



FAIR Universe



BERKELEY LAB



W

UNIVERSITY of
WASHINGTON



Universiteit
Leiden

FAIR Universe 2024: Higgs ML Uncertainty Challenge

Yulei Zhang

University of Washington, Seattle

Team: Wahid Bhimji, Paolo Calafiura, Ragansu Chakkappai, Yuan-Tang Chou, Sascha Diefenbacher, Jordan Dudley, Steven Farrell, Aishik Ghosh, Isabelle Guyon, Chris Harris, Shih-Chieh Hsu, Elham Khoda, Rémy Lyscar, Alexandre Michon, Benjamin Nachman, Benjamin Thorne, Peter Nugent, Po-Wen Chang, Mathis Reymond, Benjamin Sluijter, David Rousseau, Ihsan Ullah, Daniel Whiteson, Yulei Zhang

Nov 21st, 2024



ML Hackathon



<https://fair-universe.lbl.gov/>

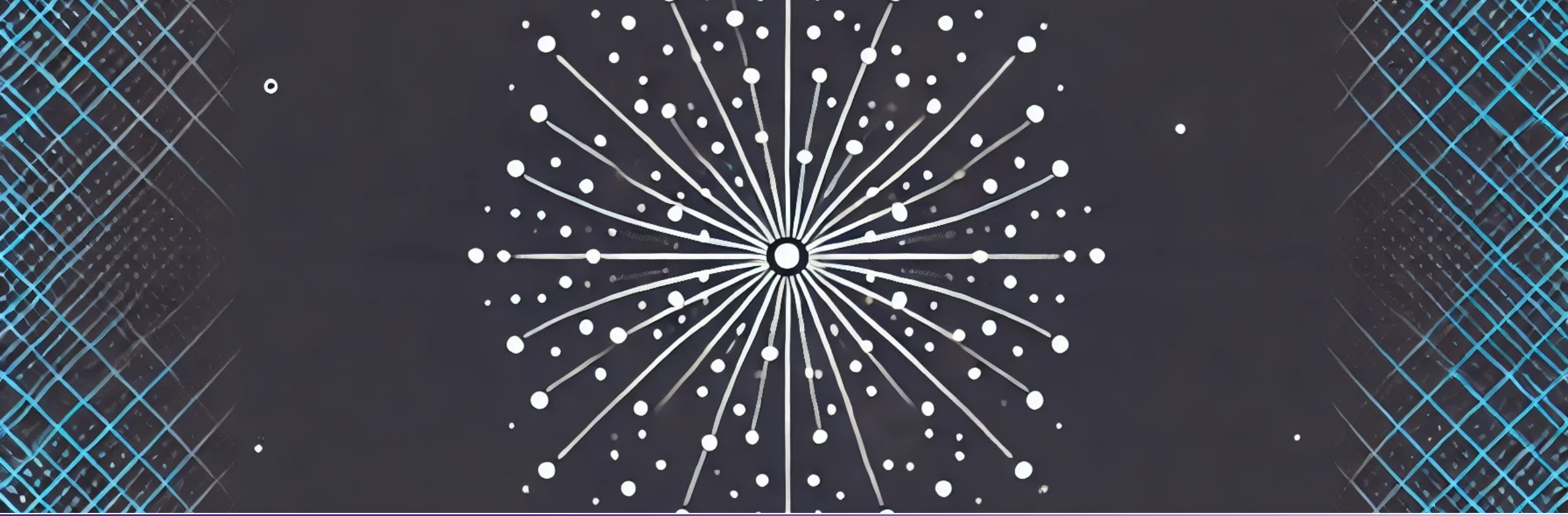
Collaborators: U. Berkeley, U. Washington, Chalearn, IJCLab-Orsay, UC Irvine, UCSD, Universiteit Leiden

Project Aims:

- An Open, **Large-Compute-Scale AI Ecosystem** for sharing datasets, training large models, and **hosting challenges and benchmarks**
- Progressive **challenge series** on **measuring and minimizing the effects of systematic uncertainties** in HEP (particle physics and cosmology)

Broad team Involvement in major AI and HEP challenges like [HiggsML](#), [TrackML](#), [LHC Olympics](#), [Fast Calorimeter Simulation Challenge](#), and wider (e.g [NeurIPS competition](#), [MLPerf HPC](#))

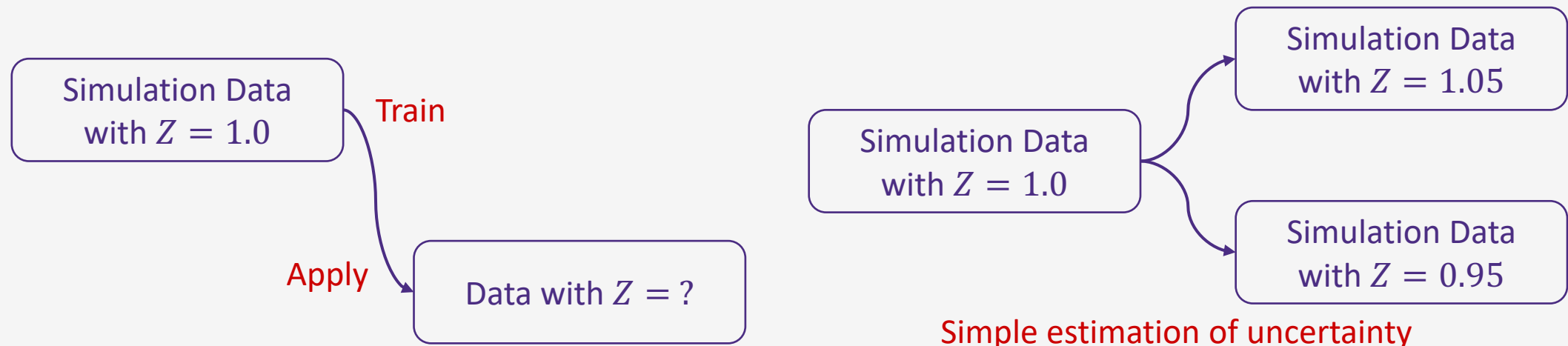
➤ **Fair Universe HiggsML Uncertainty Challenge : NeurIPS competition** 🎉



Measuring and minimizing the effects of systematic uncertainties in HEP

Bias and Uncertainty in Fundamental Sciences

- Machine learning models in HEP are typically **trained using simulations**, which include **certain assumptions** and **systematic uncertainties** (called “epistemic” uncertainties, **labeled as “Z”**).
- However, when these models **are applied to real-world data**, the conditions (detector states) may differ, leading to an unknown Z value.
- **Common Approach:** First, train the model on standard simulation data ($Z = 1$). Then, estimate uncertainties by comparing results from simulations with different Z values. Shift Z slightly and examine how it affects the results or use a **full profile likelihood** to evaluate the impact.



FAIR Universe: HiggsML Uncertainty Challenge

- **Focus on novel approaches** to address model uncertainty, including decorrelation, adversarial training, and uncertainty-aware techniques.
 - Example techniques: “Pivot” by Louppe et al. ([arXiv:1611.01046](#)), Ghosh et al. ([PhysRevD.104.056026](#)), Inferno ([arXiv:1806.04743](#)), and others.
 - **Challenge:** Scaling methods to handle multiple values of systematic uncertainties (Z), which increases training complexity and cost.
- **Key gaps:** Current benchmarks rely on **single systematic uncertainties** and **limited datasets** (based on [HiggsML 2014](#)), which restrict scaling and broader adoption.
- **New dataset for the challenge:** Extension of the original HiggsML dataset.
 - **Improvements:** **Larger dataset** (from 800k to **~300M events**), faster simulation, **parameterized systematics** (nuisance parameters).
 - **Task:** Provide a confidence interval on signal strength in a pseudo-experiment with a given signal.

Dataset Overview

- **Simulated Dataset (280 million events)**: Representative of high-energy proton collision data from the **ATLAS experiment** at the Large Hadron Collider (LHC).
- **Pythia 8.2** and **Delphes 3.5.0** for simulation, and data organized in a tabular format with **28 features** per event.
- Includes a **biasing script** introducing systematic uncertainties (Nuisance Parameters) for realistic challenges.

Process	Number Generated	LHC Events	Label
Higgs	52101127	1015	signal
Z Boson	221724480	1002395	background
Di-Boson	2105415	3783	background
$t\bar{t}$	12073068	44190	background

Public

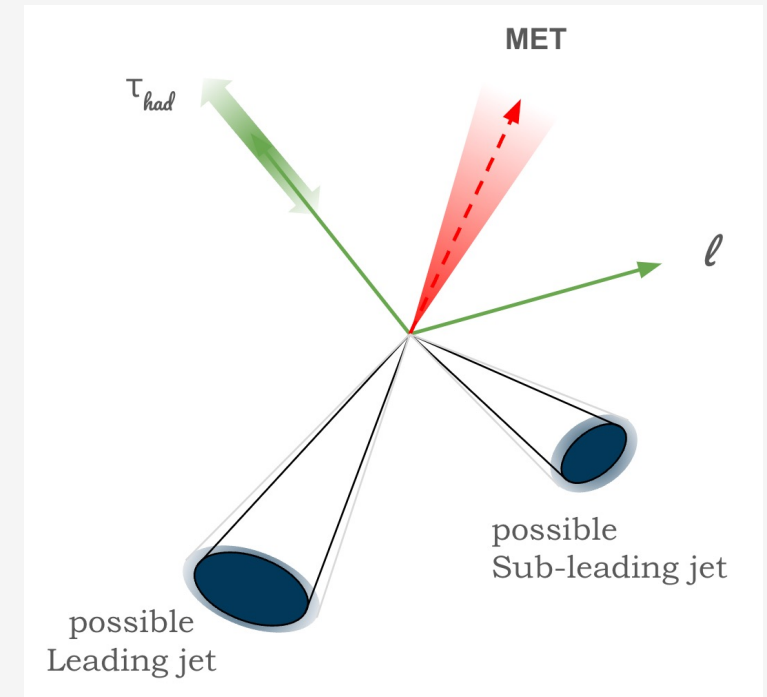
- **Train/Val: 100** \times LHC @ 10 fb⁻¹
- **Pseudo: 60** \times LHC @ 10 fb⁻¹

Private

- **Pseudo: 60** \times LHC @ 10 fb⁻¹

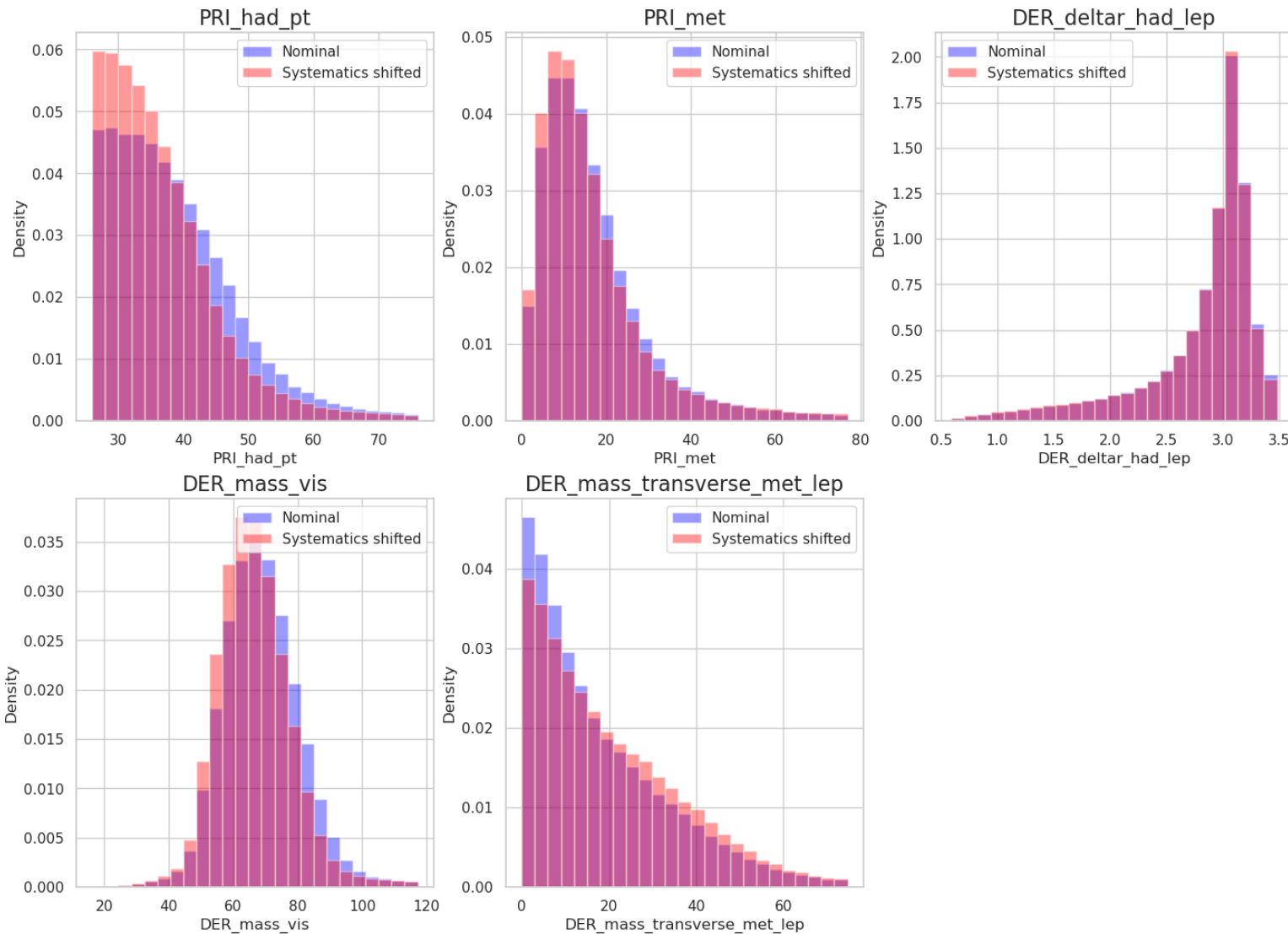
Systematics Parameterization: Nuisance Parameters

- New systematic parameterization method [[white paper](#), [github](#)]
- 6 systematic uncertainties!
- **Object-level uncertainties:**
 - Tau Energy Scale (and correlated MET) [0.9, 1.1, $\sigma = 0.01$]
 - Jet Energy Scale (and correlated MET impact) [0.9, 1.1, $\sigma = 0.01$]
 - **Additional randomized Soft MET [0, 5, $\sigma = 1.0$]**
- **Event category normalization:**
 - Overall Background norm [0.99, 1.01, $\sigma = 0.001$]
 - VV background norm [0, 2, $\sigma = 0.25$]
 - $t\bar{t}$ background norm [0.8, 1.2, $\sigma = 0.02$]



We operate within the framework of ‘**known unknowns**,’ while addressing ‘**unknown unknowns**’ remains a more complex challenge beyond this scope.

Example of one NP: Tau Energy Scale Impact

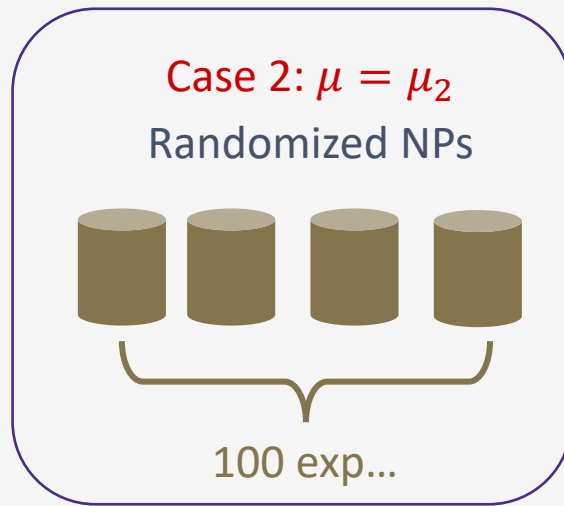
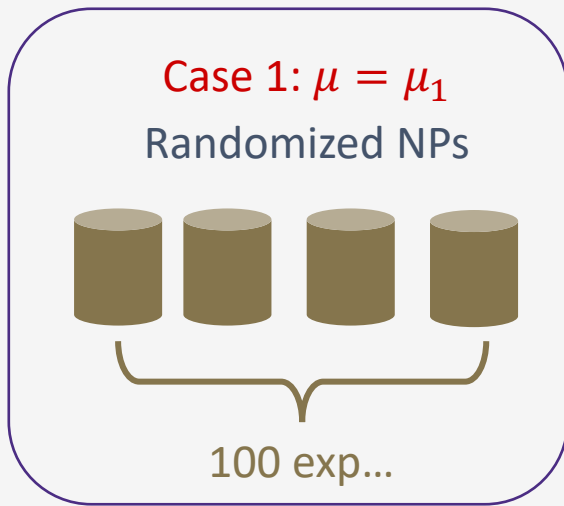


- Histogram between nominal ($TES = 1$) and shifted ($TES = 0.9$)
- $TES = 0.9$ is an exaggeration, in practice it is sampled with $\mathcal{G}(1, 0.01)$ and with boundary $[0.9, 1.1]$

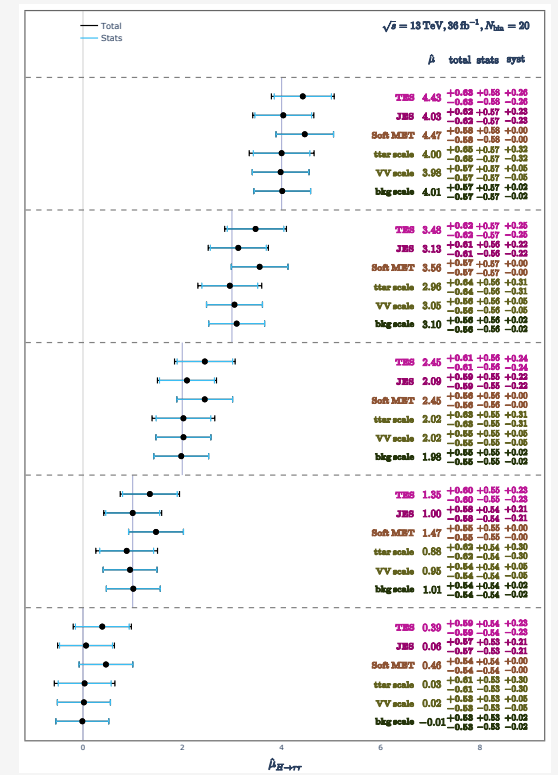
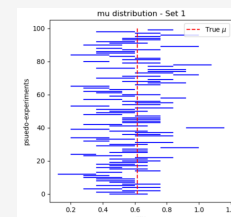
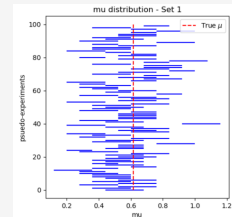
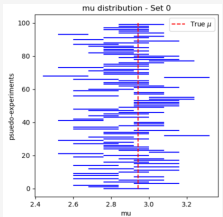
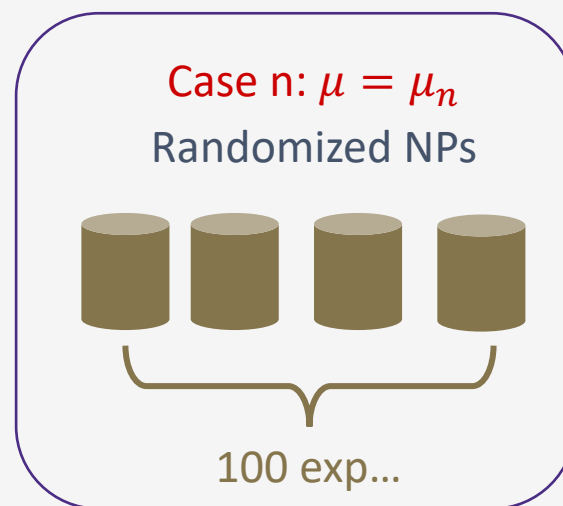
Evaluation Metrics

- Pseudo-experiments

- dataset representative of what would be measured from $10\text{fb}^{-1} \sim 800$ billion LHC pp collisions for a given value of μ and of the Nuisance Parameters



...



Evaluation Metrics

- **Interval width (ω)** averaged over N test sets
- **Coverage (c)**: fraction of time μ is contained
- Combined using a **coverage function $f(c)$** :

$$\omega = \frac{1}{N} \sum_{i=0}^N |\mu_{84,i} - \mu_{16,i}|$$

$$c = \frac{1}{N_{test}} \sum_{i=1}^N 1 \text{ if } \mu_{true,i} \in [\mu_{16,i} - \mu_{84,i}]$$

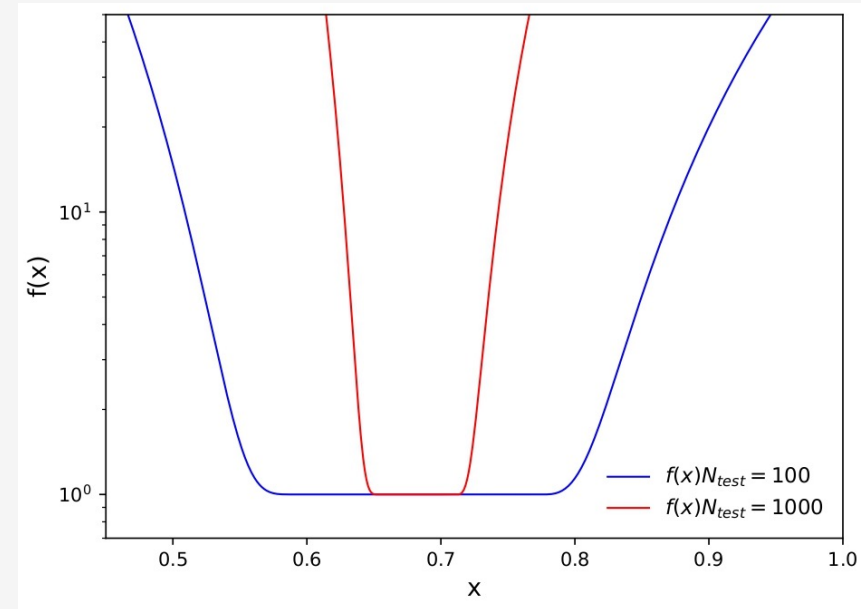
$$c \in [0.6827 - 2\sigma_{68}, 0.6827 + 2\sigma_{68}] : f(c) = 1$$

$$c < 0.6827 - 2\sigma_{68} : f(c) = 1 + \left| \frac{c - (0.6827 - 2\sigma_{68})}{\sigma_{68}} \right|^4$$

$$c > 0.6827 + 2\sigma_{68} : f(c) = 1 + \left| \frac{c - (0.6827 + 2\sigma_{68})}{\sigma_{68}} \right|^3$$

$$\sigma_{68} = \sqrt{\frac{(1 - 0.6827)0.6827}{N_{test}}}$$

- N dependance for equivalent ideal coverage
- Penalizes undercoverage more
- **Final score (s)** designed to avoid large values or gaming: $-\ln((\omega + \epsilon)f(c))$ ($\epsilon = 10^{-2}$)
- More details in [White paper](#)

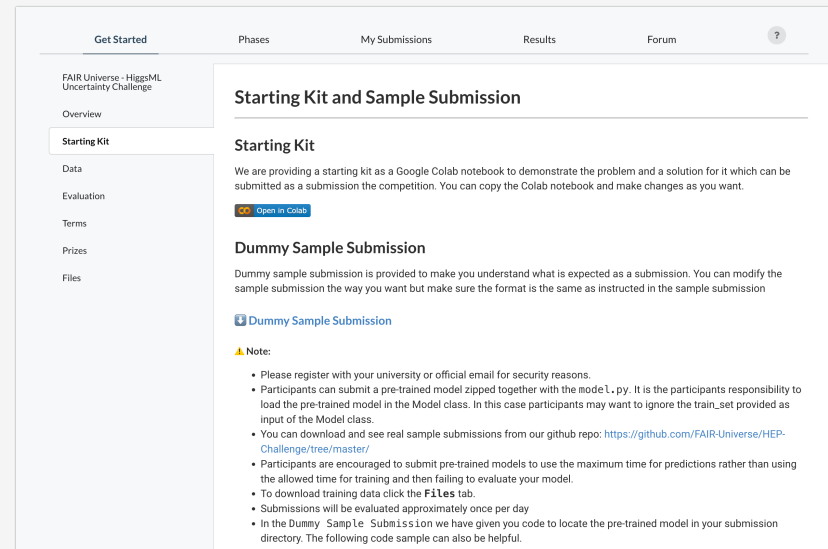
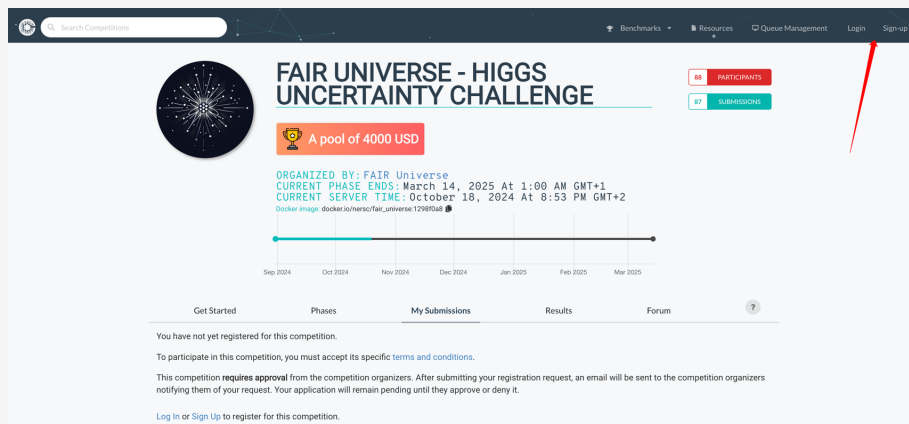


Enter the HiggsML Uncertainty Challenge!

- **Competition page:** <https://www.codabench.org/competitions/2977/>
- **Current Phase ends:** March 14, 2025 @ 0:00 GMT+0
- **Documentations:** <https://fair-universe.lbl.gov/docs/>
- **White paper:** this serves as a full breakdown of the competition in detail [[Arxiv:2410.02867](https://arxiv.org/abs/2410.02867)]
- Please apply to the competition with your institute's email address.

Total pool of 4000 USD

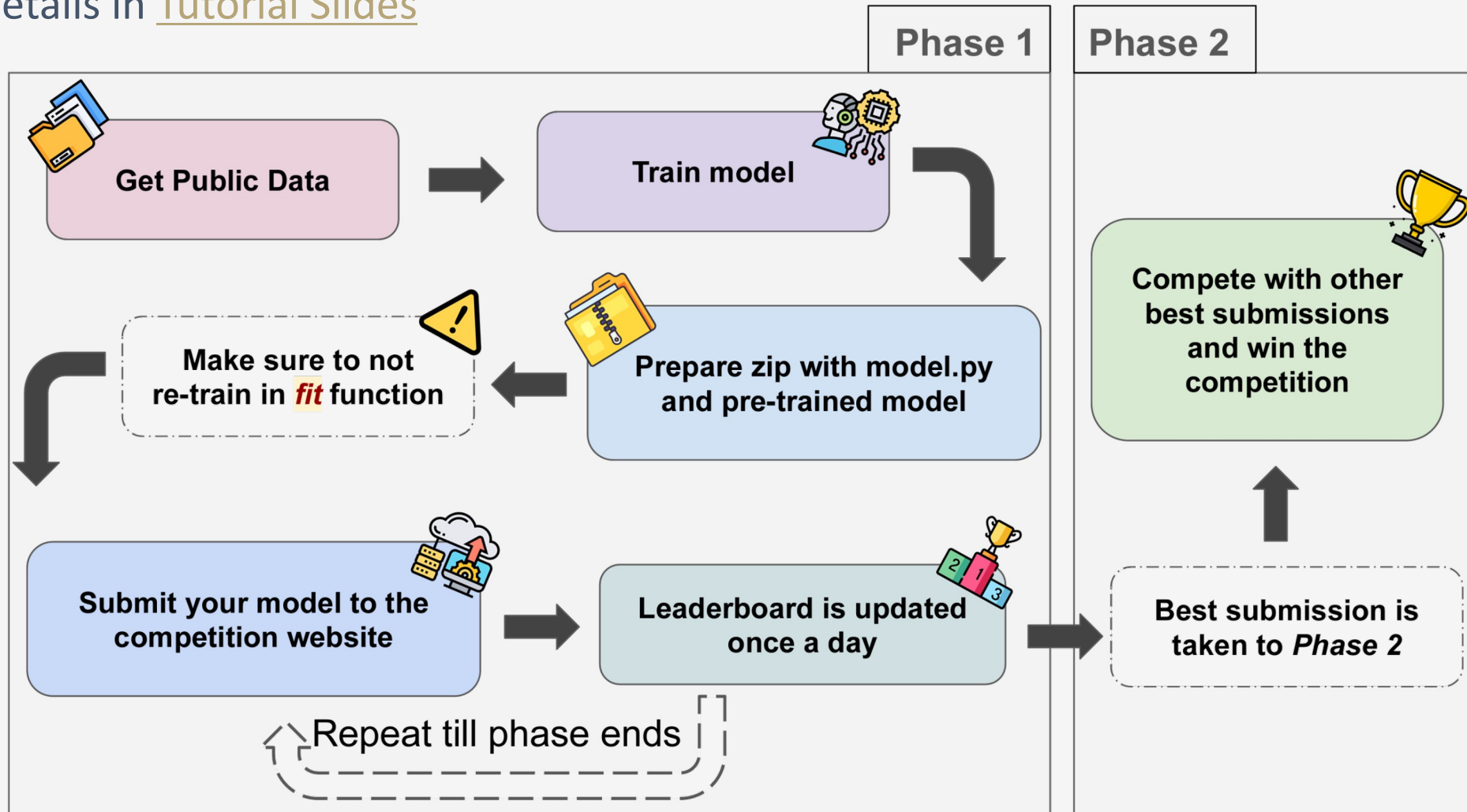
- 1 First Place: \$2000**
- 2 Second Place: \$1500**
- 3 Third Place: \$500**



FAIR Universe

Competition Flow

- More details in [Tutorial Slides](#)



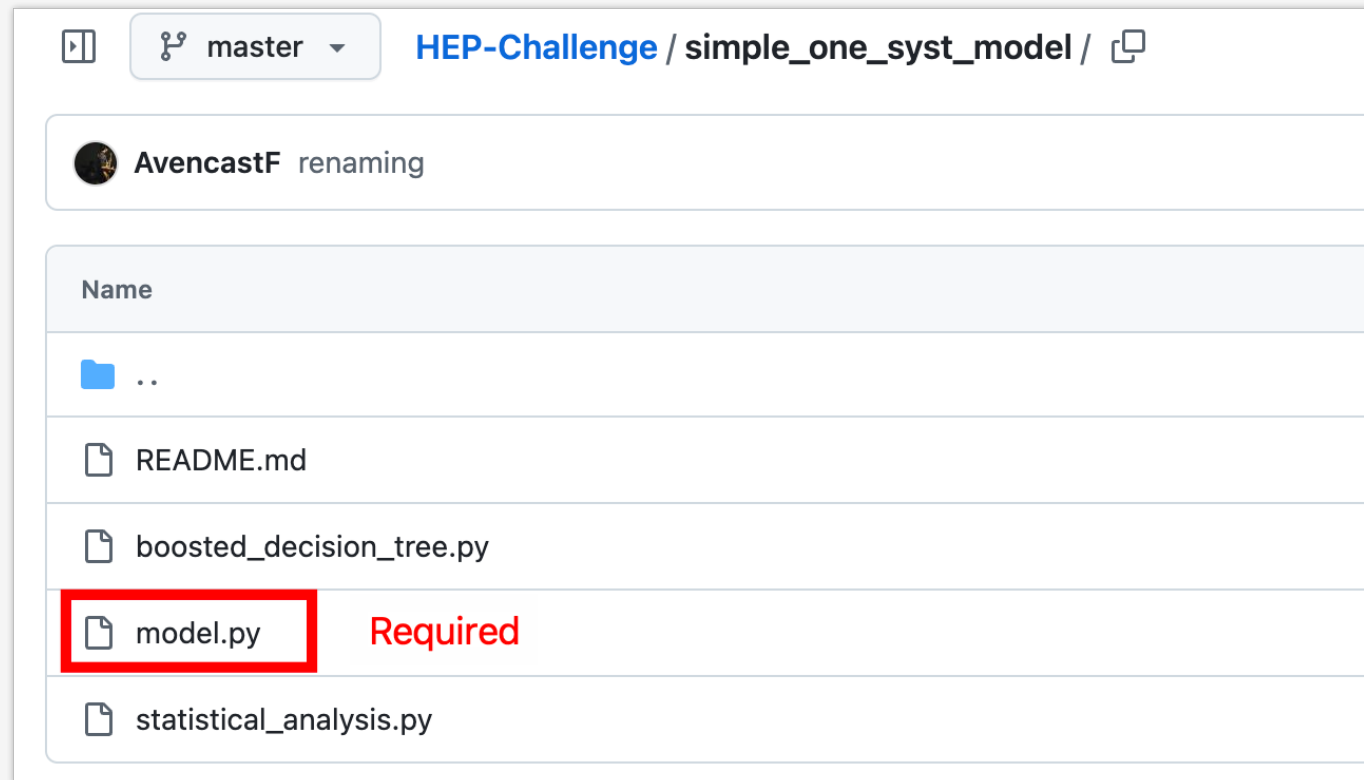
Hands-On

- Login <https://www.codabench.org/competitions/2977/#/participate-tab>

The screenshot shows the Codabench website interface for a competition. At the top, there is a dark navigation bar with a search bar on the left and links for 'Benchmarks', 'Resources', 'Queue Management', 'Login' (highlighted with a red box), and 'Sign-up' on the right. The main content area features a large circular graphic of a starburst on the left. To its right, the competition title 'FAIR UNIVERSE - HIGGS UNCERTAINTY CHALLENGE' is displayed in large, bold, black letters. Below the title, there is a red button with a trophy icon and the text 'A pool of 4000 USD'. Further down, the text 'ORGANIZED BY: FAIR Universe' is followed by 'CURRENT PHASE ENDS: March 14, 2025 At 1:00 AM GMT+1' and 'CURRENT SERVER TIME: November 21, 2024 At 4:57 PM GMT+1'. A 'Docker image' link is also present. A horizontal timeline at the bottom of the main content area shows the competition's duration from September 2024 to March 2025, with a teal bar indicating the current phase. At the very bottom, a navigation menu includes 'Get Started', 'Phases', 'My Submissions', 'Results', 'Forum', and a help icon.

Hands-On

- Example model: https://github.com/FAIR-Universe/HEP-Challenge/tree/master/simple_one_syst_model
- Required script: `model.py`



The screenshot shows a GitHub repository interface for the path `HEP-Challenge / simple_one_syst_model /`. The current branch is `master`. A commit by `AvencastF` is shown with the message `renaming`. The file list includes:

Name
..
README.md
boosted_decision_tree.py
model.py Required
statistical_analysis.py

Hands-On

- Required script: `model.py`
 - `__init__(self)`: -> None
 - `fit(self)`: -> None (To train model)
 - `predict(self, test_set)`: -> dict (predict $\hat{\mu}$ with uncertainties)

```
def predict(self, test_set):
    """
    Predicts the values for the test set.

    Args:
        * test_set (dict): A dictionary containing the data and weights

    Returns:
        dict: A dictionary with the following keys:
            - 'mu_hat': The predicted value of mu.
            - 'delta_mu_hat': The uncertainty in the predicted value of mu.
            - 'p16': The lower bound of the 16th percentile of mu.
            - 'p84': The upper bound of the 84th percentile of mu.
    """

    test_data = test_set["data"]
    test_weights = test_set["weights"]

    print("[*] -> test weights sum = ", test_weights.sum())

    predictions = self.model.predict(test_data)

    result = self.stat_analysis.compute_mu(
        predictions,
        test_weights,
    )

    print("Test Results: ", result)

    return result
```

Hands-On

- Prepare for submission: zip the model

My_awesome_model/

- model.py
- necessary_scripts.py



My_awesome_model.zip

Hands-On

- Submit your model

Get Started Phases **My Submissions** Results Forum ?

Public Phase **Step 1. Click My Submission**

?	Number of submissions used for the day	Number of total submissions used
	0 out of 5	12 out of 100

Submission upload

Metadata or Fact Sheet

Method Name: *

Submit as: ?

Yourself

Step 3. Upload your model



Hands-On

Summary



- We've built a **flexible platform** for hosting challenges and benchmarks, extending Codabench, and powered by **HPC at NERSC**.
- Launching a series of challenges focused on **uncertainty-aware methods for High-Energy Physics (HEP)**.
- A NeurIPS competition running from September 2024 to March 2025
- **Enter the HiggsML Uncertainty Challenge** now: <https://www.codabench.org/competitions/2977/>
- **We welcome feedback** on the challenge to keep it engaging and beneficial for advanced methods.
- **Get involved and stay updated:**
 - Help and feedback: Join the **#higgsml-uncertainty-challenge** channel on the [Fair Universe Slack workspace](#).
 - Ongoing updates: Subscribe to the **Fair-Universe-Announcements Google Group**.
 - Questions or collaborations: Contact **fair-universe@lbl.gov**.



Backup

NERSC-AI Ecosystem

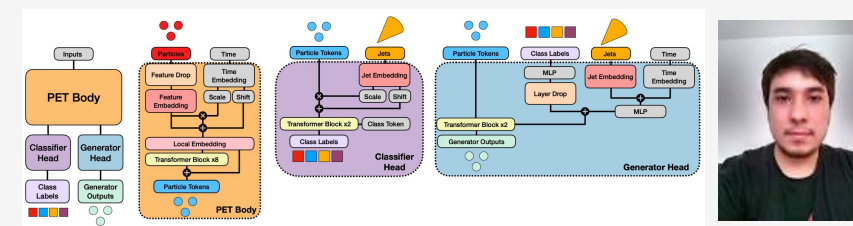
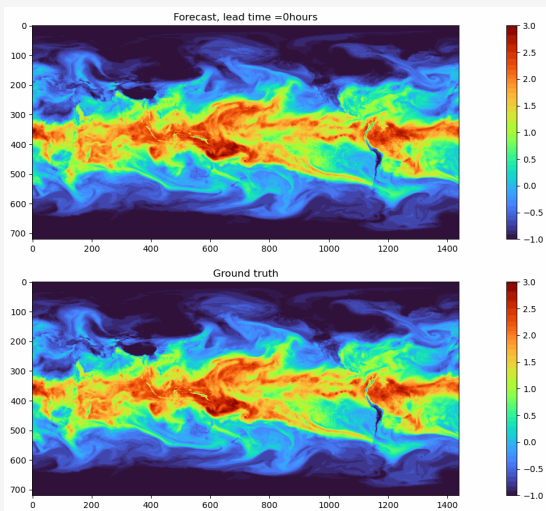
- **Deploy** optimized hardware and software (working with vendors)
 - Improve performance, e.g through benchmarking ([MLPerf HPC](#))
- **Apply** cutting edge AI for science: e.g., “NESAP” program with postdocs
- **Empower** through e.g., over 20 DL@Scale tutorials, 1000s of total participants: ([SC23](#))
- Many AI for science highlights not covered here.



FourCastNet- [Pathak et al. 2022](#)

Collab with Nvidia, Caltech

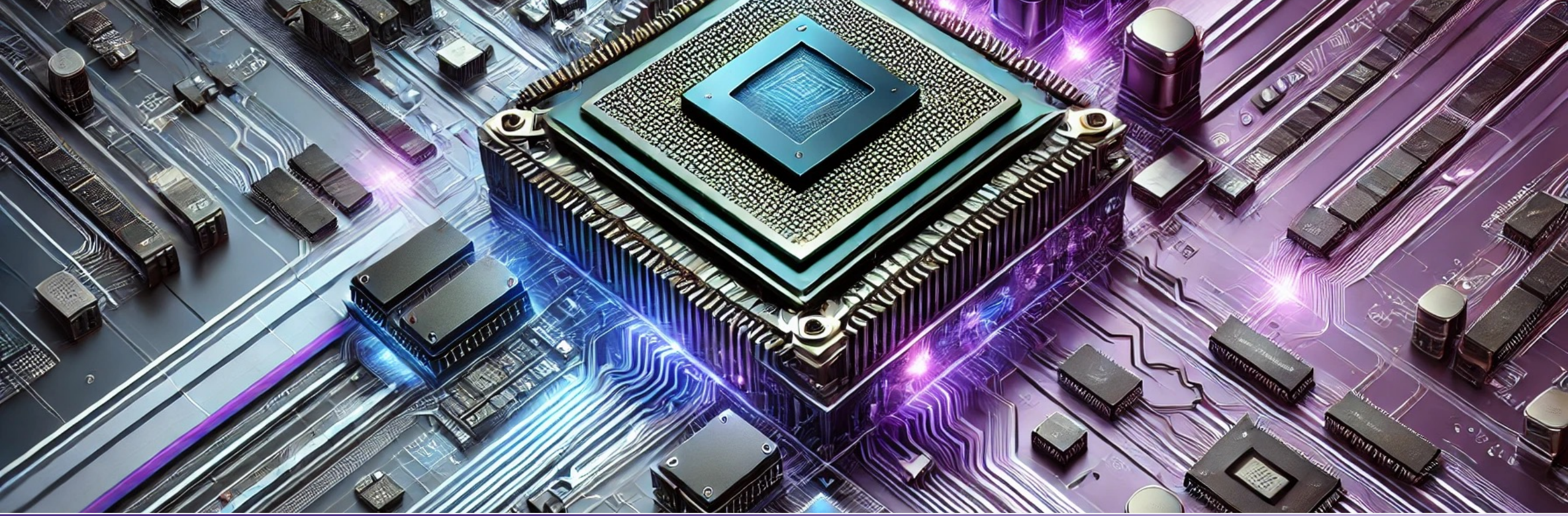
- First DL model with skill of numerical weather prediction (NWP)
- Train up to 1000s of GPUs
- Forecasts 1000s times faster than NWP



OmniLearn

H1 Collaboration ([Mikuni et. al.](#)):

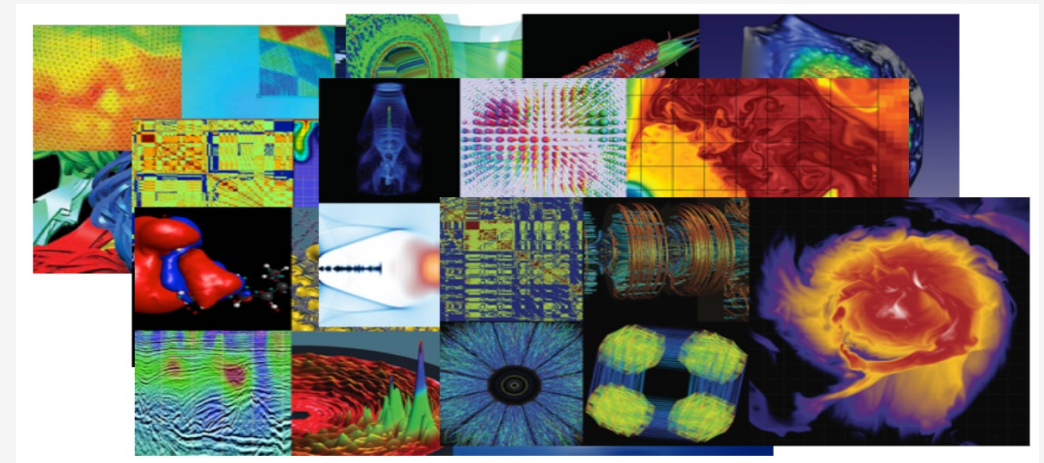
jet physics analyses by enhancing accuracy, precision, and speed across multiple tasks using a **unified machine learning model**.



Large-compute-scale AI Ecosystem for hosting challenges and benchmarks

NERSC: Mission HPC for the Dept. of Energy Office of Science

- Large compute systems
 - Perlmutter: ~7k A100 GPUs, Also high-capacity / fast filesystems, 1 Tbit/s WAN and flexible services
 - SPIN: Rancher/K8s platform for user-defined services
- Broad science user base
 - > 10,000 users
 - > 1,000 projects
 - Across all DoE Science e.g. HEP; NP; Climate; Fusion Chemistry; Materials; Genomics; etc ...



Condabench/FAIR Universe Platform

Based on <https://www.codabench.org/>

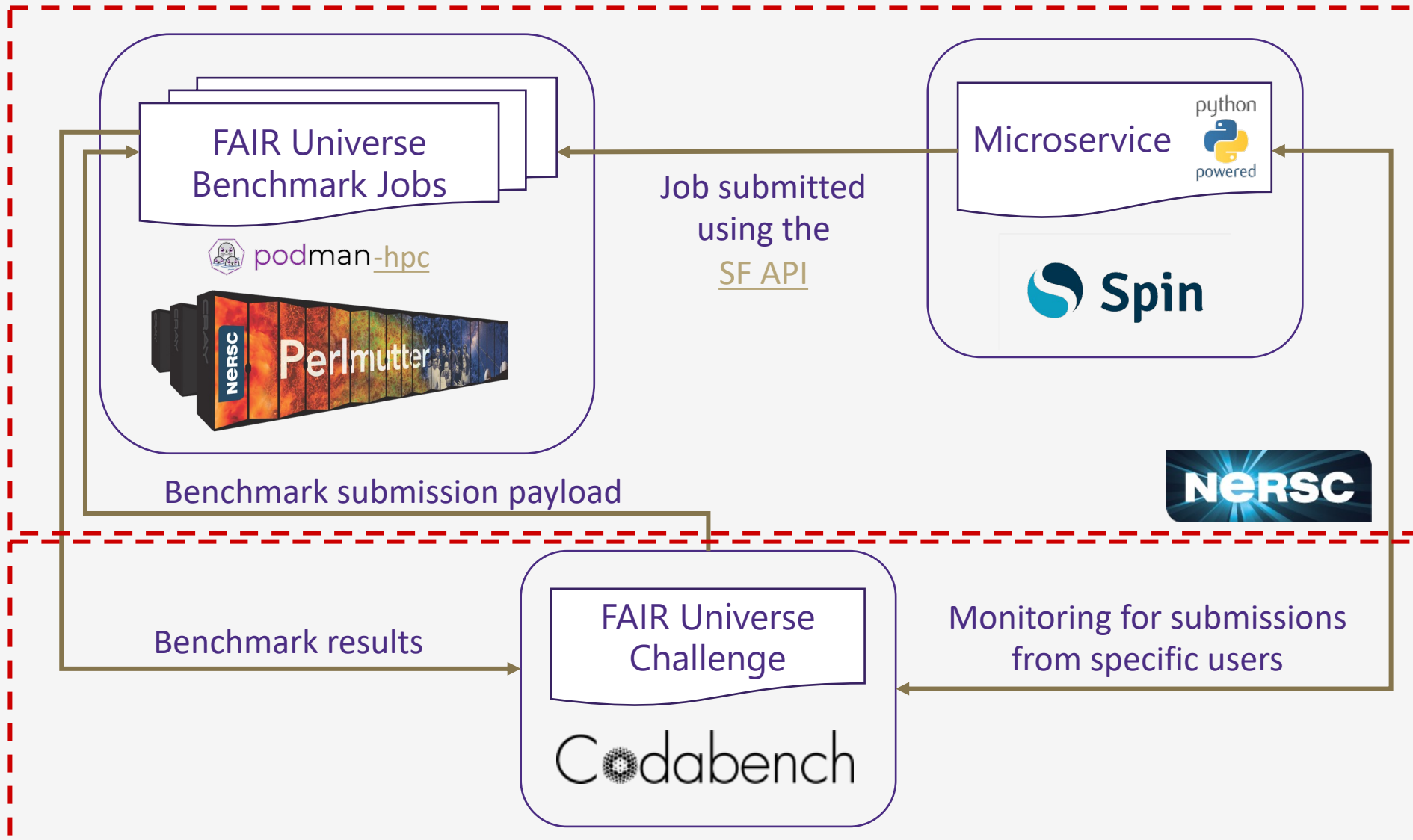
- Codabench - open-source platform for AI benchmarks and challenges
 - Originally (CodaLab) Microsoft/Stanford now a Paris-Saclay/[LISN](#) led community
 - > **500 challenges** since 2013
 - Allows code submission as well as results e.g., for evaluation timing or reproducibility
 - Also, data-centric AI “inverted competitions”
 - Organizers can define scoring functions
 - Queues for evaluation can run on **diverse compute resources**
 - Platform itself can be deployed on **different compute resources**

The screenshot shows the Codabench website interface. At the top, there's a navigation bar with the Codabench logo and logos for LISN, Université Paris-Saclay, and CodaLab. Below the navigation bar is an announcement box with a welcome message and a link to a getting started guide. The main content area is divided into two columns: 'Popular Benchmarks' and 'Featured Benchmarks'. Each benchmark card includes a title, a date, the number of participants, and a brief description. For example, the 'SenEval-2024, Task 8: Multigenerator, Multidomain, and Multilingual Black-Box Machine-Generated Text Detection' benchmark has 359 participants and is organized by Altagwan. The 'Style Trans-Fair Competition' has 11 participants and is organized by ayoubhammal.

“Fair Universe” brings Codabench to HPC at NERSC!

The screenshot shows the 'Get Started' section of the Codabench website. It features three main icons: a trophy for 'Participate', a group of people for 'Organize', and a code symbol for 'Contribute'. Below each icon is a brief description of the action. The 'Participate' section encourages finding benchmarks of interest. The 'Organize' section explains how to create a benchmark. The 'Contribute' section invites users to join the development team. At the bottom, there are three large statistics: 194 PUBLIC COMPETITIONS, 10678 USERS, and 59161 SUBMISSIONS.

FAIR Universe Platform: Codabench/NERSC Integration



FAIR Universe Platform: Codabench/NERSC Integration

- Benchmark submissions pulled to workers running on Perlmutter:
 - Use `podman(-hpc)` container runtime: secure and scalable
 - Enable parallelism/scale for
 - **Intensive methods** - use multiple A100 GPUs for training or evaluation
 - **Many participants** - through running many parallel workers
 - **Many evaluations** - e.g for Uncertainty Quantification
- Workers submitted as needed by microservice on SPIN service platform
 - NERSC's "[SF API](#)" for job submission
 - Monitor/filter submissions
- Also deploy instances of Codabench platform itself within SPIN
 - Customization and future OIDC integration with NERSC authorization

