Automating Open Science for Big Data

The Harvard community has made this article openly available. Please share how this access benefits you. Your story matters.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Published Version</td>
<td>doi:10.1177/0002716215570847</td>
</tr>
<tr>
<td>Accessed</td>
<td>August 13, 2018 10:29:37 AM EDT</td>
</tr>
<tr>
<td>Citable Link</td>
<td><a href="http://nrs.harvard.edu/urn-3:HUL.InstRepos:25290370">http://nrs.harvard.edu/urn-3:HUL.InstRepos:25290370</a></td>
</tr>
<tr>
<td>Terms of Use</td>
<td>This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Open Access Policy Articles, as set forth at <a href="http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#OAP">http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#OAP</a></td>
</tr>
</tbody>
</table>

(Article begins on next page)
Automating Open Science for Big Data

Mercè Crosas, Gary King, James Honaker, Latanya Sweeney
Harvard University

Abstract

The vast majority of social science research presently uses small (MB or GB scale) data sets. These fixed-scale data sets are commonly downloaded to the researcher's computer where the analysis is performed locally, and are often shared and cited with well-established technologies, such as the Dataverse Project (see Dataverse.org), to support the published results. The trend towards Big Data -- including large scale streaming data -- is starting to transform research and has the potential to impact policy-making and our understanding of the social, economic, and political problems that affect human societies. However, this research poses new challenges in execution, accountability, preservation, reuse, and reproducibility. Downloading these data sets to a researcher’s computer is infeasible or not practical; hence, analyses take place in the cloud, require unusual expertise, and benefit from collaborative teamwork and novel tool development. The advantage of these data sets in how informative they are also means that they are much more likely to contain highly sensitive personally identifiable information. In this paper, we discuss solutions to these new challenges so that the social sciences can realize the potential of Big Data.
1. Introduction

As with all science, science derived from Big Data research must be reproducible and transparent. A growing number of research claims are based on increasingly large data sets - starting with several GBs ($10^9$ bytes) to TBs ($10^{12}$ bytes) or even PBs ($10^{15}$ bytes) and 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 data sets are not easily shared, reused, or referenced with available data publishing software, or easily analyzed with mainstream statistical packages. Big data sets have limited value without applicable analytical tools, and the analytical results have limited value without known provenance. Presently, data sets 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 finally 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 is not a new phenomena, but 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 computers 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 comes 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 data sets are immense in size and often streaming or continually updated in real time, and may contain masses of private confidential information.

Dynamic data sets larger than a few GBs present new challenges for data sharing, citation and analysis. One challenge is that the analysis of large data often 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 data sets to their personal machine. Relatedly, any large data set 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 their data sets. Furthermore, there are no standard solutions yet to cite a subset of the 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 is more prominent in large, diverse data sets, that increasingly track nuanced detail of participant behavior, than in small data sets that can be more easily de-identified.

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

2. Extensible Framework for Long-Term Access of Big Data

2.1 Sharing, Citing and Reusing Big Data

\textsuperscript{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.
Accessible and reusable data are fundamental to science in order to continuously validate and build upon previous research. Progressive expansive scientific advance rests upon access to data accompanied with 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 through the enforcement of open data policies. Furthermore, standards and broader use of formal data citations (Altman and King, 2007; Altman and Crosas, 2013) are helping 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 data set used in a research study, and any researcher should be able to find, access and reuse that data set, 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.**
- **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.**

### 2.2 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 data sets, and find, cite and reuse data sets 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 data sets 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 a tab delimited text file).
- Discrete versioning of data sets, with full trace of all previous versions and changes made in each version.
- Workflows to integrate article submission into scientific journals with data submission to the repository.
- Integration with data exploration and analysis (see section 3).
• 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 way:

**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 quickly access 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 only be based on a traditional file system or relational database. Better approaches to managing and storing continuously growing, very large data sets include: (1) an abstract file management system such as the Integrated Rule-Oriented Data System, iRODS, (Ward et al. 2011) that can serve as a collaborative platform for working with large amounts of raw data, (2) NoSQL databases, such as the document-based MongoDB (Bonnett 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 et al. 2007).

Depending on the type of Big Data, one of these solutions or a combination of them will be 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 data set 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, as discussed in section 3.

Some of this work is already underway. 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 data set is published in Dataverse with a formal data citation and extensive metadata. Alternatively, instead of replicating the entire data set to a Dataverse repository, only a selected subset of the data stored in iRODS can be made publicly available through Dataverse, when is ready to be published.

**Citation of a subset of a large, dynamic data set:** Support for citation for large, dynamic data sets presents many problems not encountered in the bibliographic citation of literature or manuscripts (Van de Sompel, 2012). Contrary to most written publications, data sets 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 cite a subset of the data by inserting in the citation format the specific variables that define the subset. For large, dynamic data sets, 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 describes every column.
Preferably, a published data set 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 data set (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 in a web site).

3. Extensible Framework for Analysis of Big Data

3.1 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 can not be presently implemented 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, that achieve 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.  

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 large numbers 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 (Kliener 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 MCMC chains on small samples of the data and combines sampled draws.

However, many statistical models that are highly independent across different groups or strata of the data can only be estimated 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 on the behavior of individuals in complex social systems-- and yet have no simple solution for distributed computation across sharded data. Solutions for these models,

---

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

3 See also related approaches such as Maclaurin and Adams (2014), Ahn et al. (2013)
and for crucial techniques such as multiple imputation and matching, are urgently required for Big Data science.

### 3.2 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 in the emergence of the R language as the focal open sourced tool for applied statistics; here 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 contains 5698 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, how formulas should be expressed, and arguments named, meaning every researcher has to learn each package's unique calls and notation, and possibly restructure their data, before seeing if that package has any useful application to their 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 the exact task at hand, and tailored to make best approach to the style of data 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 increasingly more models into the collective library (Imai, King, and Lau 2008, 2007). 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 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 Wittenberg 2000).

3.3 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 each other. Often initiated as solutions to specific big data applications, the present open source methods available may each expect different data formats, and use different call structures or notations, not to mention 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 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 the ability for writers of new tools to make them more generally available. Critically, such a framework must:

- Allow users to use one call structure, and have access to all 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.
- Allows 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.

4. 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 increasing ethical responsibilities to safeguard privacy. We need to find solutions to preserve privacy, while still providing science the fundamental ability to learn, access and replicate findings.

4.1 Curator Models and Differential Privacy

A curator model of an architecture 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 remains in secure storage and 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 further 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 does 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 (non-private) patterns found in the data (Reiter 2009). The advantage of releasing simulated data is that researchers familiar with exploring raw tabular data can use the tools they are most familiar with, while one 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 tasked with private data will have to develop curator architectures which shield raw private data from users, and report back only privacy preserving results of user queries, such as for example, differential privacy provides, or synthetic datasets allow.

4.2 DataTags and PrivateZelig as a Privacy Preserving Workflow

DataTags and PrivateZelig, in collaboration between 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 of our solutions towards 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 (DataTags.org) aims to enable researchers to 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 dataset through a machine-actionable Tag, that 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 DataTag level, each Tag provides a well-defined prescription that defines how data can be legally shared. The DataTags application will provide an API to integrate with a Dataverse Network 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 re-identifying individuals by combining multiple data sets or using public data (Sweeney, 2000) that are beyond what DataTags addresses. However, this project gives an initial assertion of what a repository for research data must do to protect legally a sensitive data set, while making that data set still accessible, under the prescribed requirements.

Once DataTags has coded a dataset as private, the curator model described previously, releasing 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 ensuring 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, or calculating any answers that reveal individual level information about respondents.

5. Conclusion

The social sciences should embrace the potential of Big Data. But it should be done in a responsible and open way with tools accessible to the scientific community and following scientific high standards; claims based on Big Data should provide access to the data and analysis to enable validation and reusability. In this paper, we show 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 them 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, adaptive storages), an API to submit and query large amounts of data at high seed, a data citation that supports referencing a subset of dynamic data, and data curation tools that help annotate and describe Big Data. For the data analysis frameworks, extensions are two-fold: implementation of 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 preserve the privacy of sensitive data, while still allow researchers to validate previous analysis. Our team is working towards a solution by first
assessing the sensitivity of the data using a new application named DataTags, and then allowing 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.

Acknowledgements

The authors thank Michael Bar-Sinai for numerous insightful discussions about some of the technical solutions presented here. DataTags has developed from joint collaboration of the authors with Urs Gasser, Michael Bar-Sinai, David O'Brien and Alexandra Wood. Continued Zelig development is a collaboration with Christine Choirat, and this article benefits from continuous discussions with her. The PrivateZelig project comes from the ongoing collaboration of the authors with Salil Vadhan, Vito D'Orazio, Kobbi Nissim, Or Sheffet and Adam Smith. Portions of the work on Dataverse and Privacy tools are funded by the NSF (CNS-1237235), the Alfred P. Sloan Foundation, and Microsoft Research.

References


M. Altman and M. Crosas. The Evolution of Data Citation: From Principles to Implementation. IASSIST Quarterly, Forthcoming.


