*Curation and Reproducibility in an Artificial Intelligence World: Challenges and Solutions for Scientific Research*

Author: Lars Vilhuber, Cornell University

Much has been written about artificial intelligence, with astonishingly rapid progress in computer sciences. In the social sciences, concerns have been raised that artificial intelligence may impact the actual production of scientific output. Most of the discussion has been about the writing of texts, and estimates suggests that the number of articles created and possibly submitted with the help of AI systems is non-trivial. Less interest has been devoted to the use of AI as part of the legitimate scientific production process. Yet use of AI methods in legitimate scientific work is also increasing. With the earlier “replication crisis” still in mind, the question for curators is whether and how to curate AI-supported research tools, input data, and outputs. This article will approach the topic from the perspective of a “data editor”, responsible for verifying reproducibility and supporting curation of research compendia for a prominent learned society in economics.

As a preview of the results, the key to understanding these challenges is that they are not new. Yet challenges remain. In the economics literature, the use of online systems as part of a scientific workflow is astonishingly small, with most analysis still occurring with local computing resources and software tools. When APIs are used, they primarily serve data acquisition and encoding, with discrete outputs that can be curated. Many economics software tools remain proprietary. The curation of licensed software, and access to online computational resources for statistical processing is not dissimilar to the use of large language models via commercial providers, whether these are called AI or not. Thus, I will focus on similarities with problems researchers already face when working with black-box systems, commercial software, and external APIs. However, AI systems magnify these challenges considerably compared to traditional economic research. I address how researchers may alleviate some of these challenges, similar to those existing solutions.

**Fundamental Reproducibility Targets**

Effective reproducibility verification requires not only conducting the re-analysis of data via provided code, but also establishing that research materials can be accessed by others, in the future, within reasonable timeframes. To this end, the standard solution is to use trusted repositories to create partial research compendia, comprised of all code and any shareable data. When data cannot be openly shared, the access mechanism itself must be described, and is often tested, to ensure that at least in the short terms, such data can be obtained by others. In economics, established tools and standards guide this process, including the [Template README](https://social-science-data-editors.github.io/template_README/) framework that describes data provenance and access conditions for all raw data, documents all data transformations beginning with raw data, and provides complete code including processing scripts for non-shareable data.

However, AI research introduces additional complexity layers that traditional frameworks must accommodate. The integration of algorithmic transparency, data dependencies, and machine learning model archiving creates new categories of reproducibility challenges that existing methodologies inadequately address.

**Data Provenance and Preservation Challenges**

Research involving AI systems typically encounters three distinct types of data: training data used to develop models, analysis data used for research purposes, and output data generated by algorithms. Each category presents unique challenges for reproducibility. The fundamental questions surrounding data provenance become more complex in AI contexts: determining the origins of training and analysis data, establishing whether data can be shared with others, assessing continued accessibility, and evaluating preservation capabilities.

The question of whether large language models constitute data or software becomes particularly relevant. While this analysis treats models as software, their data-like characteristics in terms of size and access patterns create preservation challenges more similar to large datasets than traditional code repositories.

In the social sciences, the typical workflow starts with pre-trained LLMs, which may be fine-tuned with specific data to create customized models, or used as-is for direct (zero-shot) inference. The curation of the raw training data, and often of the weights generated by such processes, is relegated to data and computer sciences [[\*](https://doi.org/10.1162/99608f92.a50ec6e6)]. Social scientists do fine-tune existing LLMs to generate analysis data. I will not exclude from the present paper the very important discussion about transparency of the models themselves, of their embedded prompts, training data, and the licensing of their weights.

Both the tuned models and resulting analysis data should ideally be preserved, yet this presents significant challenges. The size of these datasets often exceeds traditional storage capabilities, and questions arise regarding their preservation location and associated licensing requirements. The general purpose and social-science focused repositories tend to have relative small quotas for data storage, often exceeded by large language models. Specialized platforms exist, such as Hugging Face, which provide some level of model preservation, but their commitment to long-term preservation is weak, and their focus is on sharing models, not necessarily preserving them. While Hugging Face assigns DOIs to some models and provides presumptions of preservation, no formal guarantees exist. The transition of organizations from non-profit to commercial status, as seen with OpenAI’s evolution from releasing GPT-2 openly to restricting access to GPT-3, illustrates how access to specific model versions may become limited over time.

Consider the more traditional example of immigration research utilizing historical documents. Effective data preservation might involve hosting raw scanned PDFs of primary sources on platforms like Harvard Dataverse, alongside processed replication datasets. This approach demonstrates how substantial computing resources required for data processing can be balanced against more manageable storage requirements for preservation. In the context of AI, this might mean preserving the raw model, and its potentially larger fine-tuned version.

In addition, for LLM-specific applications, researchers must consider whether tuned models can be released given privacy constraints,

**Computational Requirements and Software Dependencies**

The computational environment presents another critical dimension for reproducibility. Documentation must describe both the hardware and software configurations used by researchers, including relevant hardware specifications, memory requirements, storage needs, and necessary software with specific library versions. This is even more relevant for research using LLMs, which evolve rapidly, and proper environment preservation becomes essential. While this has well-accepted solutions, such as virtual machines, Docker containers, or python environments, social science researchers have historically not been quick to adopt these methods. In addition, LLM applications often require stricter operating conditions than typical economic research, potentially necessitating specialized hardware configurations such as specific and expensive graphics cards (GPUs), which must be purchased or rented.

The requirement to rely on expensive computing resources, however, is not new. Currently, most economists utilize commercial software with relatively standardized access, and sometimes rely on commerically collected data, with subscriptions costing many thousands of dollars. Commercial software costs vary dramatically, from thousands to hundreds of thousands of dollars. Some research requires substantial computational time, creating significant costs. Free open-source software only alleviates a small part of those costs. The use of LLMs via commercial providers or running on rented cloud resources is simply an extension of that practice.

While these are real impediments to greater accessibility, and thus possibly to “open science”, these impediments are not new. Creating scientific output costs money, and the use of LLMs does not change that. It may not even cost more than other methods.

**Reproducible Execution and Documentation**

The most fundamental test of reproducibility requires that code execute completely from beginning to end without error and ideally without user intervention. This should recreate all figures, tables, and numerical results included in research publications. Existing challenges in the social sciences include the poor dissemination of “push button reproducibility”, the ability to run the entire analysis, from data acquisition to artifact generation, with a single controller script. Only approximately 31% of replication packages in top economics journals contain main controller scripts, indicating widespread deficiencies in basic reproducibility practices even before considering AI-specific challenges.

AI research introduces additional complexities. Even within short time periods, inadequate software version control combined with API changes, as demonstrated by transitions in OpenAI’s interface specifications within only a few months, can render previously functional code inoperable. Rapidly evolving APIs create reproducibility barriers independent of research methodology quality. More importantly, many earlier LLMs were fundamentally probabilistic, without the ability to fix a seed for PseudoRNGs, and were updated “on-the-fly”, leading to both software and data inputs changing over time, in uncontrolled fashion. More recent models have improved on this, providing stronger (albeit apparently not perfect) guarantess of quasi-deterministic output.

One possible solution to this is to use downloadable models, whether openly licensed or those subject to restrictive licenses. If the model supports it, it guarantees that the input data is under researcher control. This, however, takes us back to the problem of preserving these physically large models. Furthermore, researchers are drawn to the “latest and greatest” models, which may not be available for download. Most downloadable models may lag in performance and quality. Researchers wishing to improve reproducibility face a tradeoff between said reproducibility and the quality of their inference.

**Some Guidance**

***Logging and Documentation Practices***

Comprehensive logging can serve as crucial evidence of code execution, particularly important for computationally expensive operations or when working with data that cannot be shared. This is true even before considering LLMs. Most statistical software provides mechanisms for creating execution records, though some require explicit instruction to generate verbose output or command-line options for log file creation. For Python applications, custom wrapper functions can track function calls with timestamps, arguments, and return values. These approaches provide documentation that code has executed while capturing relevant metadata about the computational process. Decorators can automatically log function calls with parameters and timing.

These logs serve dual purposes: documenting successful execution for researchers and providing evidence of reproducibility for journals and reviewers. In cases where replication requires expensive computational resources or access to restricted data, log files may provide the only feasible method for demonstrating code functionality. The credibility of such log files, which are not impervious to manipulation, can be enhanced by using tools that generate cryptographic hashes of log contents, ensuring integrity and authenticity. The TRACE project <https://transparency-certified.github.io/> provides one approach, adaptable to certain circumstances.

***Environmental Management and Dependency Specification***

Most AI-based research in the social sciences uses Python, so I will focus this discussion to that programming language, but similar methods exist for Julia and R, among the programming languages in use by social scientists. Python environments are a potential solution for many reproducibility challenges, but their implementation can present particular challenges for reproducibility. While “pip freeze” is recommended for documentation, it may not generate robust reproducible environments due to platform-specific dependencies. The solution involves identifying minimal package requirements corresponding to explicit imports in code, then pruning requirements files to essential components. The goal involves creating environments that capture necessary dependencies while allowing package managers to resolve secondary dependencies appropriately.

In the context of AI research, the environment must also account for ubitiquous use of API keys. Secure programming methods are not well established in the social sciences, and many researchers have very manual methods, if at all, of managing API keys. Furthermore, most Python libraries do not allow to pin the LLM used within. This is relegated to the user-generated code, which must be careful to document and use the specific model versions used. The use of “default” values or simply using the “latest” model is strongly discouraged, as it is detrimental to even short-term reproducibility. This parallels API data access issues, where “latest” data may be revised between research and publication.

***Cost Considerations and Resource Planning***

AI research involves significant computational costs that must be documented and planned. Traditional economic research might require expenditures for software licenses, data access, or travel for restricted data. Consider on-site or controlled access to data enclaves. The Federal Statistical Research Data Centers provide access to researchers physically present in the United States, and in many cases, only via physical secure rooms. Much European data is only available either on-site or from researchers physically present in Europe. For researchers, this can create considerable travel costs, which must be covered by research funds or grants. AI applications also may have substantial costs for model training, inference operations, and repeated executions for variability assessment. These costs must be quantified for future researchers wishing to replicate results, even if such costs may be declining rapidly over time for access via commercial providers. Arguably, these costs exceed “traditional” economic research by orders of magnitude, when such traditional research uses relatively small public-use datasets that can be analyzed by students on cheap laptops. Training models in cloud environments, running models on datasets multiple times, and assessing result variability can each cost thousands of dollars. These expenses make replication economically challenging and may limit reproducibility testing to well-funded research groups.

**Summary of Best Practices**

Effective reproducibility in AI research requires implementation of several key practices. Environmental management from project inception ensures consistent computational conditions. Comprehensive logging provides evidence of execution, particularly valuable when repetition is expensive. Version precision applies to input data, software dependencies, and critically, model versions used - avoiding references to “latest” models that may change over time. Complete code inclusion encompasses prompts, intermediate responses, and processing scripts, even when underlying data cannot be shared. In AI contexts, the code itself, including specific prompts used with models, constitutes crucial methodological information that affects result interpretation. Metadata documentation should specify random seeds where possible, hyperparameters, “temperature” settings, and other configuration details. Prompts themselves should be considered metadata requiring preservation, as slight variations in prompt formulation can substantially affect model outputs.

Data preservation strategies must consider licensing constraints and storage requirements. Industry repositories may serve adequately for sharing purposes, while academic repositories like Zenodo and Dataverse provide formal preservation commitments. The toolkit for preserving large datasets exceeding 200GB remains underdeveloped, presenting ongoing challenges for AI research that routinely generates datasets of this scale.

**Conclusion**

While AI and LLMs are not fundamentally different from other computational approaches regarding reproducibility principles, they may appear to magnify existing difficulties compared to “typical” economic research practices. The solutions involve maintaining reproducible practices from project initiation, exercising computational empathy by considering others’ technical constraints, ensuring precision in version specification, and utilizing existing resources like template README guides and self-checking reproducibility tools.

The field continues evolving rapidly, requiring researchers to adapt traditional reproducibility practices to accommodate new technological constraints while maintaining scientific rigor. Success depends on recognition that reproducibility challenges, while amplified in AI contexts, remain addressable through careful attention to documentation, environmental management, and transparent reporting practices. The fundamental goal remains unchanged: ensuring that scientific findings can be verified, understood, and built upon by other researchers, thereby maintaining the integrity and advancement of scientific knowledge in an (increasingly?) AI-dependent research landscape.