Optimizing Probes of New Physics Models

Generative Algorithm-Driven Nested Sampling for Parameter Estimation

Rajneil Baruah
Department of Physics
Bennett University

rajneilb.physics@gmail.com

From Big Bang to Now : A Theory-Experiment Dialogue
SRM University-AP

23 January 2025

Preliminary Results of a work in progress
2501.xxxxxx hep-ph

Collaborators:
- Subhadeep Mondal (Bennett U.)
- Sunando Kumar Patra (Bangabasi Evening Coll.)
- Satyajit Roy (Bangabasi College)

Talk Outline

Sampling in BSM Physics

Nested Sampling

ML-Assissted Nested Sampling

Results

BSM Physics

New physics (NP) models are *'Physics Inspired'*

Standard Model (SM) provides highly accurate and experimentally validated predictions.

NP models have limited parameter space available.

Assessing the statistical consistency of new physics models is critical.

Sampling High-Dimensional Space

Curse of Dimensionality: Space Volume grows exponentially. Makes uniform sampling expensive.

Sparse Data Distribution: Have most points far from each other

Complex Landscapes: Often have multiple local minima, complicating efficient exploration.

Computational Cost: Evaluating functions or models for high-dimensional inputs is resource-intensive.

Bayes' Theorem

Likelihood *Function*

Prior *Distribution*

Posterior *Distribution*

$$\boxed{P\left(B_J \mid A\right)} = \frac{\boxed{P\left(A \mid B_J\right)}\boxed{P\left(B_J\right)}}{\boxed{\sum_{i=1}^{n} P\left(A \mid B_i\right) P\left(B_i\right)}}$$

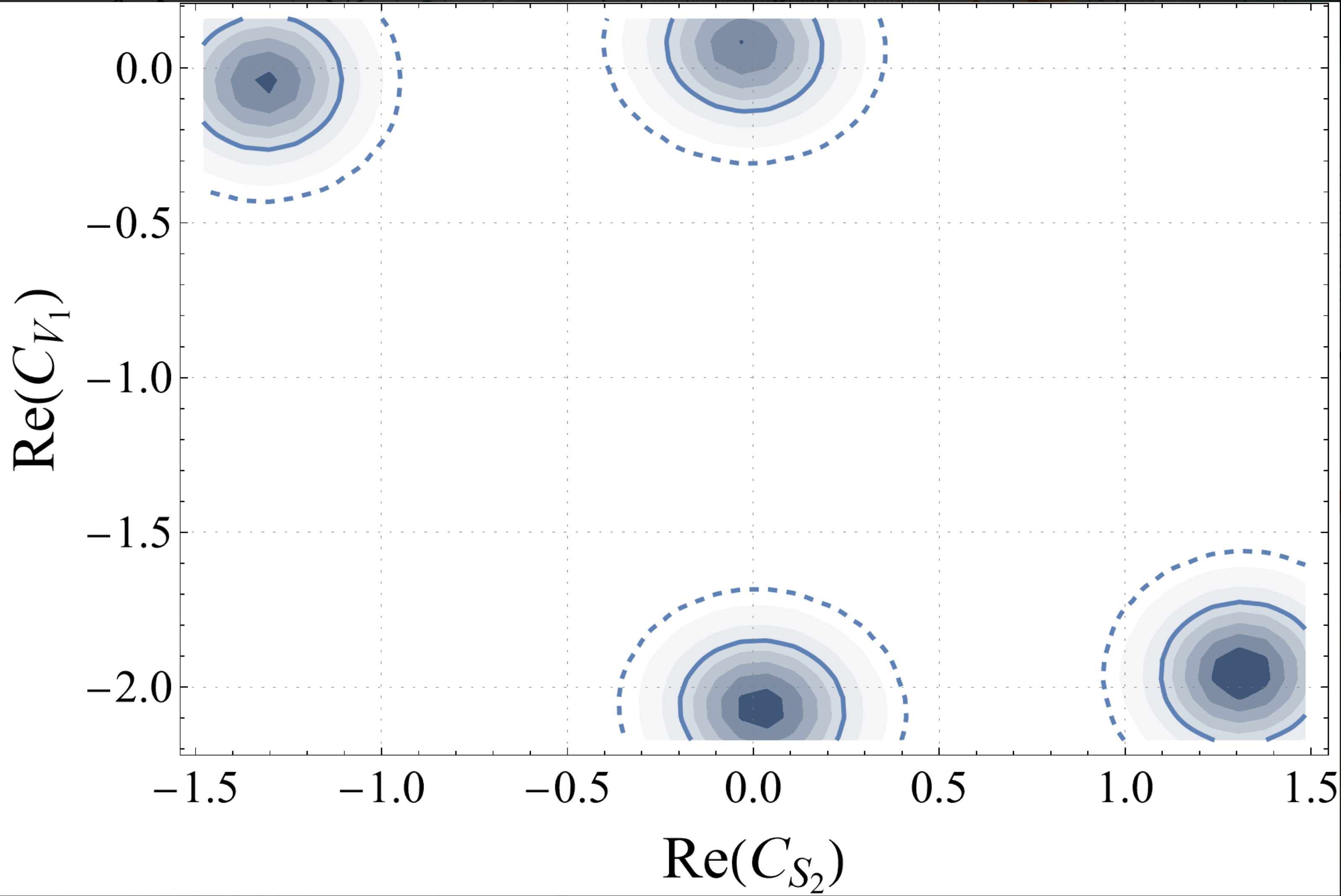
Evidence

Why?

No more "*good points, bad points.*"

If "good", we want to know "how good?"

MCMC



2404.02698 [hep-ph]

Nested Sampling

Nested Sampling

Nested Sampling is a powerful algorithm for calculating ***Bayesian Evidence*** and exploring ***posterior distributions***.

Nested Sampling

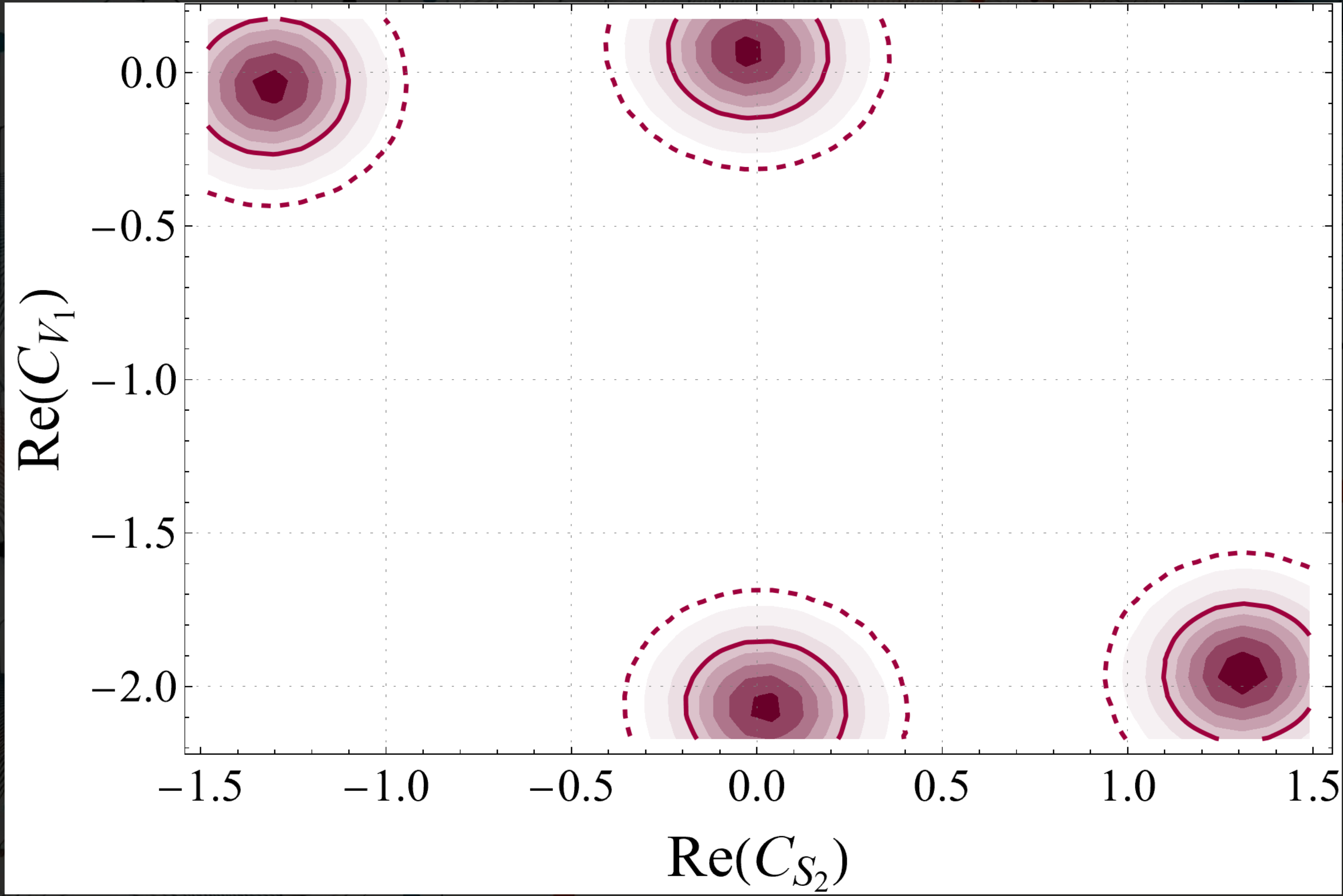
Simultaneous Z calculation for model comparison and parameter inference, unlike classical MCMC.

Effective in multi-modal problems, sampling entire prior space, capturing all modes.

Self-tuning with minimal monitoring, easy application to new problems.

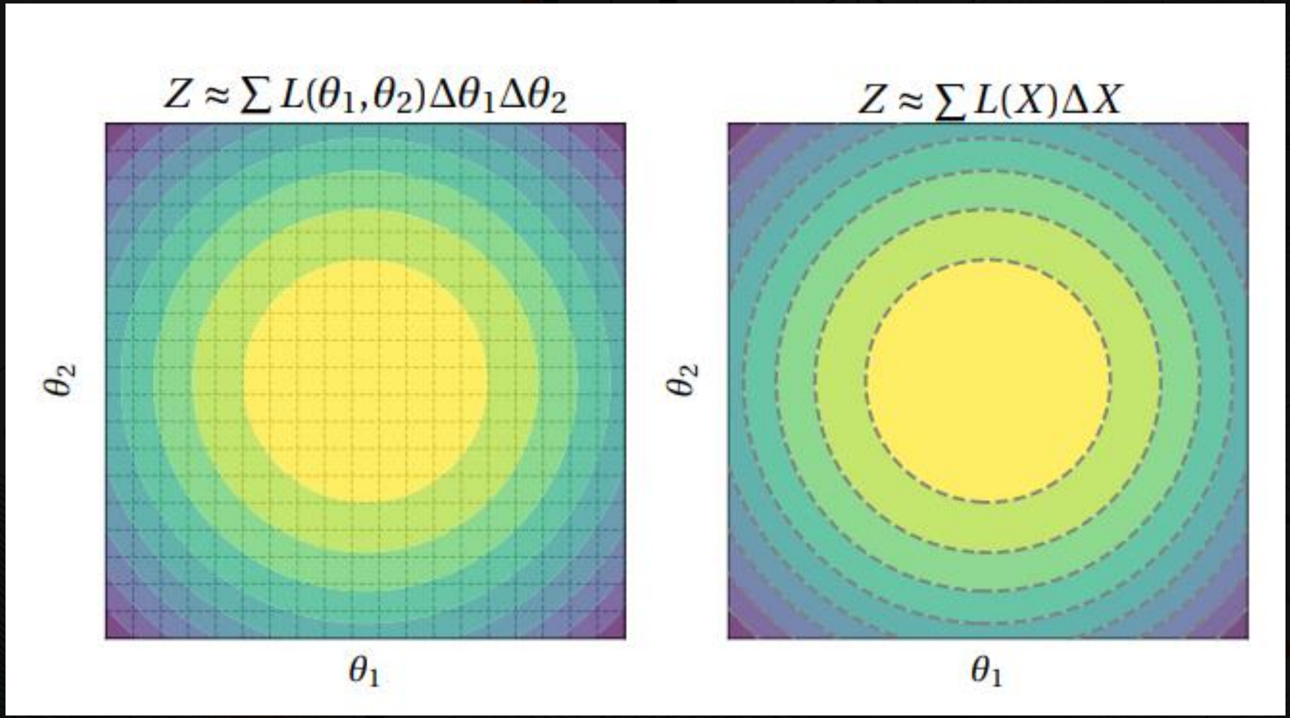
Parallelizable: Combine runs for higher precision, addressing MCMC limitations.

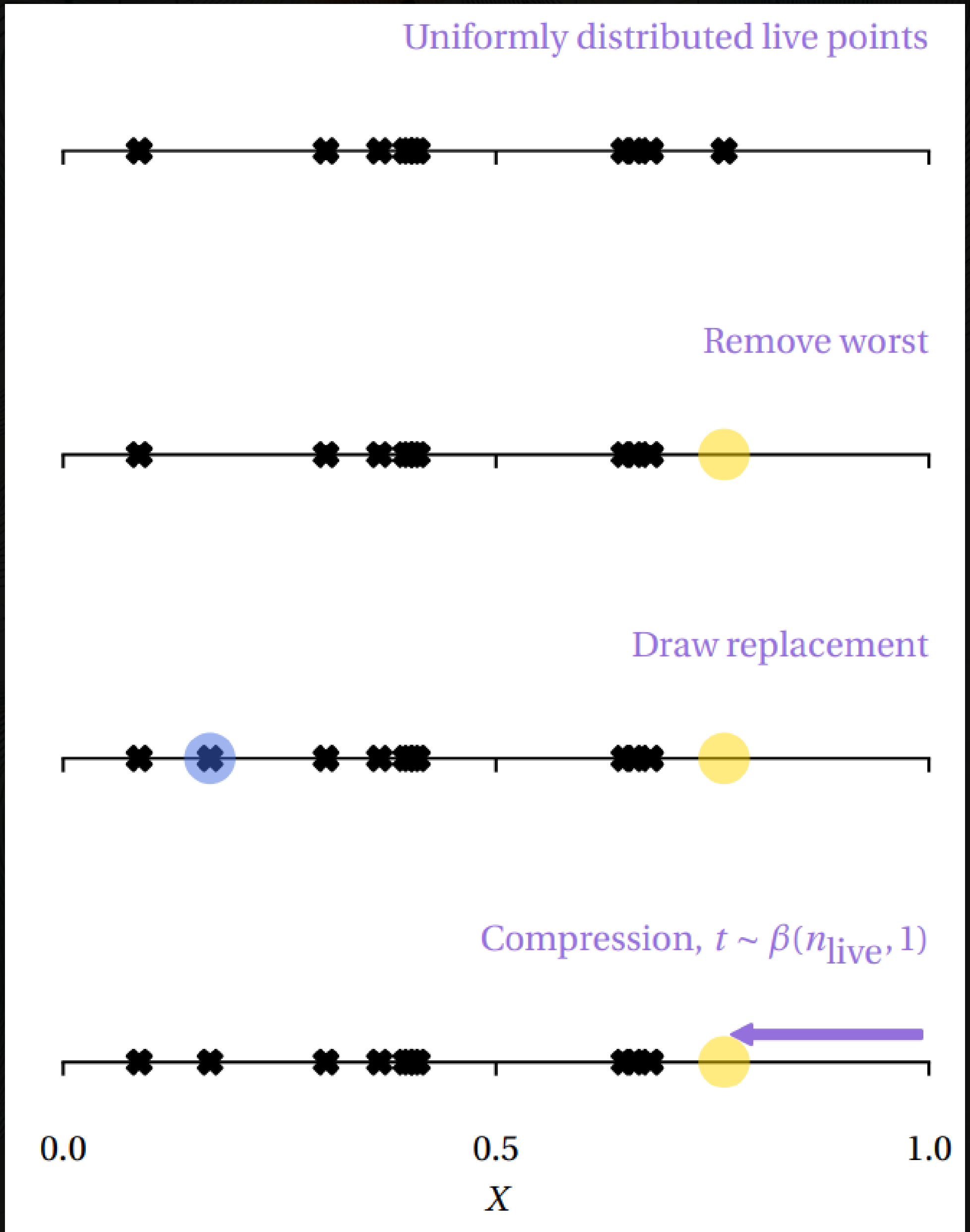
Nested Sampling Result



2404.02698 [hep-ph]

Nested Sampling

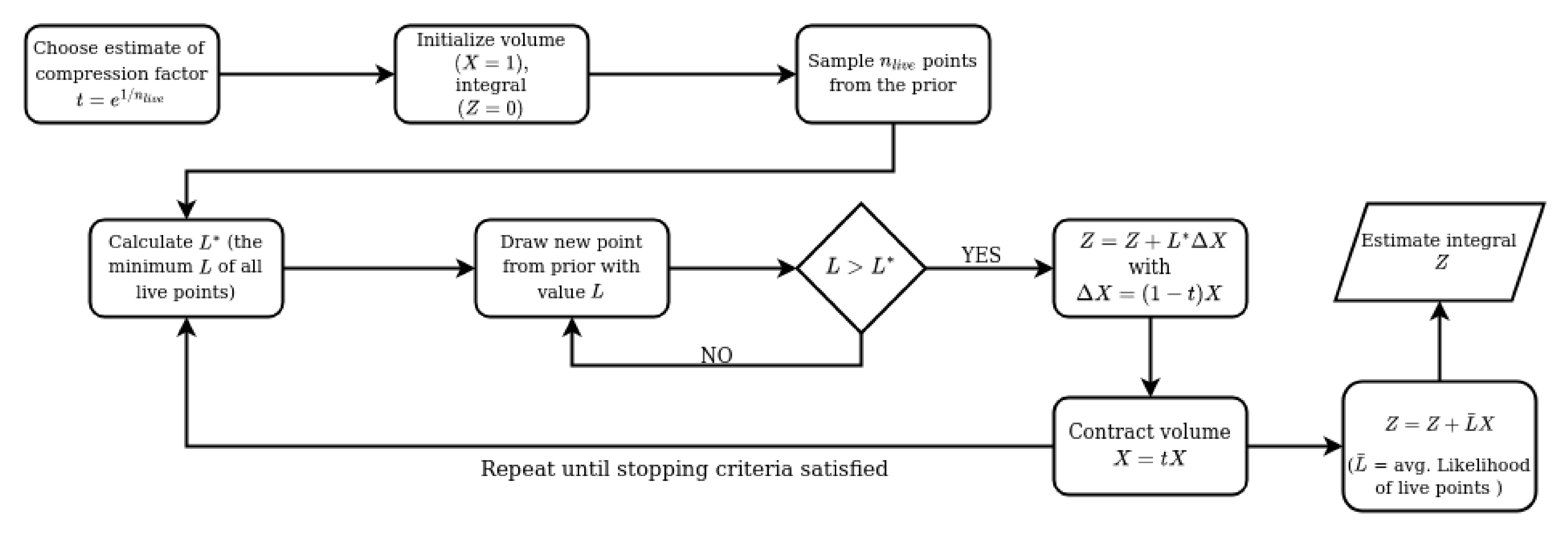




2205.15570[stat.CO]

Nested Sampling

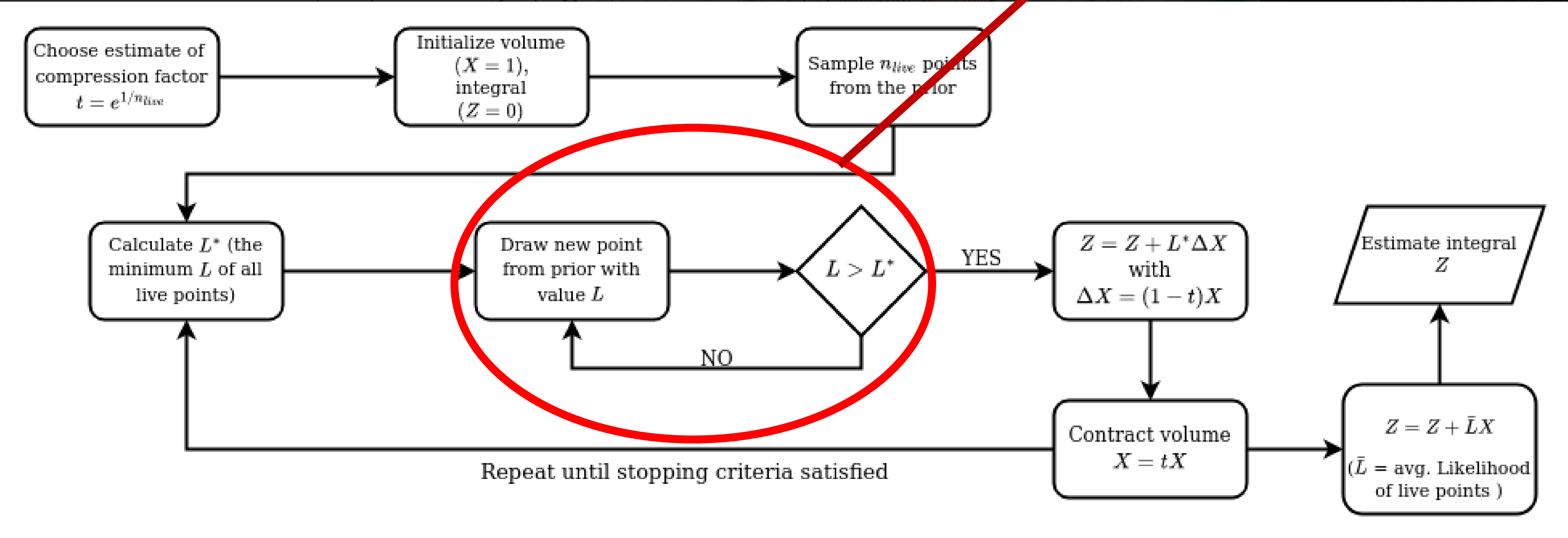
2404.02698 [hep-ph]



Nested Sampling

Extremely Time Inefficient

2404.02698 [hep-ph]



Generative Algorithms

Algorithms that generate new data samples resembling a given dataset, such as images, text, or audio.

Examples: Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and Diffusion Models.

Applications: Image synthesis, text generation, drug discovery, and data augmentation.

Challenges: Balancing realism and diversity, computational cost, and mitigating biases in generated outputs.

Normalizing Flows

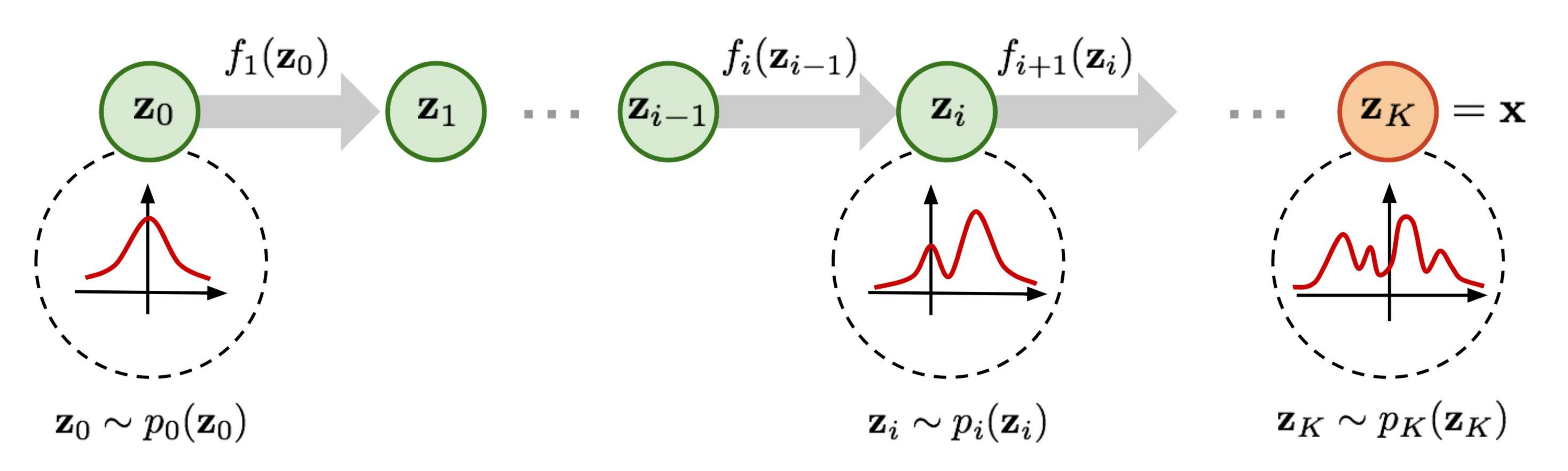
Designed to transform a simple probability distribution into a more complex target distribution: sequence of invertible and differentiable transformations.

Generates realistic samples by sampling from the simple base distribution and applying the learned transformations.

Used in tasks requiring high-quality density estimation or likelihood-based generation.

Normalizing flows use a **deterministic and invertible mapping** to learn the data distribution explicitly.

ML-Assisted NS



https://ankurdhuriya.medium.com/

How do we make use of it?

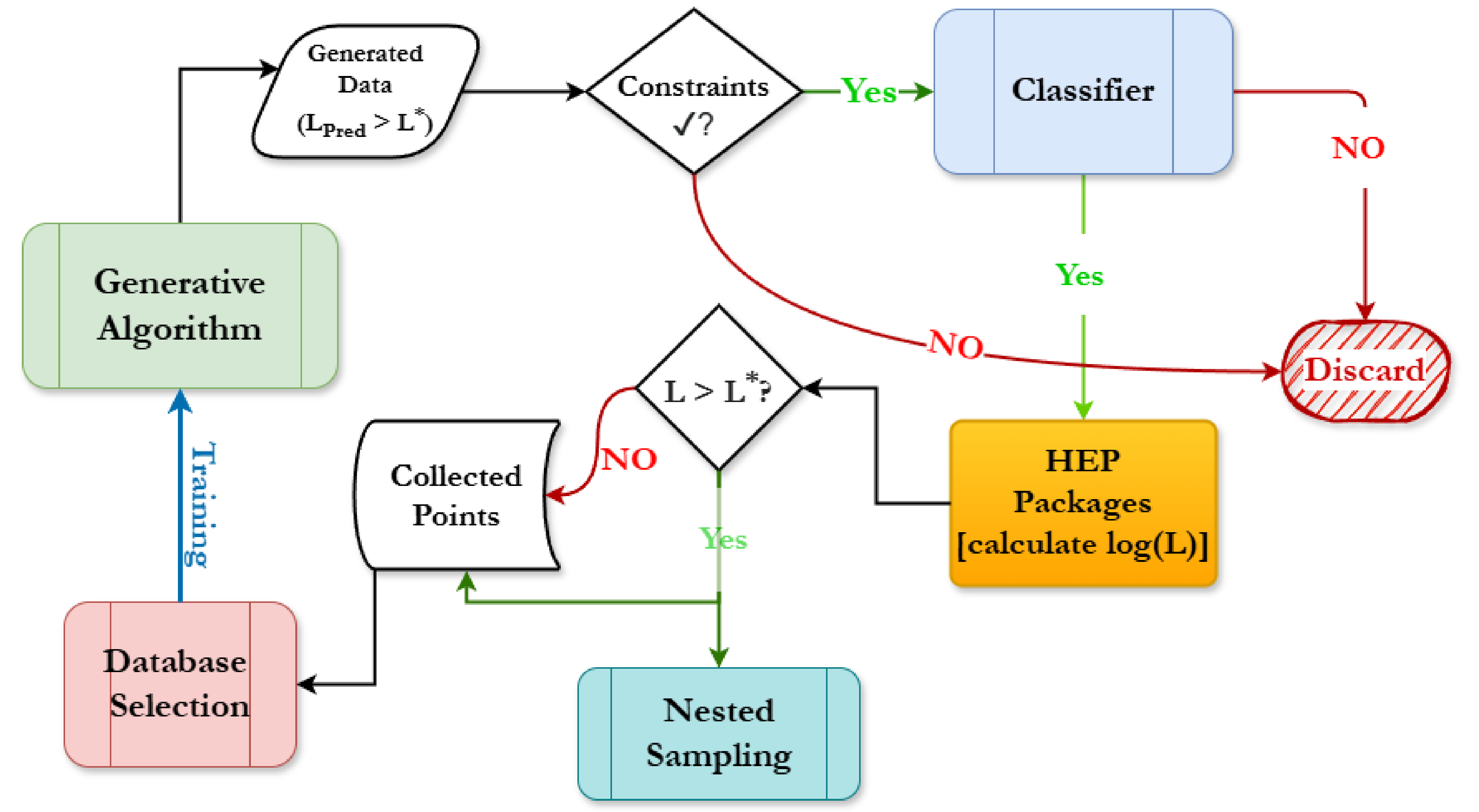
Likelihood Calculation is a Time Inefficient task.

We sample a training dataset consisting of parameters, observables, and chi-square values.

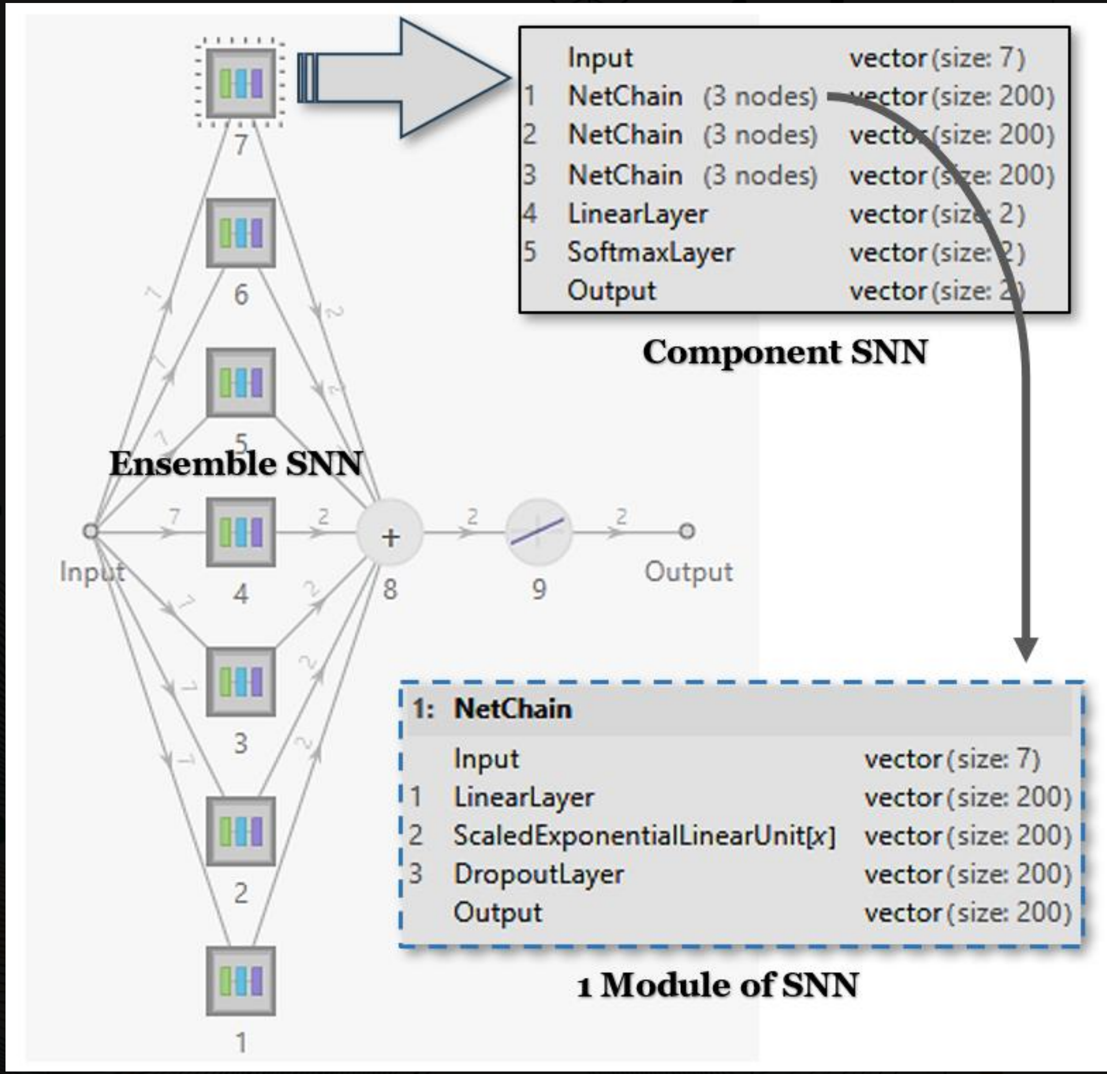
Train a generative algorithm to learn the distribution of the dataset.

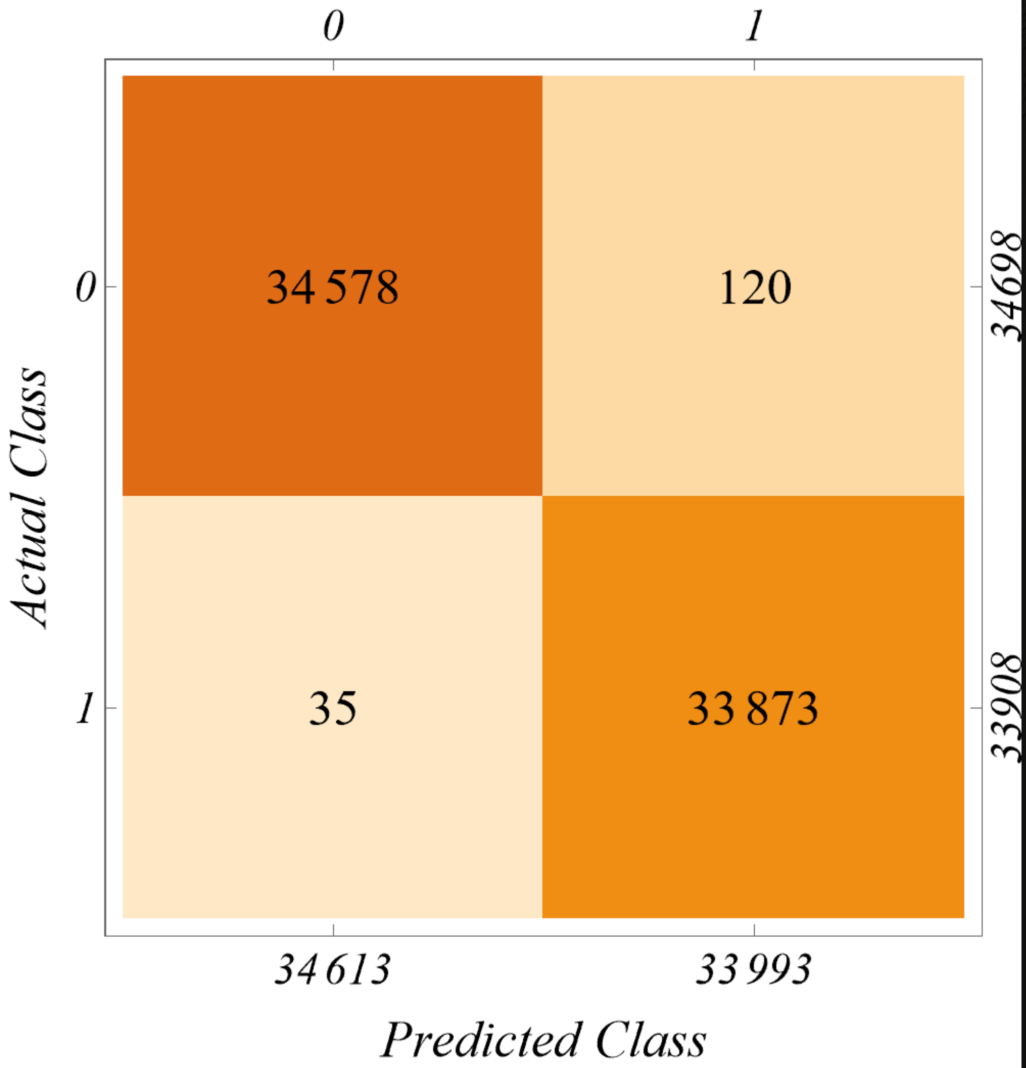
During Nested Sample, we sample from this new *Learned Distribution* rather than the prior itself.

ML-Assisted NS



Classifier


Input vector (size: 7)
1 NetChain (3 nodes) vector (size: 200)
2 NetChain (3 nodes) vector (size: 200)
3 NetChain (3 nodes) vector (size: 200)
4 LinearLayer vector (size: 2)
5 SoftmaxLayer vector (size: 2)
Output vector (size: 2)
Component SNN
Ensemble SNN
Input
Output
1: NetChain
Input vector (size: 7)
1 LinearLayer vector (size: 200)
2 ScaledExponentialLinearUnit[x] vector (size: 200)
3 DropoutLayer vector (size: 200)
Output vector (size: 200)
1 Module of SNN

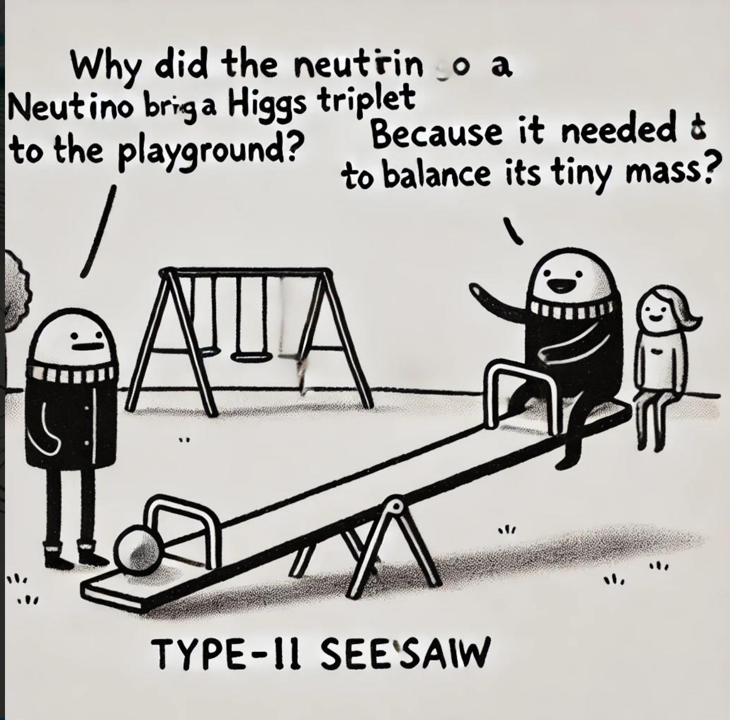

0
1
Actual Class
34 578
120
35
33 873
34698
33908
34 613
33 993
Predicted Class

Type-II Seesaw Model

$$V(H,\Delta) = -m_H^2 H^\dagger H + \frac{\lambda}{4}(H^\dagger H)^2 + M_\Delta^2 Tr(\Delta^\dagger \Delta) + [\mu(H^T i\sigma^2 \Delta^\dagger H) + \text{h.c.}]$$
$$+\lambda_1 (H^\dagger H) Tr(\Delta^\dagger \Delta) + \lambda_2 (Tr\Delta^\dagger \Delta)^2 + \lambda_3 Tr(\Delta^\dagger \Delta)^2 + \lambda_4 H^\dagger \Delta \Delta^\dagger H$$

Extends the SM by introducing a scalar triplet

Explains Neutrino Mass through a Majorna Mass term leading to Lepton Number Violation



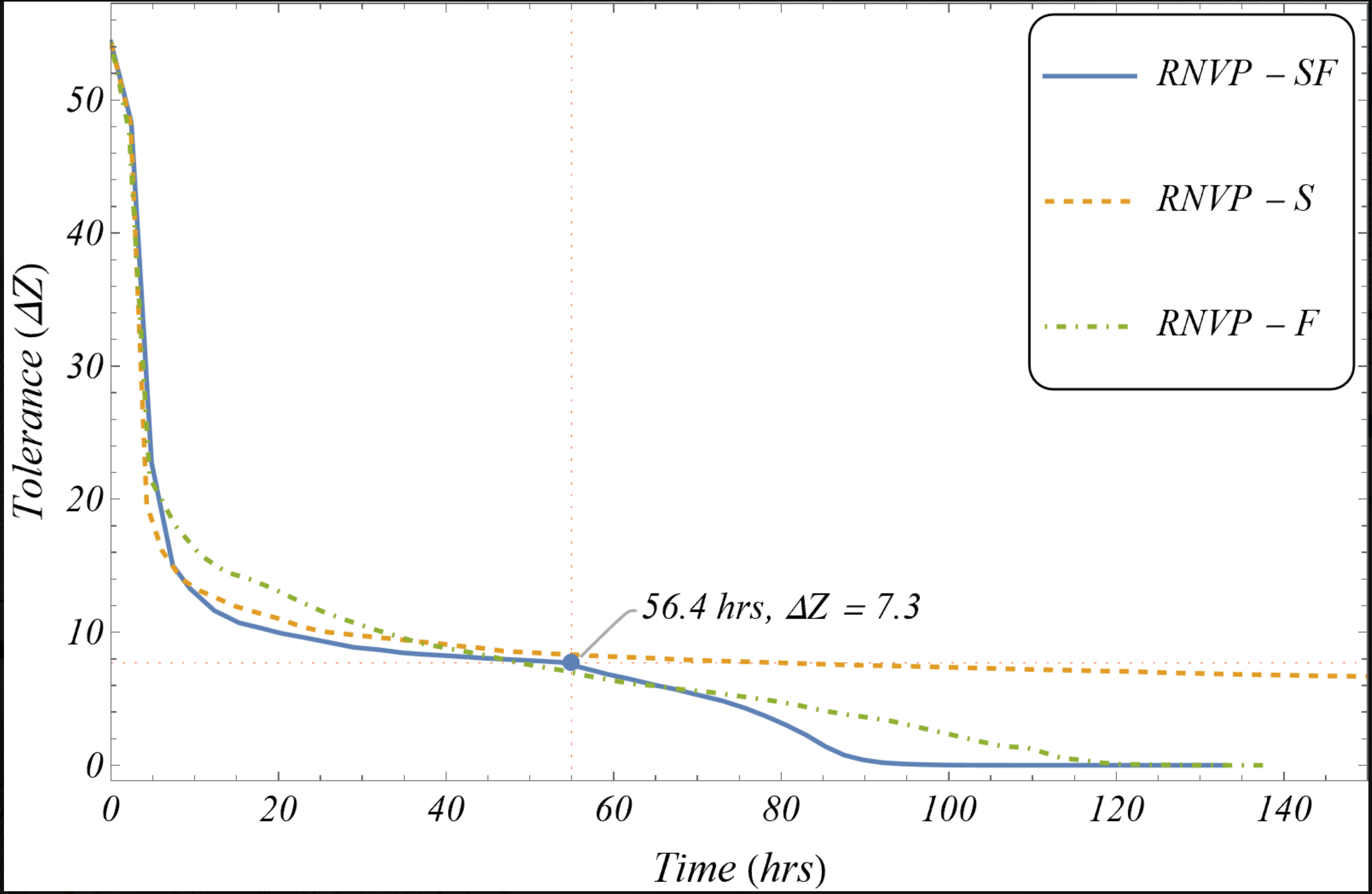
Real NVP – S/F/SF

RealNVP-S: Pick a Representative Sample from the pool of collected points.

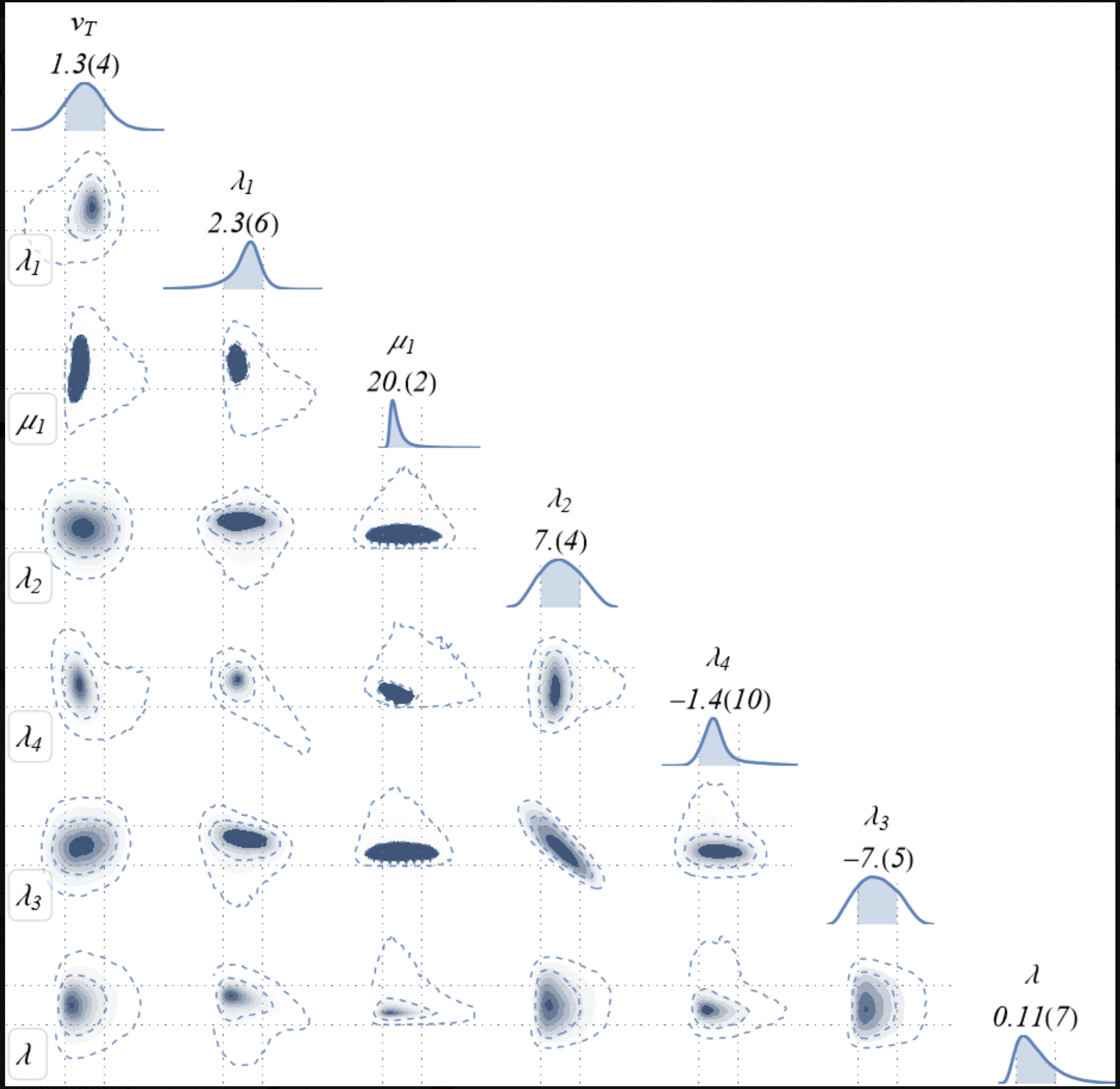
RealNVP-F: Pick sample with *iso-likelihood* contour slightly larger than the one of live points

RealNVP-SF – Start with RNVP-S and after saturation, RNVP-F

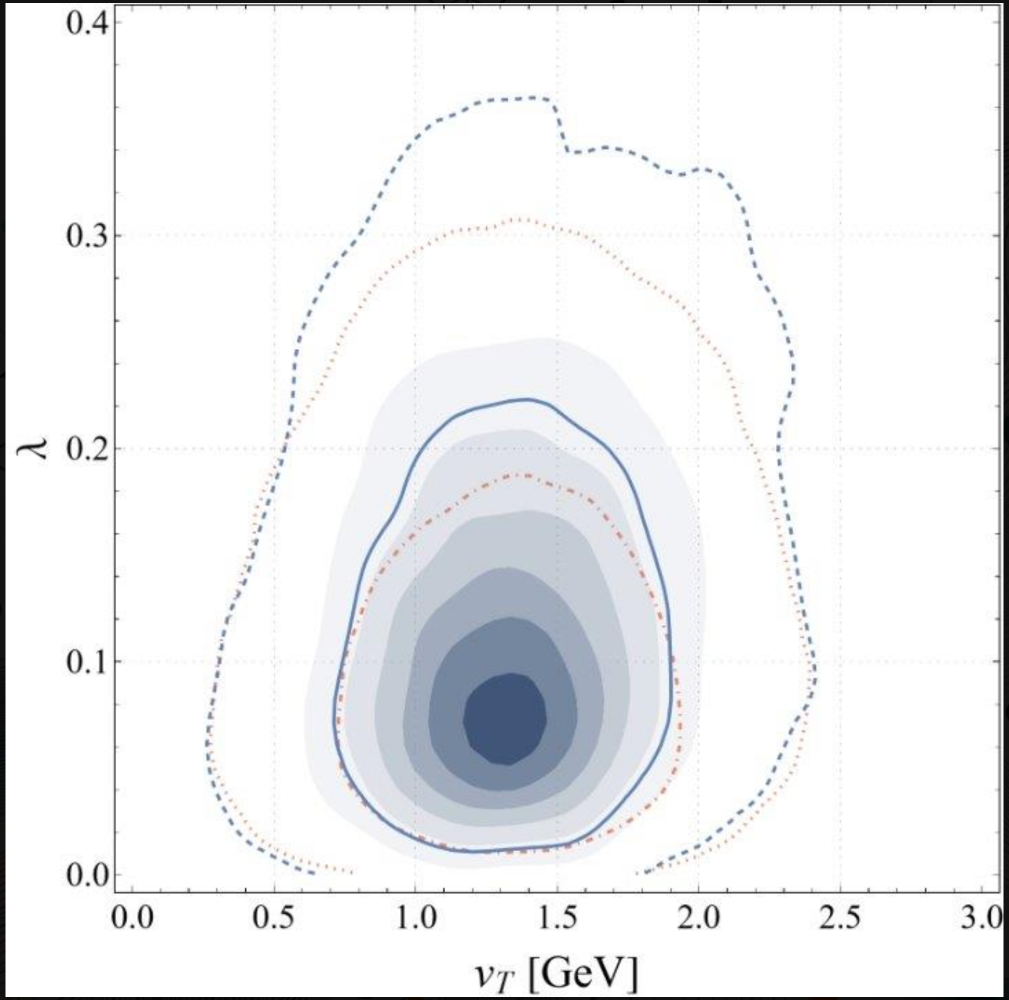
Computation Time

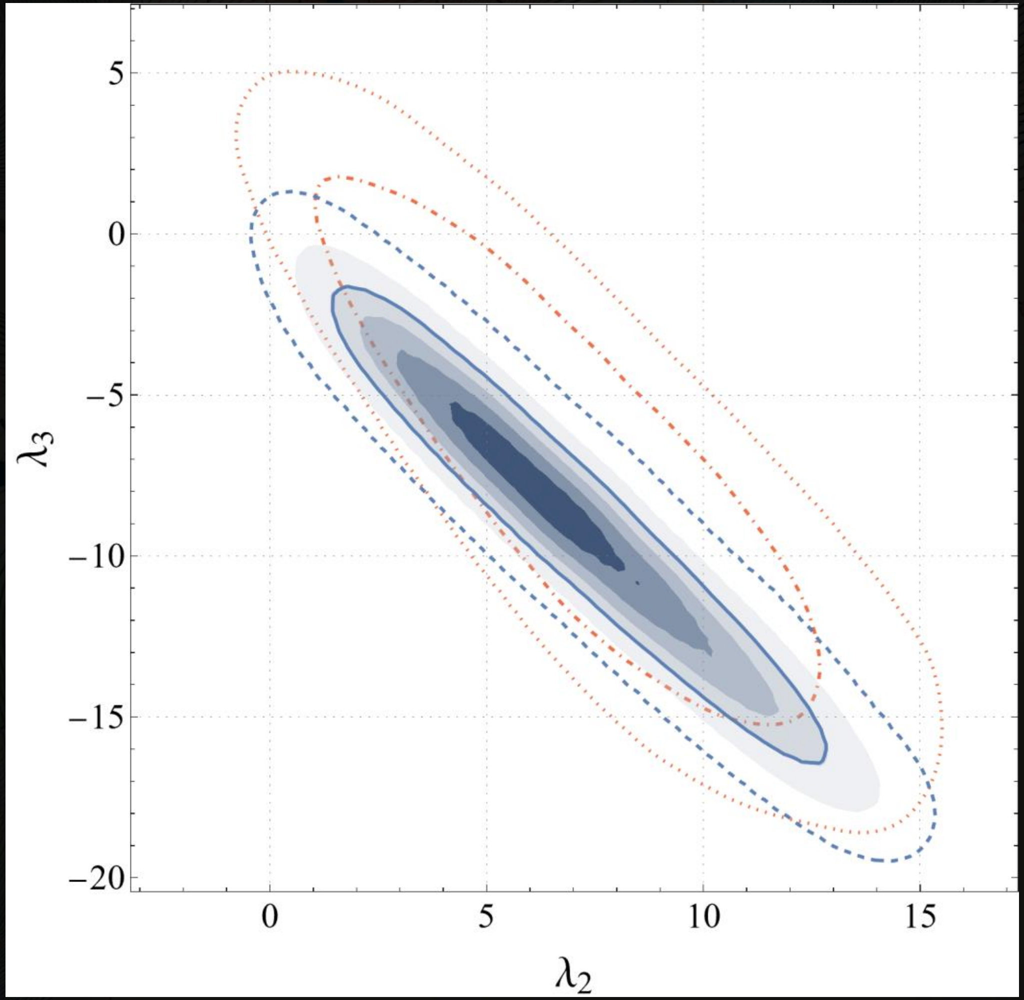


ML-NS Results

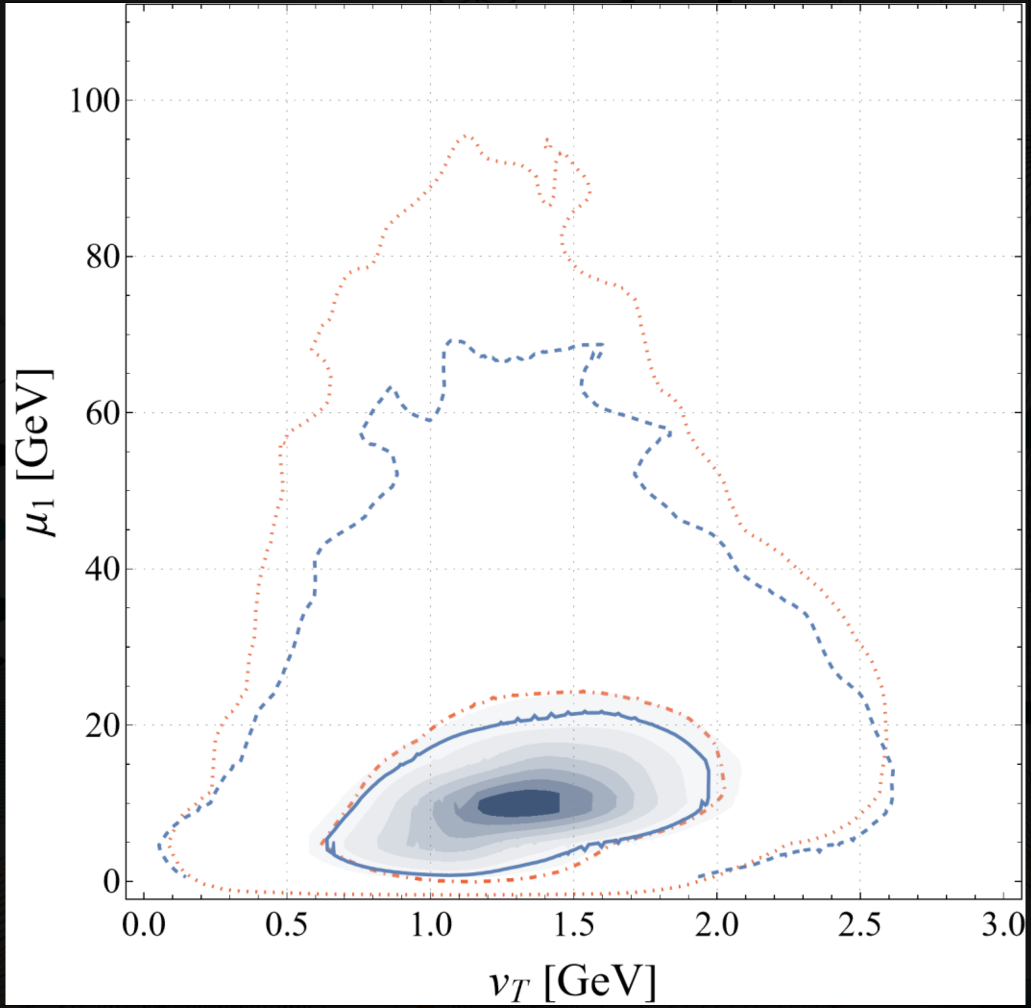


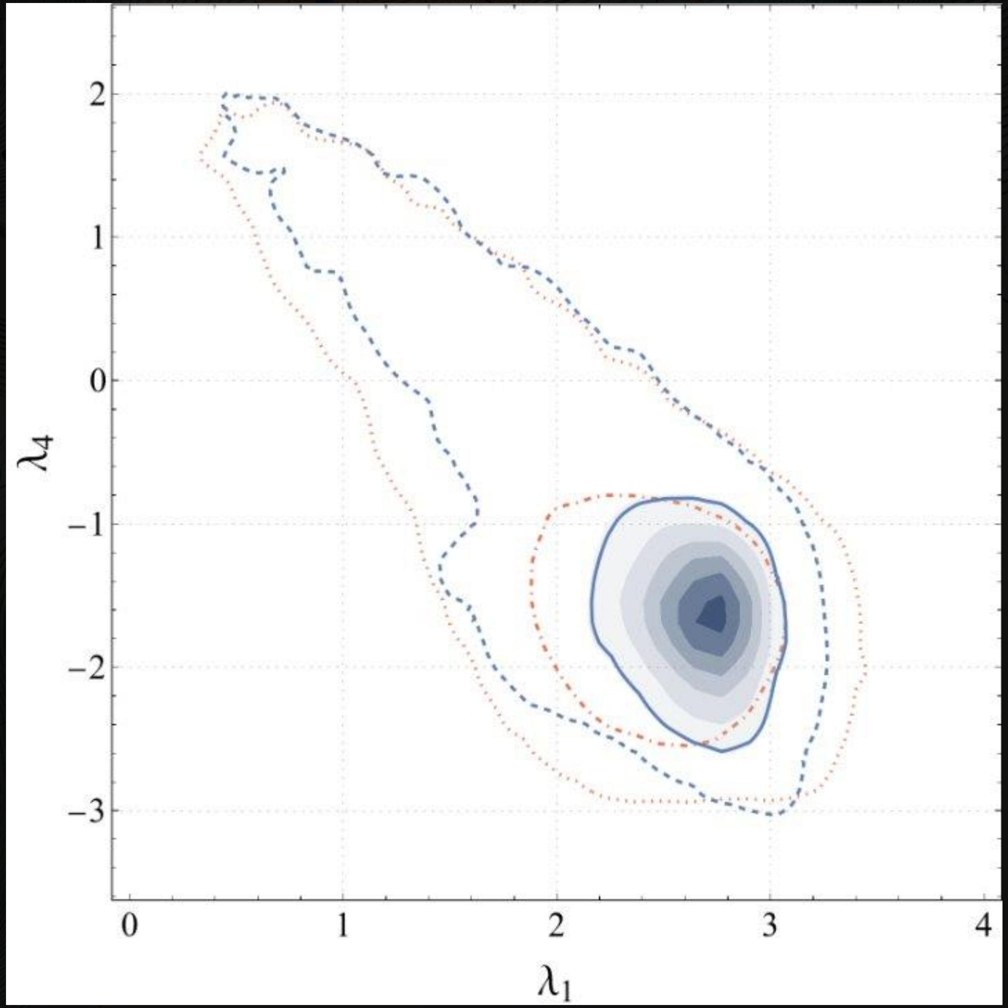
ML-NS Results



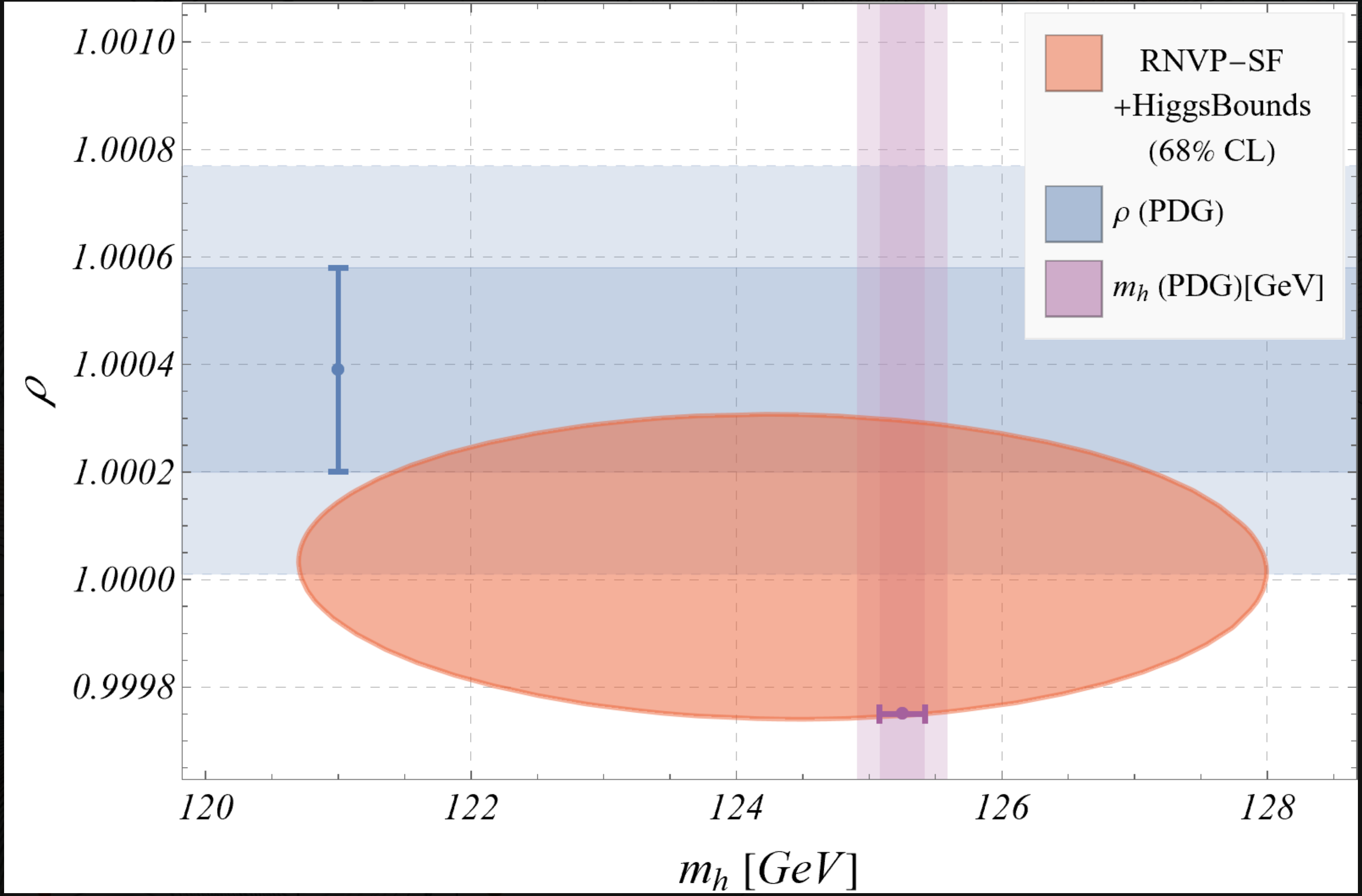


ML-NS Results

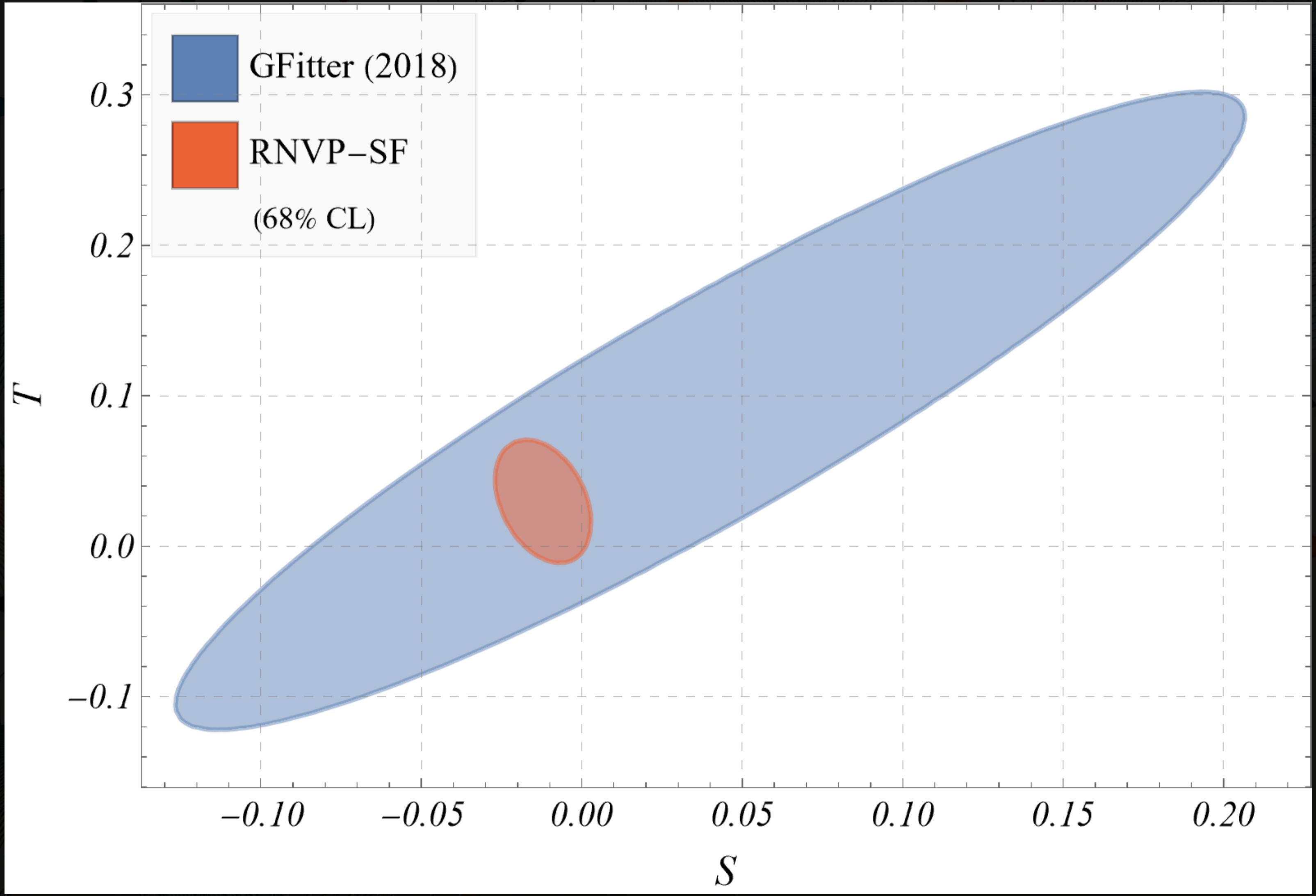




ML-NS Results



ML-NS Results



Conclusion

Sampling BSM spaces with high-dimensional parameter spaces is difficult.

Standard MCMC and NS have their own flaws.

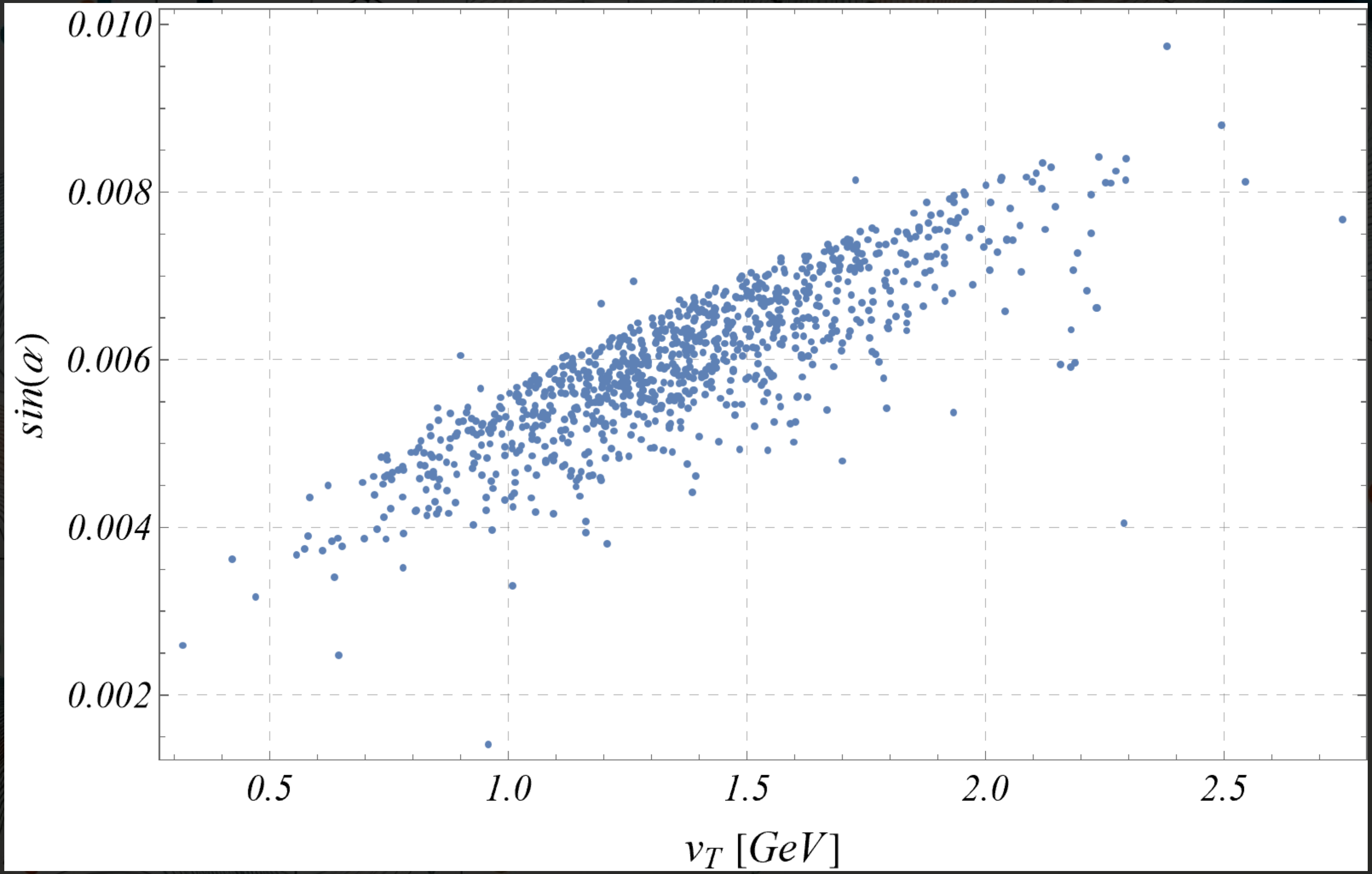
Using generative algorithms to <u>*assist*</u> NS makes the algorithm faster.

ML-NS overcomes this and samples the space accurately and in much lesser time.

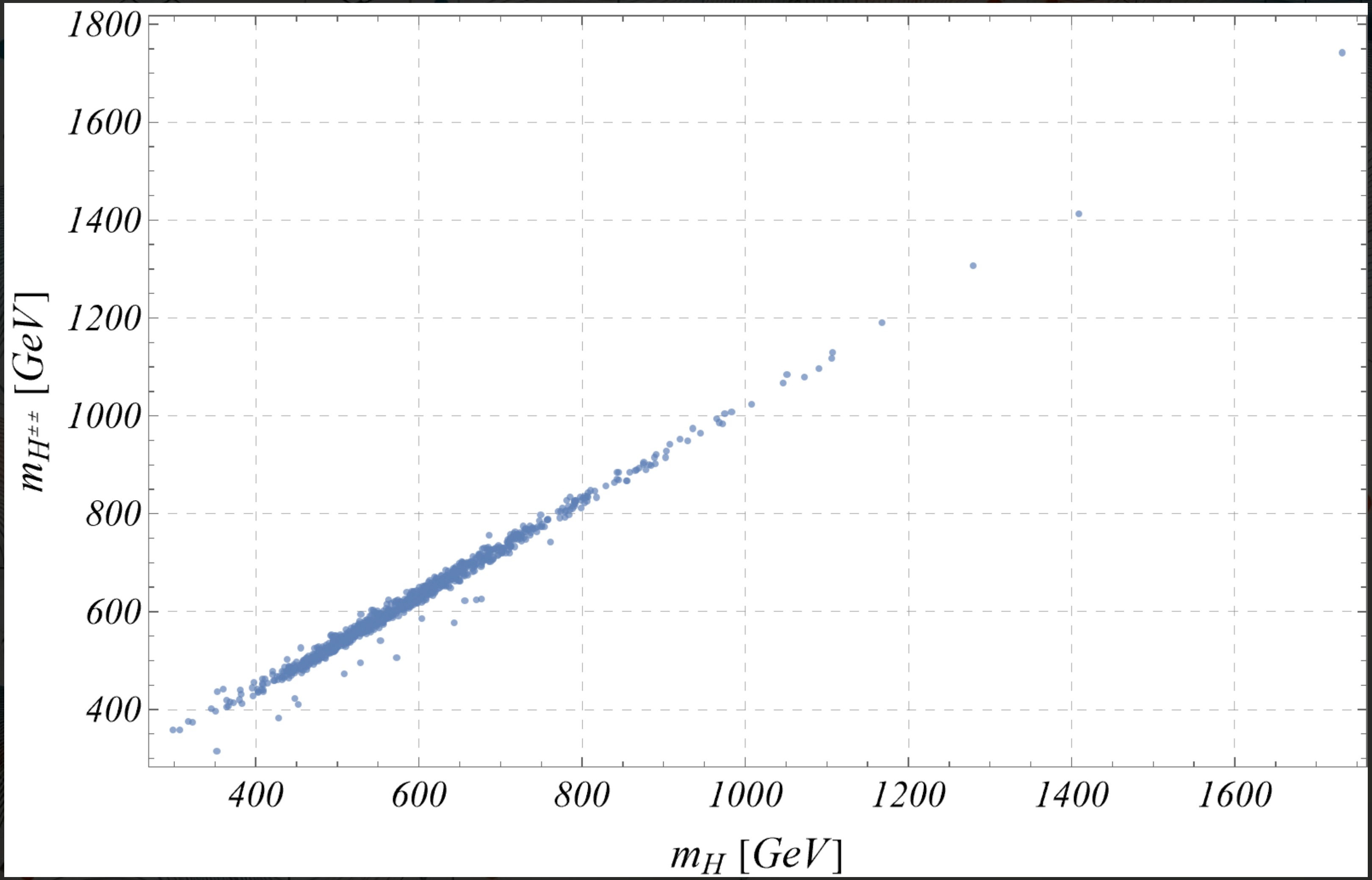
Thank You

I would like to acknowledge ANRF(erstwhile DST-SERB) India for providing Core Research Grant no. CRG/2022/003208.

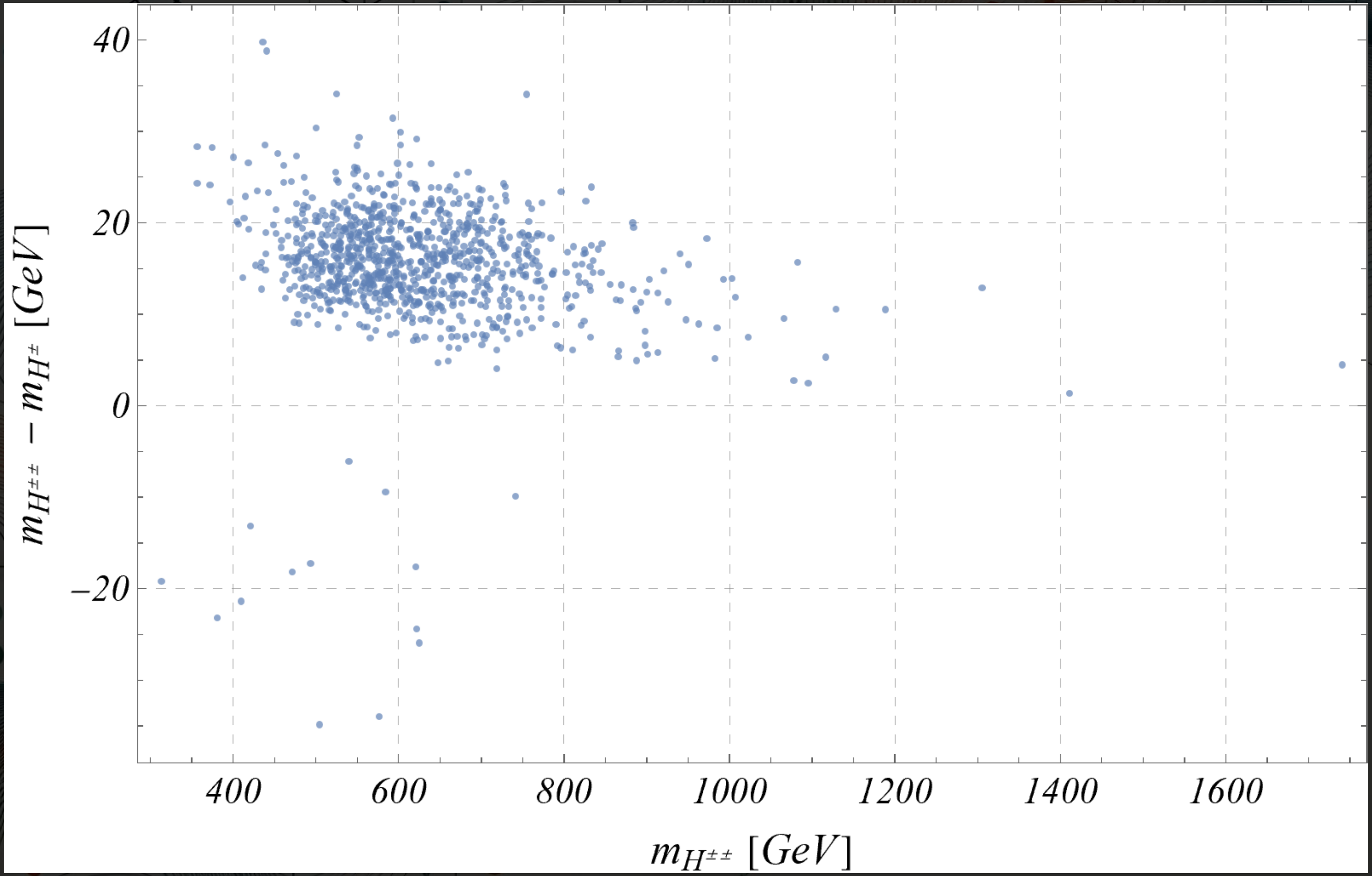
Extra Slide 1



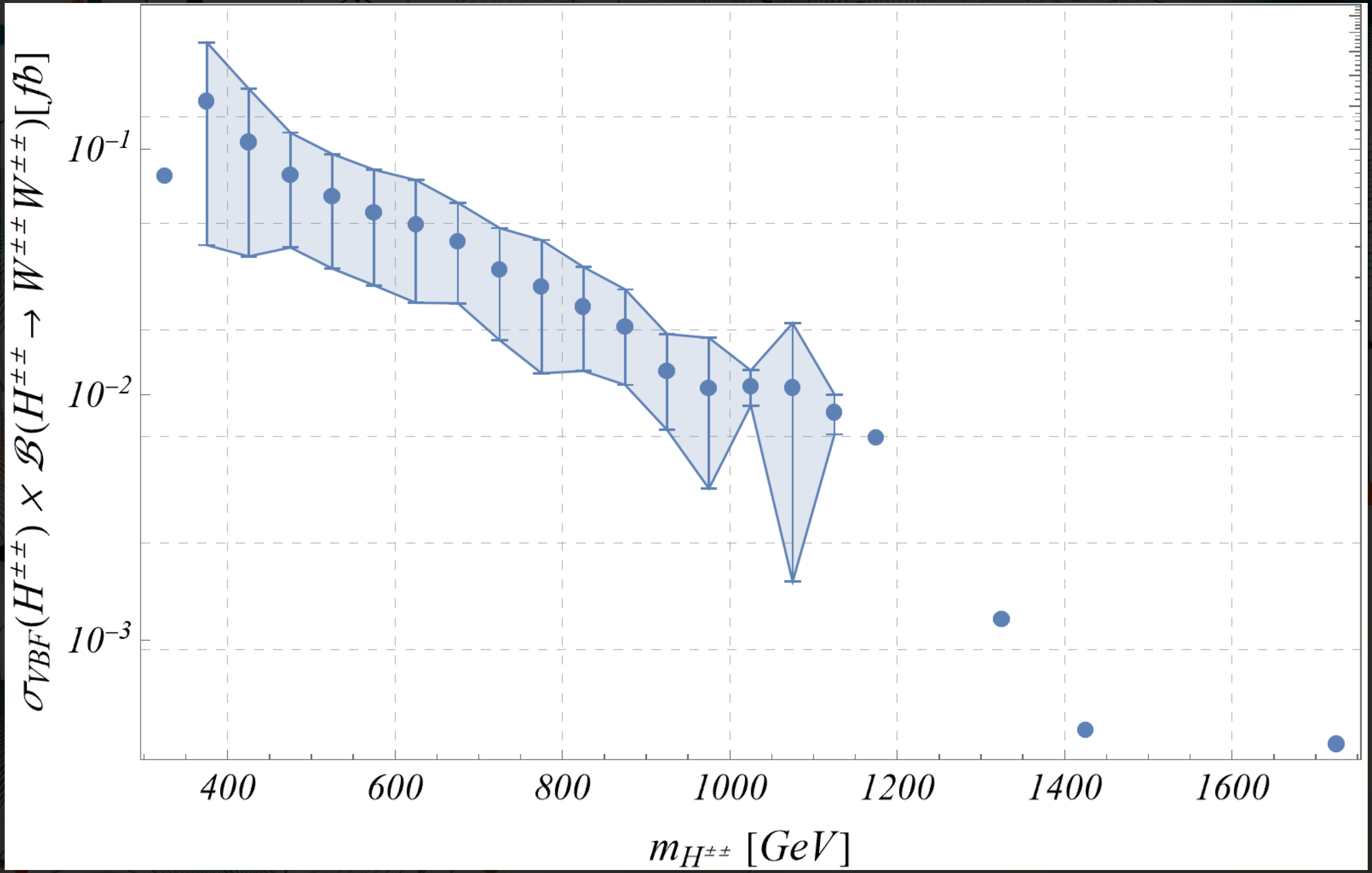
Extra Slide 2



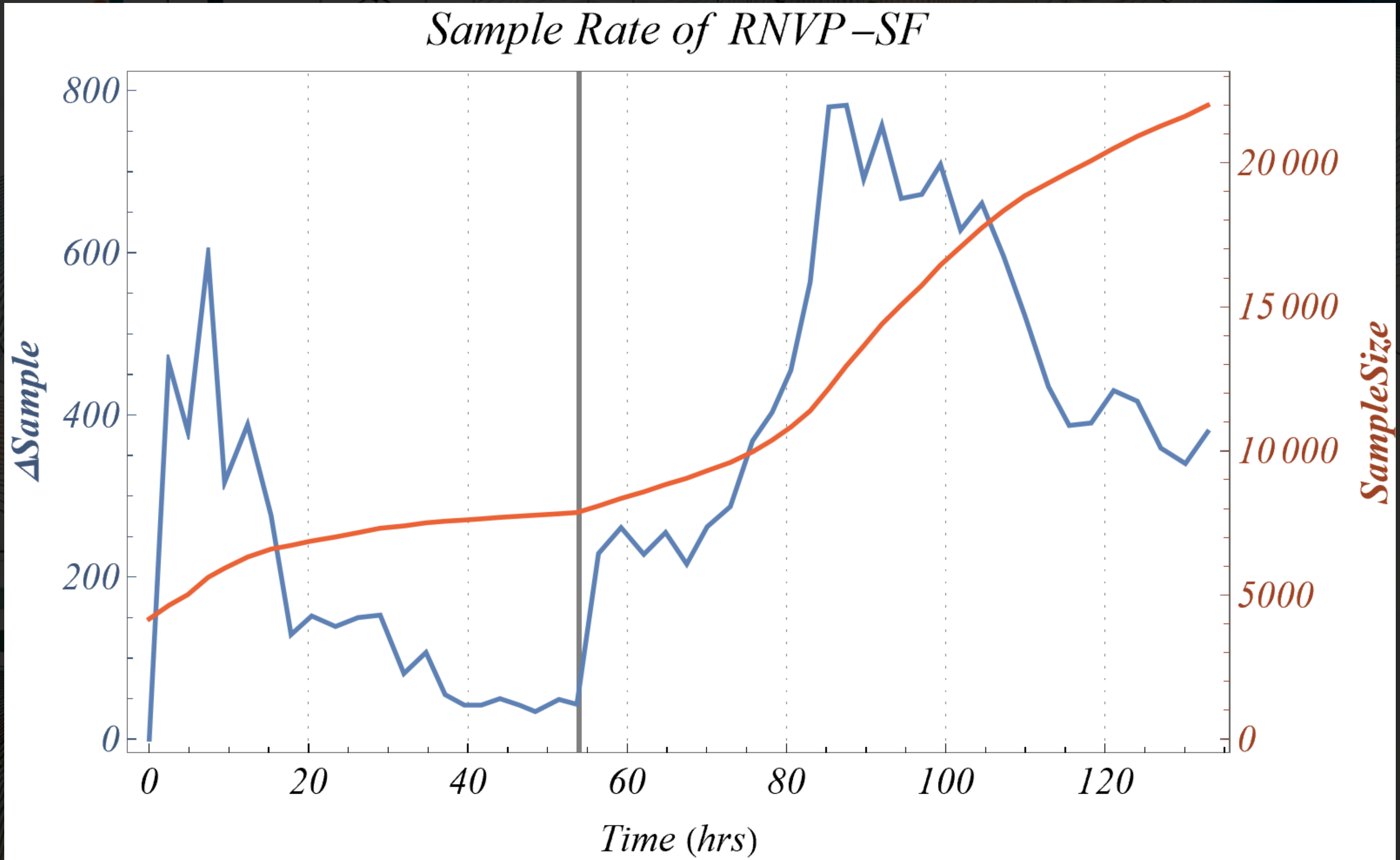
Extra Slide 3

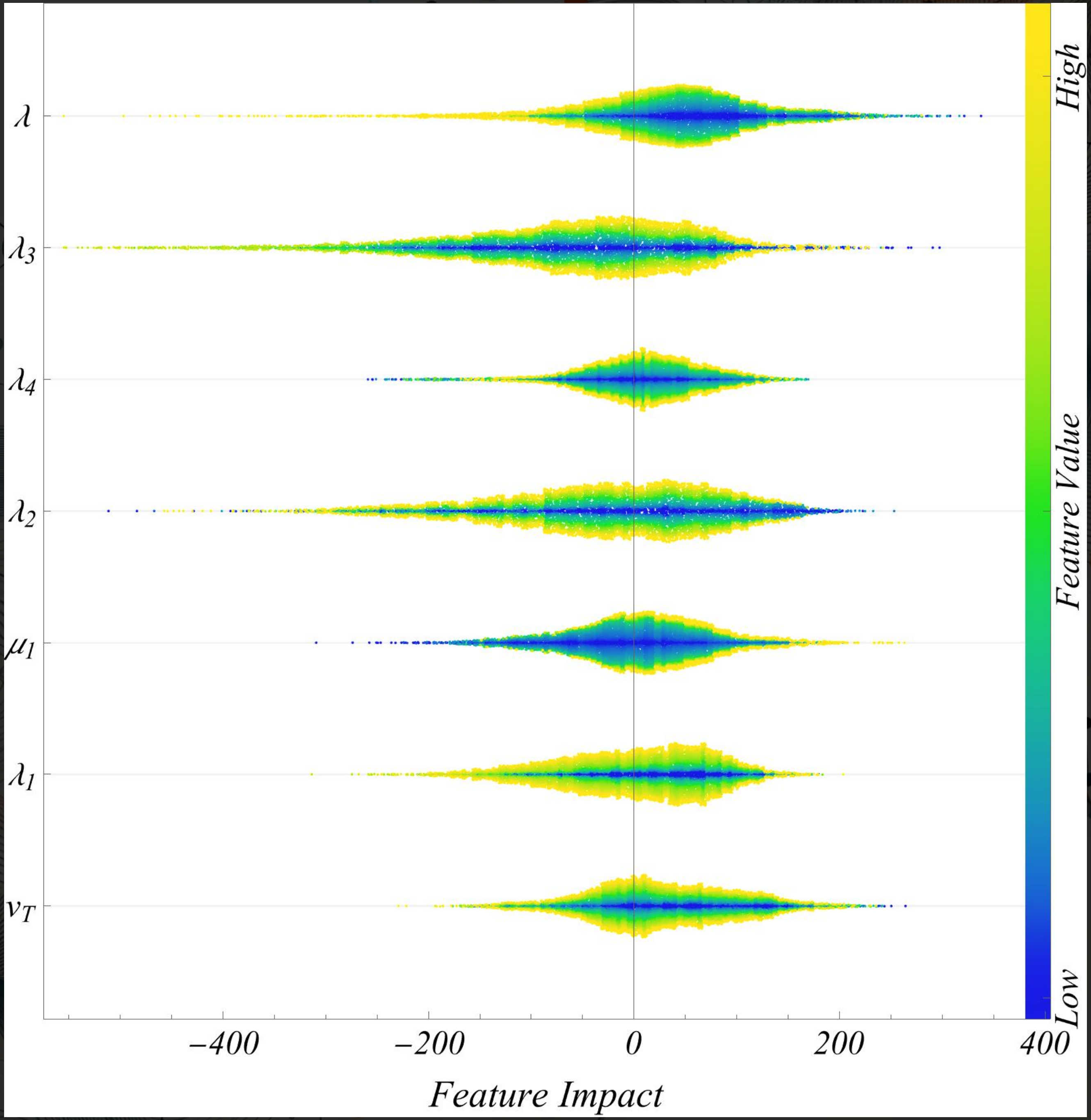


Extra Slide 4



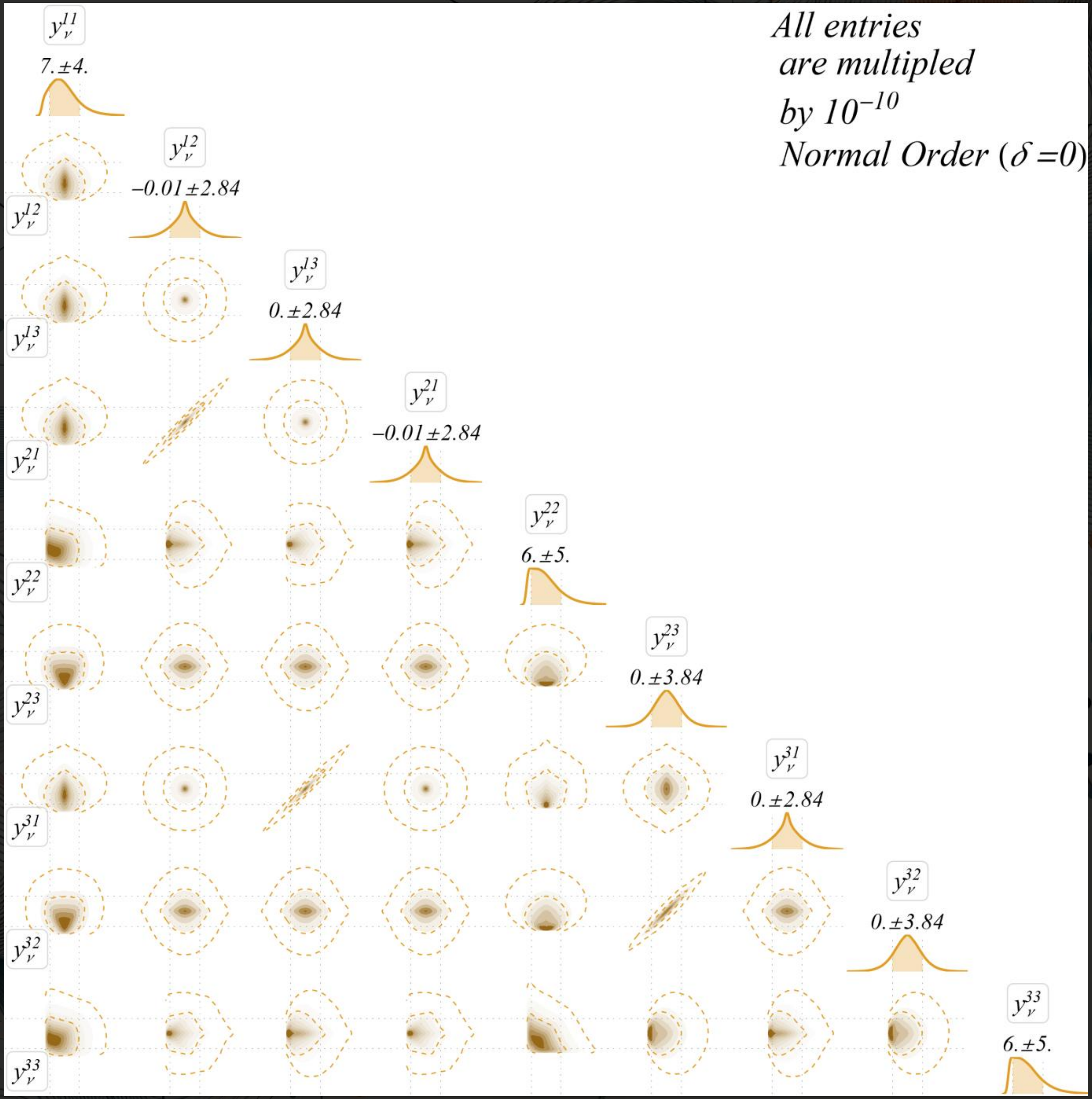
Extra Slide 5





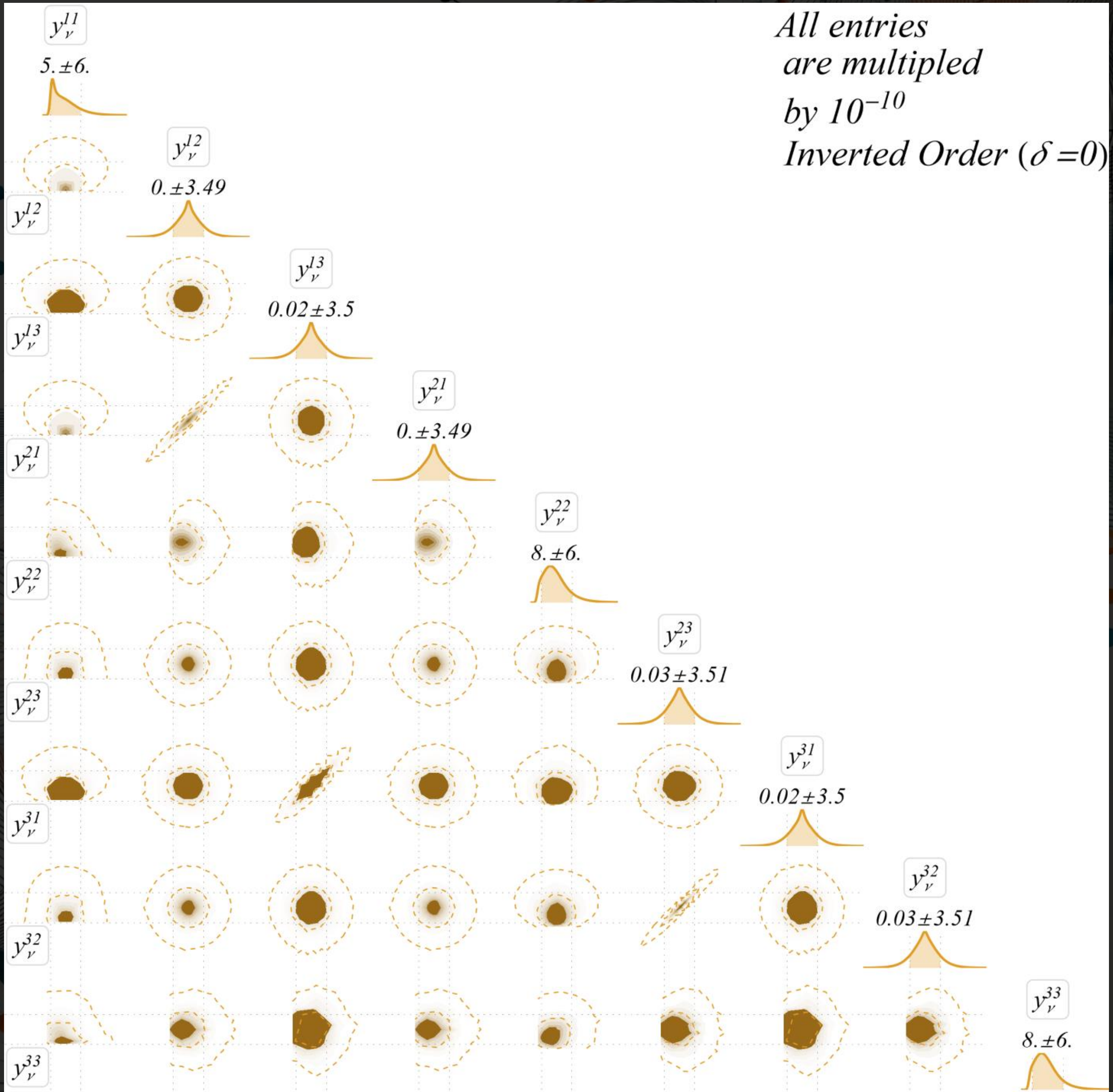
λ
λ_3
λ_4
λ_2
μ_1
λ_1
v_T
−400
−200
0
200
400
Feature Impact
High
Feature Value
Low

E6



All entries
are multipled
by 10^{-10}
Normal Order (δ =0)
y_ν^{11}
$7.\pm4.$
y_ν^{12}
-0.01 ± 2.84
y_ν^{13}
$0.\pm2.84$
y_ν^{21}
-0.01 ± 2.84
y_ν^{22}
$6.\pm5.$
y_ν^{23}
$0.\pm3.84$
y_ν^{31}
$0.\pm2.84$
y_ν^{32}
$0.\pm3.84$
y_ν^{33}
$6.\pm5.$

E7



All entries are multipled by 10^{-10}
Inverted Order ($\delta = 0$)
y_ν^{11}
$5. \pm 6.$
y_ν^{12}
$0. \pm 3.49$
y_ν^{13}
0.02 ± 3.5
y_ν^{21}
$0. \pm 3.49$
y_ν^{22}
$8. \pm 6.$
y_ν^{23}
0.03 ± 3.51
y_ν^{31}
0.02 ± 3.5
y_ν^{32}
0.03 ± 3.51
y_ν^{33}
$8. \pm 6.$

E8