have to develop curator architectures that shield raw, private data from users and report back only privacy-preserving results of user queries, such as what differential privacy provides or synthetic datasets allow.

DataTags and PrivateZelig as privacy preserving workflows

DataTags and PrivateZelig, collaborations among our Data Science group and Data Privacy Lab at IQSS, the Center for Research on Computation in Society (CRCS) at Harvard's School of Engineering and Applied Sciences, and the Berkman Center for Internet and Society at Harvard's Law School, are two solutions working toward a workflow and platform that facilitate careful understanding of the privacy concerns of research data, and a system of curated, differentially private access when warranted.

The DataTags project[4] aims to enable researchers to be able share sensitive data in a secure and legal way, while maximizing transparency. DataTags guides data contributors through all legal regulations to appropriately set a level of sensitivity for datasets through a machine-actionable *tag*, which can then be coupled, tracked, and enforced with that data's future use. The tags cover a wide range of data sharing levels, from completely open data to data with highly confidential information, which need to be stored in a double-encrypted repository and accessed through two-factor authentication. Even though the difficulty to share the data increases with each *DataTags* level, each tag provides a well-defined prescription that defines how the data can be legally shared. The *DataTags* application will provide an API to integrate with a Dataverse repository, or any other compliant repository that supports the multiple levels of secure transfer, storage, and access required by the tags.

The DataTags project does not provide a full solution for handling all privacy concerns in sharing research data. There might be additional ethical considerations, not covered by legal regulations; or concerns about reidentifying individuals by combining multiple datasets or using public data (Sweeney 2000) that are beyond what DataTags addresses. However, this project provides an initial idea of what a repository for research data must do to protect a sensitive dataset, while still making that dataset accessible.

Once DataTags has coded a dataset as private, the curator model described previously that releases differentially private statistics, can be implemented within the Zelig architecture. PrivateZelig is such a project. In this framework, any reported results generated by Zelig would be processed through an algorithm that ensures differential privacy, to the degree of privacy required, and as elicited from the DataTags interview. A Zelig package with the ability to report back differentially private answers could sit on a server containing encrypted data that was shielded from a researcher. The researcher could pass models to PrivateZelig, functioning as a curator on data securely stored in Dataverse, possibly by means of a thin-client web interface that does not have access to any data (Honaker and D'Orazio 2014), and in return view only the differentially private answers that were generated. Thus, the researcher can generate statistically meaningful, scientifically valid, and replicable results without seeing the underlying raw, private data, and without calculating any answers that reveal individual-level information about respondents.

Conclusion

The social sciences should embrace the potential of big data. But they should do so in a responsible and open way with tools accessible to the scientific community and by following high scientific standards; claims based on big data should provide access to the data and analysis to enable validation and reusability. In this article, we showed that, with a reasonable amount of incremental effort, we can extend the Dataverse repository software and the Zelig statistical software

package to offer a data-sharing framework and analytical tools for big data and thus provide extensible, open-source software tools to help automate big data science and put it in the hands of the entire scientific community. For the data-sharing framework, the extensions include a layer in Dataverse to support multiple types of storage options more suitable for big data (such as integration with iRODS, non-SQL databases, and adaptive storages), an API to submit and query large amounts of data at high speeds, a data citation that supports referencing a subset of dynamic data, and data curation tools that help to annotate and describe big data. For the data-analysis frameworks, extensions are twofold: implement models required to analyze big data using distributed computation for performance, and enable Zelig to make use of other programming languages that can handle data processing and computing faster than R. Finally, to fully support big data research, it is critical to provide tools that help to preserve the privacy of sensitive data, while still allowing researchers to validate previous analysis. Our team is working toward a solution by first assessing the sensitivity of the data using a new application named DataTags, and then allowing researchers to run summary statistics and analysis extending Zelig with differential privacy algorithms.

This work not only helps to make big data research more accessible and accountable but also fosters collaboration across scientific domains. The work requires inputs from and collaborations with computer science, statistics, and law, making social science for big data a truly interdisciplinary enterprise.

Notes

1. At this writing, we are about to change the branding of our project from the Dataverse Network Project at thedata.org to the Dataverse Project at Dataverse.org. Since we plan to make the change not long after publication, we use the new branding in the text.

2. Embarrassingly parallel algorithms, where no communication is necessary between processors, exist for Multiple Imputation (such as Honaker and King 2010; Honaker, King, and Blackwell 2011), but even these require all processors to have datasets of the size of the original data.

3. See also related approaches by Maclaurin and Adams (2013) and Ahn, Chen, and Welling (2013).

4. DataTags.org.

References

Ahn, Sungjin, Yutian Chen, and Max Welling. 2013. Distributed and adaptive darting Monte Carlo through regeneration. In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics*, eds. Carlos M. Carvalho and Pradeep Ravikumar, 108–16. n.p., NJ: Society for Artificial Intelligence and Statistics.

Altman, Micah, and Mercè Crosas. 2013. The evolution of data citation: From principles to implementation. *IASSIST Quarterly* 62:62–70.

Altman, Micah, and Gary King. 2007. A proposed standard for the scholarly citation of quantitative data. *D-Lib Magazine* 13 (3–4). Available from http://www.dlib.org/dlib/march07/altman/03altman.html.

Bonnet, Laurent, Ane Laurent, Michel Sala, Benedicte Laurent, and Nicolas Sicard. 2011. Reduce, you say: What NoSQL can do for data aggregation and BI in large repositories. In *Proceedings of 22nd*

International Workshop on Database and Expert Systems Applications (DEXA), eds. F. Morvan, A. M. Tjoa, and R. R. Wagner, 483–88. Los Alamitos, CA: IEEE Computer Society.

Choirat, Christine, James Honaker, Kosuke Imai, Gary King, and Olivia Lau. 2015. Zelig: Everyone's statistical software. Available from http://zeligproject.org.

Crosas, Mercè. 2011. The Dataverse network: An open-source application for sharing, discovering and preserving data. *D-Lib Magazine* 17 (1–2). doi:1045/january2011-crosas.

Crosas, Mercè. 2013. A data sharing story. *Journal of eScience Librarianship* 1 (3): 173–79.

Dwork, Cynthia, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography*, eds. Shai Halevi and Tal Rabin, 265–84. Berlin: Springer.

Dwork, Cynthia, Moni Naor, Omer Reingold, Guy N. Rothblum, and Salil Vadhan. 2009. On the complexity of differentially private data release: Efficient algorithms and hardness results. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, 381–90. New York, NY: ACM.

Dwork, Cynthia, and Adam Smith. 2009. Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality* 1 (2): 135–54.

Foster, Ian. 2011. Globus Online: Accelerating and democratizing science through cloud-based services. *Internet Computing* 15 (3): 70–73.

Ho, Daniel, Kosuke Imai, Gary King, and Elizabeth Stuart. 2007. Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. *Political Analysis* 15:199–236.

Honaker, James, and Vito D'Orazio. 2014. Statistical modeling by gesture: A graphical, browser-based statistical interface for data repositories. In *Extended Proceedings of ACM Hypertext*. New York, NY: ACM.

Honaker, James, and Gary King. 2010. What to do about missing values in time-series cross-section data. *American Journal of Political Science* 54 (2): 561–81.

Honaker, James, Gary King, and Matthew Blackwell. 2011. Amelia II: A program for missing data. *Journal of Statistical Software* 45 (7): 1–47.

Idreos, Stratos, Martin L. Kersten, and Stefan Manegold. 2007. Database cracking. In *Proceedings of the 3rd International Conference on Innovative Data Systems Research*, 68–78.

Imai, Kosuke, Gary King, and Olivia Lau. 2008. Toward a common framework for statistical analysis and development. *Journal of Computational Graphics and Statistics* 17 (4): 892–913.

King, Gary. 1995. Replication, replication, replication. *PS: Political Science and Politics* 28:443–99.

King, Gary. 2007. An introduction to the Dataverse network as an infrastructure for data sharing. *Sociological Methods and Research* 36:173–99.

King, Gary. 2014. Restructuring the social sciences: Reflections from Harvard's Institute for Quantitative Social Science. *PS: Political Science and Politics* 47 (1): 165–72.

King, Gary, James Honaker, Anne Joseph, and Kenneth Scheve. 2001. Analyzing incomplete political science data: An alternative algorithm for multiple imputation. *American Political Science Review* 95 (1): 49–69.

King, Gary, Michael Tomz, and Jason Wittenberg. 2000. Making the most of statistical analyses: Improving interpretation and presentation. *American Journal of Political Science* 44 (2): 347–61.

Kleiner, Ariel, Ameet Talwalkar, Purnamrita Sarkar, and Michael I. Jordan. 2014. A scalable bootstrap for massive data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 76 (4): 795–816.

Lakshman, Avinash, and Prashant Malik. 2010. Cassandra: A decentralized structured storage system. *ACM SIGOPS Operating Systems Review* 44 (2): 35–40.

Maclaurin, Dougal, and Ryan P. Adams. 2013. Firefly Monte Carlo: Exact MCMC with subsets of data. Paper presented at the 13th Conference on Uncertainty in Artificial Intelligence (UAI), Bellevue, WA.

Reiter, Jerome P. 2010. Multiple imputation for disclosure limitation: Future research challenges. *Journal of Privacy and Confidentiality* 1 (2): 223–33.

Sanderson, Robert, and Herbert Van de Sompel. 2012. Cool URIs and dynamic data. *Internet Computing, IEEE* 16 (4): 76–79.

Schafer, Joseph L. 1997. *Analysis of incomplete multivariate data*. London: Chapman & Hall.

Scott, Steven L., Alexander W. Blocker, Fernando V. Bonassi, Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. 2013. Bayes and big data: The Consensus Monte Carlo algorithm. Paper presented at the EFaB@Bayes250 Conference.

Stuart, Elizabeth. 2010. Matching methods for causal inference: A review and a look forward. *Statistical Science* 25 (1): 1–21.

Sweeney, Latanya. 2000. Simple demographics often identify people uniquely. Carnegie Mellon Data Privacy Working Paper 3. Pittsburgh, PA.

Ward, Jewel H., Michael Wan, Wayne Schroeder, Arcot Rajasekar, Antoine de Torcy, Terrell Russell, Hao Xu, and Reagan W. Moore. 2011. *The Integrated Rule-Oriented Data System (iRODS) micro-service workbook*. La Jolla, CA: Data Intensive Cyberinfrastructure Foundation.

Xu, Hao, Mike Conway, Arcot Rajasekar, Reagan Moore, Akio Sone, Jane Greenberg, and Jonathan Crabtree. 2014. Data book architecture: A policy-driven framework for discovery and curation of federated data. Paper presented at the 1st International Workshop on Big Data Discovery & Curation, New York, NY.

Zhang, Yuchen, John C. Duchi, and Martin J. Wainwright. 2012. Communication-efficient algorithms for statistical optimization. In *Advances in neural information processing systems 25*, eds. Fernando Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger. La Jolla, CA: Neural Information Processing Systems Foundation.