evidence. In addition, engineering applications of ML often operate in a rapidly changing context and have access to large datasets, so small differences in performances are often not as important, whereas scientific claims are sensitive to small performance differences between ML models.

Why do we call it a reproducibility crisis?

We say that ML-based science is suffering from a reproducibility crisis for two related reasons. First, our results show that reproducibility failures in ML-based science are systemic. In nearly every scientific field that has carried out a systematic study of reproducibility issues, papers are plagued by common pitfalls. In many systematic reviews, a majority of the papers reviewed suffer from these pitfalls. Similar problems are likely to arise in many fields that are adopting ML methods. Second, despite the urgency of addressing reproducibility failures, there are no systemic solutions that have been deployed for these failures. Scientific communities are discovering the same failure modes across disciplines but have yet to converge on best practices for avoiding reproducibility failures.

Calling attention to and addressing these widespread failures is vital to maintaining public confidence in ML-based science. At the same time, the use of ML methods is still in its infancy in many scientific fields. Addressing reproducibility failures pre-emptively in such fields can correct a lot of scientific research that would otherwise be flawed.

Toward a solution: A taxonomy of data leakage

We now provide our taxonomy of data leakage errors in ML-based science. Such a taxonomy can enable a better understanding of why leakage occurs and inform potential solutions. Our taxonomy is comprehensive and addresses data leakage arising during the data collection, pre-processing, modeling, and evaluation steps. In particular, our taxonomy addresses all cases of data leakage that we found in our survey (Figure 1). Some of the categories in our taxonomy, e.g., sampling bias [L3.3], were not considered types of leakage in prior work, but they have the same cause as other categories of leakage: spurious correlations between the outcome variables and the features. They also have the same effect: they lead to overestimates of model performance.

[L1] Lack of clean separation of training and test dataset. If the training dataset is not separated from the test dataset during all pre-processing, modeling, and evaluation steps, the model has access to information in the test set before its performance is evaluated. Because the model has access to information from the test set at training time, the model learns relationships between the predictors and the outcome that would not be available in additional data drawn from the distribution of interest. The performance of the model on these data therefore does not reflect how well the model would perform on a new test set drawn from the same distribution of data. This can happen in several ways, such as:

[L1.1] No test set. Using the same dataset for training and testing the model is a textbook example of overfitting, which leads to overoptimistic performance estimates.[54]

[L1.2] Pre-processing on training and test set. Using the entire dataset for any pre-processing steps, such as imputation or over/under sampling, results in leakage. For instance, using oversampling before splitting the data into training and test sets leads to an imperfect separation between the training and test sets because data generated using oversampling from the training set will also be present in the test set.

[L1.3] Feature selection on training and test set. Feature selection on the entire dataset results in using information about which feature performs well on the test set to make a decision about which features should be included in the model.

[L1.4] Duplicates in datasets. If a dataset with duplicates is used for the purposes of training and evaluating an ML model, the same data could exist in the training set and the test set.

[L2] Model uses features that are not legitimate. If the model has access to features that should not be legitimately available for use in the modeling exercise, this could result in leakage. One instance when this can happen is if a feature is a proxy for the outcome variable.[10] For example, Filho et al.[55] find that a recent study included the use of anti-hypertensive drugs as a feature for predicting hypertension. Such a feature could lead to leakage because the model would not have access to this information when predicting the health outcome for a new patient. Further, if the fact that a patient uses anti-hypertensive drugs is already known at prediction time, the prediction of hypertension becomes a trivial task.

The judgment of whether the use of a given feature is legitimate for a modeling task requires domain knowledge and can be highly problem specific. As a result, we do not provide subcategories for this sort of leakage. Instead, we suggest that researchers clearly specify which features are suitable for a modeling task and justify their choice using domain expertise.

[L3] Test set is not drawn from the distribution of scientific interest. The distribution of data on which the performance of an ML model is evaluated differs from the distribution of data about which the scientific claims are made. The performance of the model on the test set does not correspond to its performance on data drawn from the distribution of scientific interest.

[L3.1] Temporal leakage. When an ML model is used to make predictions about a future outcome of interest, the test set should not contain any data from a date before the training set. If the test set contains data from before the training set, the model is built using data "from the future" that it should not have access to during training and can cause leakage.

[L3.2] Nonindependence between training and test samples. Nonindependence between training and test samples constitutes leakage, unless the scientific claim is about a distribution that has the same dependence structure. In the extreme (but unfortunately common) case, training and test samples come from the same people or units. For example, Oner et al.[56] find that a recent study on histopathology uses different observations of the same patient in the training and test sets. In this case, the scientific claim is being made about the ability to predict gene mutations in new patients; however, it is evaluated on data from old patients (i.e., data from patients in the training set), leading to a mismatch between the test set distribution and the scientific claim. Similarly, for predicting protein function, the family of the protein can lead to dependencies if proteins from the same family are split across the training and test sets.[57] The train-test split should account for the dependencies in the data to ensure correct performance evaluation. Methods such as "block cross-validation" can partition the dataset strategically so that the performance evaluation does not suffer from data leakage and

overoptimism.[58,59] Handling nonindependence between the training and test sets in general (i.e., without any assumptions about independence in the data) is a hard problem, because we might not know the underlying dependency structure of the task in many cases.[60]

[L3.3] Sampling bias in test distribution. Sampling bias in the choice of test dataset can lead to data leakage. One example of sampling bias is spatial bias, which refers to choosing the test data from a geographic location but making claims about model performance in other geographic locations. Another example is selection bias, which entails choosing a non-representative subset of the dataset for evaluation. For example, Bone et al.[61] highlight that in a study on predicting autism using ML models, excluding the data corresponding to borderline cases of autism leads to leakage because the test set is no longer representative of the general population about which claims are made. In addition, borderline cases of autism are often the trickiest to diagnose, so excluding them from the evaluation set is likely to lead to overoptimistic results. Cases of leakage caused by sampling bias can often be subtle. For example, Zech et al.[62] find that models for pneumonia prediction trained on images from one hospital do not generalize to images from another hospital because of subtle differences in how images are generated in each hospital.

A model may have leakage when the distribution about which the scientific claim is made does not match the distribution from which the evaluation set is drawn. ML models may also suffer from a related but distinct limitation: the lack of generalization when we try to apply a result about one population to another similar but distinct population. Several issues with the generalization of ML models operating under a distribution shift have been highlighted in ML methods research, such as fragility toward adversarial examples,[63] image distortion and texture,[64] and overinterpretation.[65] Robustness to distribution shift is an ongoing area of work in ML methods research. Even slight shifts in the target distribution can cause performance estimates to change drastically.[66] Despite ongoing work to create ML methods that are robust to distribution shift, best practices to deal with distribution shift currently include testing the ML models on the data from the distribution we want to make claims about.[52] In ML-based science, where the aim is to create generalizable knowledge, we should take results that claim to generalize to a different population from the one models were evaluated on with caution.

Other issues identified in our survey

Computational reproducibility issues. Computational reproducibility of a finding refers to sharing the complete code and data needed to reproduce the findings reported in a paper exactly. This is important to enable external researchers to reproduce results and verify their correctness. Five papers in our survey outlined the lack of computational reproducibility in their field.

Data quality issues. Access to good-quality data is essential for creating ML models.[67,68] Issues with the quality of the dataset could affect the results of ML-based science. Ten papers in our survey highlighted data quality issues such as not addressing missing values in the data, the small size of datasets compared with the number of predictors, and the outcome variable being a poor proxy for the phenomenon being studied.

Metric choice issues. A mismatch between the metric used to evaluate performance and the scientific problem of interest leads to issues with performance claims. For example, using accuracy as the evaluation metric with a heavily imbalanced dataset leads to overoptimistic results, because the model can get a high accuracy score by always predicting the majority class. Four papers in our survey highlighted metric choice issues.

Use of standard datasets. Reproducibility issues arose despite the use of standard, widely used datasets, often because of the lack of standard modeling and evaluation procedures such as fixing the train-test split and evaluation metric for the dataset. Seven papers in our survey highlighted that issues arose despite the use of standard datasets.

Model info sheets for detecting and preventing leakage

Our taxonomy of data leakage highlights several failure modes that are prevalent in ML-based science. To detect cases of leakage, we provide a template for a model info sheet to accompany scientific claims using predictive modeling as a supplemental document (supplemental experimental procedures, section S4). The template consists of 21 questions that elicit precise arguments needed to justify the absence of leakage.

Prior work on model cards and reporting standards

Our proposal is inspired by prior work on model cards and checklists, which we now review. Mitchell et al.[40] introduced model cards for reporting details about ML models, with a focus on precisely reporting the intended use cases of ML models. They also addressed fairness and transparency concerns: they require that the performance of ML models on different groups of users (e.g., on the basis of race, gender, and age) is reported and documented transparently. These model cards complement the datasheets introduced by Gebru et al.[69] to document details about datasets in a standard format.

The use of checklists has also been impactful in improving reporting practices in the few fields that have adopted them.[70] Although checklists and model cards provide concrete best practices for reporting standards,[38–40,71] current efforts do not address pitfalls arising because of leakage. Further, even though several scientific fields, especially those related to medicine, have adopted checklists to improve reporting standards, most checklists are developed for specific scientific or research communities instead of ML-based science in general.

Scientific arguments to surface and prevent leakage

When ML models are used to make scientific claims, it is not enough to simply separate the training and test sets and report performance metrics on the test set. Unlike research in ML methods, where a model's performance on a hypothetical task (i.e., one that is not linked to a specific scientific claim) is still of interest to the researcher in some cases,[72] in ML-based science, claims about a model's performance need to be connected to scientific claims using explicit arguments. The burden of proof for ensuring the correctness of these arguments is on the researcher making the scientific claims.[73]

In our model info sheet, we ask researchers to answer 21 questions. These questions help them present three arguments that are essential for determining that scientific results that use ML methods do not suffer from data leakage. Note that most ML-based science papers do not present any of the three