Data driven background estimation in HEP using Generative Adversarial Networks

CHEP 2023 - May 9th, 2023

Victor Lohezic (victor.lohezic@cern.ch)
Fabrice Couderc, Julie Malclès, Özgür Şahin
IRFU - CEA Saclay

Eur. Phys. J. C 83, 256 (2023)
https://doi.org/10.1140/epjc/s10052-023-11347-8

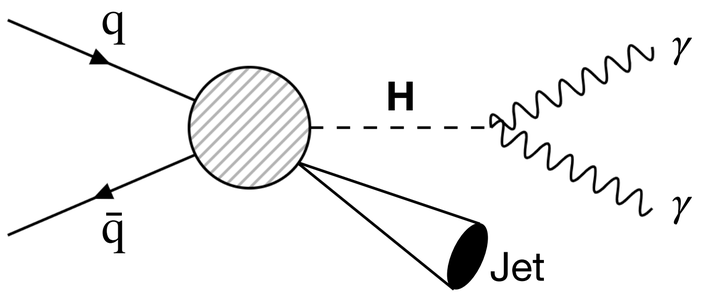
Introduction

GAN based data-driven technique to estimate background processes with a misidentified object in collider events. We will showcase this technique for the γ + Jets background process of the H → $\gamma\gamma$ analysis.

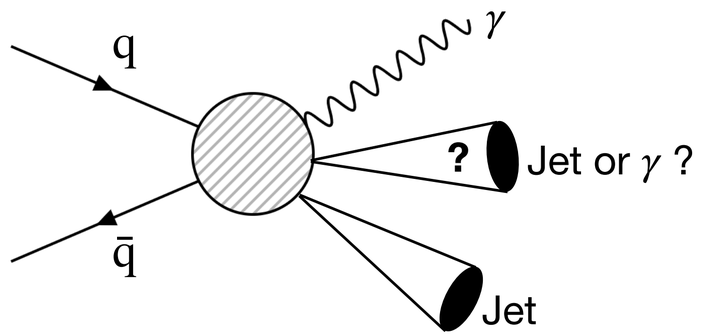
In the H → $\gamma\gamma$ analysis, dominant backgrounds are : $\gamma\gamma$ + Jets, γ + Jets, Multi Jets (MJ)

- The agreement between Data and Monte Carlo (MC) simulated samples for γ + Jets and MJ is not satisfying and the statistics is too low for the training of subsequent discriminants.

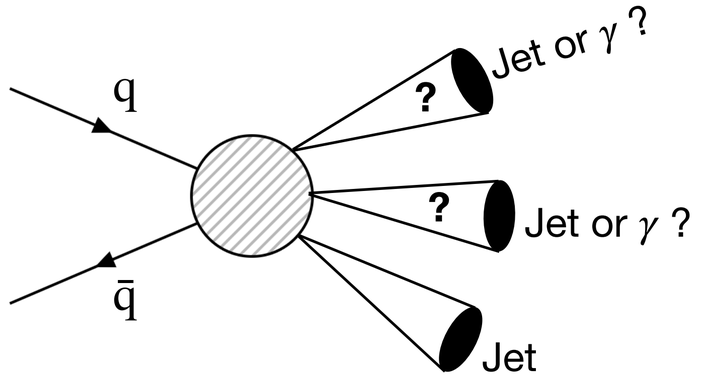
Example of H → $\gamma\gamma$ event (Signal)



Example of γ + Jets event (Bkg)



Example of MJ event (Bkg)



➡ What if we use data directly to describe those samples ?

- We would like to improve the data driven approach used in the previously published analyses using this technique.

Overview

I. **A data driven estimation of the background**

II. **Training a GAN**
   a. Generative Adversarial Network (GAN)
   b. Evaluation procedure

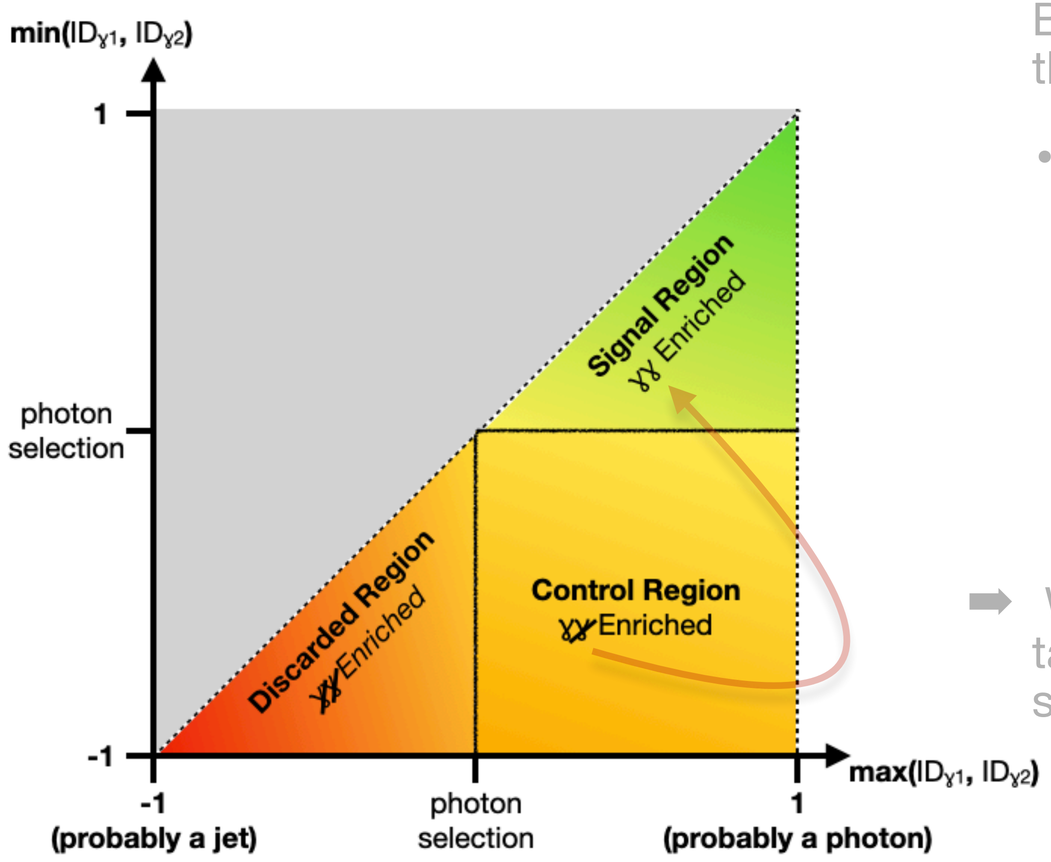
III. **Generating a full object (misidentified photon)**
   a. Optimization of training
   b. Results

IV. **Conclusions and outlooks**

I. A data driven estimation of the background

In an event each photon is given a score (photon ID) representing its likelihood to be a photon. Control region in data based on photon ID is used to replace MC γ + Jets / MJ samples (better agreement, more statistics).

➡ Need one photon with very low photon ID : probably a misidentified photon γ (as opposed to a prompt photon γ



➡ Many analyses already use data driven background estimation.
By reverting the cut on the min photon ID, one needs either to get rid of the photon ID variable or to generate a new min photon ID !

- This procedure was used in published analysis from CMS experiment [1], new ID was generated by :
  1. Deriving a 1D probability density function (PDF) from the misidentified photon ID distribution
  2. Generating a random min photonID following this PDF, in the signal region but below the max photonID
  3. However correlations are not preserved

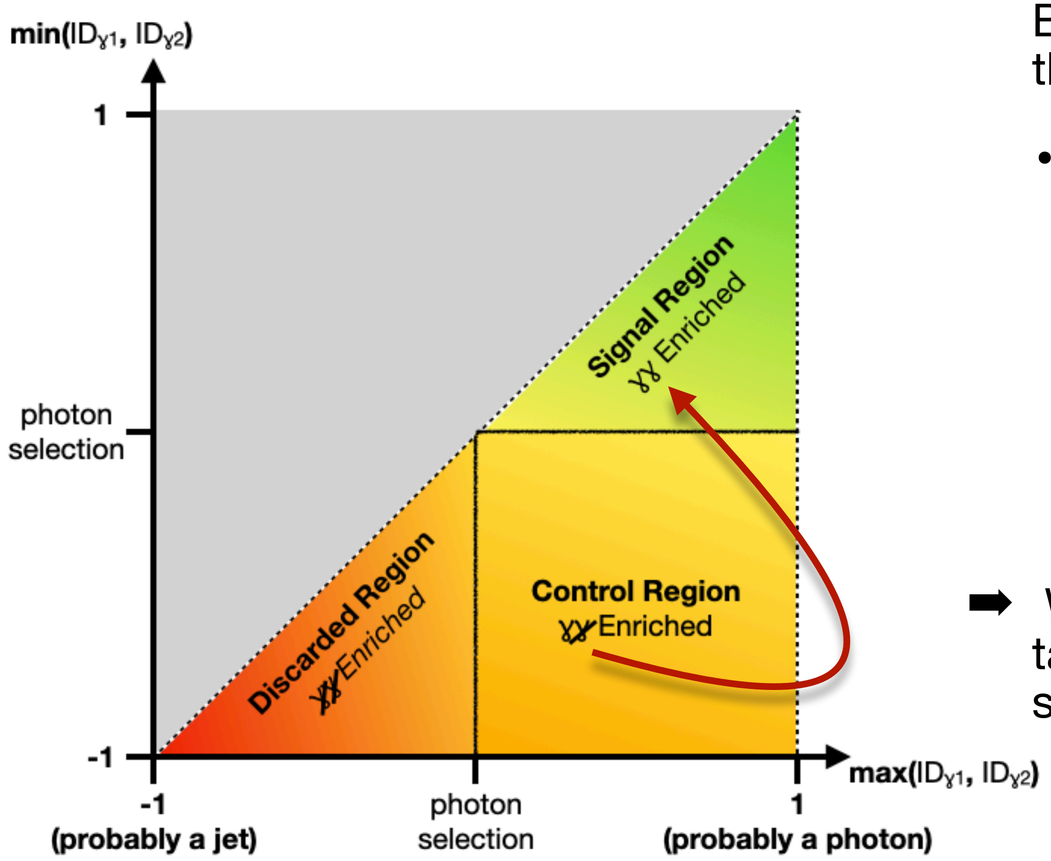
➡ We propose a new method to generate a suitable photon (not only ID) taking into account these correlations thanks to ML and more specifically GAN (Generative Adversarial Networks)

[1] Measurements of ttH production and the CP structure of the Yukawa interaction between the Higgs boson and the top quark in the diphoton decay channel, CMS collaboration

I. A data driven estimation of the background

In an event each photon is given a score (photon ID) representing its likelihood to be a photon. Control region in data based on photon ID is used to replace MC γ + Jets / MJ samples (better agreement, more statistics).

➡ Need one photon with very low photon ID : probably a misidentified photon γ (as opposed to a prompt photon γ



➡ Many analyses already use data driven background estimation.
By reverting the cut on the min photon ID, one needs either to get rid of the photon ID variable or to generate a new min photon ID !

- This procedure was used in published analysis from CMS experiment [1], new ID was generated by :
  1. Deriving a 1D probability density function (PDF) from the misidentified photon ID distribution
  2. Generating a random min photonID following this PDF, in the signal region but below the max photonID
  3. However correlations are not preserved

➡ We propose a new method to generate a suitable photon (not only ID) taking into account these correlations thanks to ML and more specifically GAN (Generative Adversarial Networks)

[1] Measurements of ttH production and the CP structure of the Yukawa interaction between the Higgs boson and the top quark in the diphoton decay channel, CMS collaboration

Overview

I. A data driven estimation of the background

II. **Training a GAN**
   a. Generative Adversarial Network (GAN)
   b. Evaluation procedure

III. Generating a full object (misidentified photon)
   a. Optimization of training
   b. Results

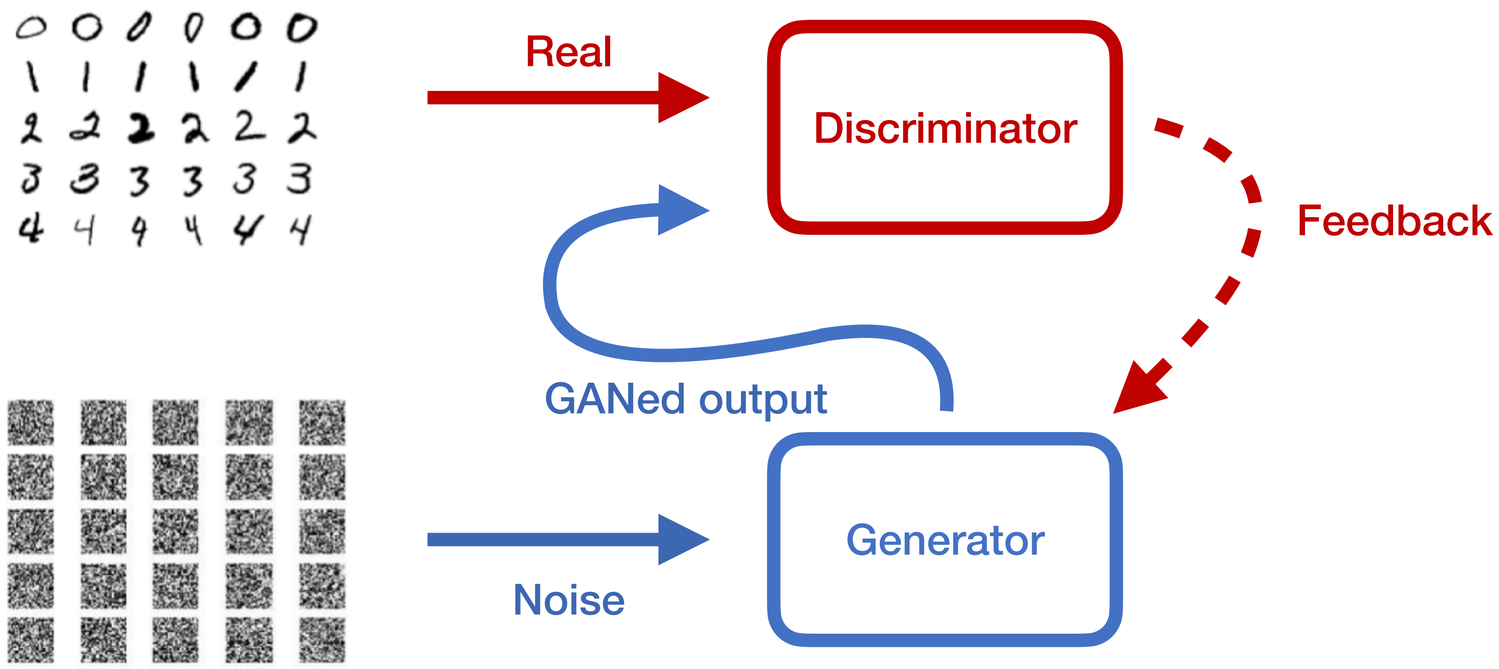
IV. Conclusions and outlooks

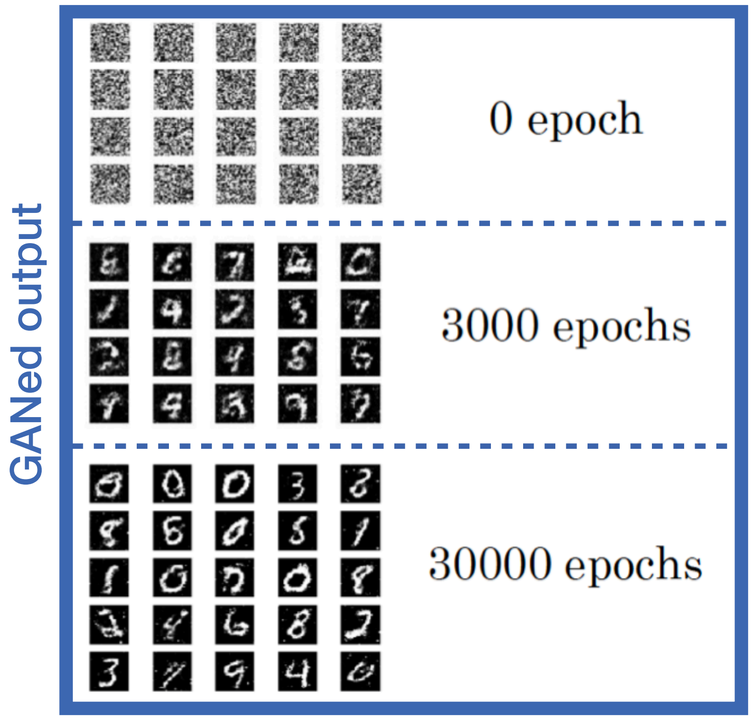
II. Training a GAN

II.a - Generative Adversarial Networks (GANs)

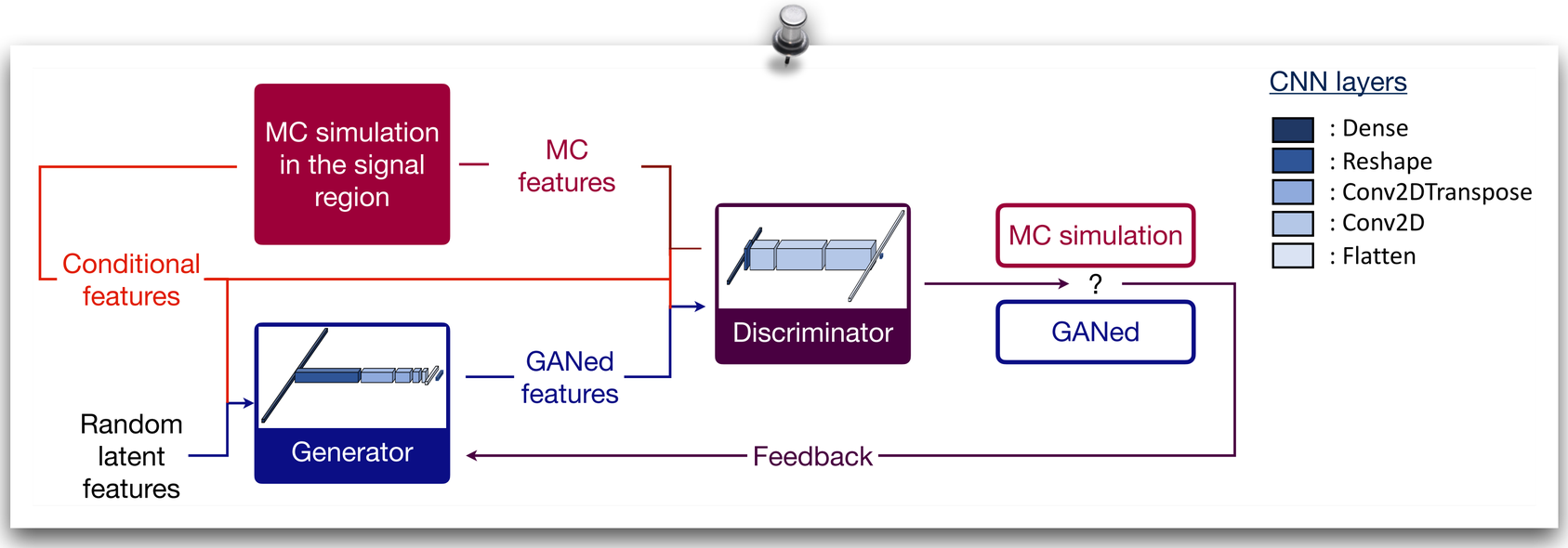
Would it be possible to create an algorithm capable of learning underlying correlations and capable of generating a sample statistically independent from the training sample ?

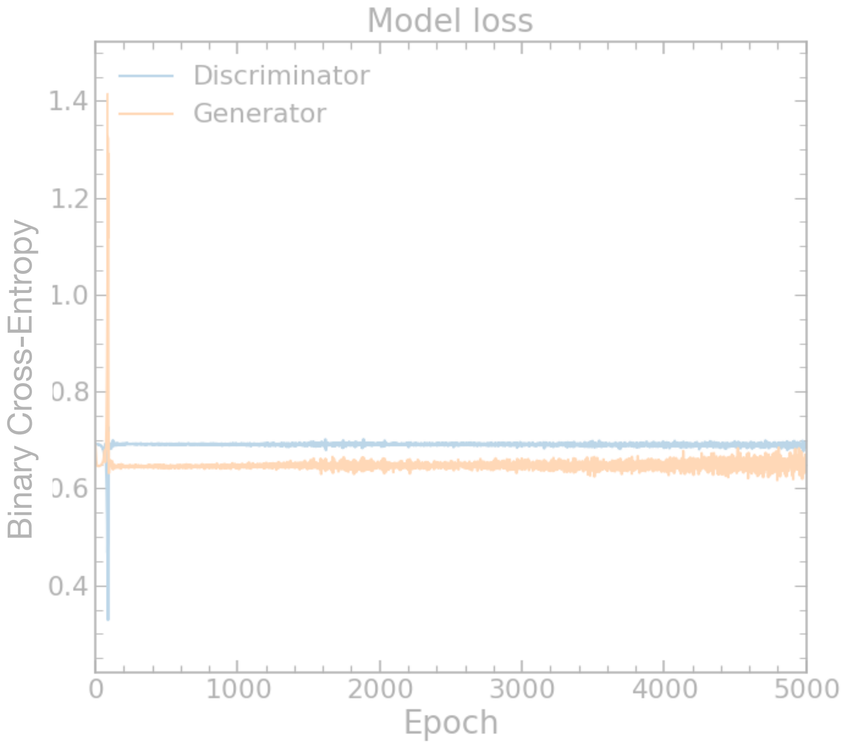
- *Goodfellow et al.* suggested a model consisting of two neural networks competing against each other :
  - the "discriminator" sorts samples between real and generated ones - *i.e.* discriminates fakes
  - the "generator" tries to produce samples which will fool the discriminator





We use a conditional version of a GAN and we train on the misidentified photon (ID, p_T, η, ϕ) :

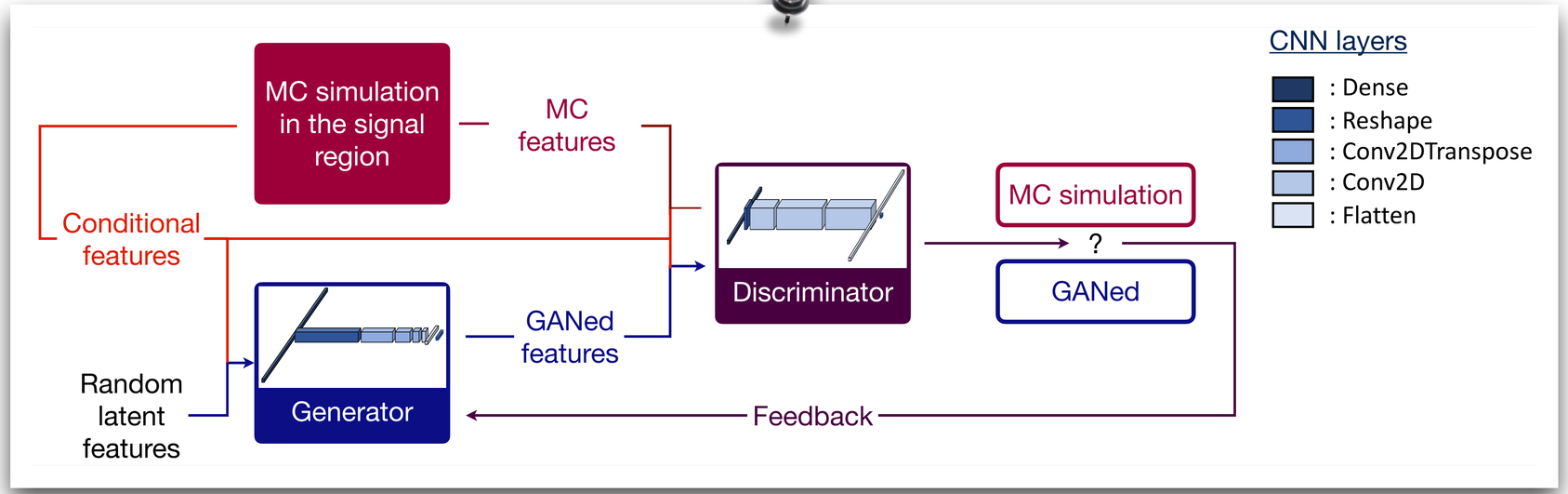


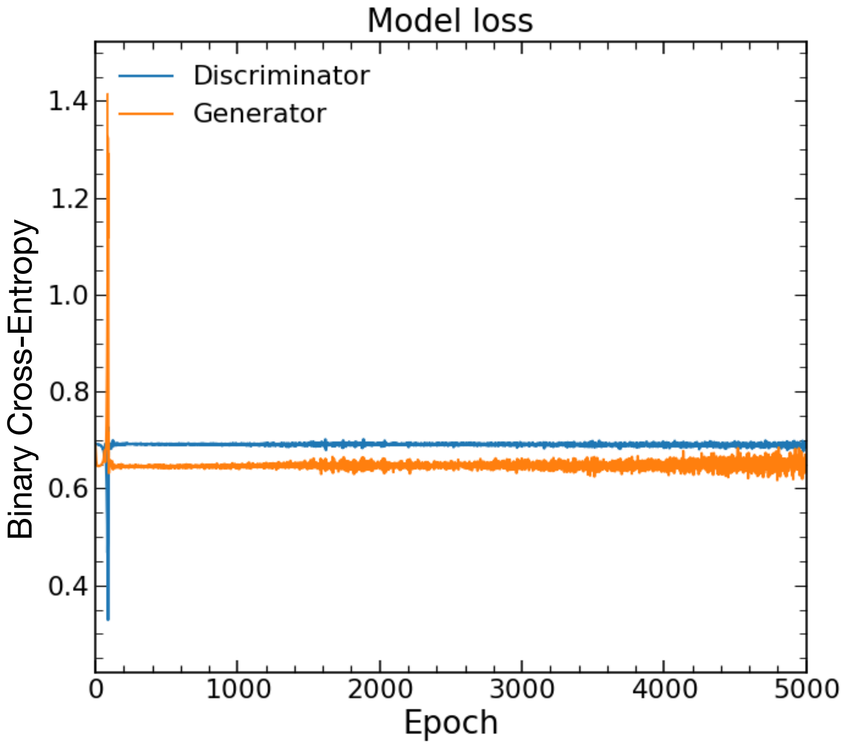


- Usually, monitoring the loss of a neural network is enough to evaluate its performance. It is not the case for GAN where both networks need to perform well against the other so their loss stays flat.

➡ We need to set up a more elaborate evaluation procedure

We use a conditional version of a GAN and we train on the misidentified photon (ID, p_T, η, ϕ) :





- Usually, monitoring the loss of a neural network is enough to evaluate its performance. It is not the case for GAN where both networks need to perform well against the other so their loss stays flat.

➡ We need to set up a more elaborate evaluation procedure

II.b - Evaluation procedure

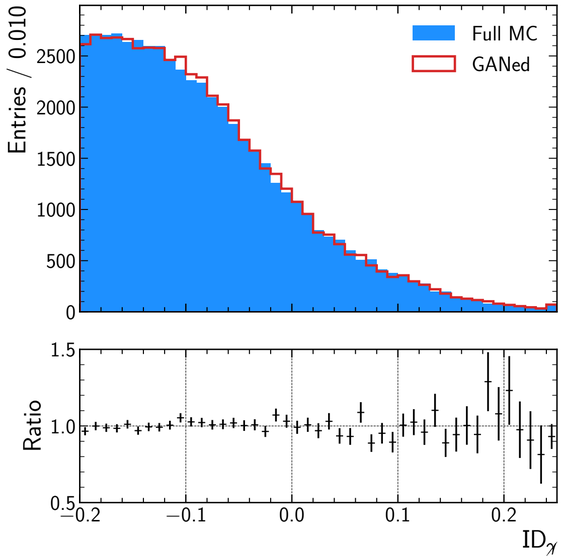
- To evaluate the performance of a given model, we rely on different metrics computed for each training epoch on the training sample and on a validation sample :

χ^2 metric :

$$\chi^2 = \sum_{k=1}^{\#\text{bins}} \frac{(n_k - N_k)^2}{N_k^2}$$

n_k : sum of the weights of **GANed events** in bin k

N_k : sum of the weights of **original events** in bin k



Log Likelihood metric :

$$-2\ln(\Lambda) = -2\sum_{k=1}^{\#\text{bins}} N_k \cdot \log(p_k), \quad p_k = \frac{n_k}{\sum n}$$

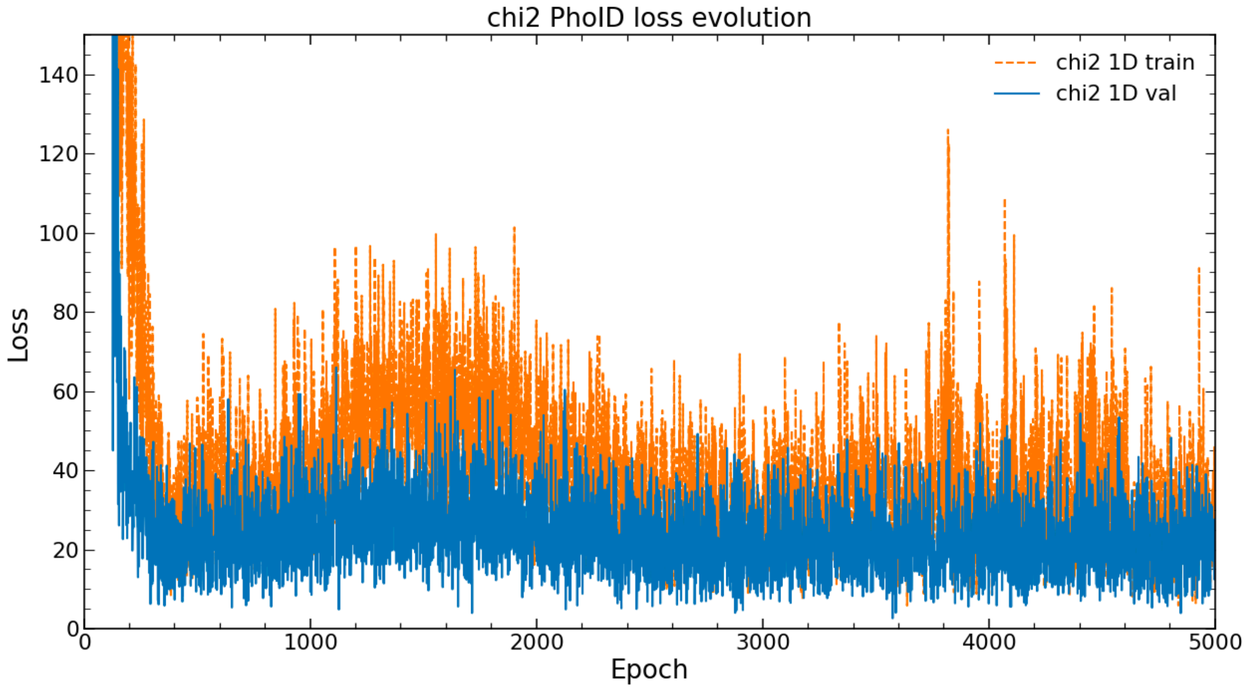
n_k : sum of the weights of **GANed events** in bin k

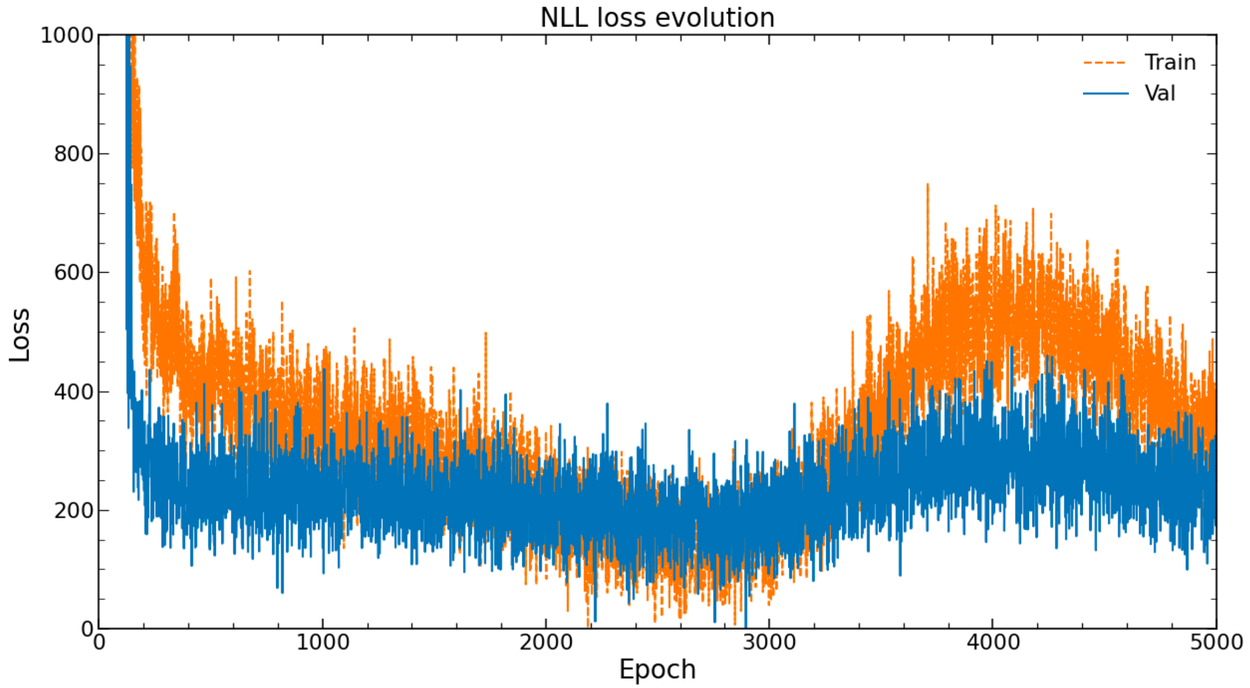
N_k : sum of the weights of **original events** in bin k

For the NLL we histogram our events in 4D :

- transverse momentum of misidentified photon p_{T_γ}
- pseudorapidity of misidentified photon η_γ
- p_T of diphoton pair over its mass $\frac{p_{T_{\gamma\gamma}}}{m_{\gamma\gamma}}$
- ID of misidentified photon ID_γ

Takes into account correlations by construction

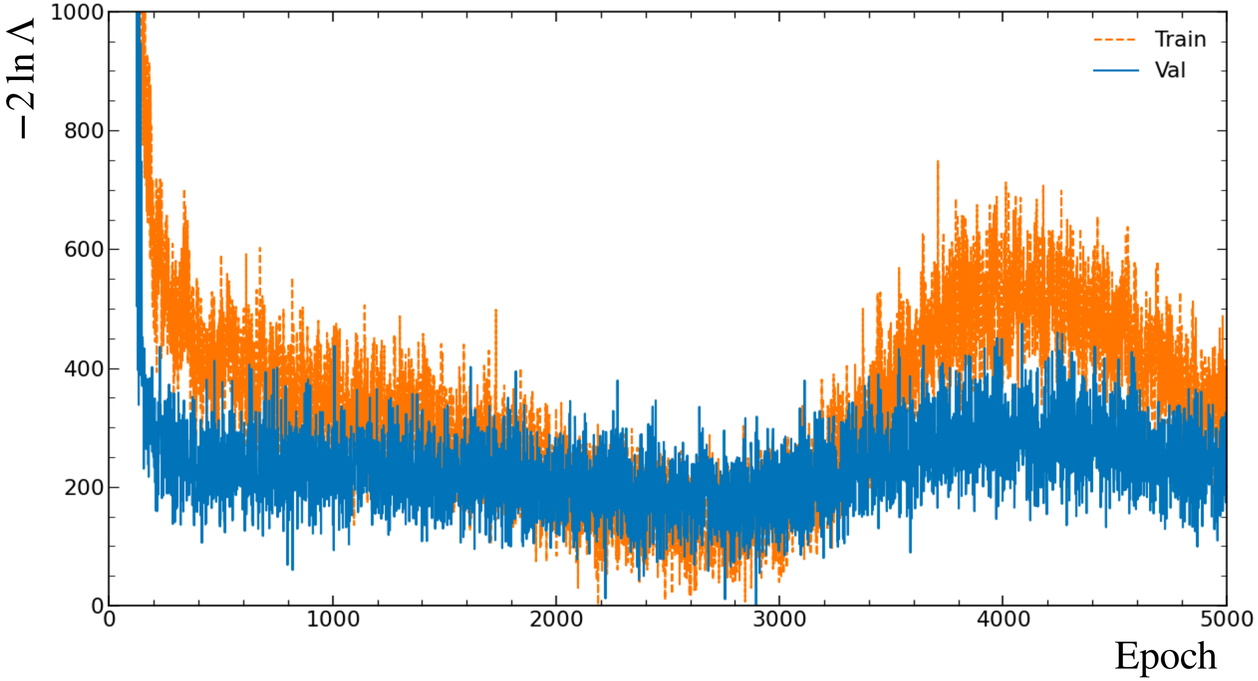




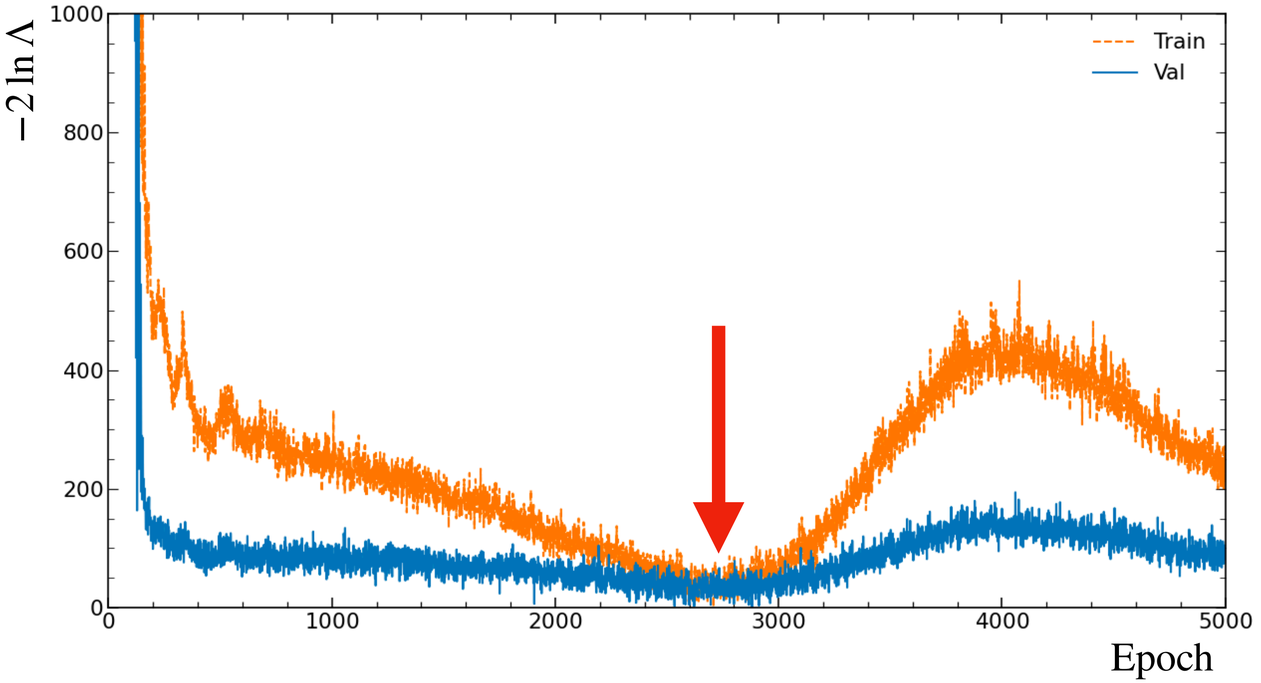
➡ Metrics are fluctuating a lot !

- p_k (see slide 9) estimation is statistically limited creating fluctuations in the NLL. These fluctuations can be reduced by increasing the number of generation per event :

1 generation per event



10 generations per event



- Seeing how the fluctuations decrease, we decide to go to 100 generation per event

- Then we can find epochs where the model is reaching minima for these metrics and take a closer look at its performance

Overview

I. A data driven estimation of the background

II. Training a GAN
   a. Generative Adversarial Network (GAN)
   b. Evaluation procedure

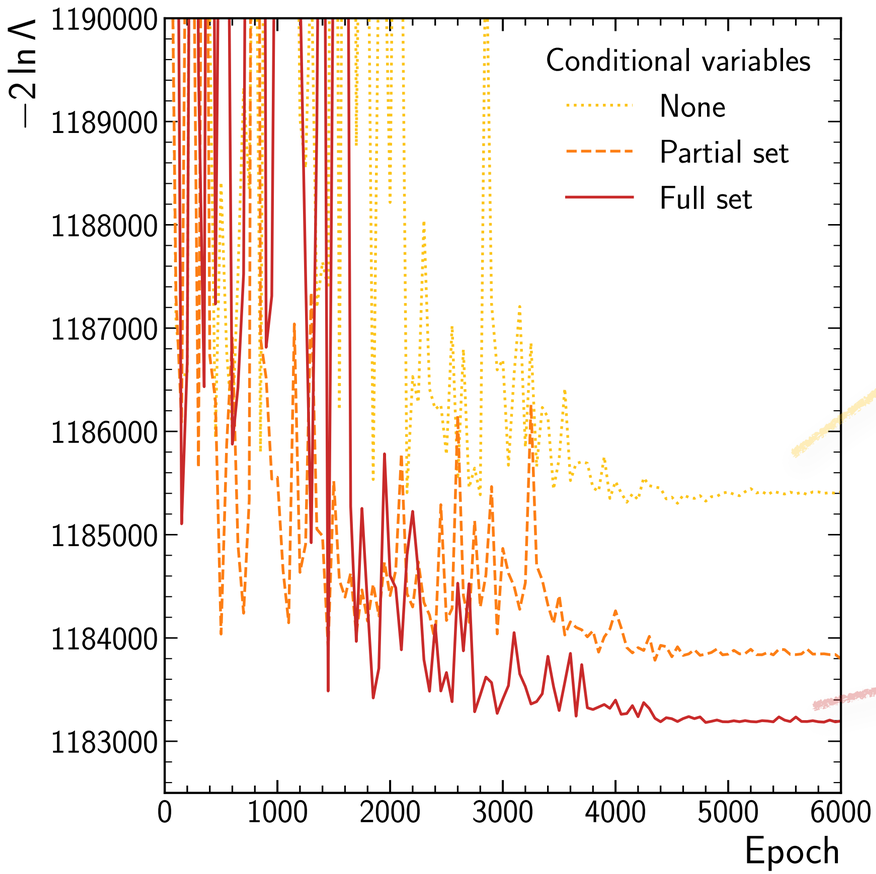
III. Generating a full object (misidentified photon)
   a. Optimization of training
   b. Results

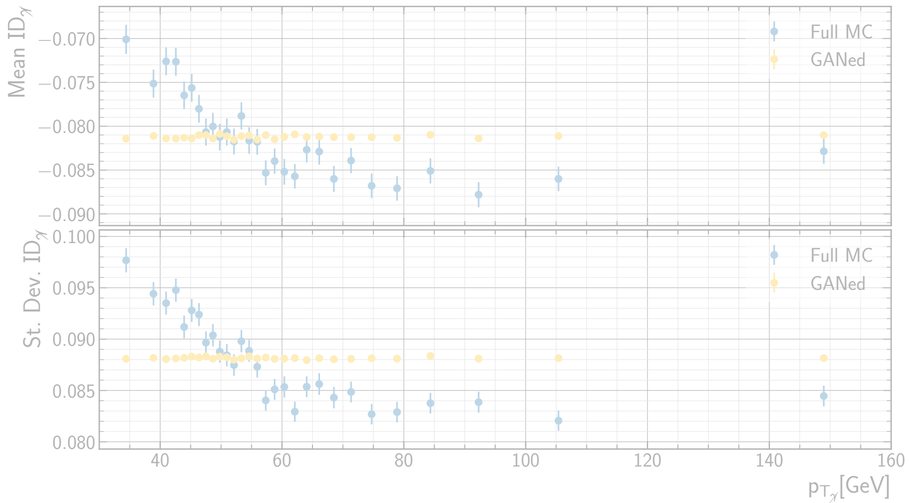
IV. Conclusions and outlooks

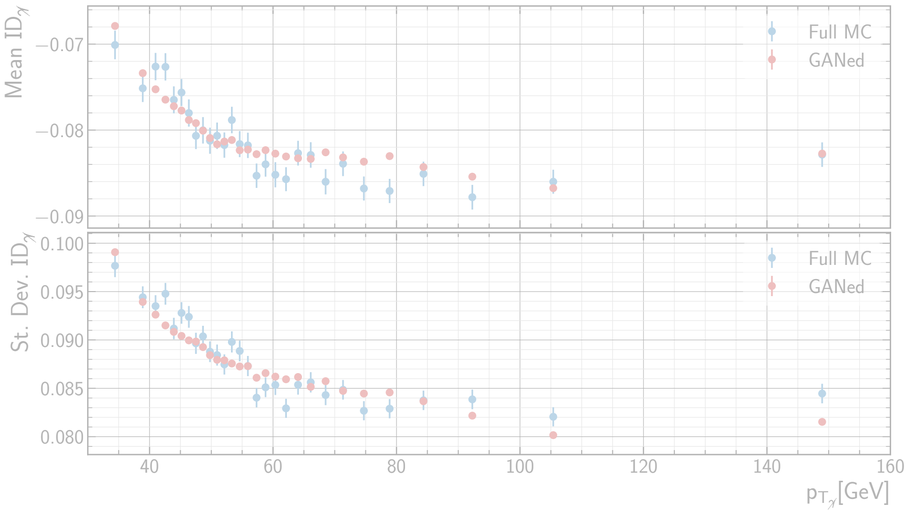
III. Generating a full object (misidentified photon)

III.a - Optimization of training

- Trying to find the correct set of observables to train our GAN, we can clearly see how hiding information from the network affects its ability to learn correlations



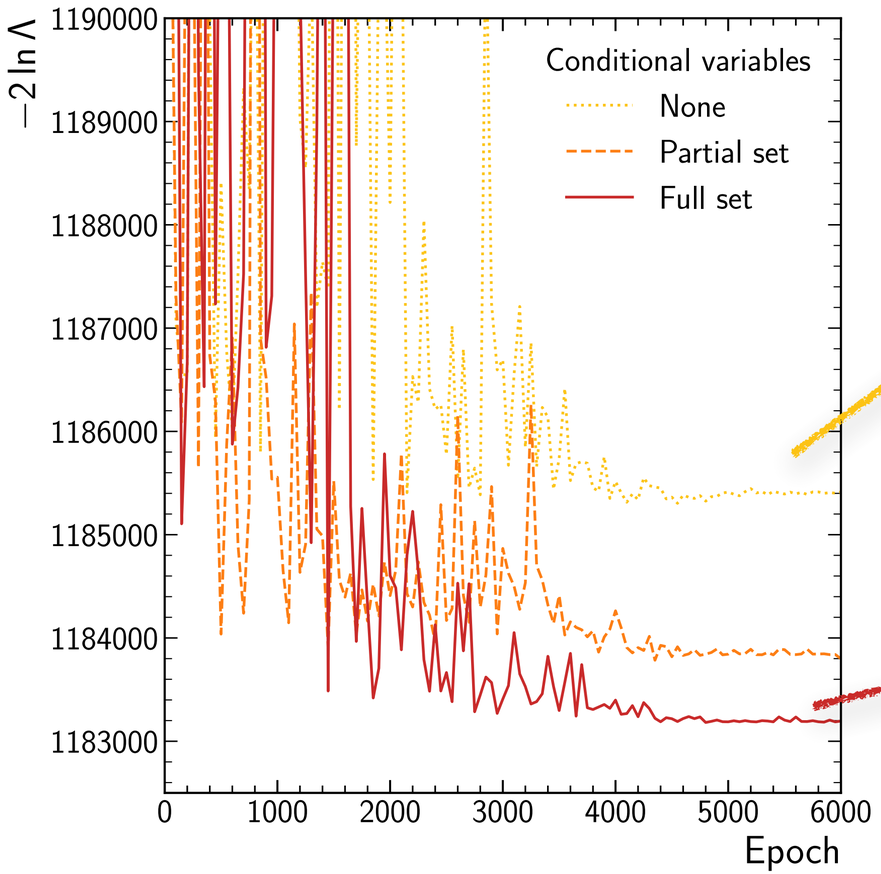


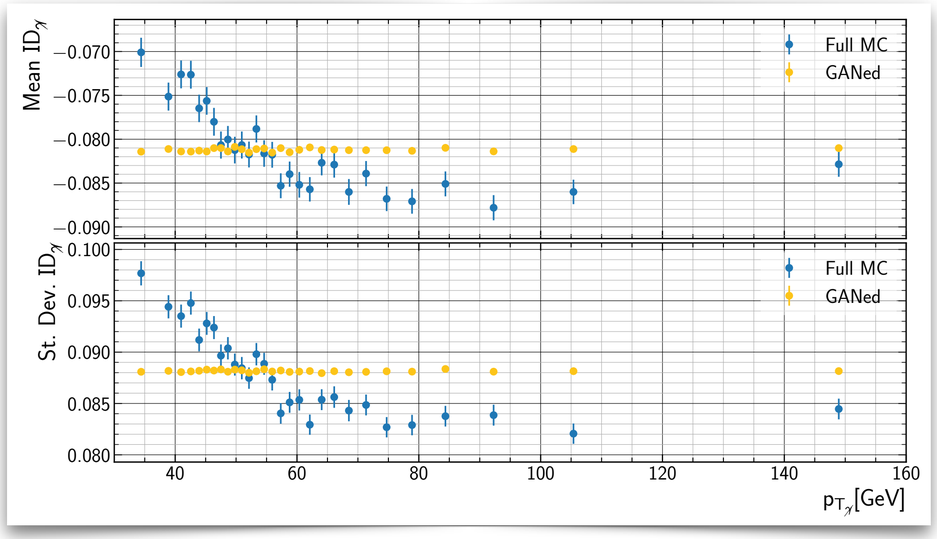


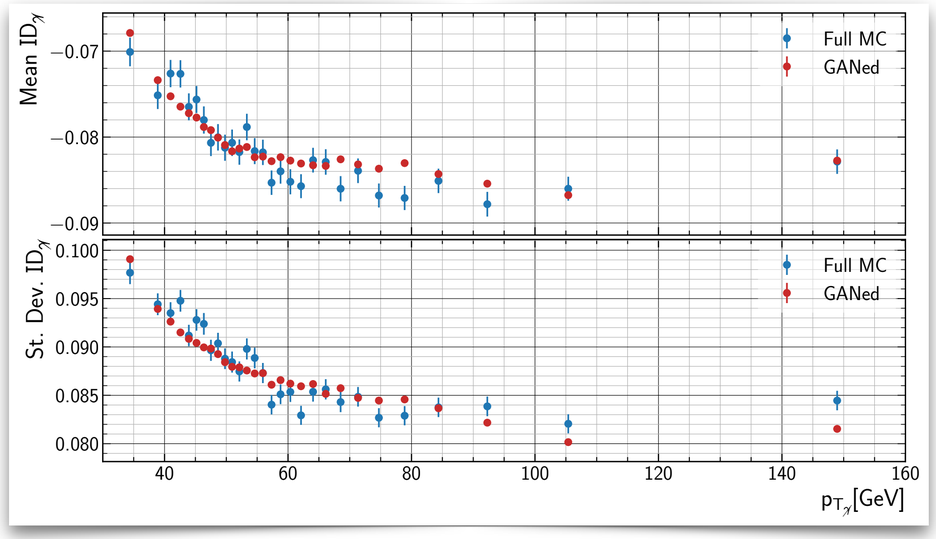
III. Generating a full object (misidentified photon)

III.a - Optimization of training

- Trying to find the correct set of observables to train our GAN, we can clearly see how hiding information from the network affects its ability to learn correlations



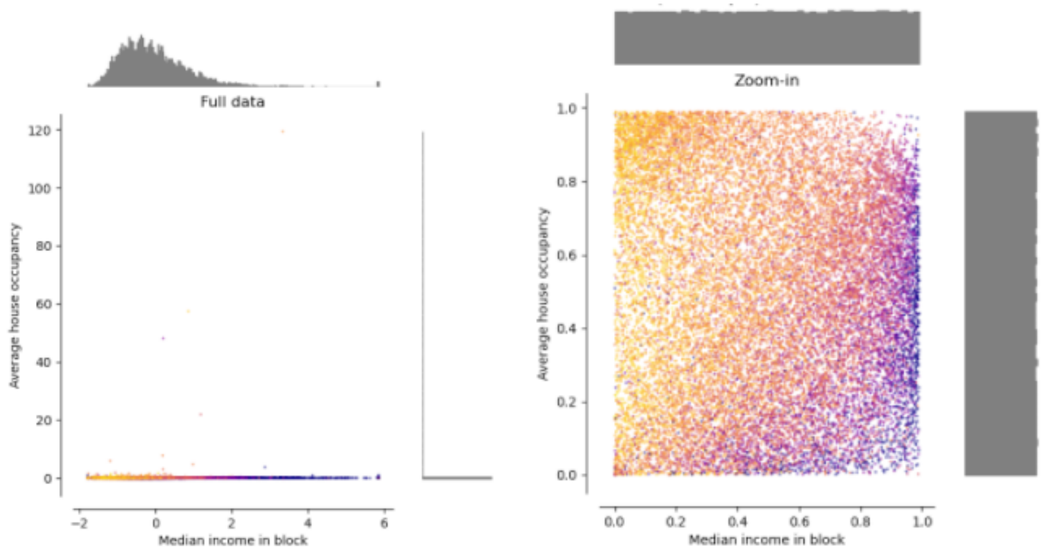


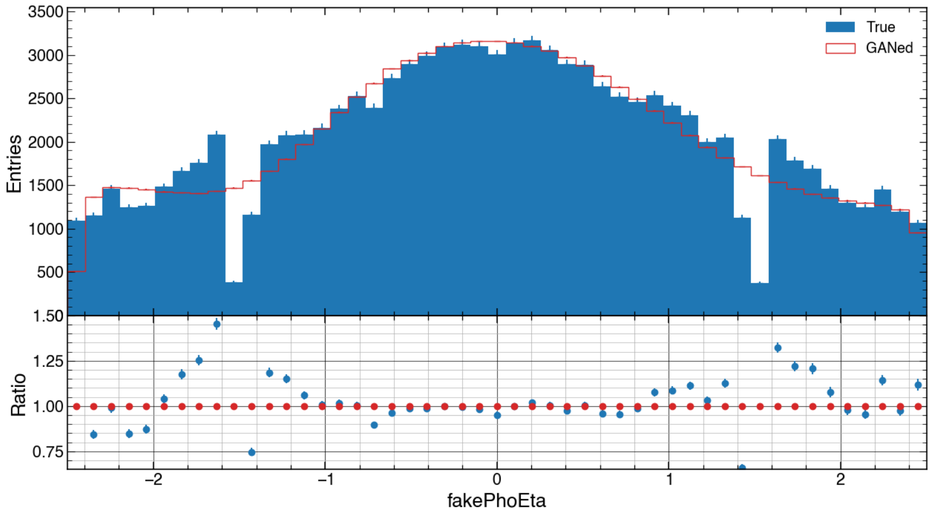


- Discrepancies for low p_T, low photonID and also in the η gaps, a better preprocessing could help

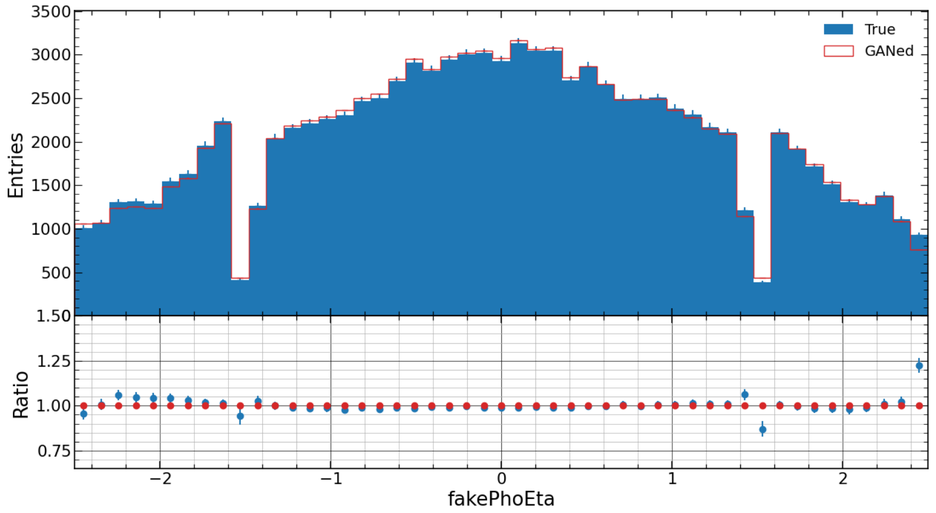
- Preprocessing the input data using a quantile transformation

Example from scikit-learn's documentation





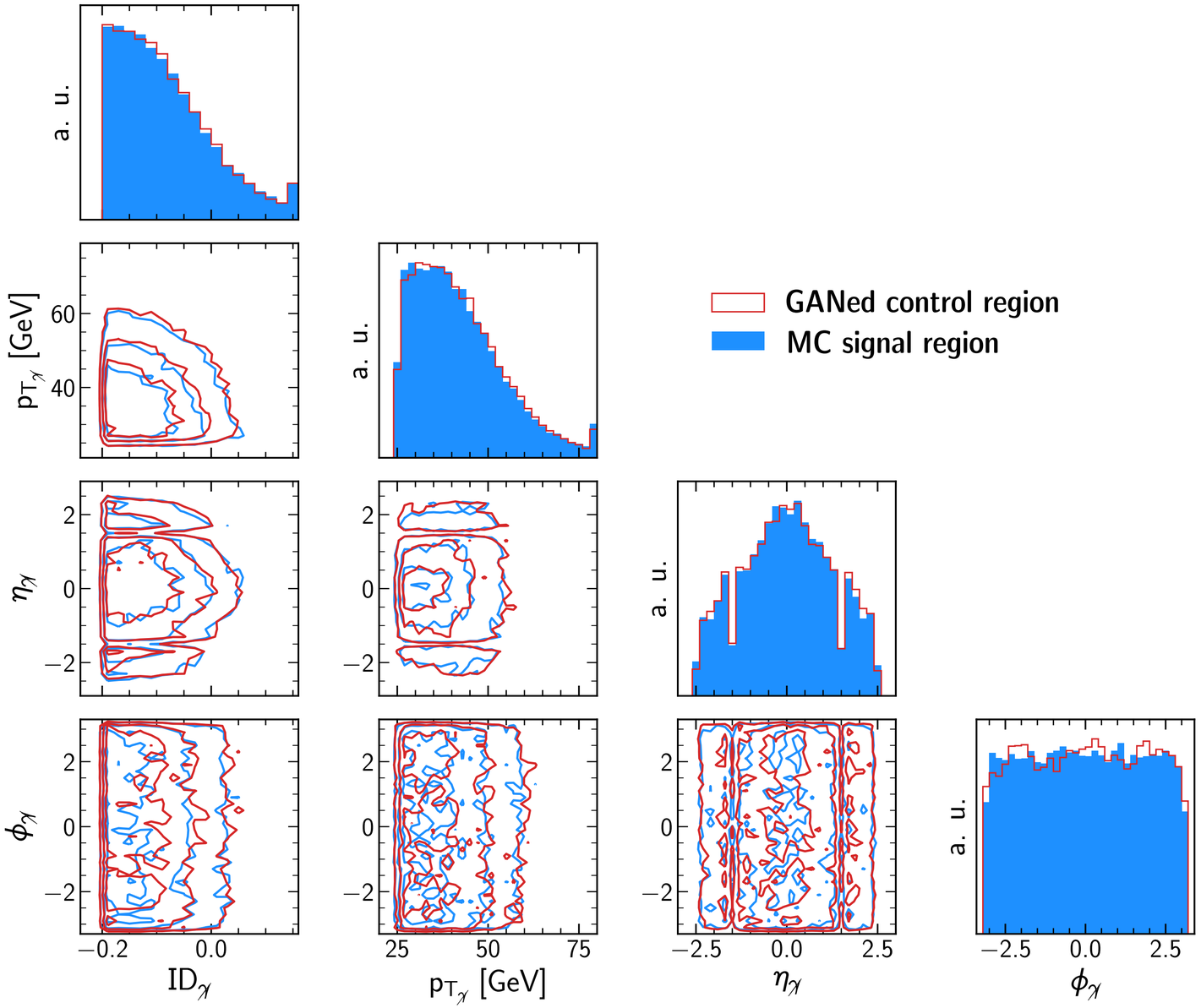
Quantile Transformation used in preprocessing



➡ Transformation helps the GAN recover the gaps in η and the core of the ID and p_T distributions

II.b - Results

- GAN is able to generate a full misidentified object that would pass the selection criteria *(see 1D distributions on diagonal)*
- GAN learns correlations between observables of the objects *(see contours on off-diagonal plots)* but also correlations with the rest of the event *(see distance correlation coefficients matrix)*
- This method could be used as a general tool to generate other objects for other use cases

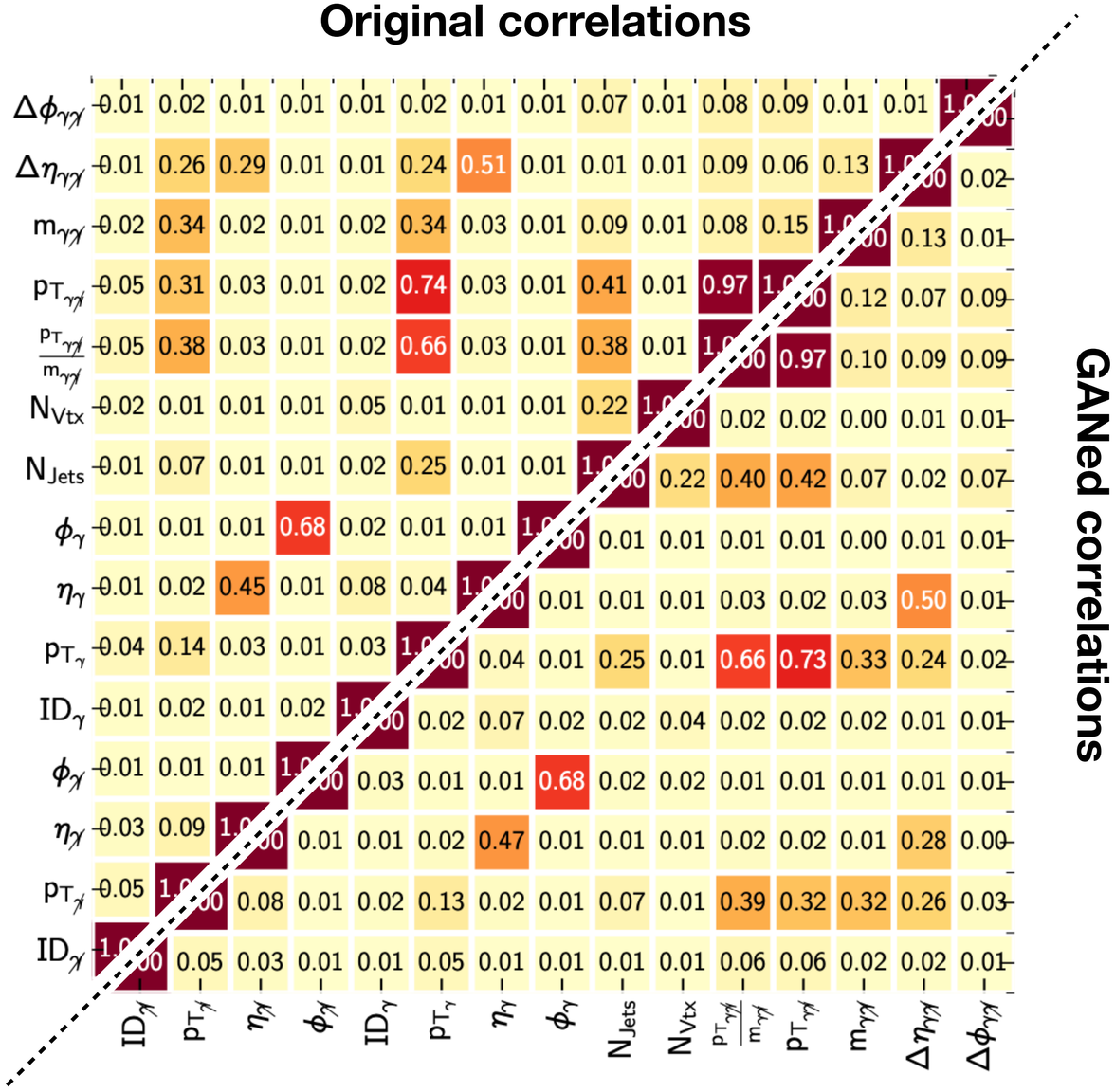


II.b - Results

- GAN is able to generate a full misidentified object that would pass the selection criteria
(see 1D distributions on diagonal)

- GAN learns correlations between observables of the objects *(see contours on off-diagonal plots)* but also correlations with the rest of the event *(see distance correlation coefficients matrix)*

- This method could be used as a general tool to generate other objects for other use cases



IV. Conclusions and outlooks

- We developed an evaluation procedure to test the GAN's generator performance and pick the best performing one
- Thanks to GAN we can generate a misidentified photon mimicking the behaviour of an object passing the photon selection criteria
- The sample produced for this showcase could be used for any H → $\gamma\gamma$ analysis
- This method can be used as a general tool to generate other objects for analyses dealing with background coming from misidentified objects