FAIR Universe

U.S. DEPARTMENT OF ENERGY
Office of Science

ATLAS EXPERIMENT

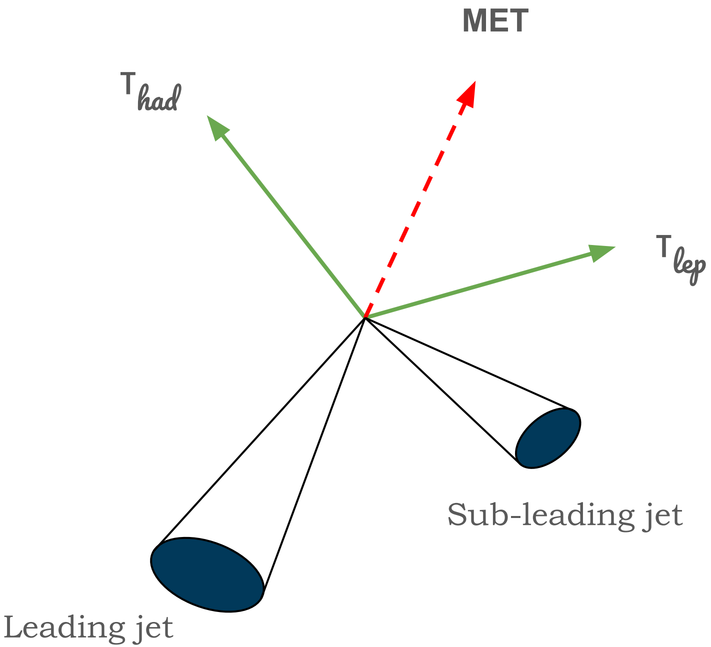
FAIR Universe : HiggsML Uncertainty Challenge

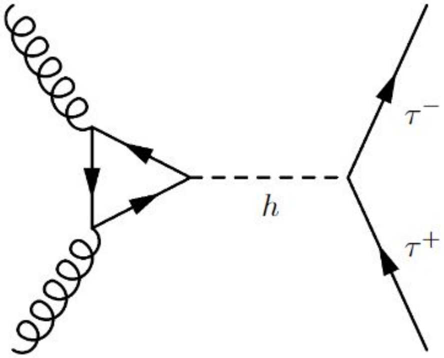
*Wahid Bhimji, Paolo Calafiura, **<u>Ragansu Chakkappai</u>**, Yuan-Tang Chou, Sascha Diefenbacher, Steven Farrell, Aishik Ghosh, Isabelle Guyon, Chris Harris, Shih-Chieh Hsu, Elham Khoda, Benjamin Nachman, Peter Nugent, Mathis Reymond, David Rousseau, Ihsan Ullah, Daniel Whiteson,*

université PARIS-SACLAY

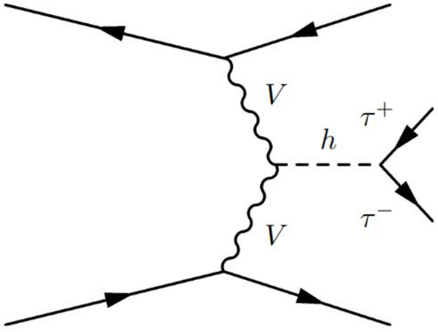
IJCLab Irène Joliot-Curie

Introduction

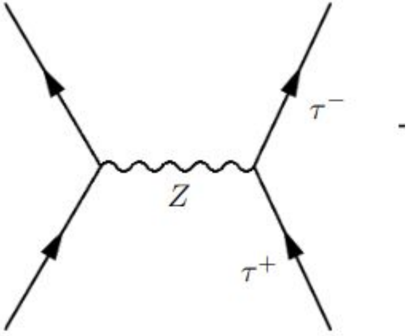




(a) ggF Production

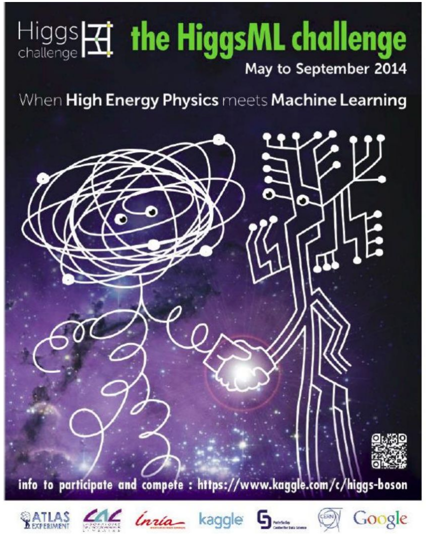


(b) VBF Production



(a) $Z \rightarrow ll$ Background

Introduction

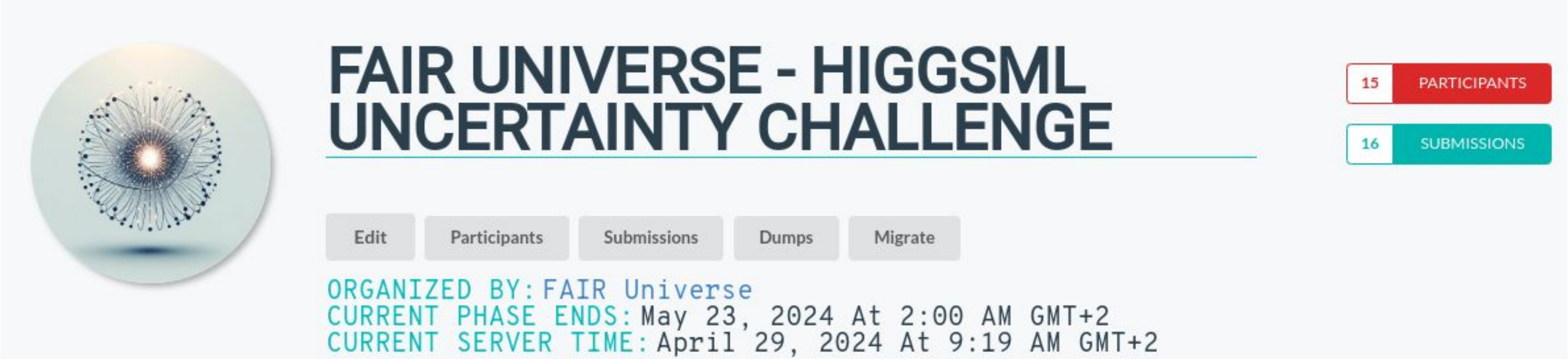


- $H \rightarrow \tau + \tau -$ is an important channel to look for standard model discrepancy
- HiggsML challenge 2014
- Aim : Improve cross-section significance in higgs to tau tau data

---

- Signal processes : $H \rightarrow \tau + \tau -$
- Background processes:
    - Z $\rightarrow \tau + \tau -$,
    - ttbar
    - W + jets
    - Diboson, top etc.

Fair Universe: HiggsML Uncertainty Challenge



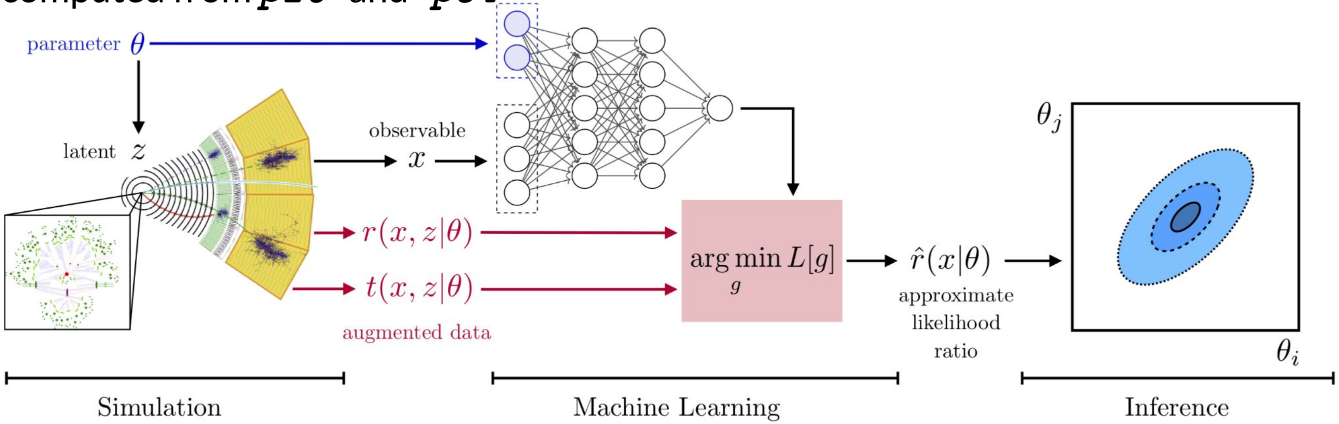
- Extension of HiggsML Kaggle challenge (H→$\tau\tau$) organised by David Rousseau and Isabelle Guyon and others in 2014
- HiggsML Uncertainty Challenge Pilot - launched on March 11 2024
- Full HiggsML Uncertainty Challenge planned for June - Oct 2024
    - Submitted as NeurIPS competition - results to be presented at NeurIPS in December

Background on Fair Universe Project

- 3 year US Dept. of Energy, AI for HEP project. Aims to:
  - Provide an open, **large-compute-scale AI ecosystem** for sharing datasets, training large models, fine-tuning those models, and **hosting challenges and benchmarks.**
  - **Organize a challenge series**, progressively rolling in tasks of increasing difficulty, based on novel datasets.
  - Tasks will focus on **measuring and minimizing the effects of systematic uncertainties** in HEP (particle physics and cosmology).
- This funding went to LBL, NERSC, U Washington, and Chalearn (Isabelle Guyon's Non-Profit US Organisation).
  - Chalearn paid CNRS for my PhD

Challenge Objective

- Train a AI model to improve cross section significance
- The model will be tested with datasets with unknown systematics and signal strength μ. (μ=1 if Standard Model)
- For each test set participants must predict :
  - ***mu_hat***
  - ***p16 = mu - sigma_mu***
  - ***p84 = mu + sigma_mu***
- Score is computed from ***p16*** and ***p84***



Challenge Datasets

- The data is generated with Pythia (event generation) and Delphes (Detector simulation).
- Using the updated Delphes ATLAS card
- Goal is to have a very large public dataset not necessarily accurate.
- Less Accurate than Madgraph/Sherpa + Geant4, but much faster
- Generates 1000 events/min
- Aims to generate ~150 Million Events after initial cuts
- Data generated using NERSC supercomputer.

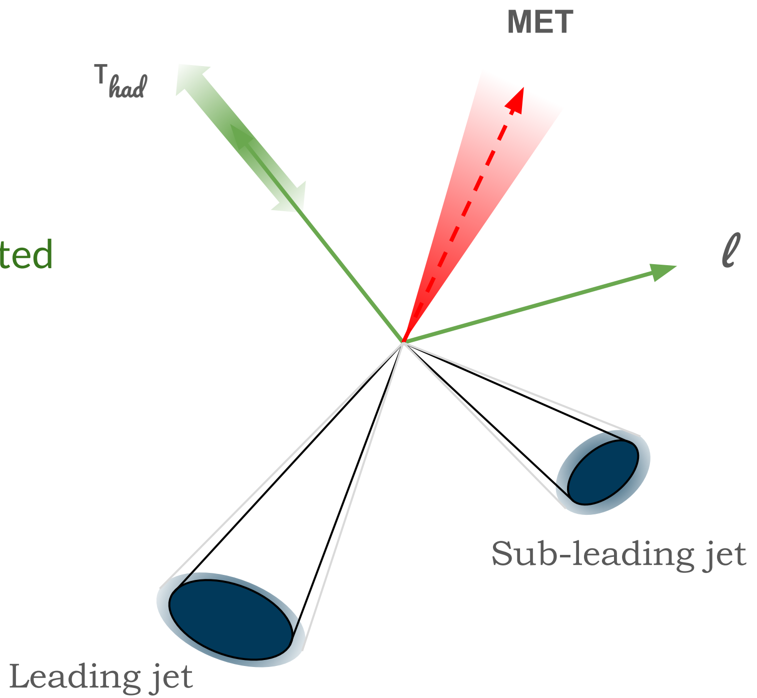
Process	Number Generated	LHC Events	Label
Higgs	16214520	9220	signal
Z Boson	14135841	2569787	background
W Boson	287514800	2964267	background
$t\bar{t}$	31921500	320318	background

DELPHES
fast simulation

Challenge Datasets - Systematics

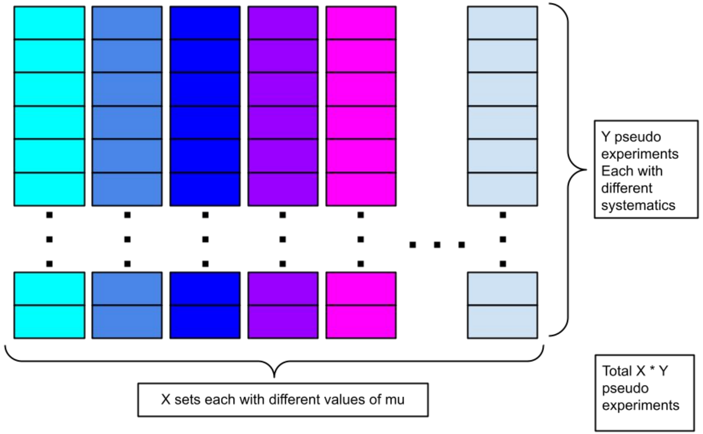
Apply parameterized systematics (Nuisance Parameters) :

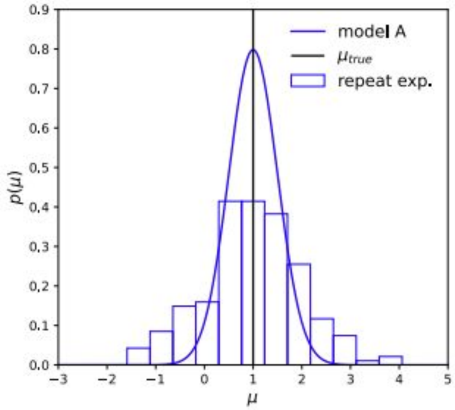
- Tau Energy Scale - shift hadronic Tau Pt (and correlated MET)
- Jet Energy Scale (and correlated MET impact)
- Additional randomised Soft MET
- Background normalisation
- W-boson background normalisation

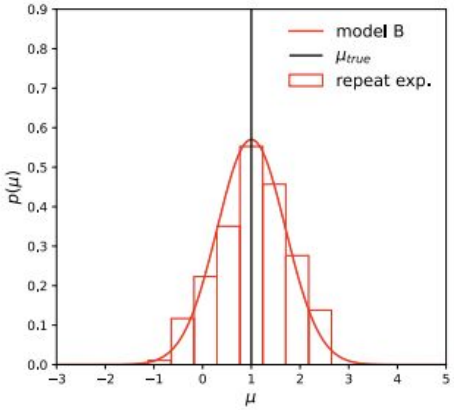


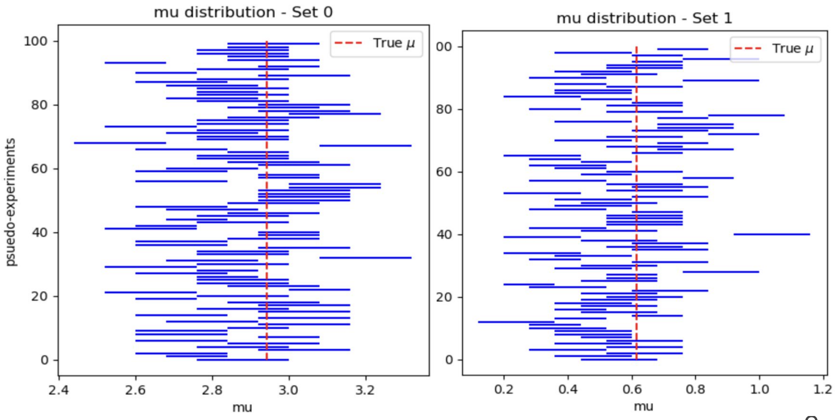
Evaluation

- Form multiple pseudo-experiment test sets: different signal strengths (μ) and systematics
  - Current pilot - 4μ and 50 pseudo-experiments
- Task: predict uncertainty interval $[\mu_{16}, \mu_{84}]$
  - E.g. 68% quantile of likelihood or assume 1σ





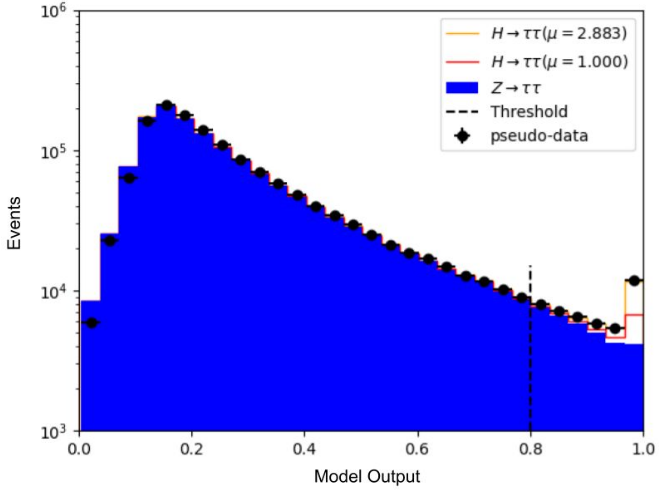




Basic Algorithm

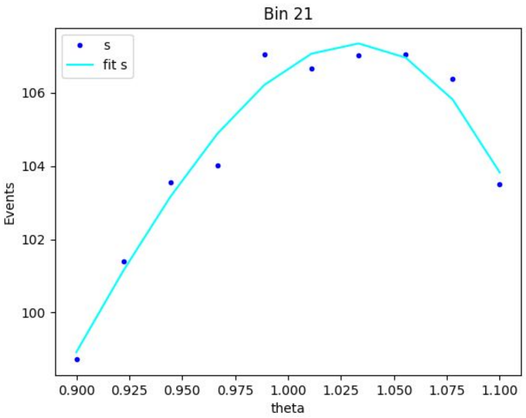
1. Divide data into ***train_set*** and ***holdout_set***
2. Use ***train_set*** to Train the simple dense NN
3. Define S_i and B_i : predicted score bin content
4. Construct for S_i (α) and B_i(α) functions from ***holdout_set***
5. Combine define Binned Negative Log Likelihood function as function of NPs and μ
6. For Each pseudo experiment

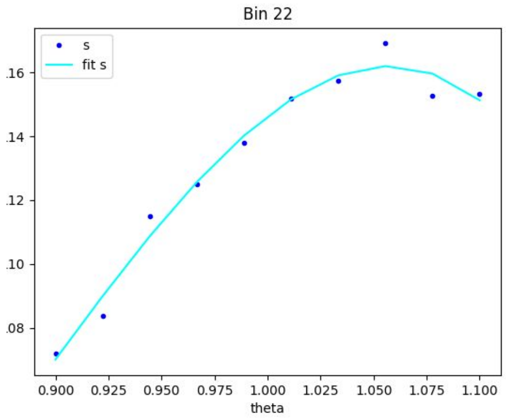
    a. Predict score for pseudo experiment
    b. Use Minuit to find value of mu, sigma_mu and NP
    c. Returns
        - ***mu_hat***
        - ***p16 = mu - sigma_mu***
        - ***p84 = mu + sigma_mu***

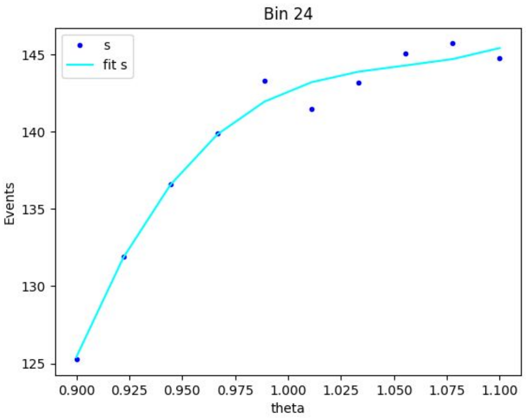


Parameterisation of S(α)

With the help of the *holdout_set* for we get values of S and B for each NP in each bin. A polynomial function is used to fit them. This function is later used in the NLL formalism







Profile μ and α simultaneously

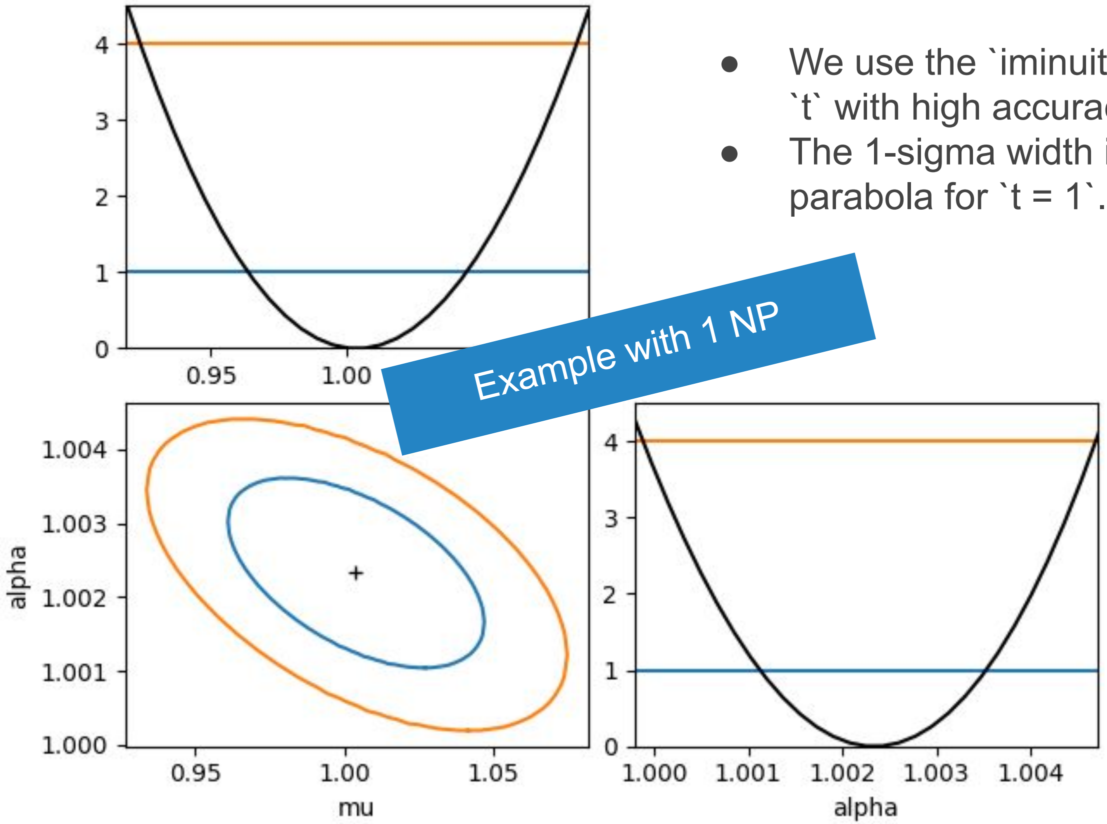
$$L(\mu, \vec{\alpha}|\mathcal{D}) = \prod_{i=1}^{N_{\text{bins}}} \frac{(\mu S_i(\vec{\alpha}) + B_i(\vec{\alpha}))^{n_i} e^{-(\mu S_i(\vec{\alpha})+B_i(\vec{\alpha}))}}{n_i!}$$

$$\Rightarrow t_{\mu,\vec{\alpha}} = -2\log\left(L(\mu, \vec{\alpha} \mid \mathcal{D})\right)$$

$$= -2\sum_{i}^{N_{\text{bins}}} n_i \log(\mu S_i(\vec{\alpha}) + B_i(\vec{\alpha})) + (\mu S_i(\vec{\alpha}) + B_i(\vec{\alpha}))$$

L here is the likelihood estimator which depends on μ and α, where α is a vector of 5 NP thus the μ at which L is maximum or *t* is minimum is the predicted μ,

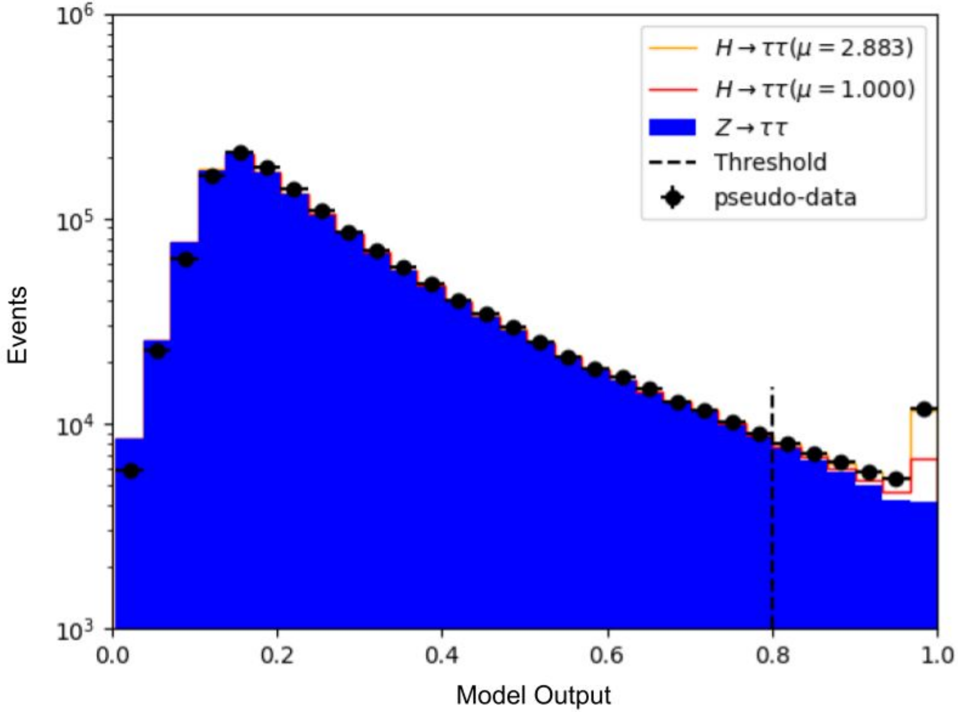
NLL curve and contour



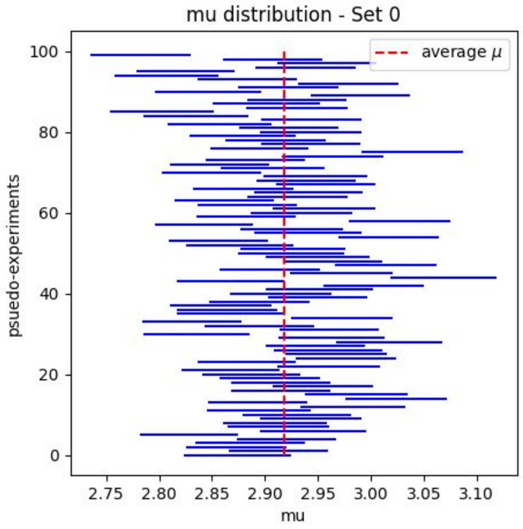
- We use the `iminuit` package to find the minimum of `t` with high accuracy.
- The 1-sigma width is the width between points on the parabola for `t = 1`.

Example with 1 NP

Results



$H \to \tau\tau(\mu = 2.883)$
$H \to \tau\tau(\mu = 1.000)$
$Z \to \tau\tau$
Threshold
pseudo-data
Events
Model Output



mu distribution - Set 0
average μ
psuedo-experiments
mu

Conclusion

- AI challenge which addresses Systematic Uncertainty in HEP problem.
- Large Data Set with ~150M Events (signal + background)
- New Scoring to take Coverage and Confidence interval into account.
- Taylor made ingestion algorithm to test multiple pseudo-experiments in parallel.
- Large Computing Infrastructure as back_end
- You can enter the HiggsML Uncertainty Challenge Pilot now!
  - http://go.lbl.gov/fair-universe-higgsml-spring24
  - https://www.codabench.org/competitions/2164/

Help and feedback: #higgsml-uncertainty-challenge-spring-24 channel on the Fair Universe Slack workspace
Ongoing information Google Group: Fair-Universe-Announcements
Collaborations, questions, comments: fair-universe@lbl.gov

Thank you for your attention!

Back-up

Uncertainty Quantification Metric

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

$x \geq 0.68 - 2\sigma_{68}$ and $x \leq 0.68 + 2\sigma_{68}$: $1.$

$x < 0.68 - 2\sigma_{68}$: $1 + \left|\frac{x-(0.68-2\sigma_{68})}{\sigma_{68}}\right|^4$

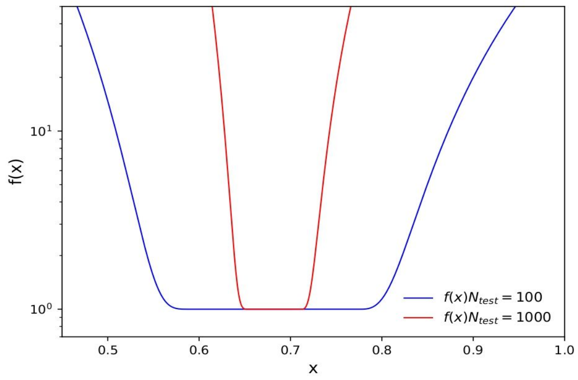
$x > 0.68 + 2\sigma_{68}$: $1 + \left|\frac{x-(0.68+2\sigma_{68})}{\sigma_{68}}\right|^3$

with $\sigma_{68} = \frac{\sqrt{(1-0.68)0.68N}}{N}$

- N dependance for equivalent ideal coverage
- Penalizes undercoverage more
- Final score (s) designed to avoid large values or gaming

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

$$c = \frac{1}{N}\sum_{i=0}^{N} 1 \text{ if}(\mu_{true,i} \in [\mu_{84,i} - \mu_{16,i}])$$

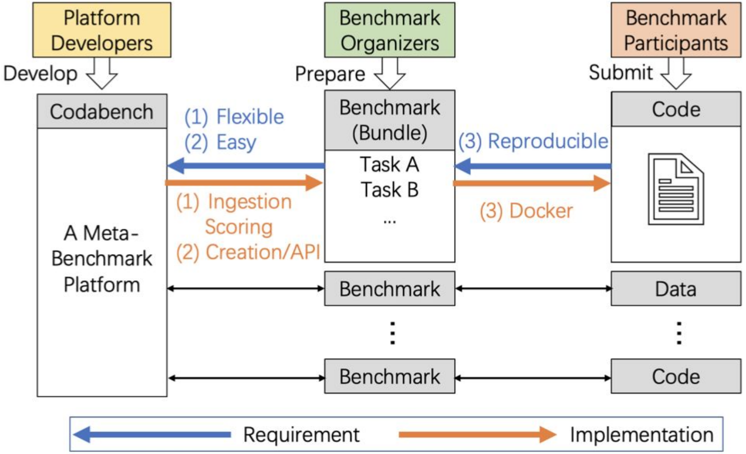


$$s = -\ln\left((w+\epsilon)f(c)\right)$$

See also Sascha Diefenbacher's AISSAI Workshop presentation

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

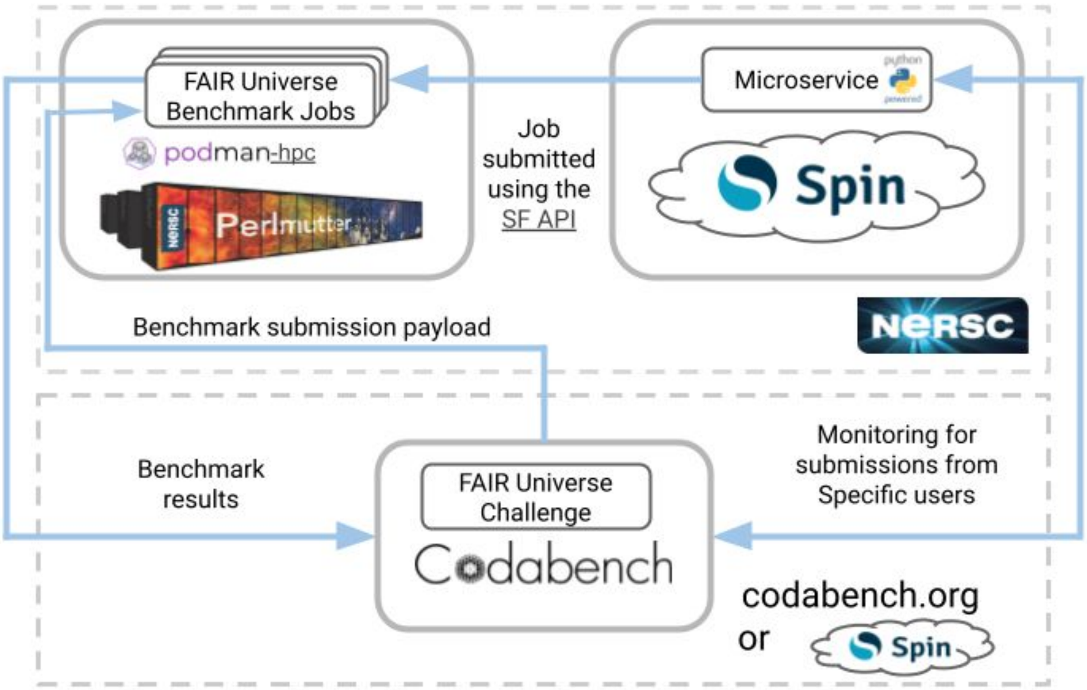
Codabench Platform



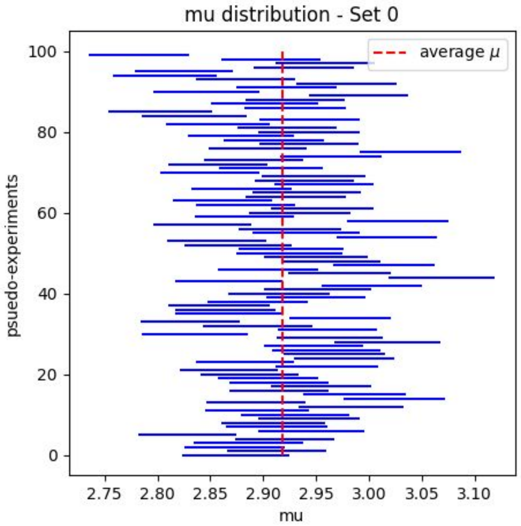
Codabench - open source platform for AI benchmarks and challenges

- Originally (CodaLab) Microsoft/Stanford now a Paris-Saclay/LISN led community
- > 600 challenges since 2013
- Completely open-ended competition design.
- Allows code submission as well as results e.g. for evaluation timing or reproducibility
- Also data-centric AI “inverted competitions”
- Queues for evaluation can run on diverse compute resources
- Platform itself can be deployed on different compute resources
- Ranked best challenge platform for ML by ML contests

Fair Universe Platform: Codabench-NERSC integration



Signal Strength and Coverage - Pytorch



====================

Overall Score

====================

[*] --- RMSE: 0.054

[*] --- MAE: 0.098

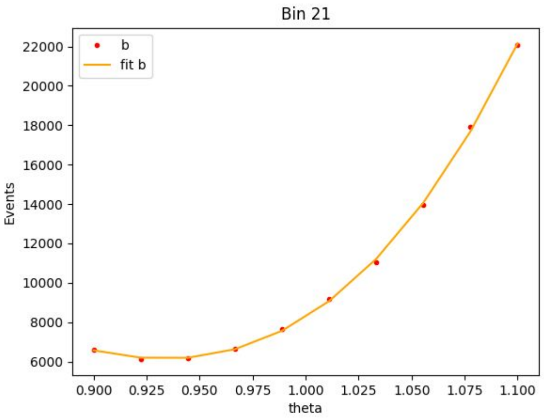
[*] --- Interval: 0.084

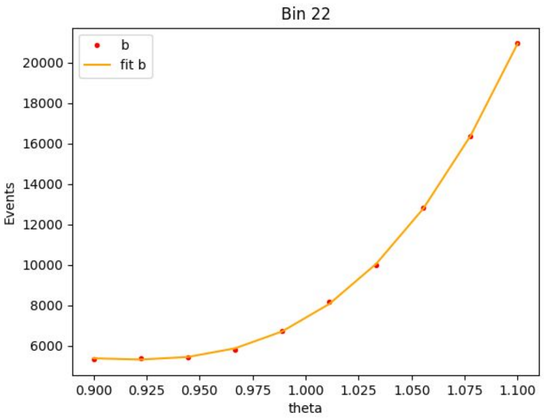
[*] --- Coverage: 0.46

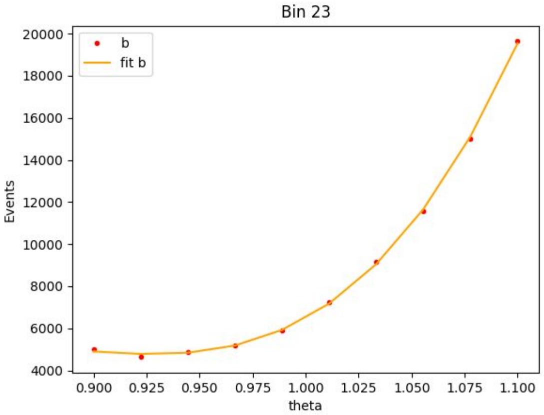
[*] --- Quantiles score: -1.651

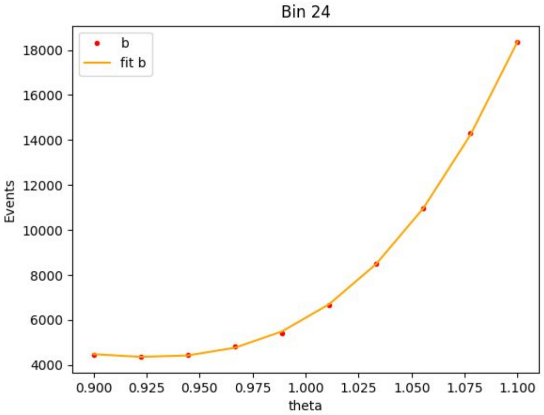
Coverage plot for NN pytorch (syst) [100 PX]

Parameterisation of B(alpha)









NN with L2 regularization using PyTorch

- PyTorch NN Classifier is Trained to distinguish Signal (Higgs) from Background (Z)
- 32 features,
- Architecture
  - 4 Hidden layers with 200 nodes
  - 1 Output node
  - Sigmoid Activation between layers
  - L2 Regularization during training
- Model return score between 0 (background) and 1(signal),