SDSC SAN DIEGO SUPERCOMPUTER CENTER

NTNU | Norwegian University of Science and Technology

Sources of Irreproducibility in Machine Learning

FARR Workshop 2024

Kevin Coakley - 10/10/2024

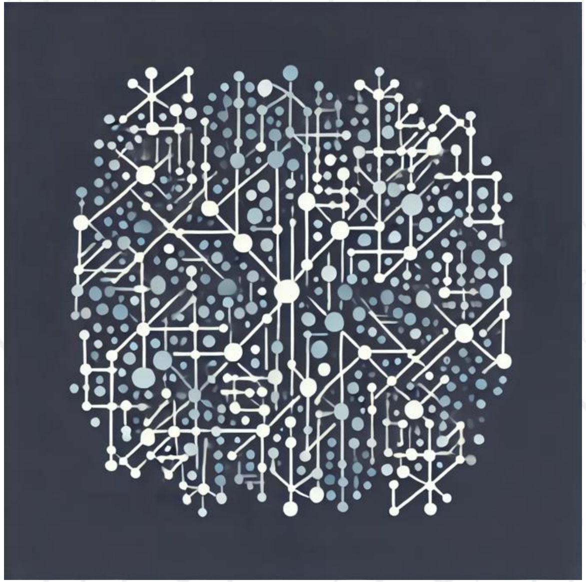
Download Slides Here

Reproducibility Crisis in ML

- Reproducibility is critical for trust in scientific findings, particularly in fields with high-stakes applications like healthcare, autonomous systems, and finance[1].
- In one study, the accuracy of models from 16 identical training runs varied by as much as 10.8%, even after removing weak models[2].
- Machine learning models that produce high variance in results challenge the reliability of findings[1].
- A survey of 901 researchers and practitioners found many respondents were unaware of (31.9%) or unsure about (21.8%) any variance and 83.8% were unaware or uncertain of variance caused by implementation choices[2].

1. Gundersen, Odd Erik, et al. "Sources of Irreproducibility in Machine Learning: A Review." arXiv e-prints (2022): arXiv-2024.
2. H. V. Pham *et al.*, “Problems and Opportunities in Training Deep Learning Software Systems: An Analysis of Variance,” p. 13, 2020.

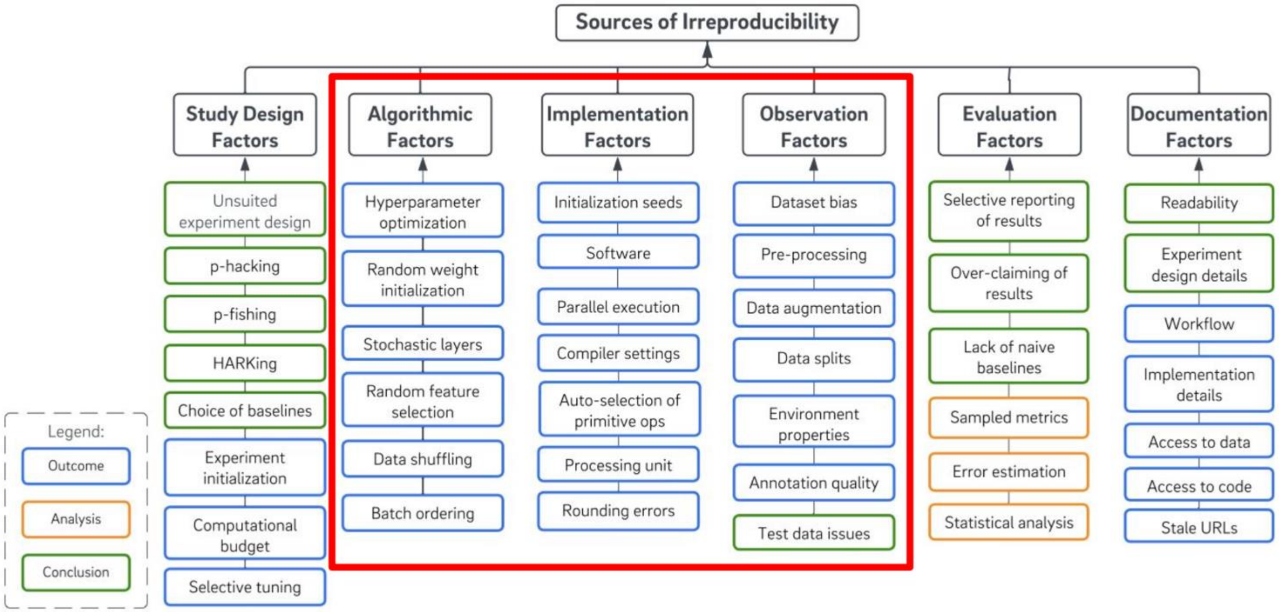
Reproducibility Crisis in ML



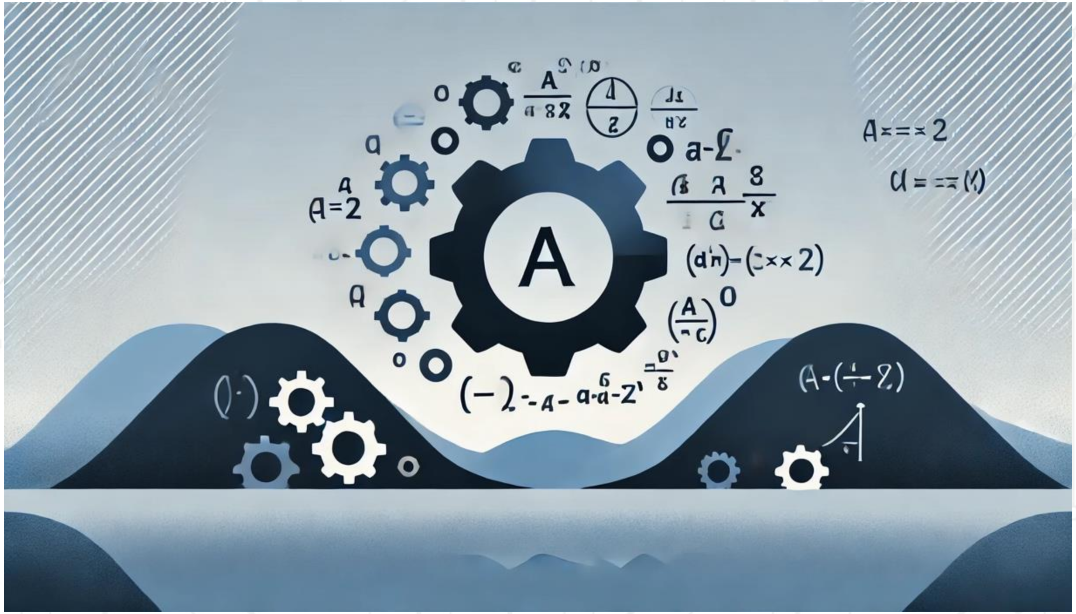
- When developing or evaluating ML models it is critical to understand the sources of variation that can cause ML results to be irreproducible.

Sources of Irreproducibility

Sources of Irreproducibility - Overview



Gundersen, Odd Erik, et al. "Sources of Irreproducibility in Machine Learning: A Review." arXiv e-prints (2022): arXiv-2024.



Algorithmic Factors

Algorithmic Factors Causing Irreproducibility - Part 1

AF - Hyperparameter Optimization

Different hyperparameter optimization methods (random, grid, Bayesian optimization, intuition) and optimization budgets (study design factor) will affect outcome.

AF - Random Weight Initialization

The random initialization of weights in neural networks can lead to the model to converge to different local minima.

Algorithmic Factors Causing Irreproducibility - Part 3

AF - Data Shuffling

Random data shuffling done during training so learning converges faster can cause outcomes to differ.

AF - Batch Ordering

Due to memory limitations, data samples are fed into DL algorithms in batches. Randomizing batch order between epochs results in different outcomes between training runs.

Algorithmic Factors Causing Irreproducibility - Part 2

AF - Stochastic Layers

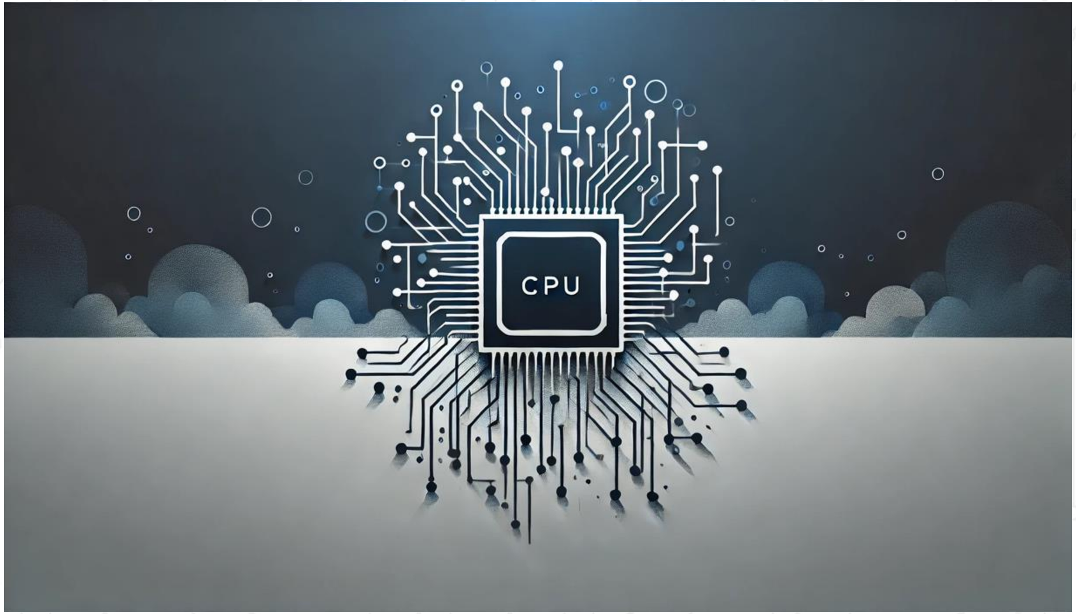
Stochastic model layers, like Dropout, intended to make deep neural networks more robust, affect their outcome.

AF - Random Feature Selection

Many learning algorithms rely on selecting features at random during training, like Random Forests. Which randomly selected features are chosen will influence the outcome.

Algorithmic Factors - Conclusions

- **Stochasticity** in deep learning inherently leads to different outcomes across runs.
- **Significant performance variations** between runs can affect conclusions.
- **Setting and documenting random seeds** is critical for algorithmic reproducibility.
- **Consistent outcomes don't guarantee robustness** — variability must be considered.
- **Report performance variation** over multiple runs to ensure transparency and reliability.



Implementation Factors

Implementation Factors Causing Irreproducibility - Part 1

IF - Initialization Seeds

Different seeds used to initialize the pseudo-random number generator produce different outcomes. The same seed on different platforms produces different outcomes.

IF - Software

Outcomes across DL frameworks (TensorFlow, PyTorch) can vary significantly. Different software (libraries, operating systems) or versions may implement the same algorithm differently, causing different outcomes.

Implementation Factors Causing Irreproducibility - Part 2

IF - Parallel Execution

Random completion order of parallel tasks introduces variation. Truncation error of floating-point calculations introduces variability as $A + B + C \neq C + B + A$ when calculated in parallel.

IF - Compiler Settings

Hong et al[1], found severe sensitivity to Intel compiler optimization levels for weather simulations that rely on floating-point calculations.

1 S.-Y. Hong *et al.*, "An Evaluation of the Software System Dependency of a Global Atmospheric Model," *Monthly Weather Review*, vol. 141, no. 11, pp. 4165–4172, Nov. 2013, doi: 10.1175/MWR-D-12-00352.1.

Implementation Factors Causing Irreproducibility - Part 3

IF - Auto-selection of Primitive Ops

High level libraries implement DL algorithms using GPU-optimized DL primitives from low-level libraries (cuDNN and CUDA). Autotune in cuDNN automatically benchmarks several modes of operation which might change between runs.

IF - Processing Unit

Changing the processor can affect results. The same GPU chip on hardware from different manufacturers can produce different outcomes when running deterministically.

Implementation Factors Causing Irreproducibility - Part 4

IF - Rounding Errors

Different hardware architectures and software implement the rounding of floating-point numbers in different ways. These rounding errors accumulate during long-running calculations, particularly when using GPUs.

Implementation Factors - Conclusions

- Variations in **software and hardware** mirror the inconsistencies seen in physical labs.
- Treat the software and hardware environment as a **calibrated scientific instrument** for ML experiments.
- Consistent results require controlling for differences in **software libraries, hardware configurations, and parallelization**.
- Always **document and share** all configurations to support reproducibility.

Observation Factors

Observation Factors Causing Irreproducibility - Part 1

OF - Dataset Bias

The methods used to gather data (manual or automated) and the way data is captured introduce biases to datasets.

OF - Data Pre-processing

Differences in data pre-processing will change outcomes, so the applied pre-processing techniques must be well documented to facilitate reproducibility.

Observation Factors Causing Irreproducibility - Part 2

OF - Data Augmentation

Stochastic data augmentation procedures are influenced by both algorithmic and implementation factors, which leads to differences in training data and outcomes.

OF - Data Splits

Differences in data splits cause a difference in outcomes.

Observation Factors Causing Irreproducibility - Part 3

OF - Environment Properties

Stochasticity and different dynamic properties of the testing environment could affect the outcome, especially in continuous control simulators such as those used in deep reinforcement learning.

OF - Annotation Quality

Differences in annotations made by humans will affect the target value and the outcome.

Observation Factors Causing Irreproducibility - Part 4

OF - Test Data Issues

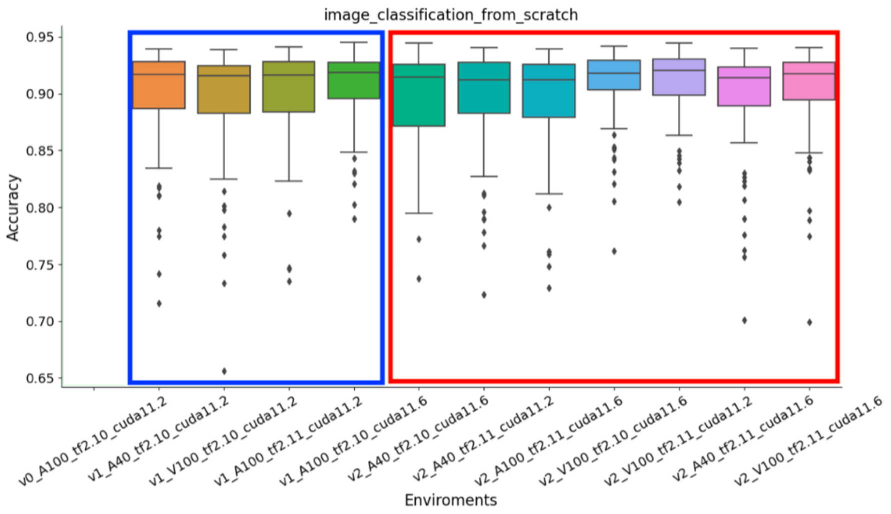
Model performance is overestimated when models are trained on data that should only be available at test time (data leakage).

Observation Factors - Conclusions

- Observation factors might affect the **outcome and interpretation** of an experiment.
- **Dataset bias and pre-processing** significantly impact model outcomes and interpretations.
- Mitigate these effects by setting **random seeds** and thoroughly documenting **data pre-processing** and **provenance**.
- **Careful handling** of duplicate data, outliers, and missing values is essential to avoid **bias**.
- **Dataset shifts** over time may cause models to become outdated—**regularly reassess and update** them.

Why Noise Control Isn't Enough

- "Simply removing noise from one part of the technical stack is not a robust way to improve training stability[1]"
- The effect of these sources of irreproducibility doesn't appear to be cumulative. Blue one source changed, red two sources changed (Sources: GPU, TensorFlow version, CUDA/CUDNN version)[2]



1 Zhuang, D., Zhang, X., Song, S., Hooker, S.: Randomness in neural network training: Characterizing the impact of tooling. Proceedings of Machine Learning and Systems 4, 316–336 (2022)

2 Coakley Unpublished

Conclusions

- **Irreproducibility is a Complex Challenge**
  - Arises from various factors across algorithms, implementations, and data handling.
- **Interconnectedness Across the Technical Stack**
  - Algorithmic, implementation, and observational factors all contribute to variability in results.
- **No Single Fix**
  - Controlling one aspect, such as random seeds, is insufficient for ensuring training stability.
  - Addressing irreproducibility requires attention to the entire technical pipeline.

Questions?

Contact Information:

Kevin Coakley

kcoakley@sdsc.edu

Download Slides Here