BERKELEY LAB
Bringing Science Solutions to the World

U.S. DEPARTMENT OF ENERGY
Office of Science

Generator Based Inference (GBI)

Chi Lung Cheng, Ranit Das, Runze Li, Radha Mastandrea, Vinicius Mikuni, Benjamin Nachman, David Shih, Gup Singh

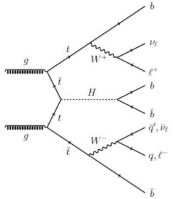
https://arxiv.org/abs/2506.00119

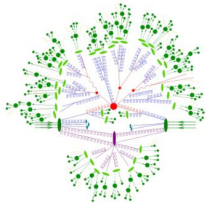
Yale

June 17, 2025
AD4HEP

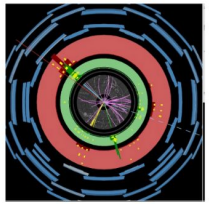
Simulation Based Inference (SBI)

Physics knowledge
(model parameters)





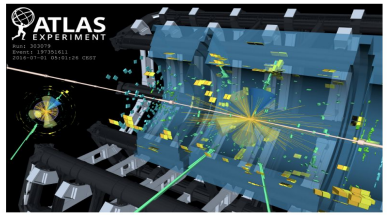
Monte Carlo Simulation

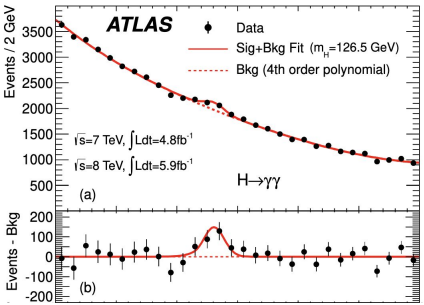


Inference
(Usually histogram based)

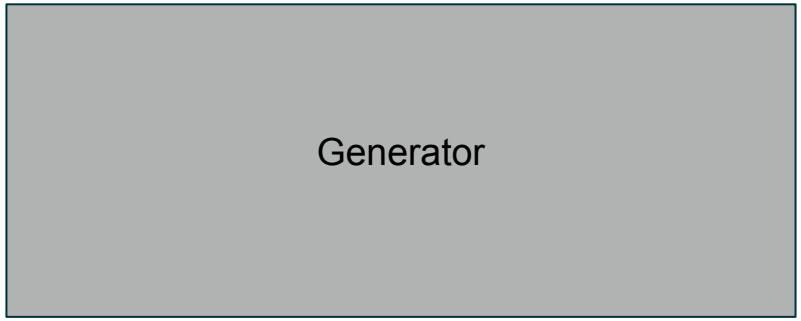
$$p(\ \boxed{}\ |\ \boxed{}\ )$$

Experiment
Results

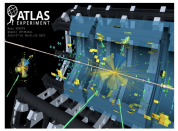




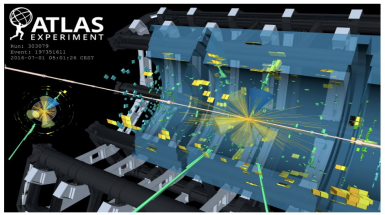
Generator Based Inference (GBI)



Inference

$p($  $|$  $)$

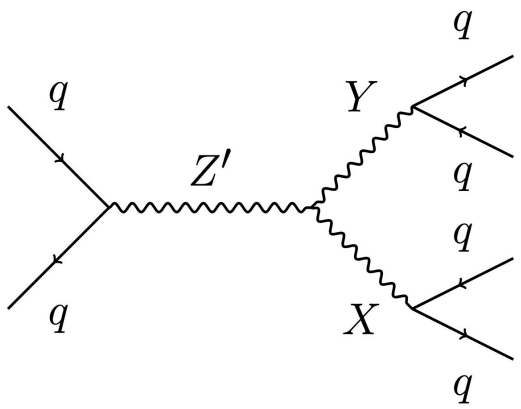
Experiment Results

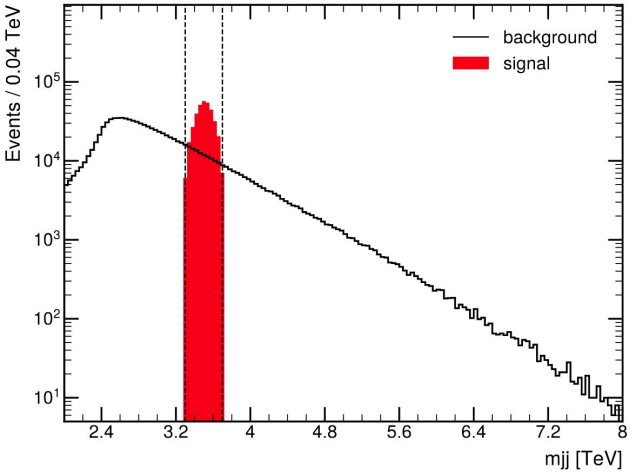


- **Apply parameter inference in Anomaly Detection (AD) style**
- SBI is a special case of GBI when MC simulation is used as the generator
- 2 generator designs will be presented here: RANODE and PAWS

LHCO Dataset

- Signal: Z′ ⇒ X + Y (Zenodo link)
- Background: QCD dijet events (Zenodo link1 link2)



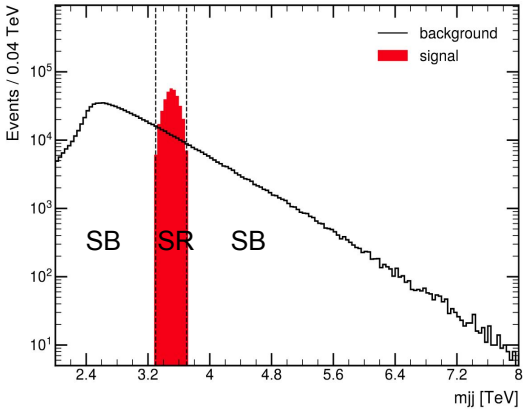


- Model parameters:
  - m_X : 500 GeV
  - m_Y : 100 GeV
  - Signal strength μ = S/(S + B)

Part I. RANODE Based GBI

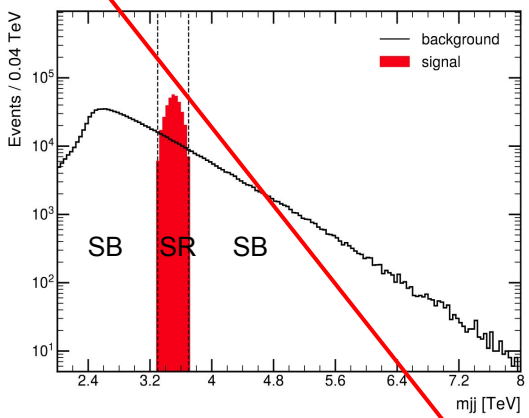
RANODE

- Train normalizing flow (NF) model p_B **on data in side band region** (SB), conditioned on m_{jj}
  - Training features are: $[m_{jmin}, \Delta_{mjj}, \tau^{j1}_{21}, \tau^{j2}_{21}]$



RANODE

- Train normalizing flow (NF) model p_B **on data in side band region** (SB), conditioned on m_{jj}
  - Training features are: $[m_{jmin}, \Delta_{mjj}, \tau^{j1}_{21}, \tau^{j2}_{21}]$

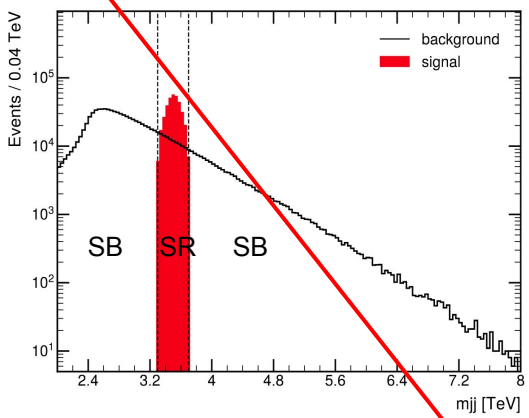


- Interpolate p_B into signal region (SR), freeze it, and train model p_S **on SR data** by maximizing the likelihood:

$$L = \sum_{x_i \in SR} \log(\mu p_S(x_i) + (1-\mu) p_B(x_i))$$

RANODE

- Train normalizing flow (NF) model p_B **on data in side band region** (SB), conditioned on m_{jj}
  - Training features are: $[m_{jmin}, \Delta_{mjj}, \tau^{j1}_{21}, \tau^{j2}_{21}]$



- Interpolate p_B into signal region (SR), freeze it, and train model p_S **on SR data** by maximizing the likelihood:

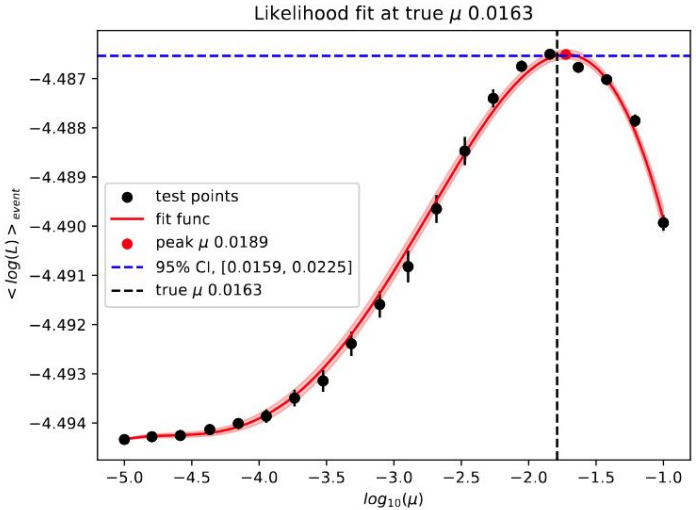
$$L = \sum_{x_i \in SR} \log(\mu p_S(x_i) + (1-\mu) p_B(x_i))$$

Inference with RANODE

- Fix μ at different test values, and train model p_S **on SR data** by maximizing the likelihood:

$$L = \sum_{x_i \in SR} \log(\mu p_S(x_i) + (1-\mu) p_B(x_i))$$

- Fit the likelihood at different test μ as a function L(μ) to find its peak with confidence interval
  - Fully data-driven

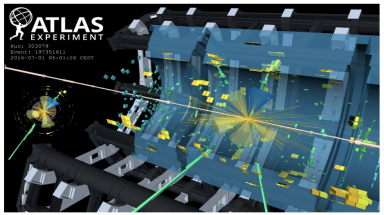


RANODE Based GBI

Data

Generator

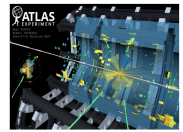
Inference



RANODE

$$L = \sum_{x_i \in SR} \log(\mu p_S(x_i) + (1-\mu) p_B(x_i))$$

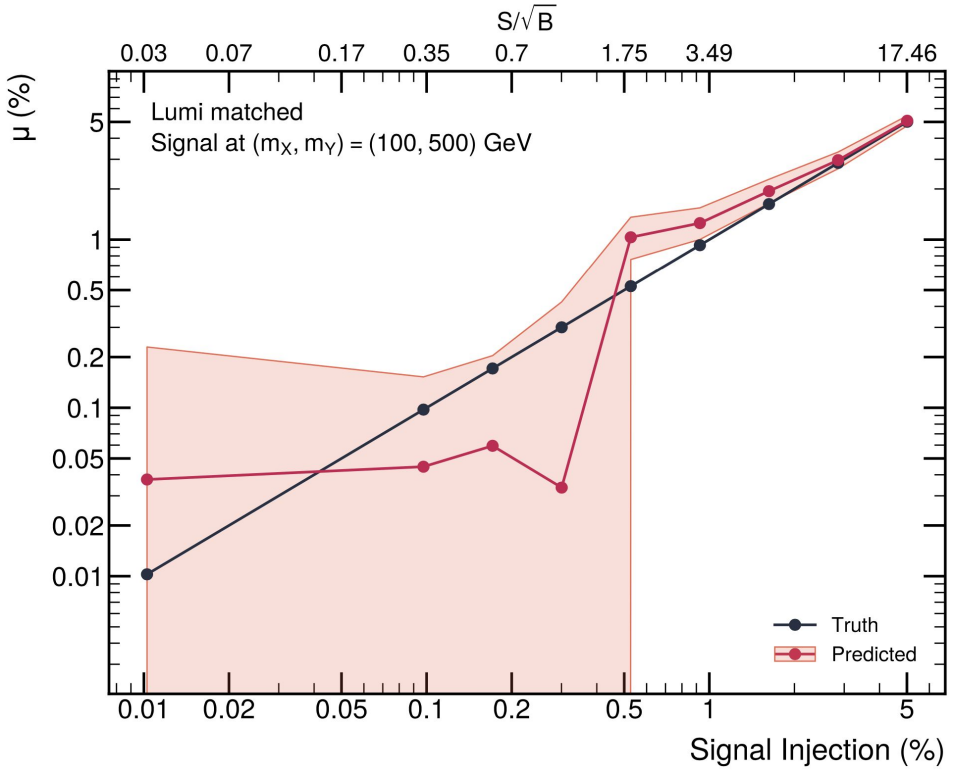
$$p(\quad | \quad)$$



- Signal model independent
- Completely data driven

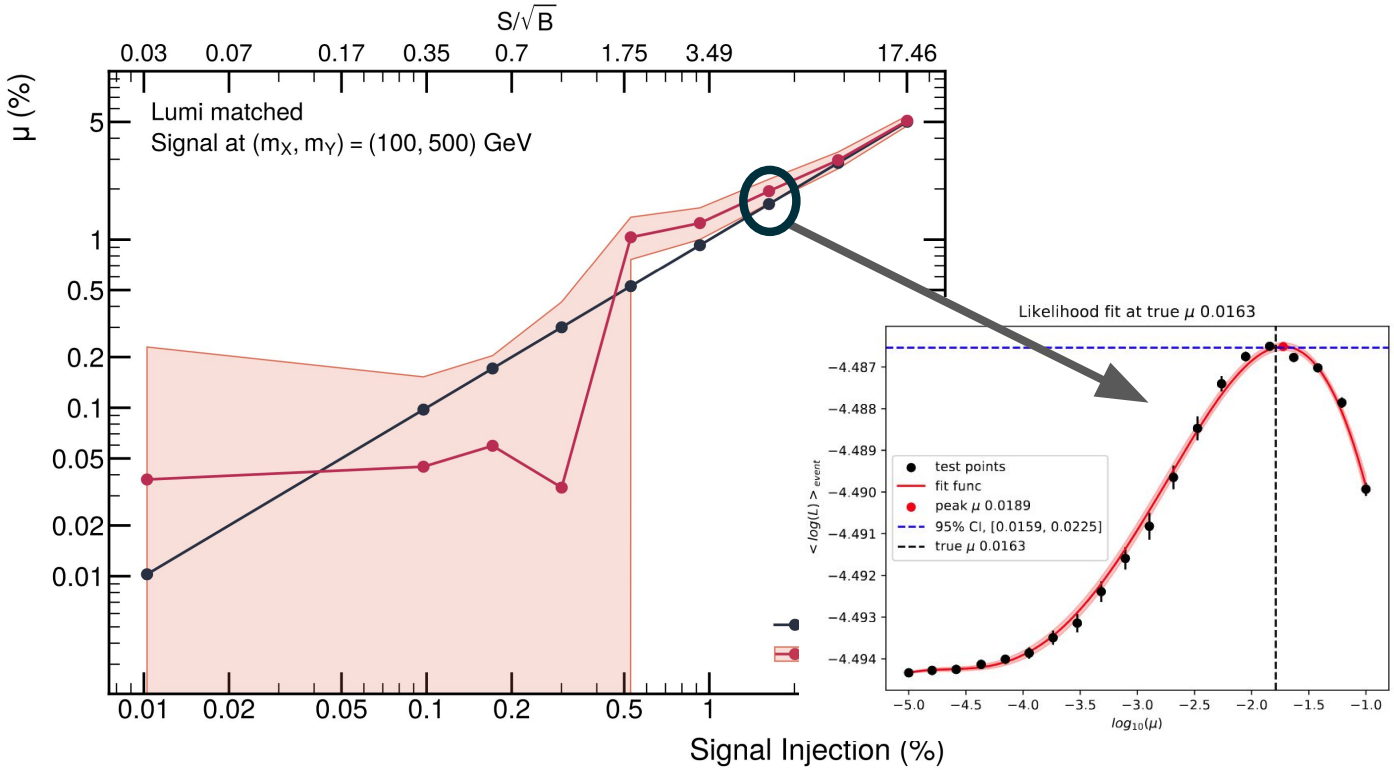
Inference with RANODE

- Applying previous method at different signal injection strengths gives us the following plot:



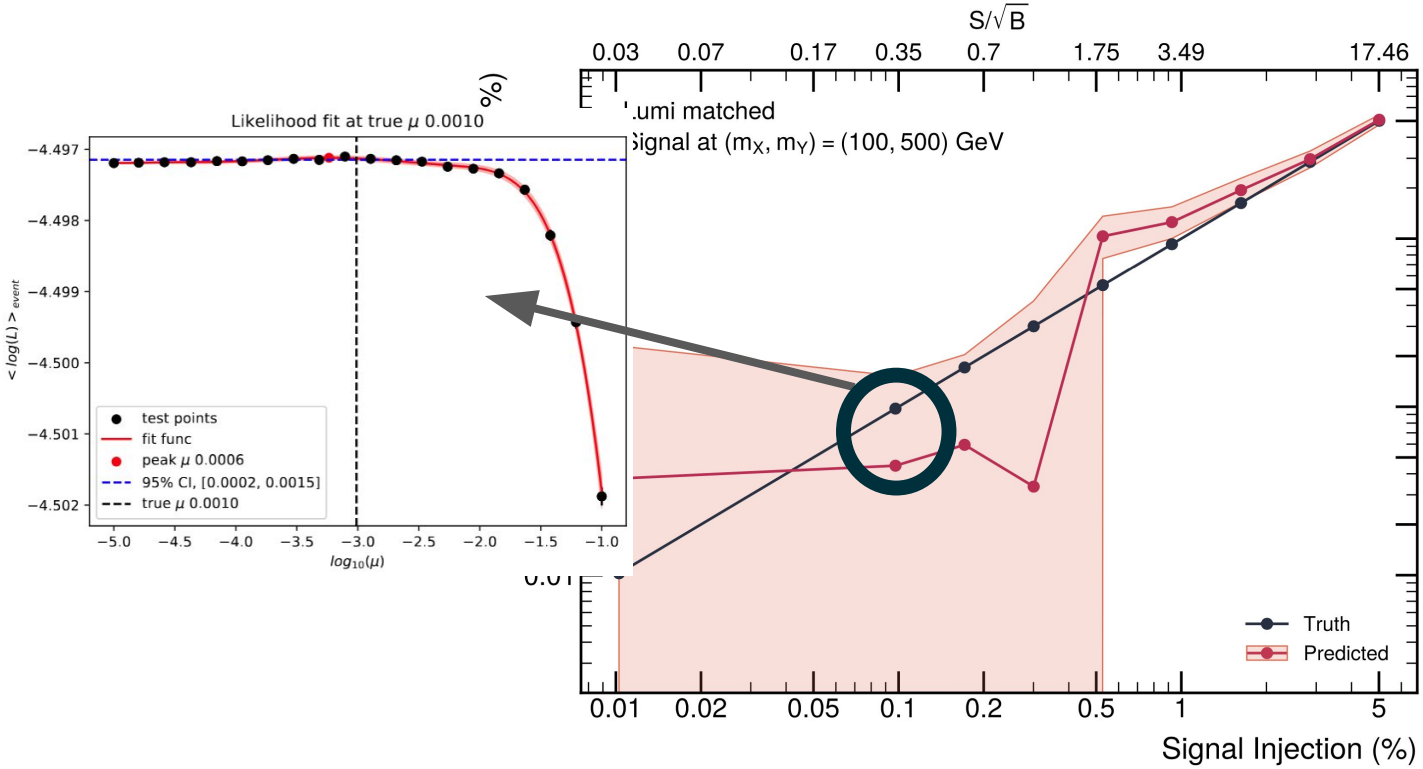
Inference with RANODE

- Applying previous method at different signal injection strengths gives us the following plot:



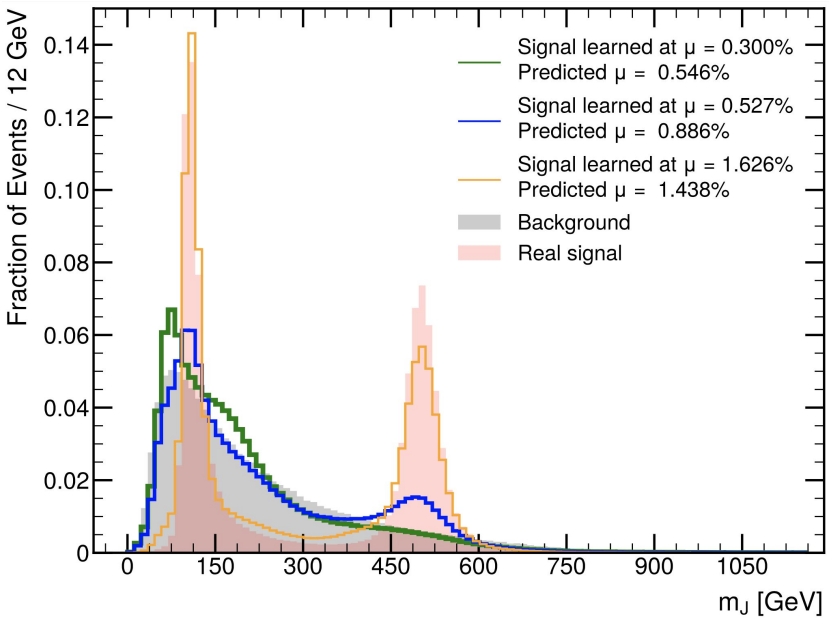
Inference with RANODE

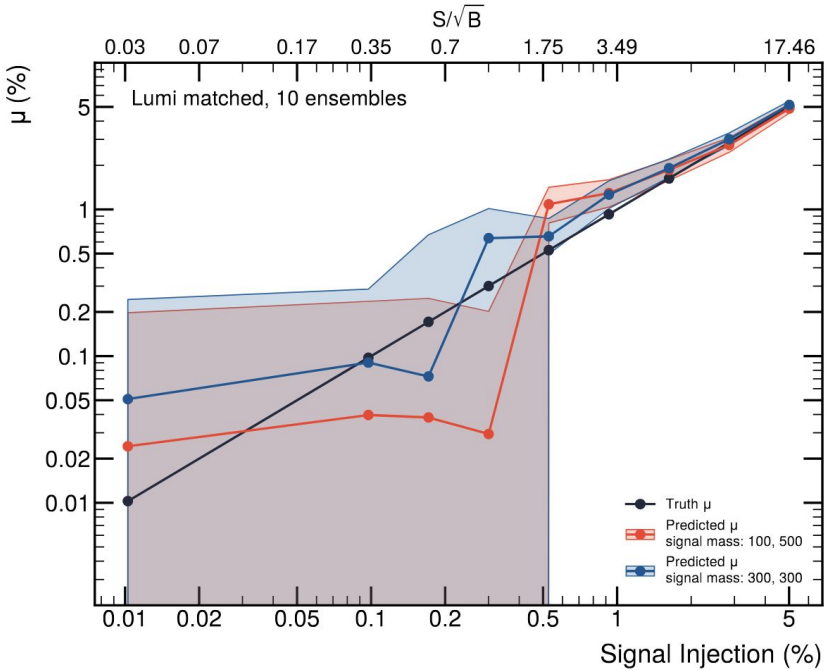
- Applying previous method at different signal injection strengths gives us the following plot:



Inference with RANODE

- RANODE provides good interpretability + is model independent:

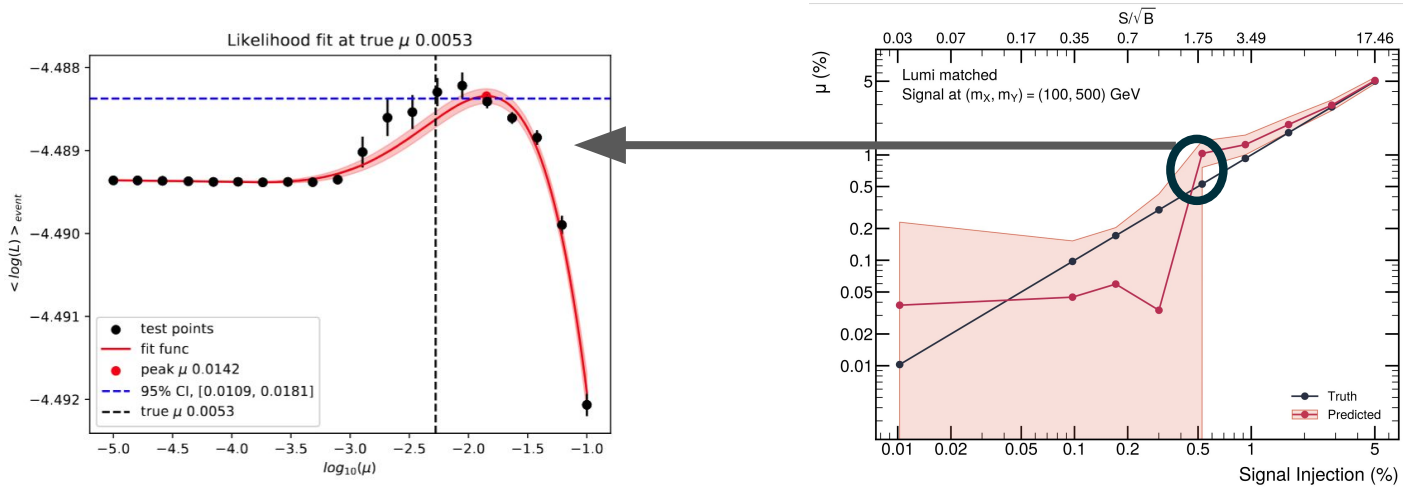




Inference with RANODE

- RANODE inference shows bias towards larger μ at small signal strength
  - Due to the imperfection of model B, what model S learns is a mixture of signal and error of model B ⇒ makes optimal μ larger

$$L = \sum_{x_i \in SR} \log(\mu p_S(x_i) + (1-\mu)p_B(x_i))$$



Part II. PAWS Based GBI

Prior-Assisted Weak Supervision (PAWS)

- Pre-train supervised classifier g(x, θ) on signal vs background
  - x : input feature vector [m_{j1} , m_{j2} , τ^{j1}_{21} , τ^{j2}_{21} , τ^{j1}_{32} , τ^{j2}_{32}]
  - θ : signal model parameter (m_X, m_Y)
  - Signals: MC simulation with different θ values
  - Backgrounds: Conditional Flow Matching (CFM) model trained in SB data and then interpolated into SR
    - Since background samples do not have signal parameter θ, they will be replicated with each θ value in signal

Prior-Assisted Weak Supervision (PAWS)

- Pre-train supervised classifier g(x, θ) on signal vs background
  - x : input feature vector [m_{j1} , m_{j2} , τ^{j1}_{21} , τ^{j2}_{21} , τ^{j1}_{32} , τ^{j2}_{32}]
  - θ : signal model parameter (m_X, m_Y)
  - Signals: MC simulation with different θ values
  - Backgrounds: Conditional Flow Matching (CFM) model trained in SB data and then interpolated into SR
    - Since background samples do not have signal parameter θ, they will be replicated with each θ value in signal
- Parameterizing g on θ allows it to interpolate on θ and be less model specific
  - g(x, θ) can be used to define the likelihood ratio (full derivation in backup):

$$\Lambda_{FS}(x|\theta) = \frac{P_S(x|\theta)}{P_B(x|\theta)} \approx \kappa(\theta) * \frac{g(x,\theta)}{1-g(x,\theta)}$$

$$\kappa(\theta) = \frac{P_B(\theta)}{P_S(\theta)}$$

Inference with PAWS

- Likelihood on data can also be expressed (derivation in backup):

$$\begin{aligned}\Lambda_{WS}(x|\theta,\mu) &= \frac{P_{Data}(x|\theta,\mu)}{P_{refB}(x)} \\ &= \mu * \Lambda_{FS}(x|\theta) + (1-\mu) \\ &\approx \mu * \kappa(\theta) * \frac{g(x,\theta)}{1-g(x,\theta)} + (1-\mu)\end{aligned}$$

Inference with PAWS

- Likelihood on data can also be expressed (derivation in backup):

$$\Lambda_{WS}(x|\theta,\mu) = \frac{P_{Data}(x|\theta,\mu)}{P_{refB}(x)}$$
$$= \mu * \Lambda_{FS}(x|\theta) + (1-\mu)$$
$$\approx \mu * \kappa(\theta) * \frac{\text{  }(x,\theta)}{1-\text{  }(x,\theta)} + (1-\mu)$$

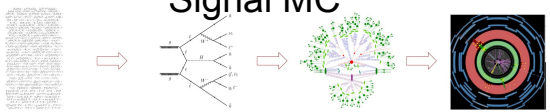
- Take the pre-trained model g(x, θ), freeze all parameters and make θ learnable, then optimize on θ and μ by maximizing data likelihood
  - Inference part only relies on data
  - Minimizing -log(L) gives the optimal signal fraction μ and signal parameter θ
  - Scanning likelihood over parameter space gives confidence intervals for μ and θ

PAWS Based GBI

Generator

CFM generated background

Signal MC

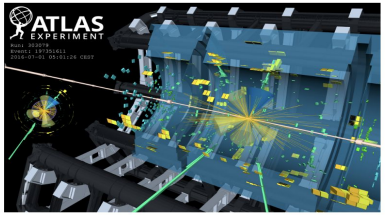


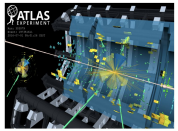
Inference

Pre-trained classifier

g(x, θ)

Data

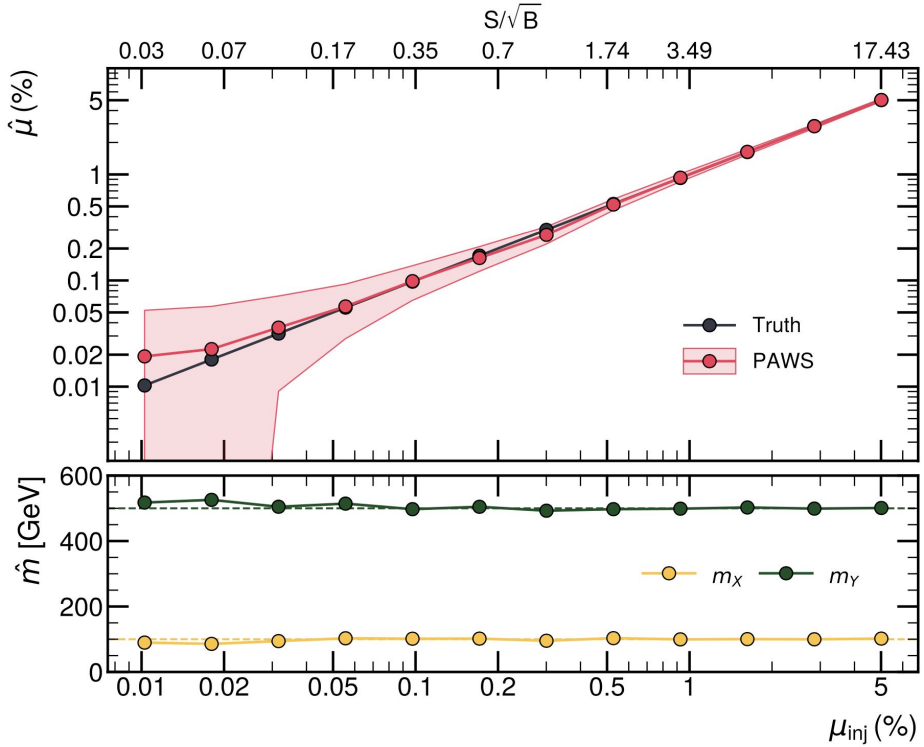


$p($  $|$  $)$

- Signal MC is needed
- Background is data-driven
- Inference only based on data

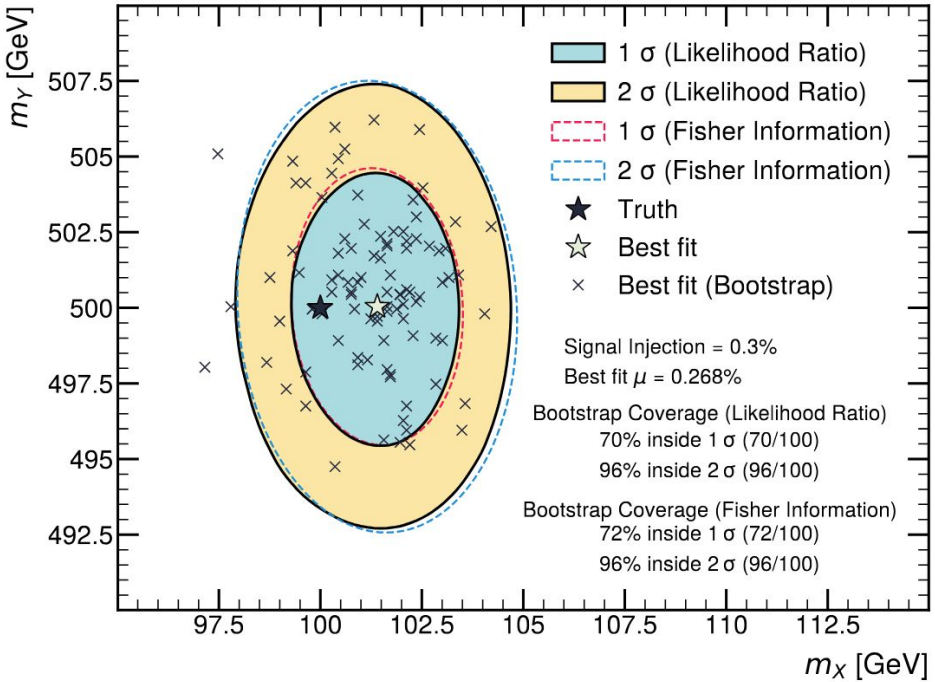
Inference with PAWS

- GBI-PAWS finds signals starting at ~ 0.1σ



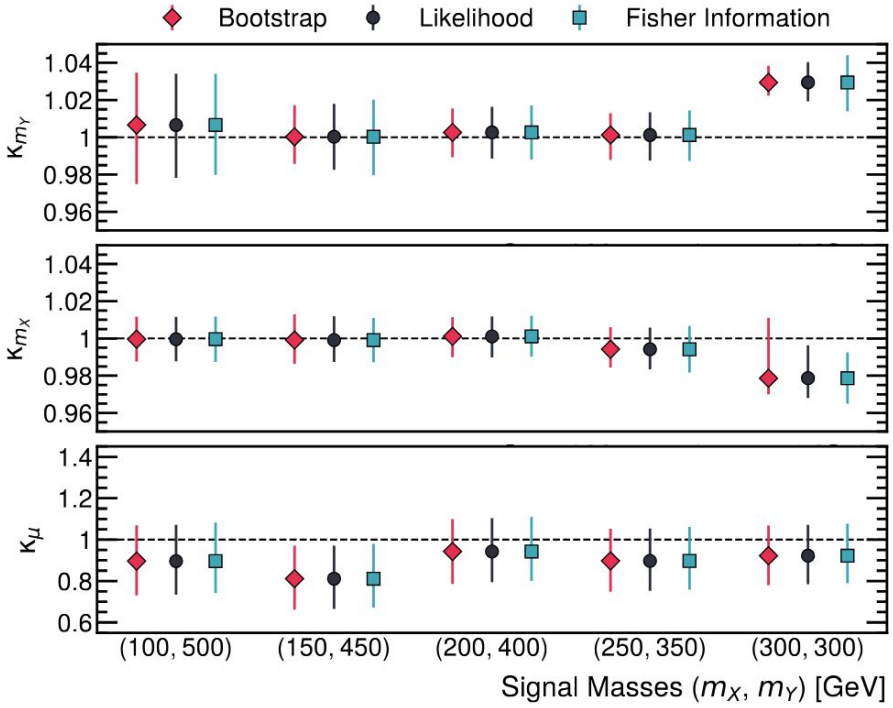
Inference with PAWS

- GBI-PAWS makes good prediction on model parameters
  - Several methods are used to retrieve the confidence interval and their results agree well



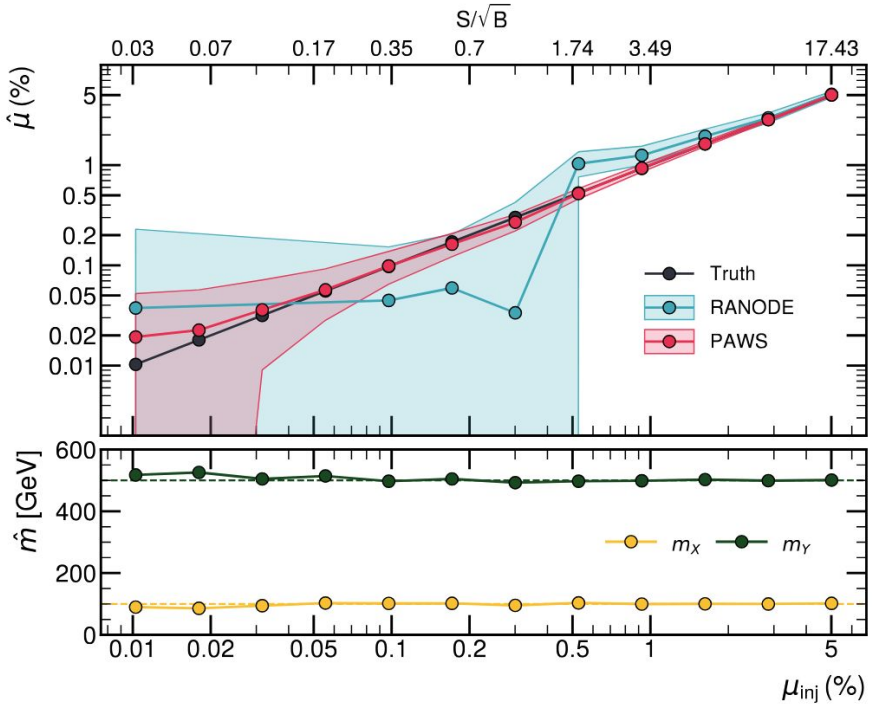
Inference with PAWS

- GBI-PAWS works well on data containing different signal mass models



Comparing PAWS with RANODE in GBI

- RANODE: Signal model independent, fully data driven
- PAWS: Higher sensitivity, accurate confidence interval, needs signal MC



Conclusion

- Generator Based Inference (GBI) (arXiv):
  - Uses generator to infer physics parameters from observed data
  - Improves SBI with less assumption on signal model and less reliance on MC simulation
  - Improves AD with more interpretable results and confidence interval

- Two approaches are introduced in this work:
  - RANODE:
    - NF models are used as generator and to do inference
    - Signal model independent
    - Discovery lower bound ~ 1σ on LHCO dataset, shows bias at smaller signal fraction
  - PAWS:
    - CFM model is used as generator, pretrained MLP is used to do inference
    - Requires pre-training and signal model MC, but background is data driven
    - Discovery lower bound ~ 0.1σ on LHCO dataset with no bias

Backup

Simulation Based Inference

Classical SBI:

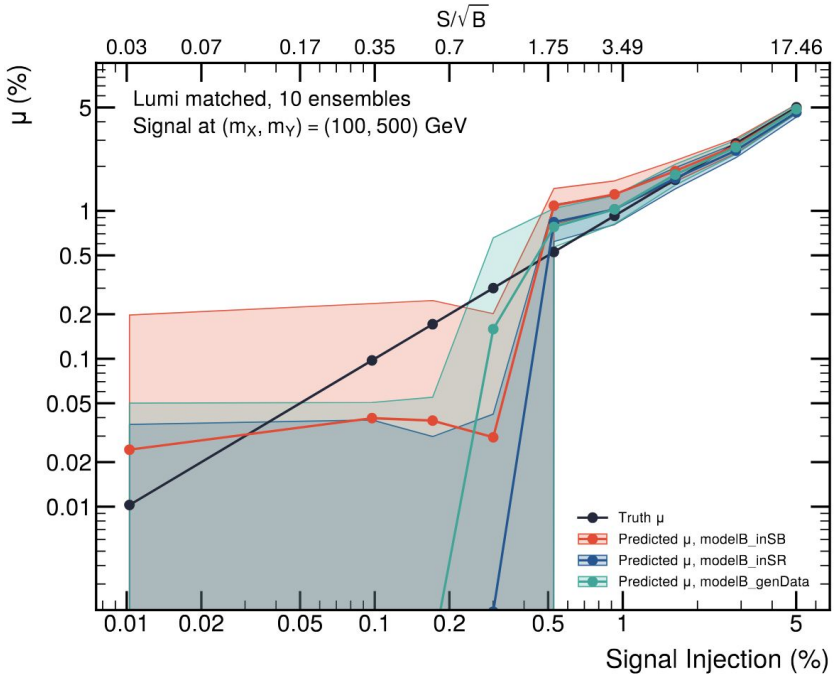
- Doing MC simulation of many possible model parameters
  - High computational cost
  - Needs good MC simulation
  - Model specific
- Fitting histograms to find the best parameters
  - Significant reduction of dimensionality

Neural network based SBI:

- Using machine learning classifiers as likelihood ratio estimator to construct confidence interval
  - Still need MC simulation to train the classifier

Inference with RANODE

- This effect can be compensated by training model B directly in SR or using model B to sample background data in SR ⇒ unrealistic



Derivation of PAWS' loss function

$$
\begin{aligned}
g(x,\theta) &\approx \frac{P_S(x,\theta)}{P_S(x,\theta)+P_B(x,\theta)} \\
&= \frac{P_S(x|\theta)P_S(\theta)}{P_S(x|\theta)P_S(\theta)+P_B(x)P_B(\theta)} \\
&= \frac{P_S(x|\theta)}{P_S(x|\theta)+P_B(x)\kappa(\theta)}
\end{aligned}
$$

For background, x and θ are independent

$$
\Lambda_{FS}(x|\theta) = \frac{P_S(x|\theta)}{P_B(x|\theta)} \approx \kappa(\theta) * \frac{g(x,\theta)}{1-g(x,\theta)}
$$

Derivation of PAWS' loss function

$$
\begin{aligned}
\Lambda_{WS}(x|\theta,\mu) &= \frac{P_{Data}(x|\theta,\mu)}{P_{refB}(x|\theta,\mu)} \\
&\approx \frac{P_{Data}(x|\theta,\mu)}{P_B(x|\theta,\mu)} \\
&= \frac{\mu P_S(x|\theta,\mu) + (1-\mu)P_B(x|\theta,\mu)}{P_B(x|\theta,\mu)} \\
&= \mu \frac{P_S(x|\theta,\mu)}{P_B(x|\theta,\mu)} + (1-\mu) \\
&= \mu * \Lambda_{FS}(x|\theta) + (1-\mu) \\
&\approx \mu * \kappa(\theta) * \frac{g(x,\theta)}{1-g(x,\theta)} + (1-\mu)
\end{aligned}
$$