Motivating contributions to public goods of uncertain future values: A pre-Analysis plan

Yan Chen Margaret Levenstein Linfeng Li Lars Vilhuber *

August 3, 2020

*We thank Tanya Rosenblat, Alain Cohn, Erin Krupka, and participants of the Behavioral and Experimental Economics Lab Group at the University of Michigan for helpful discussions and comments.

1 Introduction

Metadata is data about data. As high quality metadata increases the likelihood that a data set is found and re-used in the future, metadata can be viewed as public goods of uncertain future values. Little is known about people's motivations to contribute to public goods of uncertain future values. In this study, we investigate this question in the realm of metadata provision in a field experiment.

Domain experts' contribution to metadata is invaluable as they have deep expertise rooted in their scholarly works. However, experts may have little interest in enhancing the metadata for studies that they have published, making the data sets difficult to find and creating barriers for data reuse and reproducibility. In our field experiment, we investigate what motivates people to contribute to public goods of uncertain future values. Specifically, we examine what motivates domain experts to contribute metadata for published studies from several perspectives. First, she may care about the findability of her study. By enhancing the study-level metadata, modern search engines will be able to *find* the study more easily. Illuminating the connection between metadata and findability might reduce the uncertainty for this type of digital public goods. Second, she may be motivated to provide metadata for private benefits, such as increased citation, or social benefits, such as helping others in their research.

Generally, the metadata for a study fits into two categories, study-level and variable-level metadata. The former characterizes a study, such as the geography and time period, and enables search engines to find a data set. In comparison, providing variable-level metadata involves labelling each variable and composing a comprehensive codebook documenting the data collection and manipulation. Reusability relies on complete variable-level metadata provided directly by the original authors. When an expert provides enhanced metadata for a data deposit, it might generate different levels of benefits. Locally, all co-authors of the study benefit from a data deposit with enhanced metadata. At the intermediate level, close neighbors in the co-author network who are experts in the same field might also benefit from enhanced metadata. As close academic neighbors who share common topical or methodological interest, experts in the same field are more likely to reuse the data from a well documented and reusable study (Gregory, 2020). Globally, improving study-level metadata increases the *findability* of the data deposit (Chapman et al., 2020). Researchers in the same or neighboring disciplines are more likely to find and cite the better documented studies.

Practically, enhancing study-level metadata that improves findability is detached from producing complete variable-level metadata that improves reusability. For one published study, it may take up to 30 minutes to populate all the study-level metadata fields that are available on openICPSR.

Chapman et al. (2020) offers a survey of the state of the art *dataset search* architectures from both the commercial and research domains. Despite the advances in Machine Learning and NLP, dataset search still relies heavily on structured metadata fields. Greenberg et al. (2001) provided early evidence that resource authors are able to provide high-quality metadata with the help of simple web-forms and the resource authors are willing to provide metadata for the cause of enhancing findability of their studies. Santos et al. (2016) argues that metadata stands at the core of operationalizing an architecture that serve both the data owners and data users in a FAIR manner. Piwowar, Day, and Fridsma (2007) demonstrated a sizable correlation between citation and whether datasets are openly accessible. While it has been well-cited, the study is often criticized for its sample selection method.

Our study aims to find mechanisms that increase metadata provision. We do so by reducing the uncertainty of its future value, and by making two different kinds of benefits more salient.

2 Experiment Design

As context, the American Economics Association (AEA) is among the early movers in the social sciences to make empirical research published in its journals more transparent and reproducible. In the early days, when an empirical paper is conditionally accepted, authors must send the data, code and readme files as a zip file to the editorial office. Upon publication, these supplemental information is provided on the AEA journal website. Because of the technological constraints from using zip files, these data sets typically have variable-level metadata of various quality, but sparse study-level metadata. Starting in October 2019, the AEA migrated the entire archive of 3,073 data and code supplements into the AEA Data and Code Repository hosted at openICPSR.[1] Currently, the metadata fields for the migrated deposits are sparse, making the migrated studies hard to find.

Participants in our experiment are authors of articles published in the AEA journals, whose data sets were migrated to openICPSR. Using their publication records, we can organize these economists through a co-author-network, where we allow for multiple edges between individuals if they have co-authored multiple papers together. In total, there are $3,070$ articles and $4,320$ unique authors. The giant component of this network has $2,008$ authors. To mitigate spillovers through the co-author-network, we include a portion of the articles into this experiment, such that none of any two articles has a common author. That is, the set of articles in our sample are independent.

We have one control condition and two treatment conditions. In the control condition, we describe the context and the task, and invite authors to provide metadata through a customized experiment interface. In treatment 1, we insert one extra paragraph into the email template used in the control condition, which explains that metadata increases the findability as well as the citation of the data set and article, emphasizing the private benefit of providing metadata. Treatment 2 adds the potential social benefit to graduate students and other researchers.

Our participants are domain experts. In our field experiment, we provide a simple interface to make it easier for the authors to provide study-level metadata.

[1] `https://www.openicpsr.org/openicpsr/aea/`. ICPSR is the largest social science data repository in the world. Its data sets are managed by professional archivists.

2.1 Preparation: Units of randomization

We plan to treat each participant with one data deposit. The unit of randomization is at the data-deposit level. Furthermore, since nearly half of the participants belong to a connected giant component, we need to mitigate the potential network interference by dropping a number of data deposits. In our study, we include a subset of the migrated data deposits in a way that none of the chosen deposits has any overlapping author.

To construct the set of data deposits for random assignment, we perform the following procedure on the network of authors, where edges between any pair of authors are introduced by one co-authored data deposit. By construction, this is a multigraph where self-loops and multi-edges are permitted.

1. For a given component in the author-to-author network, we use the Leiden algorithm (Traag, Waltman, and van Eck, 2019), a community detection algorithm, to compose a partition of its nodes (authors). Each element in the partition is called a *community*.

2. Drop inter-*community* edges along with and all edges introduced by the same set of deposits[2]. All communities become stand-alone components in the reduced graph.

3. Iterate through all components with more then one deposit and repeat Step 1 and Step 2 on each component. This generates a reduced graph free of all known inter-*community* edges.

4. Repeat Step 3 until no inter-*community* edges can be found within components in the reduced graph.

5. Extract units for random assignment based on the components in the final version of the reduced graph. If there are multiple articles in a (complex) component, we pick one article along with its authors for random assignment. If the (simple) component has only one article, we pick all authors involved.

For details of the network reduction procedure, please refer to Appendix C.

2.2 Email templates

In the following email templates, <variable-text> will be replaced with the proper value. Blue and underlined texts are denoting hyper-linked texts, where participants will click on. We have one control message (C) and two treatment messages (T_{priv}, T_{social}).

2.2.1 Control Condition

In the control message, we provide the background information that data deposits previously hosted by AEA are now migrated to openICPSR. Participants are invited to

[2]Inter-community edges are identified by constructing the *Quotient graph* given the partition of the original network component. Inter-community edges coincides with the edges in the quotient graph.

provide study-level metadata for their chosen data deposit (AEA publication) through a customized experiment interface where we collect study-level metadata.

Subject line: Your AEA data set migration

> Dear Dr. <Lastname>,
>
> Since July 16, 2019, the American Economic Association has used the AEA Data and Code Repository at openICPSR as the default archive for its supplements. The migration increases the findability of your data through a variety of federated search interfaces such as Google Dataset Search, the openICPSR search interface, and the general ICPSR search interface.
>
> To further enhance the findability of your data, we ask that you spend up to 20 minutes to provide additional metadata for your AEA data deposit through a user-friendly web interface. The information will be batch-imported back to the original openICPSR deposit.
>
> - If you are interested to proceed, please click here to provide additional metadata for your study titled "<Article 1:Title>."
> - Please click here if you think your co-authors are better suited for providing metadata. We will opt you out of future communications.
> - Please click here if you are not interested in providing metadata and would like to opt out of future communications altogether.
>
> For articles with more than one author, each co-author is receiving an identical email with an individualized link. Thank you for your effort!
>
> Sincerely,
>
> Lars Vilhuber
>
> AEA Data Editor

Note that we provide three links for participants to choose to (1) provide the metadata for their study; or (2) let their co-author(s) provide the metadata; or (3) opt out. As providing metadata contributes to both local and global public goods, we separate them using two opt-out links. Lastly, telling each author that their co-author(s) receives an identical message reduces the likelihood that authors communicate with each other prior to responding to the message.

2.2.2 Treatment: Findability and Private Benefit (T_{priv})

Compared to the control message, we add a new paragraph and introduce the evidence that explains and supports the "findability" implication of enhanced metadata. This paragraph reduces the uncertainty of the future value for metadata.

The paragraph concludes with an emphasis on the private benefits of enhanced metadata. participants are invited to a customized experiment interface to populate study-level metadata. The experiment interface is identical to the one used in the control condition.

Subject line: Your AEA data set migration

Dear Dr. <Lastname>,

Since July 16, 2019, the American Economic Association has used the AEA Data and Code Repository at openICPSR as the default archive for its supplements. The migration increases the findability of your data through a variety of federated search interfaces such as Google Dataset Search, the openICPSR search interface, and the general ICPSR search interface.

Analyses of search and usage of ICPSR's data catalog indicate that most datasets are discovered because searches pick up metadata that includes citation to published articles and key concepts (geography, methods). Enhancing the metadata for your dataset will increase the likelihood that your publication and data are found and cited.

To further enhance the findability of your data, we ask that you spend up to 20 minutes to provide additional metadata for your AEA data deposit through a user-friendly web interface. The information will be batch-imported back to the original openICPSR deposit.

- If you are interested to proceed, please click here to provide additional metadata for your study titled "<Article 1:Title>."
- Please click here if you think your co-authors are better suited for providing metadata. We will opt you out of future communications.
- Please click here if you are not interested in providing metadata and would like to opt out of future communications altogether.

For articles with more than one author, each co-author is receiving an identical email with an individualized link. Thank you for your effort!

Sincerely,

Lars Vilhuber

AEA Data Editor

Please note that the italics is added for emphasis in the paper. The corresponding paragraph is not italicized in the email message.

2.2.3 Treatment: Findability, Private Benefit and Social Benefits (T_{social})

In addition to the first treatment, we add the social benefit of the enhanced metadata on graduate students and others. Participants are again invited to a customized experiment interface to populate study-level metadata. The interface is identical to the one used in the control condition.

Subject line: Your AEA data set migration

Dear Dr. <Lastname>,

Since July 16, 2019, the American Economic Association has used the AEA Data and Code Repository at openICPSR as the default archive for its supplements. The migration increases the findability of your data through a variety of federated search interfaces such as Google Dataset Search, the openICPSR search interface, and the general ICPSR search interface.

Analyses of search and usage of ICPSR's data catalog indicate that most datasets are discovered because searches pick up metadata that includes citation to published articles and key concepts (geography, methods). Enhancing the metadata for your dataset will increase the likelihood that your publication and data are found and cited, making it more useful to graduate students and others.

To further enhance the findability of your data, we ask that you spend up to 20 minutes to provide additional metadata for your AEA data deposit through a user-friendly web interface. The information will be batch-imported back to the original openICPSR deposit.

- If you are interested to proceed, please click here to provide additional metadata for your study titled "<Article 1:Title>."
- Please click here if you think your co-authors are better suited for providing metadata. We will opt you out of future communications.
- Please click here if you are not interested in providing metadata and would like to opt out of future communications altogether.

For articles with more than one author, each co-author is receiving an identical email with an individualized link. Thank you for your effort!

Sincerely,

Lars Vilhuber

AEA Data Editor

2.3 Experiment procedure

Each participant receives a customized email using one of the three templates. The links in the email for the customized web interface are individual specific, where the article we choose to include is displayed. For screenshots of the experiment interface, please refer to Appendix D.

Outcome variables We collect outcome variables both through the treatment email and the survey interface. First, we record the intention to contribute through the hyperlinks in the body of the treatment email. A positive response is recorded when a participant clicks on the link to contribute metadata. A negative response is recorded when a participant chooses either "co-authors are better suited for providing metadata" or "not interested in providing metadata". We use MailChimp to administrate the email delivery and tracking. Per our testing, the email-opening and link-clicking events are tracked even when our email reaches an old email address and gets forwarded to another email address of the same researcher. For participants who proceed to the experiment interface, we collect the individual responses for metadata fields and aggregate the responses at the study level for analysis. Appendix A contains the complete list of metadata fields we collect.

After collecting the metadata contents, participants answer a few survey questions about their data re-use activities and willingness to update the data deposit. For those interested in updating the data deposits on openICPSR and providing enhanced variable-level metadata, we collect their interest through the survey and grant them write-access to their data deposits on openICPSR. With write-access, authors can upload new files to the deposit and provide variable-level metadata. Lastly, we ask the authors to give us their motivation for providing the metadata.

Accommodation for sequential inputs For each article with multiple authors, our interface is designed to accommodate sequential inputs of metadata by co-authors. In the event that multiple co-authors contribute metadata sequentially, a new contributor can see who have contributed what, and add content accordingly. In the unlikely event that multiple co-authors edit at the same time, our interface is not able to let them see the location of each other's cursors.

As a result of sequential inputs, when accounting for individual contributions to a given metadata field, we consider both the overall contribution of all co-authors, as well as a binary variable denoting *individual improvement*. The last edit will be the overall contribution. To capture the individual improvement for a given metadata field, we compare contributions provided by authors who edited consecutively. If there is a difference, we document it as an improvement. Otherwise, we set the improvement to zero.

We plan to launch the experiment on Monday August 3, 2020, and continue to collect data for the subsequent two months.

2.4 Hypotheses

Our hypotheses are derived from a theoretical model of public goods of uncertain future values, where the public good is simultaneously a local public good that benefits co-authors of a study, and a global public good for the scientific community. The two primary outcome variables include the link(s) participants click and the quantity of metadata contributions. Our first hypothesis is based on the findability information in the treatment emails which reduces the uncertainty and increases the expected future value of the public good.

Hypothesis 1. *Participants in the treatments are less likely to opt out of the study than those in the control condition.*

Between the two treatments, we expect that marginal increase in participation from the private benefit of findability and citation will be larger than that from the social benefit of helping others, based on a field experiment designed to motivate domain experts to contribute to Wikipedia (Chen et al., 2020).

Hypothesis 2. *The marginal increase in participation rate from the Private Benefit treatment is greater than that from the additional Social Benefit.*

Conditional on reaching the metadata contribution page, we expect that effort will follow the same order as specified in our first two hypotheses.

Hypothesis 3. *At the data deposit level, the quantity of metadata is the highest under the Social&Private Benefit treatment, followed by that under the Private Benefit treatment, which, in turn, is followed by that in the control condition.*

Lastly, based on Babcock et al. (2017), we expect that women are more likely to accept tasks with low promotability, such as providing local public goods. We expect that women in our sample are less likely to click the link to let their co-author(s) take over the task of metadata contributions.

Hypothesis 4. *For articles with more than one author, women are less likely to click the link to let their co-authors contribute metadata.*

2.5 Random assignment and balancedness check

We randomly assign experimental conditions at the article level and block by three basic characteristics of the article: the network position of an article[3], the number of authors in an article, and its year of publication. Based on our selection algorithm, articles are included from simple components or randomly chosen from a complex component in the reduced graph. Further, simple components can be naturally occurring, or can be generated through the network-trimming exercise. In practice, we sort the list of articles sequentially by (network position, number of authors, year of publication), and reshuffle the ordering within each group. To complete random assignment, we enumerate through the sorted and reshuffled list and iteratively assign C, T_{priv}, T_{social} for all articles we include in the experiment. For more detailed random assignment procedures, please refer to Section C. The random assignment is summarized in Table 1.

Table 2 reports the summary statistics of pre-treatment characteristics, broken down into the three experimental conditions. Panel A presents the article attributes of the included articles and Panel B provides demographic information for the authors. Columns (1) through (3) report the mean as well as standard errors. We perform χ^2 tests on joint orthogonality across the treatments and reports the associated p-values in column (4). As none of the reported p-values is less than or equal to 0.05, we argue that our block-random

[3]As detailed in Section C, we draw articles from components of in the network. The network position of an article is defined by its generating component, which can be a simple natural component, a simple component that belongs to a complex natural component and a complex component in the trimmed graph.

Experimental Conditions:	Control	T_{priv}	T_{social}
Number of articles	487	487	486
Number of authors	1007	1013	1003

Table 1: Number of Observations after Random Assignment

assignment produced balanced experimental groups along observable characteristics. To further verify balancedness over treatment conditions, at the article level, we formulate the following regression framework and test for the joint hypothesis that $\beta_1 = \beta_2 = 0$. We use multinomial logistic specifications and the joint test has p-value = 0.23.

$$\begin{aligned}\text{Treatment}_i = \beta_0 + \beta_1 \times \text{Female}_i + \beta_2 \times \text{YearsAfterPhD}_i \\ + \mathbf{B}_{block} \times \text{Block-byVariables}_i + \epsilon_i\end{aligned}$$

Similarly, at the article level, we formulate the following regression framework and test for the joint hypothesis that $\beta_1 = \vec{\beta}_2 = 0$. We used the following multinomial logistic specifications and the joint test has p-value = 0.33.

$$\begin{aligned}\text{Treatment}_j = \beta_0 + \beta_1 \times \text{NumReference}_j + \vec{\beta}_2 \times \text{JournalDummy}_j \\ + \mathbf{B}_{block} \times \text{Block-byVariables}_j + \epsilon_j\end{aligned}$$

where j indexes articles.

Table 2: Attributes of included articles and participants, by Conditions

	Control	Findability + Private	Findability + Social	p-value
	(1)	(2)	(3)	(4)
Panel A: Article attributes				
Number of references	26.682	25.532	26.879	0.394
	(0.794)	(0.683)	(0.778)	
Journals:				
AEA Papers and Proceedings	0.047	0.066	0.045	0.290
	(0.010)	(0.011)	(0.009)	
AEJ: Applied Economics	0.109	0.119	0.146	0.192
	(0.014)	(0.015)	(0.016)	
AEJ: Economic Policy	0.144	0.117	0.160	0.144
	(0.016)	(0.015)	(0.017)	
AEJ: Macroeconomics	0.097	0.140	0.107	0.088
	(0.013)	(0.016)	(0.014)	
AEJ: Microeconomics	0.055	0.064	0.045	0.451
	(0.010)	(0.011)	(0.009)	
American Economic Review	0.507	0.441	0.451	0.084
	(0.023)	(0.023)	(0.023)	
AER: Insights	0.000	0.008	0.004	0.135
	(0.000)	(0.004)	(0.003)	
Journal of Economic Literature	0.006	0.000	0.004	0.246
	(0.004)	(0.000)	(0.003)	
Journal of Economic Perspectives	0.035	0.045	0.037	0.684
	(0.008)	(0.009)	(0.009)	
Observations	487	487	486	
Panel B: Author attributes				
Female	0.240	0.226	0.211	0.300
	(0.013)	(0.013)	(0.013)	
Years after PhD	16.889	17.915	17.685	0.109
	(0.358)	(0.370)	(0.359)	
Observations	1007	1013	1003	

[1] Columns (1) through (3) report average values and column (4) reports the p-value testing the joint orthogonality across treatments. Standard errors are provided in parentheses.
[2] Article attributes used for block-random assignment are omitted.

2.6 Power analysis

Suppose that there are two treatments X and Y. There are n_x participants who setting, $n_x < N_x$ where the later also include participants who did not open the email. Let x and y denote the number of participants responding positively in each of the two treatments. Then, we have

$$x \sim Bi(n_x, p), \qquad y \sim Bi(n_y, q)$$

We want to test whether there is a treatment effect on the probability of positive response. The null and alternative hypotheses are stated as:

$$H_0 : p = q, \qquad H_1 : p > q$$

We run simulations using the two proportion z-test over repeated realizations of the sample proportions. For a given pairs of (p, q), we set the significant level $\alpha = 0.05$, and power $1 - \beta = 90\%$. We repeatedly generate sample proportions with a given group size $n_x = n_y$ and calculate the test statistics. By iterating the exercise 10,000 times, we approximate the power using the proportion of rejected tests out of the 10,000 tests. With $n_x = n_y = 430$ and $(p, q|p \in \{0.1, ..., 0.99\}$ and $q < p - 0.1)$, i.e., the minimum detectable effect size of 10 percentage points, the simulated power for each parameter value is above 0.9.

Therefore, each of our experimental conditions should have a minimum number of 430 articles. Our random assignment leads to 487 or 486 articles in each condition, exceeding the minimum.

2.7 Pre-analysis plan

We frame all our hypotheses as pairwise comparisons. In this section, we outline the analysis we plan to run after finishing data collection.

2.7.1 Willingness to contribute

Upon receiving the email, participants first choose whether to proceed and provide additional metadata themselves, or let their co-author(s) contribute (when applicable), or opt out completely. To test Hypotheses 1 and 2, we estimate the treatment effects on the participants' willingness to contribute using the following regression framework:

$$\begin{aligned} r_i = \beta_0 + \beta_1 \times \text{Findability\&Private}_i + \beta_2 \times \text{Findability\&Private\&Social}_i \\ + \mathbf{B}_{article} \times \text{article-attributes}_i + \mathbf{B}_{subject} \times \text{author-attributes}_i + \epsilon_i \end{aligned}$$

where the dependent variable r_i is participant i's choice, which is ordinal and can be positive (1), null response (0) or negative (-1)[4]. The independent variables include the treatment dummies (Findability&Private and Findability&Private&Social), the article-level controls and the author-level attributes. Article-level controls include the year of publication, the number of co-authors on the paper, the number of references cited in the paper, network position of the article and the journal of publication. For author-level attributes, we include the gender and the number of years after the participant's PhD.

2.7.2 Metadata contribution

We measure the contribution of metadata at the article level, where individual contributions are aggregated through our metadata contribution interface that allows for sequential inputs. If all edits by a group of co-authors are performed sequentially, those who start later will be presented with the up-to-date contributions by those who start earlier. The aggregate contribution is the latest metadata content provided by the last author. In the event where multiple co-authors on a paper enter experiment interface simultaneously, we will manually aggregate the metadata contribution by merging their contents.

To test Hypothesis 3, we estimate the treatment effects on the overall contribution for metadata fields using the following regression framework:

$$\begin{aligned} Y_j = \beta_0 + \beta_1 \times \text{Findability\&Private}_j + \beta_2 \times \text{Findability\&Private\&Social}_j \\ + \mathbf{B}_{article} \times \text{article-attributes}_j + \mathbf{B}_{subject} \times \text{averaged-author-attributes}_j + \epsilon_j \end{aligned}$$

where j indexes the articles. The dependent variable, Y_j, is the contribution rate denoting the percentage of completed metadata fields for the article. Findability&Private$_j$ and Findability&Private&Social$_j$ are dummy variables representing the treatment status of article j. Article-level controls include the year of publication, the number of co-authors on the paper, the number of references cited in the paper, network position of the article and the journal of publication. For author-level attributes, we include the fraction of female participants and the average number of years after the authors' PhD.

2.7.3 Gender

To test Hypothesis 4, we conduct a sub-sample analysis by including only authors from articles with more than one author and estimate the gender effect on the participants' willingness to delegate to co-author(s) using the following regression framework:

$$\begin{aligned} r_i = \beta_0 + \beta_1 \times \text{Findability\&Private}_i + \beta_2 \times \text{Findability\&Social}_i + \beta_3 \times \text{Female}_i \\ + \beta_4 \times \text{Female}_i \times \text{Findability\&Private}_i + \beta_5 \times \text{Female}_i \times \text{Findability\&Private\&Social}_i \\ + \mathbf{B}_{article} \times \text{article-attributes}_i + \mathbf{B}_{subject} \times \text{author-attributes}_i + \epsilon_i \end{aligned}$$

where the dependent variable r_i has binary ordinal categories: delegate-to-coauthors (+1) or not (-1). One might argue that the null response of not clicking on any link might be interpreted as strategically delegating the task to a co-author. We will use both interpretations in our analysis. We capture the potentially heterogeneous treatment effect

[4] We collect email-opening events, and consider participants who open our email as treated participants. According to the email templates, we consider clicking on the first hyperlink to join the metadata contribution interface as a positive response. For participants who click on the other two opt-out hyperlinks, we consider their response as negative. The null responses include those who opened the email but did not click on anything in the email.

by interacting Gender with the treatment dummies. The independent variables include the treatment dummies (Findability&Private, Findability&Private&Social), the article-level controls and the author-level attributes. Article-level controls include the year of publication, the number of co-authors on the paper, the number of references cited in the paper, network position of the article and the journal of publication. For author-level attributes, we include the number of years after PhD.

Appendix

A Meatadata Fields

Our metadata fields are designed to take into consideration two relevant sources, the AEA guidance[5] and the current set of available fields on openICPSR, which may have been designed for curating survey research.

1. Subject Terms (e.g., “Machine Learning”, “Randomized Control Trial”, “Nudges”, ...)
2. Geographic coverage (e.g, “United States”, “Florida, U.S.”, “Indonesia”, ...)
3. Time period(s) (e.g., “1982-2008”)
4. Universe (text-field, e.g. “Adult noninstitutionalized population of the United States living in households.”)
5. Data Type(s) (a drop-down menu, include experimental data, observational data, survey data . . .)
6. Collection Notes (A description of technical details and other characteristics of the data collection (such as unique authoring, dissemination, or processing information) that cannot be recorded in the other metadata fields but constitute important information for the user.)
7. Data Source
8. Unit(s) of Observation

For each metadata field, we provide a form field on the experiment interface to collect the input. Since not all metadata fields are applicable for a given study, we instruct the authors to put in “N/A” if the metadata field does not apply. Also, each metadata field is accompanied with a “help tip”, a blue icon with question mark that provides further explanation of the field when activated.

B Data Preparation

In this section, we describe the sources used to prepare for the experiment and document how each source was collected. The goal is to build a network for random assignment, with unique nodes and valid email addresses for all participants.

[5]See https://aeadataeditor.github.io/aea-de-guidance/data-deposit-aea-guidance.htmlchecklist, Guidance on how to deposit data at the AEA Data and Code Repository.

B.1 Overview of sources

For this experiment, we consider all data deposits in the AEA Data and Code Repository that have been migrated through the AEA Repository migration[6]. As the official migration record is incomplete[7], we scrape the search page for all AEA deposits on openICPSR website and collect the URL for every deposit. From the individual openICPSR webpage for each data deposit, we are able to extract the full set of authors together with the AEA_DOI and openICPSR_ID. Migrated studies are identified by their release date, and are included if the release date falls in the following set of dates: {2019-10-11, 2019-10-12, 2019-10-13, 2019-12-06, 2019-12-07}.

We generate the following set of records from various sources:

1. From openICPSR website, we collect all publically available information on the study-page, for those studies that were migrated to openICPSR. This include:
   - All publically available metadata fields;
   - openICPSR_DOI for the data deposit and AEA_DOI for the original publication
2. From Crossref, we query using AEA_DOI, and get:
   - Structured name that are parsed into multiple fields (given name, family name, suffix ...)
   - Institutional affiliation (parsed originally from the footnote field)
3. From raw PDF versions of the paper, we look for texts on the first page in the footnote area for the following fields.
   - Lastname,
   - Institutional affiliation, and
   - Email address (usually in parentheses)

Auxiliary records include emails for corresponding authors from the publisher, which does not cover the full population of participants and contains outdated emails for those who changed institution affiliation since 2018.

B.2 Disambiguation of author names

One crucial step is to disambiguate the author names in our record, which impacts the network structure, but is labor intensive. If we treat variants of the name of the same

[6] `https://aeadataeditor.github.io/aea-supplement-migration/programs/aea201910-migration.html`

[7] The "Data files for AEA Repository migration" on openICPSR has 2000+ studies, while the total number of migrated studies is 3,073.

author as multiple individuals, which would be the case if we choose to use the "string-value" of their names from the official records, we will end up with more disconnected components which should have belonged to the same giant component. Having a different network configuration will lead us to a different set of units for random assignment.

We did not expect this to be a problem as we have the official publication record. However, as it turns out, even within the official records, duplicates can be introduced in multiple forms. In order to represent the authors uniquely in the deposit network, we resorted to assign all variants of the names for the same author with a unique SubjectID. This is done by pulling from both official publishing records available through openICPSR and Crossref.

We took two iterations to generate a minimal set of SubjectIDs for the participants in our experiment. We use results from Iteration 1 as a benchmark, and use result from Iteration 2 for production. We are able to identify 4,320 unique authors from a total of 4,503 different names.

B.2.1 The origin of duplicates

The origin of duplicates of names is clear. The source page on the AEA website contains the variants of the name for the same economist. For example, see Figure 1, where "John M. Roberts" and "John Roberts" were listed, respectively. The variants of names then propagate to records on openICPSR and Crossref.

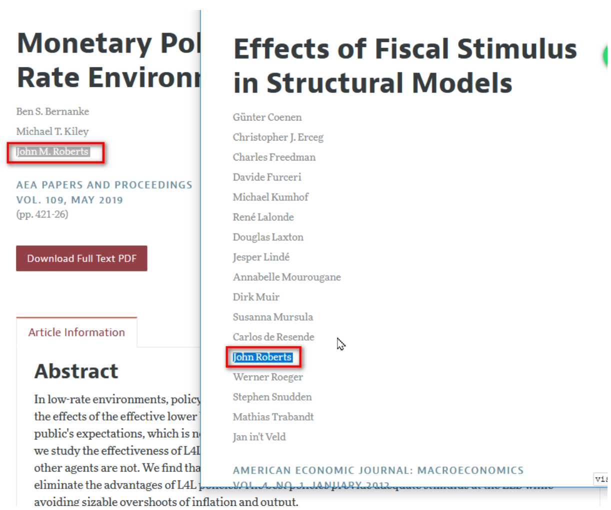


Figure 1: Source of the closed-duplicates from original record

B.2.2 Iteration 1: Name disambiguation

Since all the authors' names are from official publication record, we treat the spelling of first and last names are correct[8]and assume all variants of the name are coming from omitting middle name or wrong spelling of it. Also, we consider the edge cases where foreign names are not properly transferred.

In our disambiguation practices, we encountered the following edge-cases as well:

- Foreign names with special characters, thereby with `utf-8` encoding;
- Authors with mis-spelled firstname that has been "shortened"
- Authors postfixing/extending lastname to the record.

In bulk part, matching by first and last name identifies the same authors and we label all occurrences of the variants of name-spelling to an unique SubjectID. This attempt brought down the number of unique authors from 4,503 to 4,409.

B.2.3 Iteration 2: name-matching using openICPSR + Crossref record - match by first- and last-name

This iteration leverages on the fact that our publication record contains the list of DOI for the published papers. In particular, we assign SubjectID to records of author-name + DOI, and resolve inconsistencies along the way. In total, from the 3,070 publications, there are 7,251 author-name + DOI records. At the high level, we enumerate through the set of names and assign an ID to each unique author-name + DOI record. Practically, we attempt for 4 independent assignment of ID, and perform two rounds of consolidation of the SubjectID field.

Sources of SubjectID labeling To begin with, for both openICPSR records and Crossref records, we assign two sets of SubjectID as detailed below:

1. For names in openICPSR record, assign unique ID to unique combination of "first-name" and "lastname" field (these fields does not exist in the raw dataset and are parsed using simple rules[9]);
2. For names in openICPSR record, assign unique ID to each unique spelling of the full name, and unify the IDs belonging to the set of known "pairs of name-variants" that refer to the same individual. (This list if composed manually in Attempt 2);
3. For names in Crossref record, assign unique ID to unique combination of "firstname" and "lastname" field (these fields are provided by Crossref per the query of an DOI, yet, the middlename may enter either the firstname or the lastname field);
4. For names in Crossref record, manually assign unique ID to based on firstname, lastname and institutional affiliation.

[8] To our best knowledge, among 3070 publications we consider in our experiment, only one author has her last name spelled wrong in the original AEA record. This authors has three publications in total, and we assigned her a unique ID as we did with all other close-variants of names that represent the same author.

[9] From the raw_full_name, we parse by white spaces and keep the first word as firstname; then, from the rest of the "bag of words", we take the longest as the lastname. Of cause, this is not perfect, as "FFF M LLLL" and "FFF LLLL" will be assigned with two distinct SubjectID.

Consolidation of SubjetID For the consolidation of SubjectID field, we adopt the following steps to first identify the set of observations with buggy SubjectID from *a pair of SubjectID* and perform the fix. Let the pair of SubjectID be ID-V1 and ID-V2:

1. For each set of SubjectID, use the associated DOI field to compose the publication list under the "name" of the SubjectID;

2. For each observation, if the publication list under ID-V1 is different from the publication list under ID-V2, we extract both records for further inspection;

3. We manually inspect the discrepencies between ID-V1 and ID-V2, and "fix" the wrongly labeled SubjectID when appropriate[10];

4. Upon performing the fix (relabelling of ID) for both ID-V1 and ID-V2, we end up with a one-to-one mapping between ID-V1 and ID-V2, where the list of publications are identical regardless of how it is extracted.

In practice, we perform two rounds of consolidation exercises: first, we consolidate the two openICPSR records and the two Crossref records restively and compose openICPSR_ID and Crossref_ID. In the second round, we consolidate the two ID by first merging the records by (fuzzy) author-name + DOI [11]and then performing the consolidation steps detailed above.

Result At the end of the process, we obtain 4,320 unique SubjectIDs. Without any of the error correction, 4,503 participants would have existed based on names provided in the original record.

B.2.4 Exceptions

In our disambiguation process, we uncovered a list of distinct authors with similar names. These pairs are generated throughout multiple rounds of iteratively assigning and correcting the `SubjectID` field as mentioned in Appendix B.2.3. Here are 6 pairs of names that we have identified that belong to different individuals:

1. Michael P. Devereux and Michael B Devereux

2. Robert Gordon and Robert J. Gordon

3. Benjamin B Lockwood and Ben Lockwood

4. Benjamin M. Marx and Benjamin Marx

[10]For example, ID-V1 may assign different SubjectID for FFF LLLL and FFF M LLLL, while ID-V2 assigns identical ID to these two records. We fix ID-V1 by keeping one ID in this case. ID-V2 can be fixed in an identical manner. In rare cases, we need to fix both ID-fields.

[11]Merging by strings of names in both record can only cover 3069 records, while we have 7251 in total. We ended up pairing openICPSR record with Crossref record by pairing each name in the openICPSR record with the Crossref-author-name in the same DOI that matches the openICPSR name the best.

5. A. Banerji and Abhijit Banerjee
6. Michal Bauer and Michael D. Bauer

B.3 Email collection notes

We combined multiple sources to make sure our email database is up-to-date. In early March 2020, we scraped all available studies deposited in AEA deposits and conducted a Google search for all the authors that we found. There are a total of 4,503 names from the web-scraping exercise. From AEA, we have a separate set of contact emails for the corresponding authors in the 5 major publication outlets of AEA, which have been updated in 2018. To generate a comprehensive set of contact emails for all authors' whose AEA publications have been migrated to openICPSR AEA Data Deposit, we adopt the following data cleaning strategies:

1. For those emails that coincide between the web-scrapes and the contact list of corresponding authors, we keep these emails as "up-to-date". Further, we ask an RA to figure out the Gender and Year of PhD using the credible source where the valid email is scraped from. This is a relatively fast task, at a rate of with 30 records an hour, totaling 1,023 records.

2. For the rest of the emails, we employed two RA to manually verify the emails and generate additional labels

   - Gender
   - Year of PhD
   - Primary website

   Since our RA needs to verify the email by cross-referring various sources (CV $\succ$ Personal Website $\succ$ Departmental/Workplace/Organization Website $\succ$ NBER website), it takes a considerable amount of time to verify each record. In total, it took 211 hours to finish 3,696 rows (17 rows/hour). Two RAs were working full-time in April 2020 on this task.

All emails as well as demographic information are collected on web-pages with public access. Due to the volume of the total number of records, all email records were verified only one RA. Without double entry, around 5% of the emails we collected are no longer up-to-date by the time we plan to launch the study[12]. To make sure our contact email is valid for the participants that we include in the experiment, we conducted another round of email-validation for the 3,023 participants that we choose to include based on the the random assignment. The production rate is 50 rows/hour.

[12] In preparation for launch, we did a spot check of email qualities by revisiting the Primary Website and verify our email record against the listed contact email in CV. If no CV is found, we used the contact email from the latest working paper by the author. Among the 1,124 records we checked as of 2020-08-01, 54 needs update and amounts to 4.8%. Among those emails that we updated, less than 1% are due to human error (picking up a completely wrong email) and the remaining majority are due to change of institutional affiliations.

C Random assignment on the network

In this section, we document the set of "network-trimming" routines we adopt to generate the set of *independent* deposits we include in the experiment.

C.1 The Deposit network: construction and manipulation

We build the network of authors based on co-authorship in data deposits, where all authors are uniquely identified by their names and all data deposits are identified through the DOI of the paper. Technically, we build a multigraph which permits multiple-edges between two nodes and also allows for self-loop. Respectively, this representation of the network keeps the record of multiple co-authored papers between two authors, and also allows for multiple single-authored papers.

Practically, we scraped the existing AEA Data and Code Repository on openICPSR and obtained a complete list of authors for each study in the AEA Data Deposit. We consider only those studies that were *migrated* in a series of data dump[13]. Table 3 offers a demonstration of the raw datafile.

Article ID	Author 1	Author 2	Author 3
$\mathbb{A}_1$	A_1	A_2	A_4
$\mathbb{A}_2$	A_2	A_4	
$\mathbb{A}_3$	A_1	A_3	
$\mathbb{A}_4$	A_5		

Table 3: Original format of the scraped data

Given our raw data structure, we build such multigraph by enumerating through the list of all articles and creating an edge with the `ArticleID` between any possible combination of two authors that belongs to the same article. In essence, we transform Table 3 into the edge-list in Table 4. Through the multigraph built through the edge-list,

ArticleID	Author1	Author 2	Note
$\mathbb{A}_1$	A_1	A_2	
$\mathbb{A}_1$	A_1	A_4	
$\mathbb{A}_1$	A_2	A_4	parallel edge
$\mathbb{A}_2$	A_2	A_4	parallel edge
$\mathbb{A}_3$	A_1	A_3	
$\mathbb{A}_4$	A_5	A_5	Self-loop

Table 4: Edge-list view

we are able to track how an edge is introduced into the graph through the article-labeling of the edges. We used Netowrkx in Python for manipulating the network[14].

[13]The migration had two waves, the first wave took three days (Oct 11 - Oct 13, 2019) and the second wave took two days (Dec 7 - Dec 8, 2019).

[14]graph-tool.skewed.de/ is another alternative with strictly better performance (intended for "real-

C.1.1 Motivation

One major threat for identifying treatment effects in a network setting is "spillover effect". It is the aggregated effects introduced by network interactions and may have been fancier names. This is especially hard to pin down due to the nature of networks: the sheer presence "spillover effect" is introduced by the network structure, and it is naturally attached to the *specific* network structure.

Here, we provide one way to cut off the channels for potential "spillover effect" by isolating each treated units to be a connected-*component*[15]in a trimmed network. Alternatives random assignment schemes that also cluster the treatments at the article level are less trackable compared to our network-trimming approach.

Throughout this section, we assume the unit of randomization is at the deposit (article) level. Nevertheless, there are fairly few other feasible randomization units at our disposal. In the rest of this section, we introduce how we preform the trimming exercise.

C.1.2 Community-detection algorithms

For a given connected graph, community-detection algorithms generate the "best partition" that "maximizes" the *modularity* measure. In principle, these algorithms identify the communities composed of closely connected individuals. In our setting, we extract the unit of randomization within an element of the "best partition" and establish the independence among all selected units.

Let g_i an g_j denote the "group" to which i and j belongs, with $g_i \in 1, \cdots, N$ and N is the total number of "groups", or communities/partitions. Then, Q, the *modularity*, measures "the extent to which like is connected to the like in a network" (Newman, 2017).

$$Q = \frac{1}{2}\sum_{ij} A_{ij}\delta_{g_i g_j} - \frac{1}{2}\sum_{ij}\frac{k_i k_j}{2m}\delta_{g_i g_j} = \frac{1}{2m}\sum_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta_{g_i, g_j}$$

where A_{ij} is the adjacency matrix for the graph, δ denotes the Kroncker delta[16], k_i, k_j denote the degree for node i and j and m is the total number of edges.

In practice, we choose from two well-adopted community-detection algorithms: Louvain algorithm (Blondel et al., 2008) and Leiden algorithm (Traag, Waltman, and van Eck, 2019). The Louvain algorithm identifies a partition of the graph that maximizes the *modularity*. As reported in Table 1 in Blondel et al. (2008), the Louvain algorithm achieves a higher modularity score and out-performs standard algorithms like Newman and Girvan (2004), Pons and Latapy (2006) and Wakita and Tsurumi (2007). The Leiden algorithm, as introduced in Traag, Waltman, and van Eck (2019), prevents further partitioning of

world" networks of considerable larger sizes), but have stricter restrictions on how nodes and edges are "serialized". We stick with Netowrkx for its native Python implementation.

[15]A connected-component, in our setting, is a subgraph in which any two vertices are connected to each other by at least an edge, and which is connected to no additions vertices in the supergraph.

[16]Kroncker delta is defined as: $\delta_{g_i, g_j} = \begin{cases} 0, & \text{if } g_i \neq g_j \\ 1, & \text{if } g_i = g_j \end{cases}$

well-connected components that can be taken apart under Louvain algorithm. In Section C.3, we compare the performance of Louvain algorithm against the Leiden algorithm.

C.2 Composition of the data-deposit network of AEA authors

With the full network, we have more than 887 *components*, where the giant component taking up more than one-half of all the authors $(2,008$ out of $4,320)$. On the other end, we also have a total of 666 *simple-components* that are composed of a set of authors who have published one and only one article in AEA outlets.

C.2.1 Simple vs Complex Components

For a given component, we enumerate through all its edges and account for the number of unique "data deposits" that introduced the edges. Then:

- If all edges in a component are introduced by a unique data deposit, we call it a simple component;
- If edges in a component are introduced by a collection of data deposits, we call it a complex component.

Note, this dichotomy is defined assuming a certain underlying network (supergraph). In our "trimming exercise" that follows, we will update the underlying network by *dropping* edges.

For our experiment, we would like to include all *simple-components* in a "trimmed graph". For *complex-components* with multiple articles, we pick one article at random to include in the experiment. The following section details the operation relevant for constructing such "trimmed graph".

C.2.2 Partition the Network and trim its edges

A partition of a network is a partition over the set of nodes in a network. One typical use of the partitioning algorithm is to identify "communities" in a network, where the algorithm generates the partition and researchers try to make sense of the composition of the components. For practical purposes, we use out-of-the-shelf Louvain algorithm to generate the partition.

Notes on getting "better" partitions from the community-detection algorithm For any given undirected network, the Louvain algorithm can generate a partition based on a seeded random search: the solution tries to maximize the modularity of the partition for a given graph, and this is a NP-hard problem. Although we cannot obtain the optimal solution, we still can improve the "quality" of the generated partition by choosing which subgraph to feed to the Louvain algorithm. Note, we started the investigation with Louvain algorithm. All comparisons of performance hold true for Leiden algorithm, which we eventually choose for production.

Given that our original network have 887 independent components, we perform the following two tests for an arbitrary component:

- Generate a partition of the graph using the full graph, and extract the partitions that belong to the component that we picked;
- Take the subgraph of the component, and generate a partition for the subgraph.
- Compute modularity for the subgraph using the two partitions: M_{full} is obtained through partitioning the full graph, and M_{sub} is obtained through partitioning the subgraph.

We repeated the computation for 1000 times, with a randomly generated "seed" that guarantee reproducibility. We confirm that the Louvain algorithm will perform better with the subgraph, with $M_{full} < M_{sub}$ across all the trials.

Which deposit to keep and which deposit to drop? From a given partition of a component, we construct the "Quotient graph", whose vertices are denoting the communities identified by the partition. Then, by construction, the edges in the Quotient graph are those that were "bridging" communities in the network, thus termed inter-community edges. Removing these inter-community edges from the original component will generate several connected-components from the previously connected subgraph. Lastly, with the inter-community edges removed, the remaining edges from the deposits that generated inter-community edges will no longer provide much help for our randomization scheme. To conclude, we drop all edges introduced by deposits that generated inter-community edges and we keep all deposits that were nested within the communities.

C.2.3 Iteratively identify and remove the inter-community edges

In practice, we stick to the *best practice* [17]for applying the community-detection algorithm and iteratively build a "drop list" until the community detection algorithm fails. In Algorithm 1, we provide the psuedo code for the operation.

Algorithm 1. ***Algorithm 1*** *Collect the list of deposits to drop based on inter-community edges*

Require: *G is the full graph and L_{drop} is a set of deposits to drop*

▷ *Main Function*

function TRIM_GEN_DROP_LIST*(G, L_{drop})*
  trimmed_graph = *G without any edges introduced by the deposits in L_{drop}*
  complex_component = *complex components in* ***trimmed_graph***

[17]Note, community-detection is an NP-hard problem and all we have are approximation algorithms that are sensitive to the initial input. As one would intuitively predict, the modularity score of a partition for a given component is higher when the partitioning algorithm is only told about the precise component. This is explained in full in section C.2.2

▷ *Iterate through all complex components in the trimmed graph*

for *component in complex_component* **do**

Generate partitions using a ***community-detection algorithm***

Generate quotient graph using the subgraph of the component and the partition

l_{drop} = *deposits that introduced the* ***inter-community edges***

$L_{drop} = L_{drop} \cup l_{drop}$

end for

return L_{drop}

end function

▷ *Initialise the initial list of dropped deposits with no edge removed*

L_{drop} = TRIM_GEN_DROP_LIST(G, $\emptyset$)

▷ *Iterate until* L_{drop} *is stable*

while $\text{TRIM\_GEN\_DROP\_LIST}(G, L_{drop}) \setminus L_{drop} \neq \emptyset$ ***do***

$L_{drop} = \text{TRIM\_GEN\_DROP\_LIST}(G, L_{drop})$

end while

In summary, for a given network, we first classify the (connected) components into simple vs complex components. For each complex component, we attempt to apply the community-detection algorithm and collect a "drop list" from articles that introduced the inter-community edges in the quotient graph. We repeat the algorithm over rounds of edge-trimming, until all complex components in the reduced graph shall withstand the community-detection algorithm of choice. We will discuss our choice of community-detection algorithm in the Section C.3.

C.3 Comparison of community-detection algorithms

The core step in Algorithm 1, as highlighted, is to partition a given complex component using a community-detection algorithm. We consider two candidates: Leiden algorithm (Traag, Waltman, and van Eck, 2019) and Louvain algorithm (Blondel et al., 2008). In this section, we summarize the numerical experiments we conducted to pin down which community-detection algorithm to employ.

For production, we choose to use Leiden algorithm for multiple reasons. Overall, less articles (DOI) were dropped during the trimming exercise with Leiden algorithm, and less authors were completely dropped from the experiment. In terms of total units (components) for random assignment, Leiden algorithm gave a comparable result as well. Aside from what is shown through the comparison in Table 5, when applied to individual components, the Leiden algorithm outperforms Louvain algorithm in 155 out of 222 complex components in terms of the modularity of the derived partition of the components. More over, there are 2653 participants in the 155 components as compared to the 395 participants in the remaining 67 components where Louvain algorithm gave a "slightly better partition". To conclude, we employ Leiden algorithm for our network-trimming exercise.

Metrics	Louvain	Leiden
Part 1: Review of dropped DOI and remaining components		
UniqueDOI_Dropped	1075.63 (0.21)	929.76 (0.17)
AuthorsDropped	1003.66 (0.34)	923.24 (0.35)
# Simple Compo	1208.20 (0.14)	1058.07 (0.15)
# Complex Compo	313.85 (0.10)	399.60 (0.07)
sum(Simple, Complex)	1522.05 (0.09)	1457.67 (0.10)
Part 2: Max and random set of included participants		
Max participants*	3125.35 (0.31)	3150.52 (0.37)
Rand participants	3072.25 (0.42)	3011.80 (0.63)

Table 5: Compare trimming output from 500 distinctive random seeds
Mean value is reported, with standard error in parenthesis.
* Max participants is collected from articles with most co-authors in each complex component.
Rand participants are collected from a randomly chosen article in each complex component.

C.4 Random assignment procedures

Since the Leiden algorithm is an approximation method, we ran the proposed network-trimming algorithm 1,000 times with distinct initial seeds to look for the realization where we drop the fewest participants from the trimming exercise[18]. Upon obtaining the reduced graph without intercommunity edges, we follow the following procedure to pick who to include in the experience and how the participants are assigned into treatment:

Step 1 Choose articles from components in the reduced graph, so that each component has only one article chosen: this is trivial for simple components where only one article is involved. For complex components that remain after iterations of trim-and-drop, they are well-connected and we pick the article with *most* number of authors. For complex components that remain, we pick at random an article from each component and include all participants into the experiment.

Step 2 We assign articles/components into experimental conditions, where we block by the network position of such component and by the number of authors in the article as chosen.

Step 3 Lastly, since a very small proportion of participants have missing emails or have deceased, we drop them *after* the random assignment.

We end up with a trimmed graph of $1,460$ components and $3,023$ participants. In our population of participants, 11 participants are deceased and 6 are having missing email addresses despite our best effort to look them up. We contact all the rest of participants

[18]According to our network-trimming algorithm, we drop all articles that introduced inter-community edges. For a given author, he/she is dropped completely if all his/her articles are dropped. Among the 1,000 repetitions, the mean of dropped authors is 922.665, with standard error 0.253. The realization with fewest dropped author has dropped 893 authors.

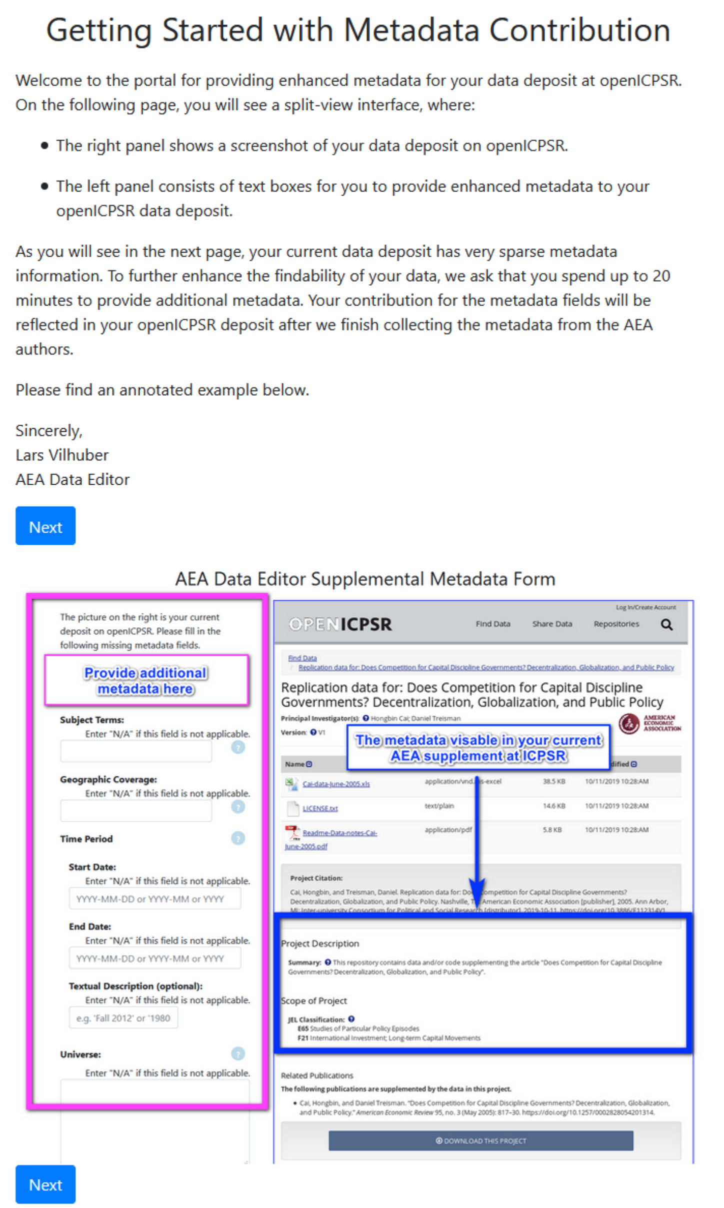
with valid email addresses. In total, there are $1,459$ articles with at least one author that has a valid email address according to our records.

D Experiment interface

We provide the screenshots of the experiment interface in this section. Each subject will enter the experiment interface with an individualized link. Participants are presented with the article they have published. We collect metadata fields on the second page of the experiment interface (Figure 3).

D.1 Mouse-over clarification text

For all the metadata fields that we collect, we provide a mouse-over "help-tip" that clarifies what *exactly* are expected for the metadata fields. The clarification text is adopted from the formal openICPSR metadata-editing interface as well as the AEA Data and Code Guidance document.



Getting Started with Metadata Contribution

Welcome to the portal for providing enhanced metadata for your data deposit at openICPSR. On the following page, you will see a split-view interface, where:

- The right panel shows a screenshot of your data deposit on openICPSR.
- The left panel consists of text boxes for you to provide enhanced metadata to your openICPSR data deposit.

As you will see in the next page, your current data deposit has very sparse metadata information. To further enhance the findability of your data, we ask that you spend up to 20 minutes to provide additional metadata. Your contribution for the metadata fields will be reflected in your openICPSR deposit after we finish collecting the metadata from the AEA authors.

Please find an annotated example below.

Sincerely,
Lars Vilhuber
AEA Data Editor

Next

AEA Data Editor Supplemental Metadata Form

Next

Figure 2: Experiment interface - Page 1: Introduction

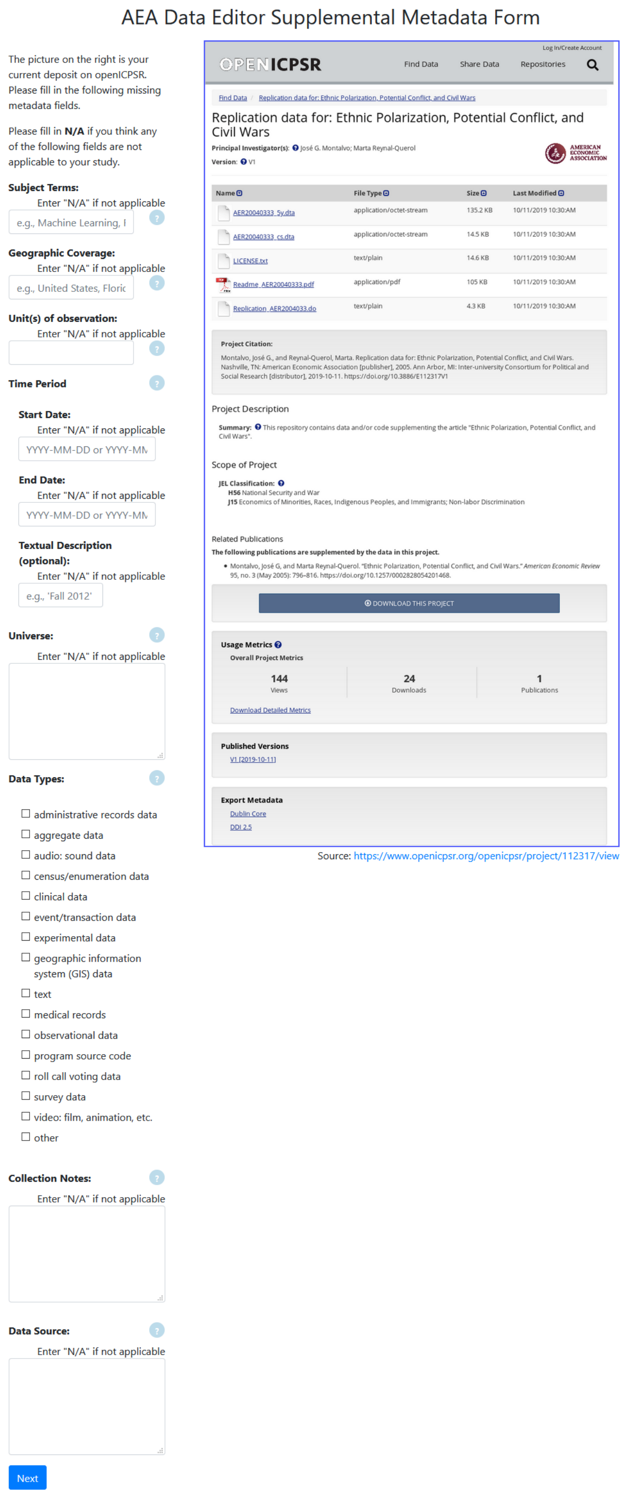
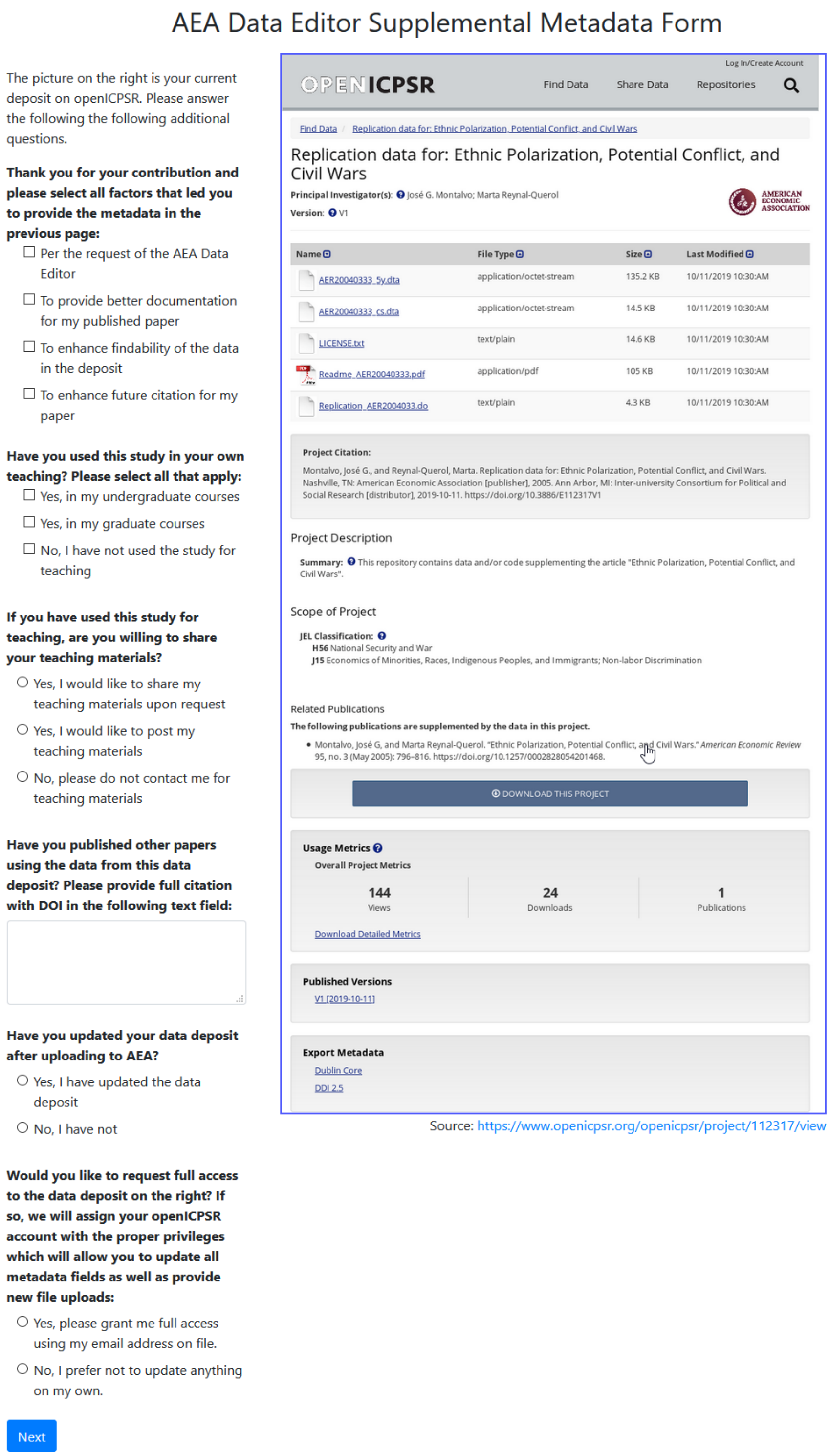


Figure 3: Experiment interface - Page 2: Collect metadata contribution



AEA Data Editor Supplemental Metadata Form

The picture on the right is your current deposit on openICPSR. Please answer the following the following additional questions.

Thank you for your contribution and please select all factors that led you to provide the metadata in the previous page:

- ☐ Per the request of the AEA Data Editor
- ☐ To provide better documentation for my published paper
- ☐ To enhance findability of the data in the deposit
- ☐ To enhance future citation for my paper

Have you used this study in your own teaching? Please select all that apply:

- ☐ Yes, in my undergraduate courses
- ☐ Yes, in my graduate courses
- ☐ No, I have not used the study for teaching

If you have used this study for teaching, are you willing to share your teaching materials?

- ○ Yes, I would like to share my teaching materials upon request
- ○ Yes, I would like to post my teaching materials
- ○ No, please do not contact me for teaching materials

Have you published other papers using the data from this data deposit? Please provide full citation with DOI in the following text field:

Have you updated your data deposit after uploading to AEA?

- ○ Yes, I have updated the data deposit
- ○ No, I have not

Would you like to request full access to the data deposit on the right? If so, we will assign your openICPSR account with the proper privileges which will allow you to update all metadata fields as well as provide new file uploads:

- ○ Yes, please grant me full access using my email address on file.
- ○ No, I prefer not to update anything on my own.

Next

Log In/Create Account

OPENICPSR Find Data Share Data Repositories

Find Data / Replication data for: Ethnic Polarization, Potential Conflict, and Civil Wars

Replication data for: Ethnic Polarization, Potential Conflict, and Civil Wars

Principal Investigator(s): José G. Montalvo; Marta Reynal-Querol

Version: V1

AMERICAN ECONOMIC ASSOCIATION

Name	File Type	Size	Last Modified
AER20040333_5y.dta	application/octet-stream	135.2 KB	10/11/2019 10:30:AM
AER20040333_cs.dta	application/octet-stream	14.5 KB	10/11/2019 10:30:AM
LICENSE.txt	text/plain	14.6 KB	10/11/2019 10:30:AM
Readme_AER20040333.pdf	application/pdf	105 KB	10/11/2019 10:30:AM
Replication_AER2004033.do	text/plain	4.3 KB	10/11/2019 10:30:AM

Project Citation:

Montalvo, José G., and Reynal-Querol, Marta. Replication data for: Ethnic Polarization, Potential Conflict, and Civil Wars. Nashville, TN: American Economic Association [publisher], 2005. Ann Arbor, MI: Inter-university Consortium for Political and Social Research [distributor], 2019-10-11. https://doi.org/10.3886/E112317V1

Project Description

Summary: This repository contains data and/or code supplementing the article "Ethnic Polarization, Potential Conflict, and Civil Wars".

Scope of Project

JEL Classification:
H56 National Security and War
J15 Economics of Minorities, Races, Indigenous Peoples, and Immigrants; Non-labor Discrimination

Related Publications

The following publications are supplemented by the data in this project.

- Montalvo, José G, and Marta Reynal-Querol. "Ethnic Polarization, Potential Conflict, and Civil Wars." *American Economic Review* 95, no. 3 (May 2005): 796–816. https://doi.org/10.1257/0002828054201468.

DOWNLOAD THIS PROJECT

Usage Metrics
Overall Project Metrics

144 Views | 24 Downloads | 1 Publications

Download Detailed Metrics

Published Versions
V1 [2019-10-11]

Export Metadata
Dublin Core
DDI 2.5

Source: https://www.openicpsr.org/openicpsr/project/112317/view

Figure 4: Experiment interface - Page 3: Collect additional information

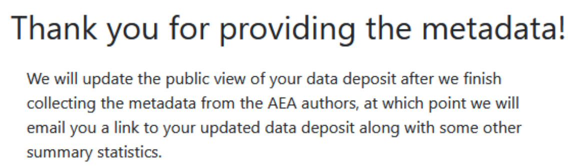


Figure 5: Experiment interface - Page 4: Finish page

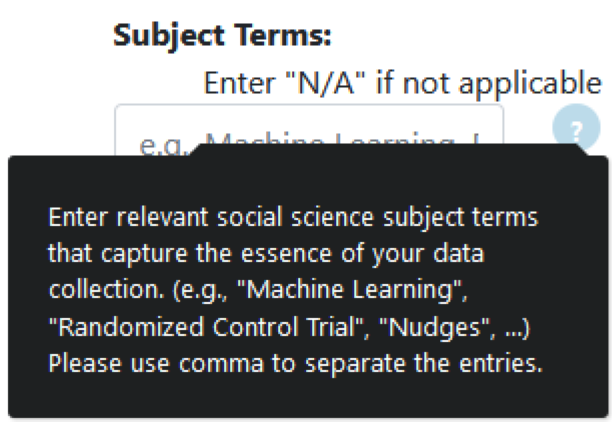


Figure 6: Mouse-over clarification text

References

Babcock, Linda, Maria P. Recalde, Lise Vesterlund, and Laurie Weingart (2017). “Gender Differences in Accepting and Receiving Requests for Tasks with Low Promotability”. In: *American Economic Review* 107.3, pp. 714–47. DOI: 10.1257/aer.20141734.

Blondel, Vincent D., Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre (2008). “Fast Unfolding of Communities in Large Networks”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.10, P10008. DOI: 10.1088/1742-5468/2008/10/P10008. arXiv: 0803.0476.

Chapman, Adriane, Elena Simperl, Laura Koesten, George Konstantinidis, Luis-Daniel Ibáñez, Emilia Kacprzak, and Paul Groth (2020). “Dataset Search: A Survey”. In: *The VLDB Journal* 29.1, pp. 251–272. DOI: 10.1007/s00778-019-00564-x.

Chen, Yan, Rosta Farzan, Robert Kraut, Iman YeckehZaare, and Ark Fangzhou Zhang (2020). “Motivating Contributions to Digital Public Goods: A Personalized Field Experiment on Wikipedia”. University of Michigan Working Paper.

Greenberg, Jane, Maria Cristina Pattuelli, Bijan Parsia, and W. Davenport Robertson (2001). “Author-Generated Dublin Core Metadata for Web Resources: A Baseline Study in an Organization”. In: *International Conference on Dublin Core and Metadata Applications* 0.0 (0), pp. 38–45.

Gregory, Kathleen (2020). “A Dataset Describing Data Discovery and Reuse Practices in Research”. In: *Scientific Data* 7.1 (1), p. 232. DOI: 10.1038/s41597-020-0569-5.

Newman, Mark (2017). *Networks: An Introduction*. 2 edition. Oxford ; New York: Oxford University Press.

Newman, Mark EJ and Michelle Girvan (2004). “Finding and Evaluating Community Structure in Networks”. In: *Physical review E* 69.2, p. 026113.

Piwowar, Heather A., Roger S. Day, and Douglas B. Fridsma (2007). “Sharing Detailed Research Data Is Associated with Increased Citation Rate”. In: *PLoS ONE* 2.3. Ed. by John Ioannidis, e308. DOI: 10.1371/journal.pone.0000308.

Pons, Pascal and Matthieu Latapy (2006). “Computing Communities in Large Networks Using Random Walks”. In: p. 28.

Santos, Luiz, Mark Wilkinson, Arnold Kuzniar, Rajaram Kaliyaperumal, Mark Thompson, Michel Dumontier, and Kees Burger (2016). “FAIR Data Points Supporting Big Data Interoperability”. In: p. 10.

Traag, V. A., L. Waltman, and N. J. van Eck (2019). “From Louvain to Leiden: Guaranteeing Well-Connected Communities”. In: *Scientific Reports* 9.1 (1), p. 5233. DOI: 10.1038/s41598-019-41695-z.

Wakita, Ken and Toshiyuki Tsurumi (2007). “Finding Community Structure in Mega-Scale Social Networks: [Extended Abstract]”. In: *Proceedings of the 16th International Conference on World Wide Web*. WWW ’07. Banff, Alberta, Canada: Association for Computing Machinery, pp. 1275–1276. DOI: 10.1145/1242572.1242805.