DATA ANALYSIS IN HIGH-ENERGY PHYSICS USING MACHINE LEARNING METHODS

Matheus Guilherme Miotto
Jun Takahashi

Department of Cosmic Rays and Chronology
University of Campinas - UNICAMP

CFNS-SURGE Summer Workshop on the Physics of the Electron-Ion Collider

Motivation and objective

Motivation and objective

- In high-energy physics, **Monte Carlo (MC)** simulations are widely employed to validate selection criteria, enhance and calculate experimental efficiency of real data analysis.

Motivation and objective

- In high-energy physics, **Monte Carlo (MC)** simulations are widely employed to validate selection criteria, enhance and calculate experimental efficiency of real data analysis.
- However, complete event simulation using MC techniques is **computationally expensive**, requiring substantial **processing time**, and compatibility to data can be an issue that generates systematics uncertainties.

Motivation and objective

- In high-energy physics, **Monte Carlo (MC)** simulations are widely employed to validate selection criteria, enhance and calculate experimental efficiency of real data analysis.
- However, complete event simulation using MC techniques is **computationally expensive**, requiring substantial **processing time**, and compatibility to data can be an issue that generates systematics uncertainties.
- Therefore, we aim to **break the cost** established by MC simulations by proposing the use of generative **Machine Learning (ML)** models for **synthetic data generation**.

Motivation and objective

- In high-energy physics, **Monte Carlo (MC)** simulations are widely employed to validate selection criteria, enhance and calculate experimental efficiency of real data analysis.
- However, complete event simulation using MC techniques is **computationally expensive**, requiring substantial **processing time**, and compatibility to data can be an issue that generates systematics uncertainties.
- Therefore, we aim to **break the cost** established by MC simulations by proposing the use of generative **Machine Learning (ML)** models for **synthetic data generation**.
- We seek to optimize the performance and statistics of standard analyses of high-energy physics by synthesizing secondary decay particles.

What exactly is Machine Learning?

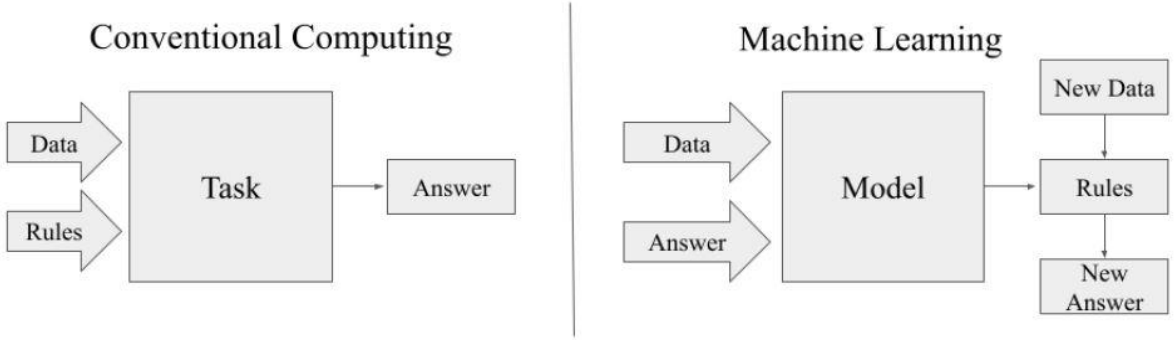
What exactly is Machine Learning?

- Definition: “A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, **improves with experience** E.” - T. Mitchell

T. M. Mitchell. “*Machine Learning*”. McGraw-Hill international editions - computer science series. McGraw-Hill Education, 1997.

What exactly is Machine Learning?

- Definition: “A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, **improves with experience** E.” - T. Mitchell



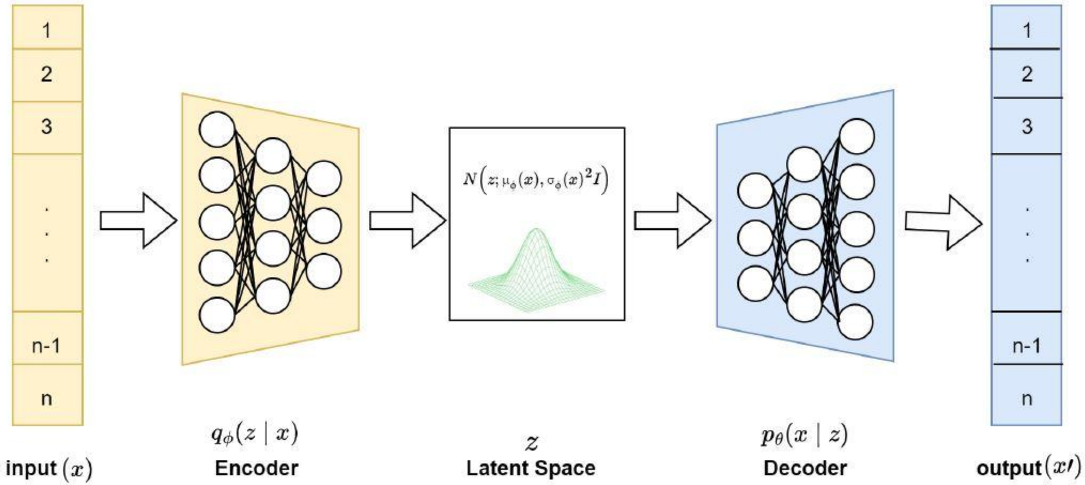
T. M. Mitchell. “*Machine Learning*”. McGraw-Hill international editions - computer science series. McGraw-Hill Education, 1997.

What can we do with ML?

What can we do with ML? Generative Models!

What can we do with ML? Generative Models!

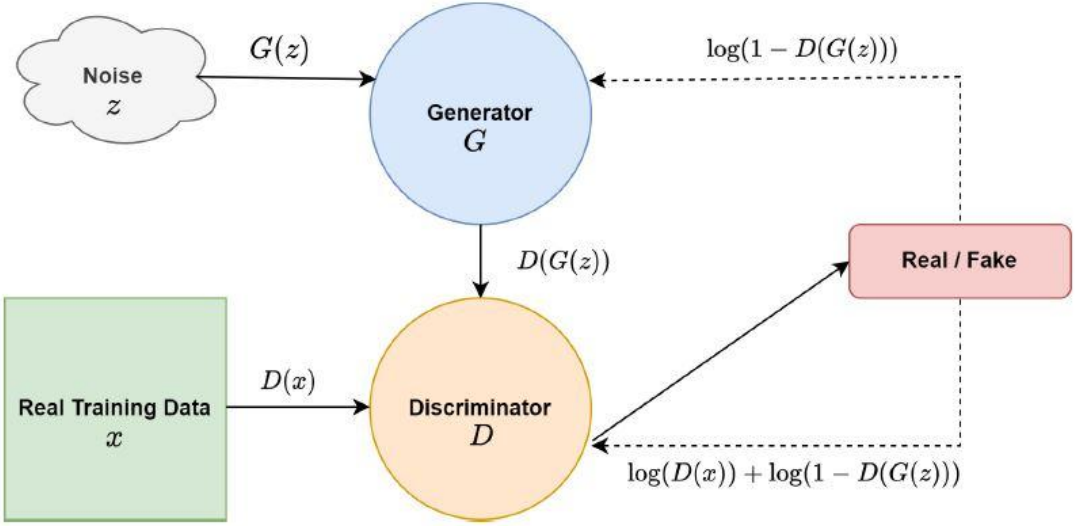
- Variational Autoencoders (Vae):



AMMARA, D.; DING, J.; TUTSCHKU, T. "*Synthetic Data Generation in Cybersecurity: A Comparative Analysis*". 2024. arXiv: 2410.16326 [cs.CR].

What can we do with ML? Generative Models!

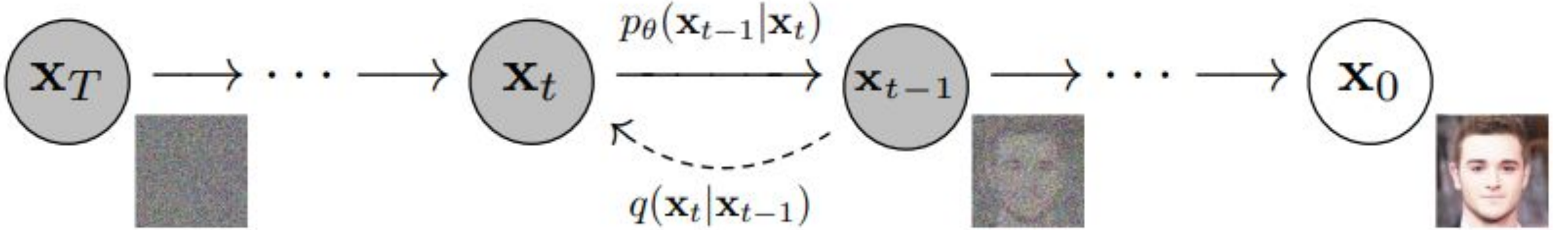
- Conditional Tabular Generative Adversarial Network (Ctgan):



AMMARA, D.; DING, J.; TUTSCHKU, T. "*Synthetic Data Generation in Cybersecurity: A Comparative Analysis*". 2024. arXiv: 2410.16326 [cs.CR].

What can we do with ML? Generative Models!

- Tabular Denoising Diffusion Probabilistic Model (TabDDPM):



HO, J; et al. "*Denoising Diffusion Probabilistic Models*". 2020. arXiv:2006.11239 [cs.LG]

More in details: model’s architecture

More in details: model's architecture

- The Ctgan model was configured with 4 fully connected layers with 256 neurons each for both the generator and discriminator. The embedding dimension was set to be equal to 32, with training performed over 500 epochs and 5 discriminator steps per generator update. Additional parameters were either optimized or kept consistent with the original implementation.

More in details: model's architecture

- The Ctgan model was configured with 4 fully connected layers with 256 neurons each for both the generator and discriminator. The embedding dimension was set to be equal to 32, with training performed over 500 epochs and 5 discriminator steps per generator update. Additional parameters were either optimized or kept consistent with the original implementation.
- The TabDDPM model employed 6 layers with 1024 neurons each. A cosine noise scheduler was applied with the number of diffusion timesteps to 1000 and a learning rate of 0,003. The remaining parameters followed the original implementation.

More in details: model’s architecture

- The Ctgan model was configured with 4 fully connected layers with 256 neurons each for both the generator and discriminator. The embedding dimension was set to be equal to 32, with training performed over 500 epochs and 5 discriminator steps per generator update. Additional parameters were either optimized or kept consistent with the original implementation.
- The TabDDPM model employed 6 layers with 1024 neurons each. A cosine noise scheduler was applied with the number of diffusion timesteps to 1000 and a learning rate of 0,003. The remaining parameters followed the original implementation.
- The Vae model architecture was optimized using the Optuna Python library for efficient hyperparameter tuning.

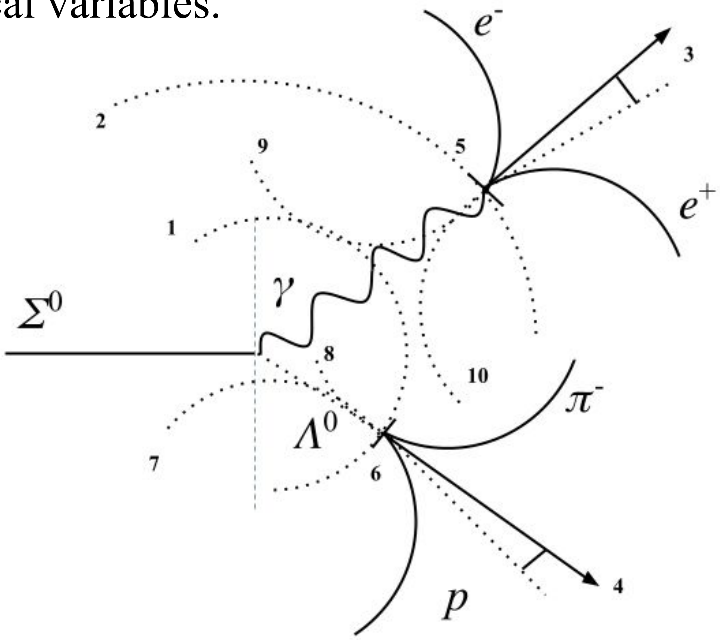
Methodology

Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions

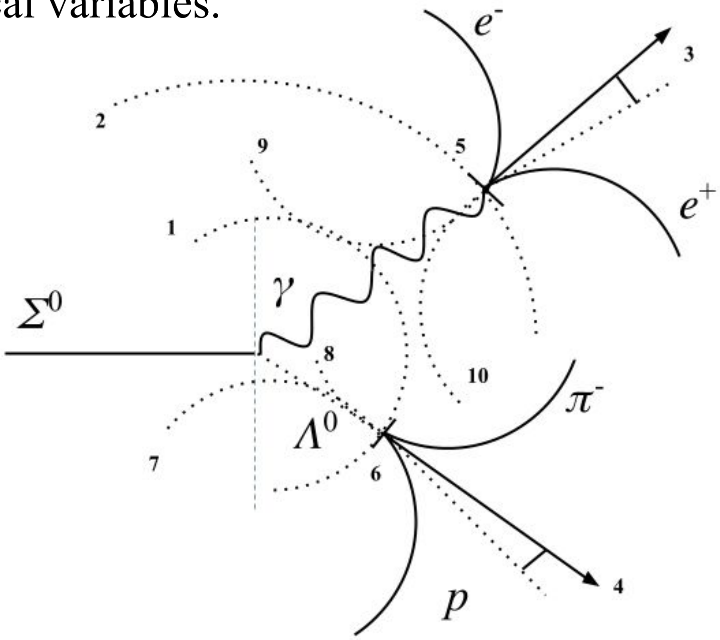
Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions
- 16 features including kinematics and topological variables.



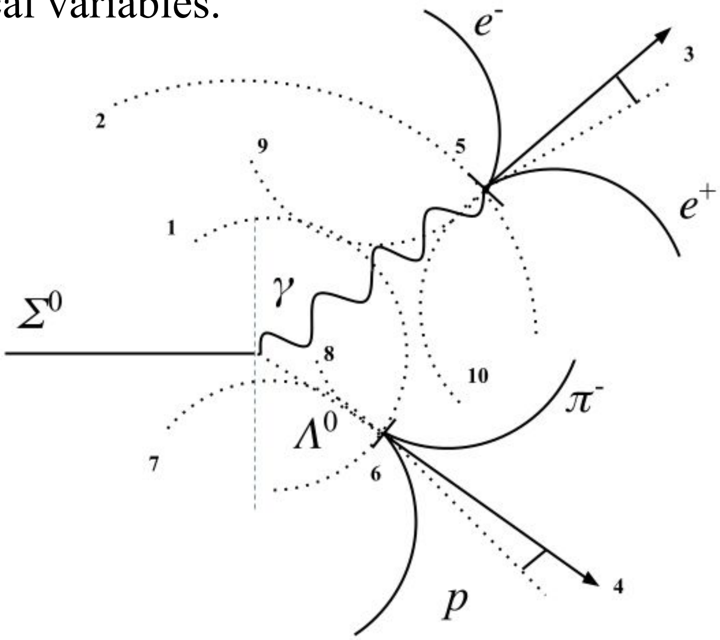
Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions
- 16 features including kinematics and topological variables.
- We're interested in Σ^0 topology, for example.



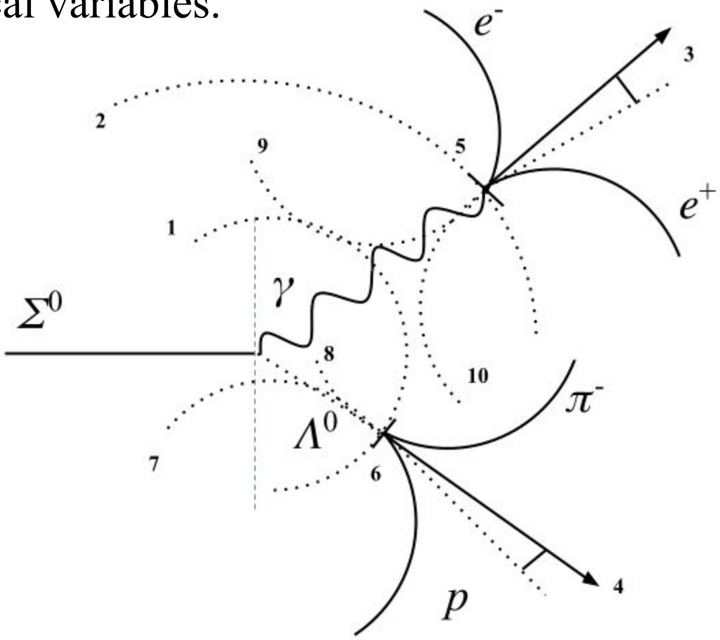
Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions
- 16 features including kinematics and topological variables.
- We're interested in Σ^0 topology, for example.
- But work exclusively with background data.



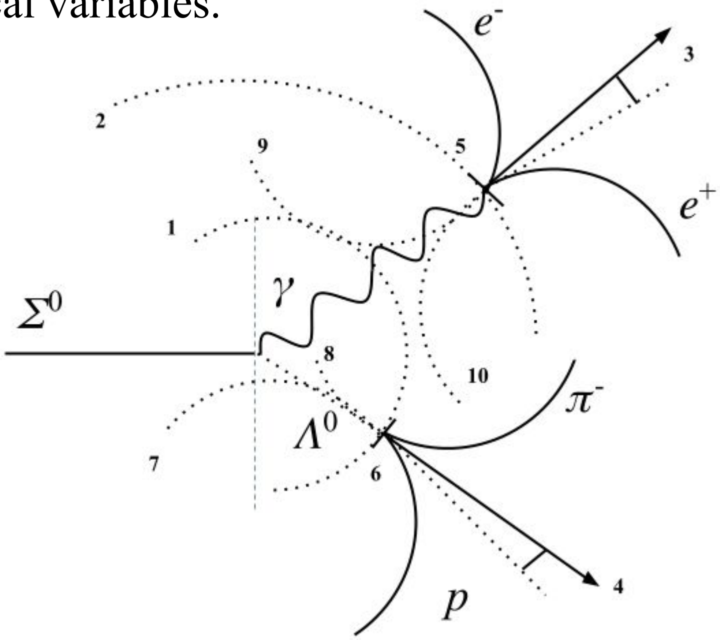
Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions
- 16 features including kinematics and topological variables.
- We're interested in Σ^0 topology, for example.
- But work exclusively with background data.
- Improve model parameters.



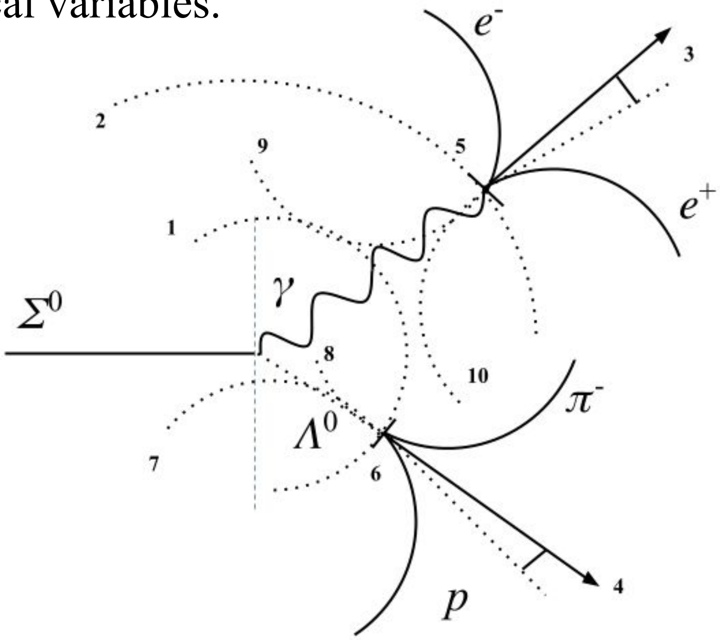
Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions
- 16 features including kinematics and topological variables.
- We're interested in Σ^0 topology, for example.
- But work exclusively with background data.
- Improve model parameters.
- Train your model.

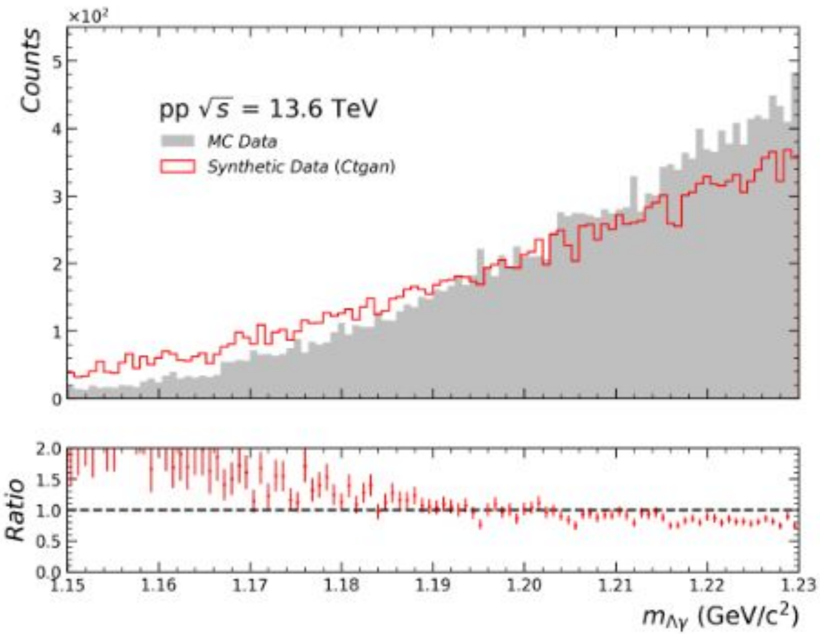


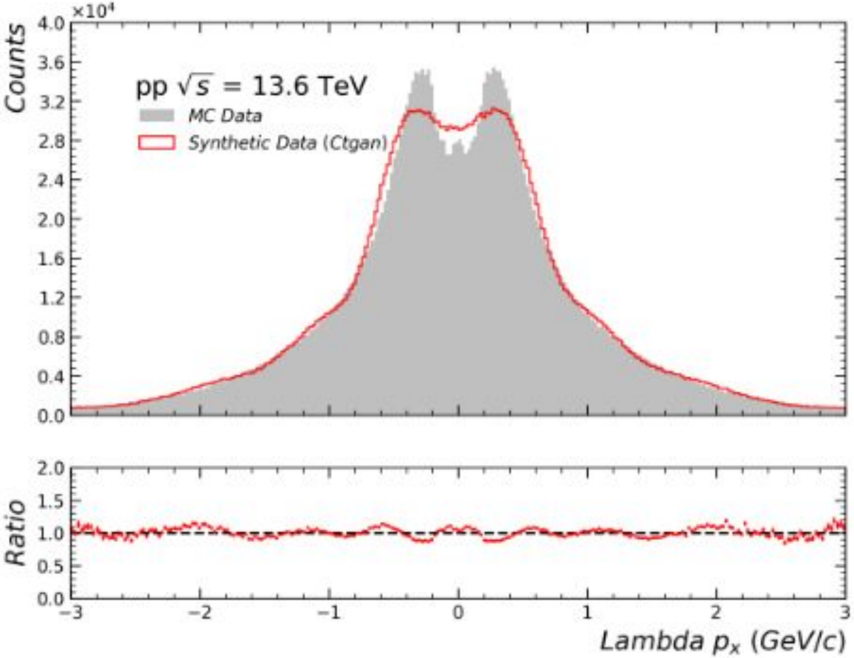
Methodology

- 2.5M-event Monte Carlo dataset of proton-proton collisions
- 16 features including kinematics and topological variables.
- We're interested in Σ^0 topology, for example.
- But work exclusively with background data.
- Improve model parameters.
- Train your model.
- Evaluate the results.

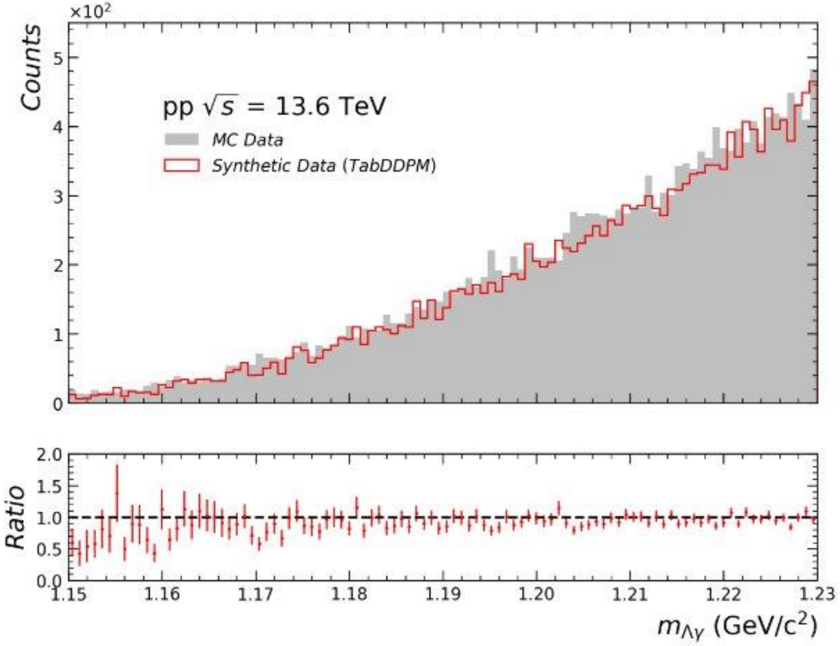


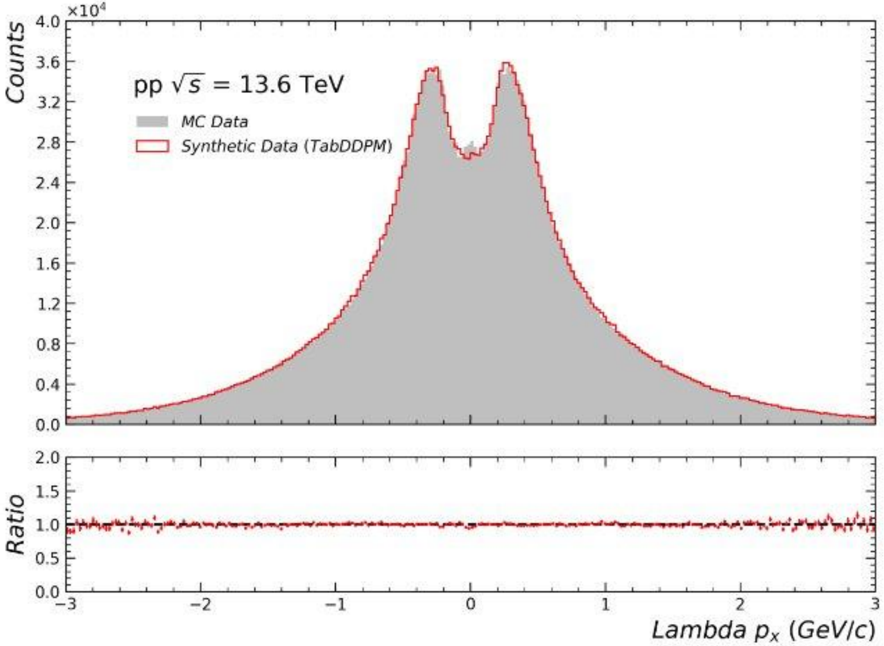
Results - Ctgan Model





Results - TabDDPM Model





- **Early results! Only momentum coordinates of particles were used in training process!**

Summary

Summary

- We have explored and implemented generative models, including Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs) and Diffusion Models.

Summary

- We have explored and implemented generative models, including Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs) and Diffusion Models.
- We have shown the ability of most of these models to learn complex, unstructured data while preserving first-order correlations and feature distributions.

Summary

- We have explored and implemented generative models, including Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs) and Diffusion Models.
- We have shown the ability of most of these models to learn complex, unstructured data while preserving first-order correlations and feature distributions.
- Therefore, we have demonstrated that CTGAN and TabDDPM can successfully generate millions of synthetic data points within seconds, thereby reducing generation time from hours to even days.

Next Steps

Next Steps

- Engage in a PhD project.

THANK YOU!

ACKNOWLEDGMENTS