ISSN 2161-3974
Full-Length Paper

Journal of eScience Librarianship 13 (2): e685
DOI: https://doi.org/10.7191/jeslib.685

Journal of eScience Librarianship
putting the pieces together: theory and practice

Toward Enhanced Reusability: A Comparative Analysis of Metadata for Machine Learning Objects and Their Characteristics in Generalist and Specialist Repositories

Stephanie G. Labou, University of California San Diego, San Diego, CA, USA, slabou@ucsd.edu
Abigail Pennington, University of California San Diego, San Diego, CA, USA
Ho Jung S. Yoo, University of California San Diego, San Diego, CA, USA
Michael Baluja, University of California San Diego, San Diego, CA, USA

Abstract

Objective: The rapidly increasing prevalence and application of machine learning (ML) across disciplines creates a pressing need to establish guidance for data curation professionals. However, we must first understand the characteristics of ML-related objects shared in generalist and specialist repositories and the extent to which repository metadata fields enable findability and reuse of ML objects.

Methods: We used a combination of API queries and web scraping to retrieve metadata for ML objects in eight commonly used generalist and ML-specific data repositories. We assessed both metadata schema and characteristics of deposited ML objects, within the context of the widely adopted FAIR Principles. We also calculated summary statistics for properties of objects, including number of objects per year, dataset size, domains represented, and availability of related resources.

Received: April 24, 2023 **Accepted**: February 23, 2024 **Published**: June 20, 2024

Keywords: FAIR, machine learning, data repository, reusability, interoperability, findability, data curation, metadata schema, metadata crosswalk

Citation: Labou, Stephanie G., Abigail Pennington, Ho Jung S. Yoo, and Michael Baluja. 2024. "Toward Enhanced Reusability: A Comparative Analysis of Metadata for Machine Learning Objects and Their Characteristics in Generalist and Specialist Repositories." *Journal of eScience Librarianship* 13 (2): e685. https://doi.org/10.7191/jeslib.685.

Data Availability: Data, analysis code, and all supplemental tables are available in UC San Diego Library Digital Collections: https://doi.org/10.6075/J0JS9QMH. Appendix is available under the article Supplementary Files.

Abstract Continued

Results: Generalist repositories excelled at providing provenance metadata, specifically unique identifiers, unambiguous citations, clear licenses, and related resources, while specialist repositories emphasized ML-specific descriptive metadata, such as number of attributes and instances and task type. In terms of object content, we noted a wide range of file formats, as well as licenses, all of which impact reusability.

Conclusions: Generalist repositories will benefit from some of the practices adopted by specialists, and specialist repositories will benefit from adopting proven data curation practices of generalist repositories. A step forward for repositories will be to invest more into use of labels and persistent identifiers to improve workflow documentation, provenance, and related resource linking of ML objects, which will increase their findability, interoperability, and reusability.

Introduction

As a result of increased access to data and computational resources, the barrier of entry for using machine learning (ML)—which involves training algorithms to identify patterns in data and make predictions based on previously unencountered data (Louridas and Ebert 2016)—in research has been substantially lowered. This has led to a corresponding growth in the use of ML across various disciplines. As a case in point, a search of arXiv for "machine learning" in September 2022 returned nearly 147,000 pre-prints, with almost 43,000 of those from the previous 12 months. Similarly, a September 2022 search of Web of Science for "machine learning" returned over 300,000 results, with approximately half published since 2020.

Many repositories, including those managed by academic libraries, are currently receiving research data submissions composed of apparent ML objects, or can anticipate receiving an increasing number of ML objects in the near term. Consequently, understanding the "ML object"—which we refer to here as the published set of components that comprise the ML research output, i.e., data, code, model, and associated documentation (see Publio et al. 2018 for example framework)—is becoming a necessity for the data curation community. Some ML practitioners are working to develop and recommend standards for sharing ML objects, as are research data communities like the Research Data Alliance. However, to date, data curation professionals have lacked clear and consistent guidance for best practices for ML specifically, although standalone guidelines exist for many data and code formats, e.g., Data Curation Network primers.

Before standardized best practice recommendations can be developed, the data curation community must first understand the current state of documentation and sharing with regards to ML research outputs. That is: what components of ML are being publicly shared and to what extent are metadata appropriately leveraged to enable findability, interoperability, and reuse of ML research outputs?

To provide a framework for this question, we use the FAIR (Findable, Accessible, Interoperable, Reusable) Principles (Wilkinson et al. 2016), which were written with research data in mind and which recent papers (Lamprecht et al. 2020; Katz, Gruenpeter, and Honeyman 2021a) have proposed extending to research software. Further, Samuel, Löffler, and König-Ries (2021) and Katz, Psomopoulos, and Castro (2021b) have advocated for applying FAIR guidelines to ML objects, which are a complex synthesis of software, data and workflows.

Findable for ML

For data to be accessible, interoperable, or reusable, they must be findable (Juty et al. 2020). In the case of ML objects, there is an additional consideration of the complex dependencies between components that make up the ML research output. All of these components, such as the training and test datasets, algorithms, and tasks, have the potential to be assigned identifiers if they are clearly and consistently labeled, and independently accessible in discrete locations (URLs). With the potential for distribution of such components scattered across multiple repositories and platforms, identifiers and linkages between related resources are especially crucial for providing context to individual, potentially reusable, ML components.

Accessible for ML

Accessibility in the framework of the FAIR principles is a repository system-level requirement, i.e., an http protocol, rather than data-level or within the purview of an individual depositor. In the context of ML, accessibility may be significantly improved via software availability as well as programmatic accessibility (and interoperability) at scale, through application programming interfaces (APIs) (see Lamprecht et al. 2020). Accessibility beyond the existence of automated access to metadata is outside the scope of this study, and is not addressed further in the analysis.

Interoperable for ML

The relatively recently proposed metadata schema ML-Schema, which leverages several state-of-the-art ML metadata schemas (MEX, OntoDM-core, Exposé, and DMOP), has the potential to improve interoperability of ML experiments regardless of computing platform (Publio et al. 2018). In addition to providing within-experiment descriptive documentation and community supported vocabularies, ML objects should explicitly link using persistent identifiers (PIDs), when available, and URLs when not, to external resources that provide context for the research therein. These resources may include training data located elsewhere, publicly available source code or algorithms used in the study, and articles reporting the results of the ML research.

Reusable for ML

Much of the research and discussion on the Reusable principle of FAIR is focused more narrowly on *reproducibility*, rather than more generic *reuse*. ML experiment reproducibility gaps are often a consequence of incomplete and/or poorly documented components (training data, source code or algorithms and code interdependencies), properties (iterations, parameters, configuration, methods, techniques, workflows), data provenance, and computing environments (software packages and versions) (Pineau et al. 2020; Samuel, Löffler, and König-Ries 2021), all of which impact reusability as well.

ML experiment documentation using a formal metadata schema can help to fill this documentation gap and enable scientists and engineers to assess their credibility (Esteves et al. 2015) and suitability for their particular application, thereby facilitating reuse. While the concept of data provenance has not gained much traction in ML communities (Gebru et al. 2021) it is a core element of richly described research data and crucial in the case of ML to avoid producing outputs based on incomplete, incorrect, or biased data (Reference Model for an Open Archival Information System, World Economic Forum Global Future Council on Human Rights, 2018). Checklists are also emerging as a tool for closing these gaps (Norgeot et al. 2020; Pineau et al. 2020; Sengupta et al. 2020; Gebru et al. 2021) and tools that support reproducibility and enhance reusability are on the rise, such as the Jupyter Notebook application extension, ProvBook, which captures and saves provenance for experiment iterations (Samuel and König-Ries 2018).

Objectives

While ML-specific tools and practices described above are useful for ML practitioners, they were not necessarily developed for research data curators unfamiliar with the intricacies of ML workflows, nor do they necessarily fit into existing repository schemas. Of more immediate concern, we don't have a strong sense of how well positioned current platforms are for hosting and sharing ML research.

Therefore, the specific objectives for this project were to survey ML research objects currently published in commonly used scientific data repositories in order to: (1) assess how metadata fields vary across data repositories likely to host ML objects, within the framework of FAIR principles, (2) document high-level characteristics of ML objects that are currently shared in these repositories, with an emphasis on characteristics related to reusability, and (3) identify strengths and growth areas for repositories as well as areas for greater awareness for data curators and researchers.

Methods

When considering where ML outputs might be deposited, we identified two main categories: specialist repositories and generalist repositories.

We define "specialist repositories" as repositories that specialize in, or are designed for, ML data, code, or other components. The repositories in this group are not an exhaustive collection of ML-centric repositories, but rather, represent some of the more popular and widely used repositories, as determined during communications with ML researchers from UC San Diego as well as our own experiences interacting with faculty and students engaged in ML research.

The specialist repositories assessed in this study are:

- **Kaggle**. Kaggle is a repository that serves the ML and data science communities, and is particularly popular with students and others interested in learning data science. The platform hosts datasets and notebooks (kernels) that perform ML or other data science tasks using the available data. Kaggle limits datasets to 100 GB.
- **OpenML**. This repository classifies ML components into five semi-hierarchical categories: datasets, tasks, flows, runs, and studies. We focus primarily on datasets, as they are most comparable to other repository content in terms of structure, use, and available metadata, with discussion of flows (ML pipelines or scripts that include library dependencies and hyperparameters) and runs (performance evaluations of a specific flow on a specific task) where relevant. OpenML has no defined limit for dataset size.
- **UC Irvine Machine Learning Repository**. Hosted by the University of California, Irvine, this repository hosts training and test datasets, with reuse as the explicit goal of the repository objects. No defined limit for dataset size. We refer to this repository by the acronym UCIMLR, henceforth.

We define "generalist repositories" as discipline-agnostic repositories with flexibility in structure and documentation. The generalist repositories included in this study are: Figshare, Zenodo, Harvard Dataverse, Dryad, and the UC San Diego Library Digital Collections, the former of which are frequently recommended by publishers (e.g., Springer Nature, PLOS (also includes Kaggle), AGU) and funders (e.g., NIH). For a comprehensive comparison of the basic characteristics of these and other generalist repositories, see Stall et al. 2020 and Supplemental Table A. The three specialist repositories, as well as Figshare and Zenodo, are non-curated, meaning that curation services are not offered at the object level after submission, while Harvard Dataverse, Dryad, and UC San Diego Library offer variable levels of curation.

Metadata retrieval

Object metadata were retrieved via official repository APIs when available. For repositories in which a public-facing API was not available, or the API returned incomplete metadata, web scraping was used to extract metadata present on an ML object webpage when doing so did not violate site terms of service (Table 1, Appendix). Records in six of the repositories were queried across indexed metadata fields for the string "machine learning"[1] with the intent to capture all objects that were apparently depositor-identified as

[1] Different repositories may index different subsets of metadata fields for string searches. We do not attempt advanced search across select fields, but rather use the default primary search for each repository.

ML-related. For OpenML and UCIMLR, in which all objects were considered explicitly ML-related, all records were retrieved (Table 1).

Metadata was retrieved from the listed repositories during the last two weeks of December 2021.[2] Code used to access all APIs was developed using Python, with tested compatibility back to Python 3.9.2. This implementation relies heavily on packages such as 'requests' for the public APIs, and 'selenium' and 'beautifulsoup4' for web scraping when necessary. The code developed to access APIs (not including web scraping) is available as a user interface called PyCurator (Baluja 2022).

Table 1: Properties of the Metadata Retrieval. Records matching "machine learning" were retrieved from the eight listed repositories via a combination of API, web scraping, and curator tools. For the specialist repositories OpenML and UCIMLR, all records were retrieved. Further details on search and return parameters are detailed in repository-specific API documentation.

Repository	Access Method	Query Term	Query Type	Granularity of Returned Records
Dryad	API, Scrape	"machine learning"	N/A	File
UC San Diego Library	Export tool (curators only)	"machine learning"	N/A	File
Harvard Dataverse	API, Scrape	"machine learning"	Dataset File	Object
Figshare	API	"machine learning"	Article (Item) Collection Project	Object
Zenodo	API	"machine learning"	N/A	Object
UCIMLR*	Scrape	N/A	N/A	Object
OpenML	API, Scrape	N/A	Dataset	Object (=File)
Kaggle	API	"machine learning"	Dataset Kernel	Object
* UCIMLR was in the process of updating their website during this project. We scraped the beta site: https://archive-beta.ics.uci.edu.				

[2] Some repositories may have made changes to metadata structure and collection since this time; results are reflective of the state of repository metadata at this moment in time.

Metadata Crosswalk

In order to compare ML object characteristics across repositories, we created a metadata crosswalk—a mapping of the equivalent properties of the metadata schemas across the eight repositories (see Supplemental Table B). Where available, we relied on publicly available documentation to determine repository metadata schema. It was necessary in some cases to compare examples of extracted data against content exposed on corresponding landing pages to understand the field type. Following completion of the crosswalk, we calculated summary statistics for metadata fields of interest in order to directly compare repository ML object characteristics such as file type, license type, dataset size, etc. See project GitHub repository (Labou and Baluja 2024) or Labou et al. 2024 for code used in calculations.

Based on findings from the crosswalk, we created a matrix to visualize FAIR Principle "compliance" by each repository, focusing on the FAIR Principle elements that are central to basic findability, interoperability, and reuse. We distinguished between full compliance, partial, and none for a subset of metadata fields as measured against the recommendations associated with each relevant element of FAIR. As an example, Figshare uses the New Zealand Standard Research Classification (ANZSRC) for Fields of Research (FoR) for research domain category ("Categories" in the object submission form and "Domain" in our matrix) and is therefore in full compliance with FAIR Principle I2. A repository was considered to be in partial compliance if they collected a particular type of metadata but did not make it easily accessible via a designated field or fully contextualize it. For instance, Harvard Dataverse collects related works but does not include a qualified reference that establishes the relationship to the item being deposited, as recommended by FAIR Principle I3. Similarly, UC San Diego Library supports a free-text field for recording funding information, but individual elements like funder and grant number are not stored in designated fields, potentially reducing findability and interoperability.

Results

FAIRness of the Repositories

The FAIR Matrix (Table 2) showed thematic variability between repositories. In general, generalist repositories collect traditional descriptive metadata, in alignment with FAIR sub-principle "F2: Data are described with rich metadata," including a PID, a key agent for findability. Of the specialist repositories, only UCIMLR generates PIDs. Generalist repositories are in compliance with the sub-principle, "R1.2: (Meta) data are associated with detailed provenance," as far as enabling citability and, in some cases, data creation or generation methods. Specialist repositories gather provenance metadata about dataset characteristics, workflows, and experiments, which are critical to evaluating ML data processing and transformation history.

Table 2: ML FAIR Matrix. Properties are grouped by metadata type. Full compliance with at least one FAIR Principle is indicated by a solid circle. Partial compliance is indicated by a half-filled circle. The primary FAIR Principle associated with each property, as determined by authors, is indicated.

	Generalist Repositories					Specialist Repositories			
Property	**Dryad**	**UC San Diego Library**	**Harvard Dataverse**	**Figshare**	**Zenodo**	**UCI MLR**	**Open ML**	**Kaggle**	**FAIR Guiding Principle**
Curation	Y	Y	Offered	N	N	N	N	N	
	Descriptive Metadata								
Title	●	●	●	●	●	●	●	●	F2
Description	●	●	●	●	●	●	●	●	F2
Language		●	●		●		●		F2
Note		●	●	●	●	●			F2
Domain	●		●	●		●			I2
Keyword	◐	◐	●	◐	●	◐	◐	◐	I2
Geographic keyword	◐	◐	◐	◐	◐				I2
Scientific keyword		◐		◐	◐				I2
Number of instances (rows)						●	●		F2/R1
Number of attributes (columns)						●	●	●	F2/R1
Missing attribute values						●	●	●	R1
Class distribution						●	●	●	R1
Recommended data split						●	●	●	R1
Target feature							●		R1
Sample size							●		R1

Table 2 Continued: ML FAIR Matrix. Properties are grouped by metadata type. Full compliance with at least one FAIR Principle is indicated by a solid circle. Partial compliance is indicated by a half-filled circle. The primary FAIR Principle associated with each property, as determined by authors, is indicated.

	Generalist Repositories					Specialist Repositories			
Property	**Dryad**	**UC San Diego Library**	**Harvard Dataverse**	**Figshare**	**Zenodo**	**UCI MLR**	**Open ML**	**Kaggle**	**FAIR Guiding Principle**
	Provenance Metadata								
Locally unique object identifier	●	●	●	●	●	●	●	●	F2
Persistent identifier	●	●	●	●	●	●			F1
Citation	●	●	●	●	●	●	●		R1.2
Original data owner		●	●					●	R1.2
Original data URL		●		●			●		R1.2
Creator/ contributor	●	●	●	●	●	●	●	●	R1.2
Creator/ contributor ORCID	●	◐	●	●	●				R1.2
Creation/ collection date		●	●	●					R1.2
Publication date	●	●	●	●	●				R1.2
Primary associated article citation		●	●	●		●			I3
Primary manuscript DOI or URL	●	●	●	●	●	●	●		I3
Related resource citation		●	●		●	●			I3
Related resource identifier	●	●	●	●	●	●	●		I3
Related resource relation type	●	●	●		●				I3
Related resource type (dataset, publication, etc.)	●	●			●				I3
Update	●	●	●	●	●		●	●	R1.2
Version	●	◐	●	●	●		●	●	R1.2
Methods	●	●							R1.2
Technical details (software, instrumentation, etc.)	●	●				●	●		R1.2
Tasks						●	●		R1
Preprocessing steps						●			R1
Estimation procedure							●		R1
Cost matrix							●		R1
Evaluation measures							●	●	R1

Table 2 Continued: ML FAIR Matrix. Properties are grouped by metadata type. Full compliance with at least one FAIR Principle is indicated by a solid circle. Partial compliance is indicated by a half-filled circle. The primary FAIR Principle associated with each property, as determined by authors, is indicated.

	Generalist Repositories					Specialist Repositories			
Property	**Dryad**	**UC San Diego Library**	**Harvard Dataverse**	**Figshare**	**Zenodo**	**UCI MLR**	**Open ML**	**Kaggle**	**FAIR Guiding Principle**
	Administrative (Rights and Preservation) Metadata								
License	●	●	●	●	●	●	●	●	R1.1
File format		●			●		●		R1
Dataset size	●	●	●	●	●			●	R1
Media type	●	●			●				R1
Checksum	●	●	●		●		●		R1.2
	Funding								
Grant title		◐		●	●				R1.2
Grant number	●	◐	●	●	●				R1.2
Funding Program		◐			●	●			R1.2
Funding Agency	●		●	●	●				R1.2
	Metrics								
Usability rating								●	R1
Views	●				●	●		●	F2
Downloads	●		●		●		●	●	F2
Citations						●			F2

Characteristics of ML objects in repositories

ML objects in repositories

Full metadata extracts, analysis code, and all supplemental tables are available at Labou et al. 2024. We retrieved a total of 38,707 objects matching our search terms from the eight repositories searched. Over 32,000 objects, representing 83% of all returned results, were from Zenodo and Figshare, both repositories with a self-deposit process, no post-deposit curation, and accepting all content types. The majority of returned objects in both repositories were more in line with traditional scholarly outputs that report results, such as journal articles, conference papers, reports, and presentations. These research outputs are useful for information sharing about ML processes and developments, but they are not reusable ML output in the practical sense we mean here. Therefore, we limited further analyses to the subset of Figshare objects tagged as "dataset," "software," or "model," and to the subset of Zenodo objects classified as "Dataset" or "Software."[3]

After filtering the Figshare and Zenodo records as described, a total of 19,127 "machine learning" objects remained and were used for subsequent analysis. Summary statistics for the metadata extracts are reported in Table 3. The ascending rank order of repositories, from fewest to greatest number of retrieved objects, was: UC San Diego Library, Dryad, Harvard Dataverse, UCIMLR, Kaggle, Zenodo, OpenML, Figshare. A total of 6,050 objects were retrieved from the three ML-specific repositories.

Trends over time

ML objects have been appearing in these repositories (as determined by date associated with object) with greater frequency of deposits in the last 10 years (Figure 1, Supplemental Table C).[4] Most repositories have seen consistently increasing numbers of deposits, especially pronounced in generalist repositories.

Common domains

Four of the repositories reported domains. The domains that were reported for 20% or more of the classified objects in any one of the reporting repositories were generally in the fields of biological sciences, computer/information sciences, medicine, chemistry, and social sciences (Table 3).

Dataset size

The median total dataset size per object for the four generalist and one specialist (Kaggle) repositories for which size was included in metadata ranged from 0.07 MB to 797 MB (Table 3). The mean dataset size ranged from 455 MB to 10,658 MB. For Kaggle, the median and mean were 1.2 MB and 639 MB, respectively.

[3] For Figshare, 42% of the retrieved "machine learning" objects were designated by depositors as "datasets" (<0.01% as "software" or "model"). For Zenodo, <20% of "machine learning" objects retrieved were designated as "dataset" and 11% were tagged as "software." Objects in Zenodo labeled as audiovisual or image resource types were spot-checked for classifiability as ML output. Since the majority of objects not tagged as data or software in the repositories were not training data, these other resource types were excluded as a whole.

[4] UCIMLR was first established in 1987 (UCIMLR, n.d.), with annual deposits in the 10s until about 2007, when a web site superseded the original ftp archive.

Table 3: Object-level summary statistics for metadata retrievals from each repository. When more than five categories for a characteristic were present, only the top five are displayed.

Characteristic	Dryad	UC San Diego Library	Harvard Dataverse	Figshare	Zenodo	UCIMLR	OpenML	Kaggle
Number of objects returned	219	25	269	10,347	2,217	583	3,460	2,007
Licenses (% of objects)	CC0 (100%)	CC BY 4.0 (92%) CC BY NC SA 4.0 (4%) MIT (4%)	CC0 (84%) No waiver (7%)	CC BY 4.0 (77%) CC BY + CC0 (8.9%) CC BY-NC 4.0 (8.5%) CC0 (2.4%) MIT (1.4%)	CC-BY 4.0 (48%) 'other open' (17%) CC0 (11%) MIT (5%) ODbL (3%)	CC BY 4.0 (99%)	Public/Public Domain/CC0 (92%) Free (1.5%) Publicly available (1.2%)	Unknown (32%) CC0 (21%) Other (11%) copyright-authors (8%) CC BY 4.0 (5%)
Number of files per object	Mean: 13 Median: 2 Max: 617	Mean: 6 Median: 5 Max: 33	Mean: 50 Median: 4 Max: 2496	Mean: 3 Median: 1 Max: 1,100	Mean: 9 Median: 1 Max: 2,600	Mean: 3 Median: 2 Max: 35	Mean: 1 * Median: 1 Max: 1	NA
Object size (MB)	Mean: 6,277 Median: 21 Max: 240,411	Mean: 11,060 Median: 991 Max: 130,855	Mean: 7,443 Median: 119 Max: 661,976	Mean: 455 Median: 0.07 Max: 675,675	Mean: 5,219 Median: 32.7 Max: 300,398	NA	NA	Mean: 639 Median: 1.2 Max: 73,678
Domain (% of objects)	Biological sciences (7.3%) Computer and information sciences (3.7%) Earth and related environmental sciences (1.4%) Clinical medicine (1.4%) Medical and health sciences (1.4%)	NA	Social Sciences (41%) Computer and Information Science (36%) Medicine, Health and Life Sciences (16%) Earth and Environmental Sciences (11%) Physics (6%) Information Science (36%) Medicine, Health and Life Sciences (16%) Earth and Environmental Sciences (11%) Physics (6%)	Biological Sciences not elsewhere classified (41%) Information Systems not elsewhere classified (29%) Genetics (21%) Biotechnology (18%) Cancer (18%)	NA	Computer (36%) Life (22%) Other (13%) Physical (10%) Business (7%)	NA	NA
Number (%) of objects with >0 related resources	162 (74%)	24 (96%)	67 (25%)	8,906 (86%)	2,077 (94%)	102 (17%)	1,661 (48%)	NA
Of objects with >0 related resources, number of related resources per object	Mean: 1.2 Median: 1	Mean: 4.8 Median: 4	Mean: 1.3 Median: 1	Mean: 1.0 Median: 1	Mean: 1.7 Median: 1	Mean: 3.6 Median: 5	Mean: 1.1 Median: 1	NA

* OpenML allows for exactly one file per object

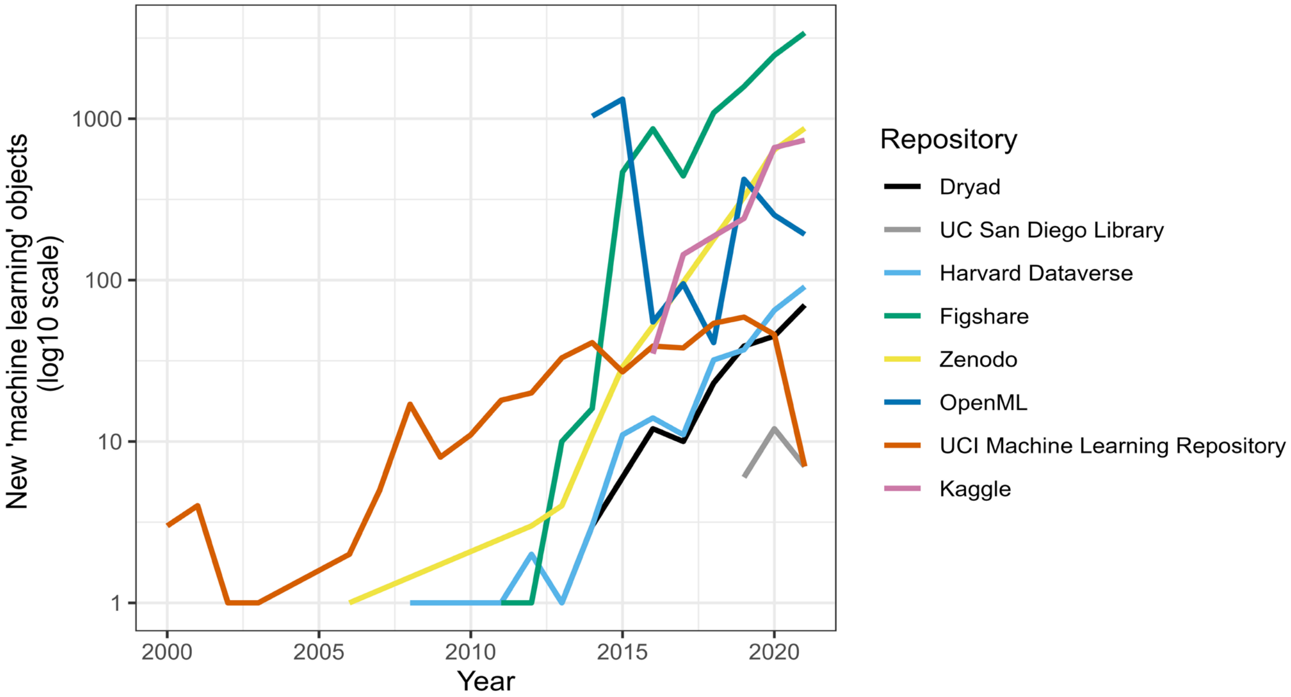

Figure 1: Upward trend over time of new "machine learning" objects published in generalist and specialist repositories, 2000-2021.

File count

Excluding OpenML, which permits exactly one file for every object, and Kaggle, whose metadata did not include file count, the median number of files per object ranged from 1 to 5, while the mean ranged from 3 to 50 (Table 3).

File type

Metadata retrievals from seven of the eight repositories yielded information about file extension (Figure 2, Appendix), with OpenML enforcing ARFF format for all objects. Basic tabular data (CSV, Excel, etc.) was the most common format, with 34% of objects in these repositories including files in this format category. Compressed formats and text formats were each present in files associated with ~20% of objects. Compression of files before upload is a common practice likely due to both usability and transfer efficiency considerations, as well as being a way to maintain original file organization; thus, the prevalence of other file formats such as tabular, textual, code, and image are likely strongly underrepresented.

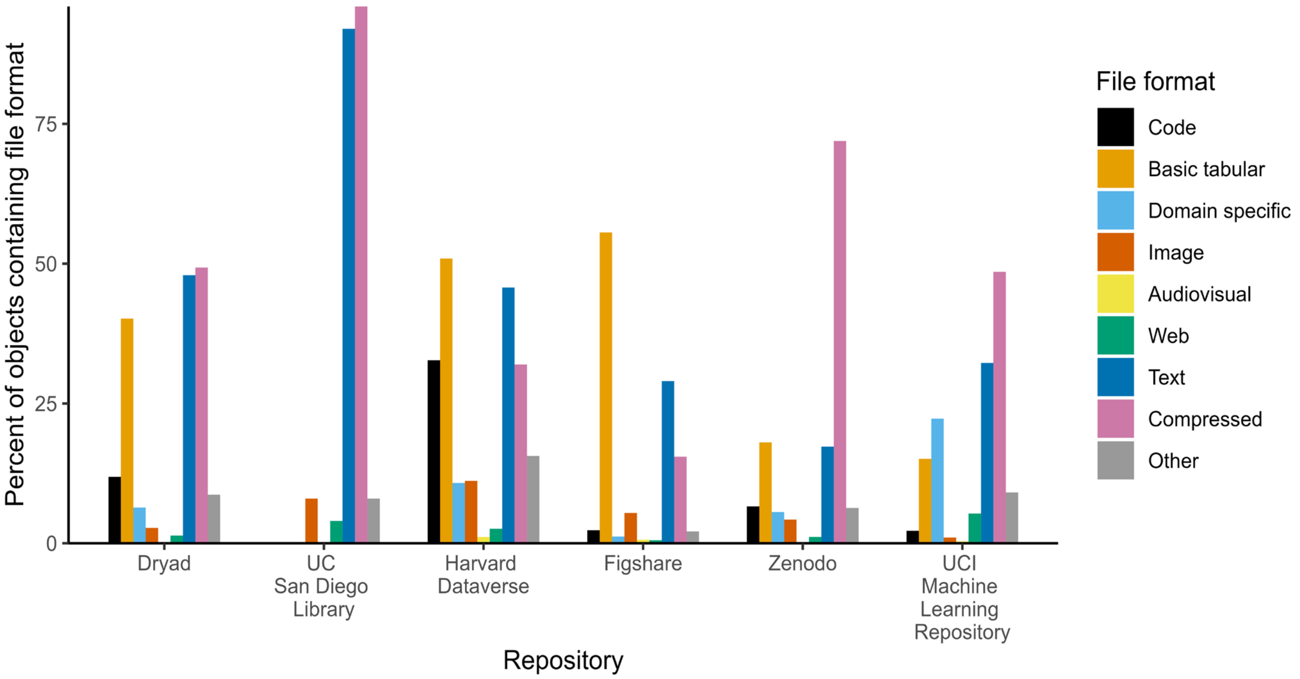

Figure 2: Percentage of objects containing file format category in repositories. Note that this excludes OpenML, which enforces ARFF format for all objects, and Kaggle, for which file format could not be determined based on extracted metadata. A full list of file extensions mapped to format category is available in Supplemental Table D. Counts and proportions used to create Figure 2 are in Supplemental Table E. Because objects can contain one or more associated files, the total for each repository may be greater than 100%.

Identifiers

All of the generalist repositories assign PIDs, specifically DOIs, to datasets. Of the specialist repositories, Kaggle and OpenML do not assign PIDs, and while UCIMLR has a field for DOI, it was largely unpopulated at the time of metadata extract. As evidence of how the absence of a PID plays out in practice, we identified several objects in Kaggle named variously "Thoracic Surgery," "Thoracic-Surgery," or "Thoracic Surgery Dataset" that replicate the UCIMLR Thoracic Surgery Data dataset (Lubicz et al. 2013). We reviewed five of these datasets—none of which have associated PIDs—and found that they have little or no additional documentation that might disambiguate them from one another or from the original dataset.

Citations

Formatted citations are displayed on the landing pages of all objects for all of the generalist repositories, although the citations themselves were not available via API for Dryad and Zenodo. Of the specialist repositories, only Kaggle lacks a citation field entirely. For OpenML and UCIMLR, only 16% and 1% of objects, respectively, have a populated citation, but for UCIMLR this looks to be an artifact of scraping the developing beta site. As of early 2023, the site reflects far more objects with citations; as such, we do not draw definitive conclusions for this repository in terms of citations.

We also examined the metadata, specifically citations, associated with several datasets that have been re-published in multiple repositories. For example, the “Iris” dataset in Kaggle (Fisher 1988) states in the “About Dataset” section that the dataset is also available in UCIMLR but only if the end-user scrolls down to the “Metadata” section of the page and clicks on “Collaborators” will they see the attribution “UCI Machine Learning (Owner).” In this case, as in many others, the elements needed to form a proper citation (author, publication date, original title, and publisher) are either not conveniently located so as to promote citability, or are entirely absent, as in the case of the original data publication date. The same dataset, entitled “iris” in OpenML, is attributed in the free text of the “Description” field: “Source: UCI - 1936 - Donated by Michael Marshall.” According to the UCIMLR Iris object landing page, Michael Marshall is indeed the dataset donor, however, the end-user will need to read the narrative associated with the “Iris Plants Database” to find a reference to the original research paper from which this data is sourced.

Data reuse permissions

Licenses associated with ML objects varied between repositories (Table 3, Supplemental Table F). All the repositories support data usage licenses, although Creative Commons licenses or waivers are specified for the vast majority of objects in the Dryad, Harvard Dataverse, UCIMLR, and UC San Diego Library repositories, while objects in the other two specialist repositories have a wider range of licenses. Part of this difference is due to the tendency of many generalist repositories to offer only a limited number of permissible license types from which depositors can select. Zenodo and Figshare also offer a range of licenses, although depositors waived rights (e.g., CC0) or assigned attribution-based licenses (e.g., CC BY, MIT, BSD) in 83% and 86% of objects, respectively. Conversely, specialist repositories (except for UCIMLR) more often allow a wider range of licenses, including “other.” In OpenML, the free text permitted in the license field made categories harder to define; nevertheless, 93% of objects were released to the public domain, 1% were assigned attribution-based licenses, and the remainder were not clearly classifiable licenses. For Kaggle, about half of objects did not have clear licenses; of those that were classifiable, 30% were CC0 or attribution-based, and 12% were more restrictive.

Related resources

Seven of the repositories contained enough Related Resource information to report the percentage of objects with links. The percentage of objects with links varied from 17% to 96% (Table 3). The median number of links per object with at least one link varied from 1 to 5.

Discussion

We identify the following as common themes across repositories and areas where further work is needed to enable findability, interoperability, and reuse of ML objects: persistent identifiers, unambiguous citations, rich technical provenance metadata, appropriate use of related resources, explicit licenses, and clear file labels.

Value of persistent identifiers

All repositories assessed here have what we consider minimally required metadata fields: locally unique object identifier, title, description, keyword, creator, and license. Although most of the repositories support a field for PID, neither Kaggle nor OpenML do so. This is perhaps not entirely inappropriate, considering the common purpose of these repositories: exploratory research, skill building, and collaboration, versus the more formal publication nature of the other repositories. However, in the context of the reuse/republishing of datasets (especially UCIMLR-derived ones) across repositories, the provenance would be more likely to endure if all of their datasets were assigned a PID. More strongly, making the source data citation field mandatory and integrating citation copying with dataset download, to encourage users to reference the citation when reusing the data, will ensure that a direct line can be drawn between the original data and its later instantiations.

Similar to datasets, we note that ML experiments as discrete entities could also include PIDs. For instance, in OpenML, the repository with the most comprehensive ML architecture, projects - when documented completely - include links to data and algorithms used and values for parameters as well as model evaluation measures. Should finer-scale identification be desired, popular algorithm packages themselves could be assigned PIDs. Algorithms are the engine of ML, driving the classification, clustering, regression, anomaly detection, or other task type, and are prime candidates for discovery and reuse. They may come from software packages that have an associated published paper with a PID, as is the case with some (but not all) commonly used Python libraries, or they may be pulled from websites like SourceForge, an open source software platform that publishes the Weka collection of algorithms developed for data mining tasks. The practice of algorithm- or package-specific PIDs would increase transparency of ML methodology and implementation, acknowledging that this would be dependent on software platforms issuing PIDs.

Importance of unambiguous citations

Similar to PIDs, citations are a valuable element of reuse because they facilitate proper and accurate attribution of data. It is perhaps as a consequence of a) a data depositor habit of republishing datasets, especially those first published in UCIMLR and b) across the repositories there is only limited support for documenting original data owner and/or linking to the original data source, that accurate attributions may be problematic. In addition, when left to their own discretion, data submitters may not provide enough information to form an accurate citation. It is not clear at this point how common it is for people to publish datasets in more than one repository, except to note that it is not rare. While one reason may be that users wish to avail themselves of the repository-specific tools, it ultimately results in murky data provenance. A simple solution to these citability issues is for repositories to collect via mandatory fields the various elements needed to construct a proper citation.

Describe ML components with rich metadata

Beyond the minimally required metadata discussed above, all repositories would benefit from expanding traditional concepts of descriptive, provenance, and technical metadata to document ML experiments to better enable reuse. Accordingly, expanded descriptive metadata would include fields such as resource type (e.g., date, image, text), number of instances, number of attributes, class, class distribution, recommended data split, target feature, sample size, task type (e.g., classification or clustering), and dataset type (e.g., training or testing).

To tell the full story of a ML experiment, include fields for reporting what purpose the data were created or collected and known limitations of the data (such as OpenML and UCIMLR's missing values properties). Equally important are provenance metadata about the ML workflows. When thoughtfully written, a *methods* field informs the ability of a user to determine if the data is actually useful in a particular context (Wilkinson et al. 2016). Examples of methods-related metadata include: data generation and collection processes, any data processing (e.g., cleaning or wrangling), ML model training parameters and hyperparameters, and any other process-related documentation.

Technical metadata for ML includes any software or instrumentation used to create, collect, or process data, including software version and instrumentation make and model. It is not uncommon to see methods, technical details, and a data creation or collection date buried in description fields, configuration files (e.g., config or yaml environment files), or code notebooks when designated fields for this type of information are not provided during data deposit. Ultimately, having both *methods* and *technical details* fields provides the fullest context, especially in cases where reuse is anticipated.

In instances where adding database fields is not feasible, "readme" files, a staple of data curation that offers a simple means of keeping data documentation and data together, can fill this gap. UCIMLR exemplifies this practice in the context of ML by creating a downloadable "readme"-style .names file containing a structured summary of dataset metadata. This practice can easily be extended to ML datasets by creating a README that includes traditional metadata as well as the expanded descriptive, provenance, and technical metadata outlined above. ML practitioners may already be doing this, but as curators we can be prescriptive about what specifically should go into a README by creating a template for researchers that includes these fields.

Exposing related resources for findability and reusability

A set of fields consisting of related work citation, PID or URL, and relation type can help clarify the provenance of datasets like "Iris" and "Thoracic Surgery." Such fields can contextualize a dataset in relation to, for instance, source data, other versions of the dataset, reference materials, or primary associated articles (DataCite, Wood-Charlson et al. 2022). While any linking of related works is better than none, the infrastructure for establishing the interconnections between datasets and other works is available through the application of the community standard DataCite Metadata Schema, *relationType* property. Stored in a designated field, this property allows repositories to characterize the relationship between the ML object being deposited and other resources, internal or external to the repository. The DataCite *relationType* field is currently used by UC San Diego Library, Dryad, and Zenodo, which further push metadata to DataCite to maximize data visibility and reuse.

Need for clear licensing

Non-specific license statements can cause potential problems, because when exceptions to a controlled value list are allowed, licenses may be unclear in practice. For instance, among the license types offered when creating a new dataset record in Kaggle is "Other (specified in description)," but a review of the *description* field of the set of records that use this license type shows that only a small subset of records actually note a specific license. While allowing flexibility in license type is potentially useful, providing a controlled list of licenses in a mandatory field forces data providers to define "clear and accessible" guidance about data usage to future users of their work (Wilkinson et al. 2016). This is especially crucial since without a clear license allowing reuse of some kind, all other metadata provided is moot in the context of reusability.

Compressed files and the need for labels

Although compressing directories can obscure their contents and increase risk to long-term preservation, this practice is done to accommodate the need to preserve large file directory structures and deliver data to the end user efficiently. This is a common tradeoff for many generalist repositories. Given the complexity of ML outputs, the best way forward for generalist repositories may be to promote the practice of labeling ML components explicitly, with controlled vocabulary, and indicating the locations of those components, within specific folders or file names. Better yet, the file names and labels should be incorporated into pipelines that instruct users on the proper order of operations for reproducing a ML workflow (e.g., the "silver" standard described by Heil et al. 2021). If labels and workflows are provided with sufficient detail, then the particular bundling method of shared ML research will be less important, or will at least be a smaller barrier to reuse.

Access at scale

We consider the presence of a publicly accessible API to access (meta)data at scale an important aspect of reusability in the context of ML. As noted in Table 1, all repositories included in this analysis except UCIMLR and UC San Diego Library had a usable public-facing API. Even for repositories with an API, web scraping was sometimes necessary to return certain metadata fields of interest (see Appendix for more details). As researchers become more interested in "big data" and accessing bulk data, a public-facing API is becoming, if not an expectation, at least a benefit and enticement for researchers to use certain repositories over others.

Conclusion

Overall, we find there is a documentation gap between the two categories of repositories: generalist repositories focus on collecting traditional provenance metadata while specialist repositories emphasize metadata about dataset characteristics, workflows, and experiments. The gaps in metadata collection can be addressed in many cases by each adopting some of the impactful metadata practices for dataset discoverability of the other.

In particular, all repositories can ensure that their schema supports related resources, as these are crucial for explicitly linking ML objects to their source materials, supplemental documentation, and related publications across various platforms. Repositories should also facilitate submission of rich metadata and PIDs associated with these resources, when available. Specialist repositories can invest in assigning PIDs to objects and ensuring the consistent collection of rich provenance metadata through use of designated, mandatory fields. Generalist repositories can add support for ML-specific fields for characterizing ML datasets and processing steps, such as number of instances and number of attributes. Alternatively, repositories can leverage standardized README files with support for ML fields, including implementation environment details, as a means of enhancing reusability without major changes to repository infrastructure. While there are trade-offs in investing time and effort for repositories to collect additional metadata, capturing this information in some form is imperative for long-term findability and reusability of ML research outputs.

Data Availability

Data, analysis code, and all supplemental tables are available in UC San Diego Library Digital Collections: https://doi.org/10.6075/J0JS9QMH.

Metadata retrieval and characteristics of ML objects are available under the article Supplementary Files:

Appendix: Metadata retrieval & Characteristics of ML objects

Acknowledgements

This work was supported by a research grant from the Librarians Association of the University of California (LAUC) and by the Research Data Curation Program at the UC San Diego Library. Kat Koziar and David Minor provided useful feedback on drafts of this paper.

Competing Interests

The authors declare that they have no competing interests.

References

AGU Open Science Leadership. 2021. "Domain-Discipline Repositories Useful to AGU Journals." Accessed September 27, 2023. https://data.agu.org/resources/useful-domain-repositories.

Australian Bureau of Statistics. 2020. "Australian and New Zealand Standard Research Classification (ANZSRC)." Accessed January 20, 2023. https://www.abs.gov.au/statistics/classifications/australian-and-new-zealand-standard-research-classification-anzsrc/latest-release.

Baluja, Michael. 2022. PyCurator (Version 0.1) GitHub repository. https://github.com/michaelbaluja/PyCurator.

Consultative Committee for Space Data Systems (CCSDS). 2012. "Reference Model for an Open Archival Information System (OAIS)." Accessed September 1, 2022. https://public.ccsds.org/pubs/650x0m2.pdf.

DataCite Metadata Working Group. 2021. "DataCite Metadata Schema Documentation for the Publication and Citation of Research Data and Other Research Outputs. Version 4.4." *DataCite*. https://doi.org/10.14454/3w3z-sa82.

DataCite. n.d. "Connecting to Works." Accessed April 15, 2022. https://support.datacite.org/docs/relationtype_for_citation.

Esteves, Diego, Diego Moussallem, Ciro Baron Neto, Tommaso Soru, Ricardo Usbeck, Markus Ackermann, Jens Lehmann. 2015. "MEX vocabulary: a lightweight interchange format for machine learning experiments." *SEMANTICS '15: Proceedings of the 11th International Conference on Semantic Systems*: 169-176. https://doi.org/10.1145/2814864.2814883.

Fisher, Ronald Aylmer. 1988. Iris. UCI Machine Learning Repository. https://doi.org/10.24432/C56C76.

Gebru, Timnit, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. 2021. "Datasheets for datasets." *Association for Computing Machinery* 64 (12): 86-92. https://doi.org/10.1145/3458723.

Global Future Council on Human Rights 2016-2018. 2018. "How to Prevent Discriminatory Outcomes in Machine Learning." *World Economic Forum*. https://www3.weforum.org/docs/WEF_40065_White_Paper_How_to_Prevent_Discriminatory_Outcomes_in_Machine_Learning.pdf.

Heil, Benjamin J., Michael M. Hoffman, Florian Markowetz, Su-In Lee, Casey S. Greene, and Stephanie C. Hicks. 2021. "Reproducibility standards for machine learning in the life sciences." *Nature Methods* 18: 1132-1135. https://doi.org/10.1038/s41592-021-01256-7.

Juty, Nick, Sarala M. Wimalaratne, Stian Soiland-Reyes, John Kunze, Carole A. Goble, and Tim Clark. 2020. "Unique, Persistent, Resolvable: Identifiers as the Foundation of FAIR." *Data Intelligence* 2 (1-2): 30-39. https://doi.org/10.1162/dint_a_00025.

Katz, Daniel S., Morane Gruenpeter, and Tom Honeyman. 2021a. "Taking a fresh look at FAIR for research software." *Patterns* 2 (3): 100222. https://doi.org/10.1016/j.patter.2021.100222.

Katz, Daniel S., Fotis Psomopoulos, and Leyla Jael Castro. 2021b. "Working Towards Understanding the Role of FAIR for Machine Learning." *DaMaLOS@ISWC Publisso*. https://doi.org/10.4126/FRL01-006429415.

Labou, Stephanie and Michael Baluja. 2024. Comparative Machine Learning Metadata. GitHub repository. https://github.com/stephlabou/comparative-machine-learning-metadata.

Labou, Stephanie, Abigail Pennington, Ho Jung S. Yoo, and Michael Baluja. 2024. "Data from: Toward Enhanced Reusability: A Comparative Analysis of Metadata for Machine Learning Objects and Their Characteristics in Generalist and Specialist Repositories." *UC San Diego Library Digital Collections*. https://doi.org/10.6075/J0JS9QMH.

Lamprecht, Anna-Lena, Leyla Garcia, Mateusz Kuzak, Carlos Martinez, Ricardo Arcila, Eva Martin Del Pico, Victoria Dominguez Del Angel, Stephanie van de Sandt, et al. 2020. "Towards FAIR Principles for Research Software." *Data Science* 3 (1): 37-59. https://doi.org/10.3233/DS-190026.

Louridas, Panos and Christof Ebert. 2016. "Machine Learning." *IEEE Software* 33 (5): 110-115. https://doi.org/10.1109/MS.2016.114.

Lubicz, Marek, Konrad Pawelczyk, Adam Rzechonek, and Jerzy Kolodziej. 2013. Thoracic Surgery Data. UCI Machine Learning Repository. https://doi.org/10.24432/C5Z60N.

National Institutes of Health (NIH). n.d. "Generalist Repositories." Accessed September 27, 2023. https://sharing.nih.gov/data-management-and-sharing-policy/sharing-scientific-data/generalist-repositories.

Norgeot, Beau, Giorgio Quer, Brett K. Beauliue-Jones, Ali Torkamani, Raquel Dias, Milena Gianfrancesco, Rima Arnaout, et al.2020. "Minimum information about clinical artificial intelligence modeling: the MI-CLAIM checklist." *Nature Medicine* 26: 1320-1324. https://doi.org/10.1038/s41591-020-1041-y.

Pineau, Joelle, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché-Buc, Emily Fox, and Hugo Larochelle. 2021. "Improving Reproducibility in Machine Learning Research (a Report from the NeurIPS 2019 Reproducibility Program)." *Journal of Machine Learning Research* 22 (1): 7459-7478. https://dl.acm.org/doi/abs/10.5555/3546258.3546422.

PLOS ONE. n.d. "Recommended Repositories." Accessed September 27, 2023. https://journals.plos.org/plosone/s/recommended-repositories.

Publio, Gustavo Correa, Diego Esteves, Agnieszka Ławrynowicz, Panče Panov, Larisa Soldatova, Tommaso Soru, Joaquin Vanschoren, and Hamid Zafar. 2018. "ML-Schema: Exposing the Semantics of Machine Learning with Schemas and Ontologies." *arXiv*. https://doi.org/10.48550/arXiv.1807.05351.

Samuel, Sheeba, and Birgitta König-Ries. 2018. "Provbook: Provenance of the Notebook." *figshare*. https://doi.org/10.6084/m9.figshare.6401096.v2.

Samuel, Sheeba, Frank Löffler, and Birgitta König-Ries. 2021. "Machine Learning Pipelines: Provenance, Reproducibility and FAIR Data Principles." In: *Provenance and Annotation of Data and Processes* IPAW IPAW 2020 2021, edited by Glavic, Boris, Braganholo, Vanessa, and Koop, David. *Lecture Notes in Computer Science* 12839. Springer, Cham. https://doi.org/10.1007/978-3-030-80960-7_17.

Sengupta, Partho P., Sirish Shrestha, Béatrice Berthon, Emmanuel Messas, Erwan Donal, Geoffrey H. Tison, James K. Min, et al. 2020. "Proposed Requirements for Cardiovascular Imaging-Related Machine Learning Evaluation (PRIME): A Checklist." *JACC: Cardiovascular Imaging* 13 (9): 2017-2035. https://doi.org/10.1016/j.jcmg.2020.07.015.

Springer Nature. n.d. "Generalist repository examples." Accessed September 27, 2023. https://www.springernature.com/gp/authors/research-data-policy/generalist-repositories/12327166.

Stall, Shelley, Maryann E. Martone, Ishwar Chandramouliswaran, Mercè Crosas, Lisa Federer, Julian Gautier, Mark Hahnel, et al. 2020. "Generalist Repository Comparison Chart." *Zenodo*. https://doi.org/10.5281/zenodo.3946720.

Wilkinson, Mark D., Michael Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, et al. 2016. "The FAIR Guiding Principles for scientific data management and stewardship." *Scientific Data* 3: 160018. https://doi.org/10.1038/sdata.2016.18.

Wood-Charlson, Elisha M., Zachary Crockett, Chris Erdmann, Adam P. Arkin, and Carly B. Robinson. 2022. "Ten simple rules for getting and giving credit for data." *PLoS Computational Biology* 18 (9): e1010476. https://doi.org/10.1371/journal.pcbi.1010476.