How to GAN away Detector Effects

Marco Bellagente

January 17, 2020

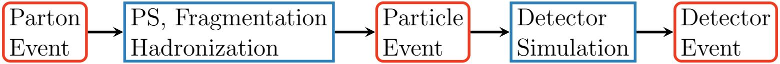
M. B., A. Butter, G. Kasieczka, T. Plehn, R. Winterhalder
arXiv:1912.00477

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

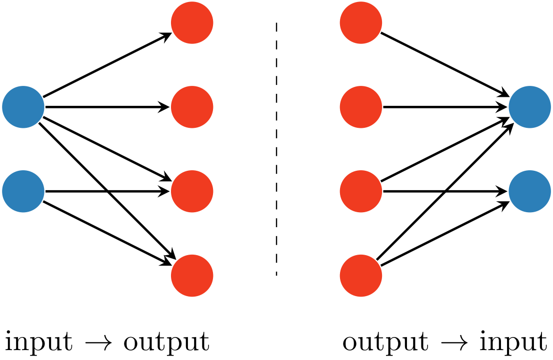
MAX-PLANCK-INSTITUT FÜR KERNPHYSIK

INTERNATIONAL MAX PLANCK RESEARCH SCHOOL PTFS FOR PRECISION TESTS OF FUNDAMENTAL SYMMETRIES

Understanding LHC data requires a chain of tools, based on fast and reliable Monte Carlo techniques:



In principle, we can invert a Monte Carlo simulations at least statistically:



Idea:

perform the inversion by training a generative model to learn the conditional probability $P\left(input|output\right)$

The GAN landscape:

- phase space generation

  S. Otten et al. arXiv:1901.00875, B. Hashemi et al. arXiv:1901.05282, R. Di Sipio et al. arXiv:1903.02433, A. butter et al. arXiv:1907.03764

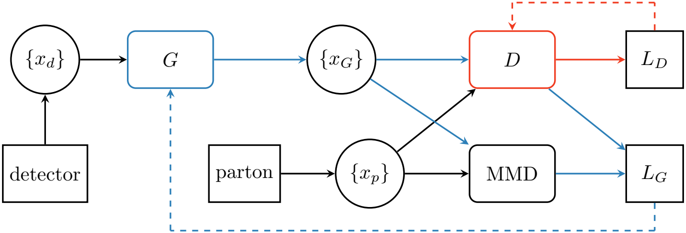
- detector simulations

  M. Paganini et al. arXiv:1705.02355, arXiv:1712.10321, P. Musella et al. arXiv:1805.00850, M. Erdmann et al. arXiv:1802.03325, arXiv:1807.01954, ATLAS Collaboration ATL-SOFT-PUB-2018-001, ATL-SOFT-PUB-2019-007

- cool stuff I could not place under a single definition

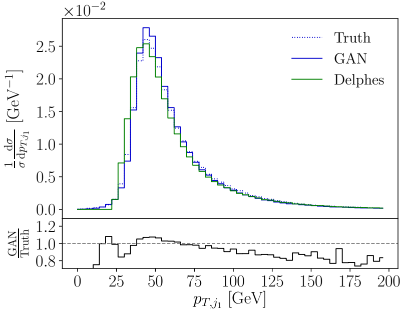
  S. Carrazza et al. arXiv:1909.01359, A. Butter et al. arXiv:1912.08824

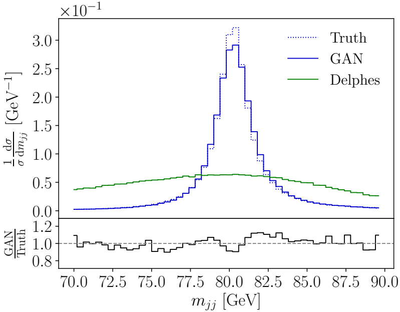
Naive GAN inversion: represent the inverse map between paired datasets with a GAN



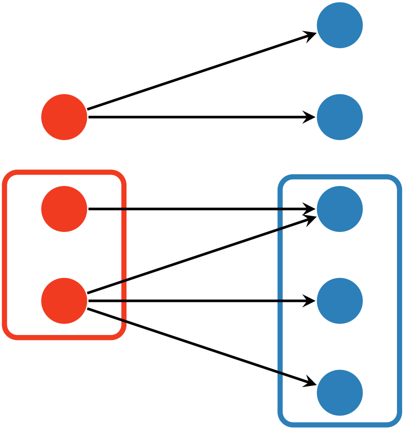
Set-up:
$pp \to ZW^{\pm} \to (l^+ l^-)\,(jj)$, 2 resolved jets after detector simulation, no ISR

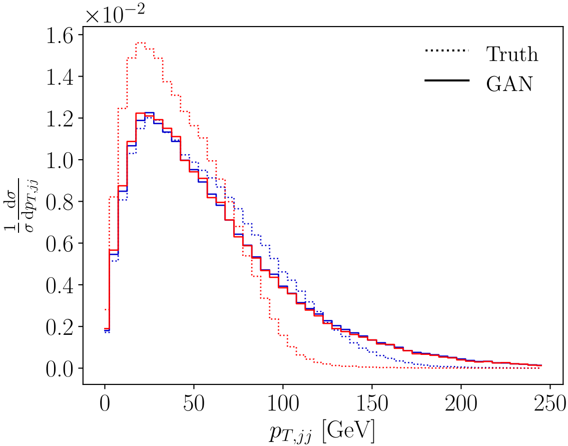
The model learns to map the entire detector dataset to the parton dataset





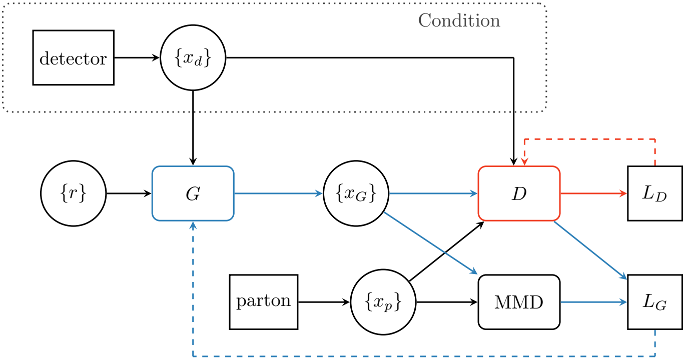
However, the model does not learn data distances/correlations:



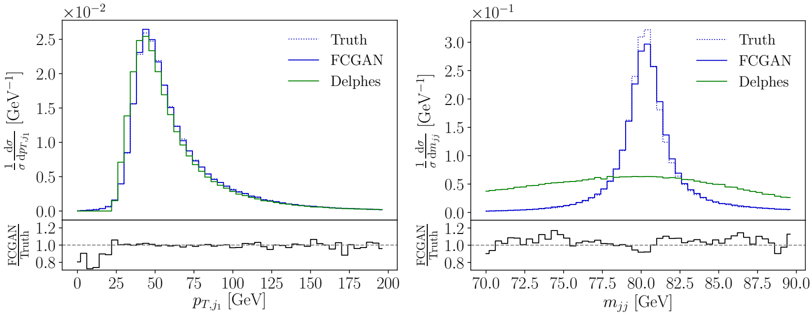


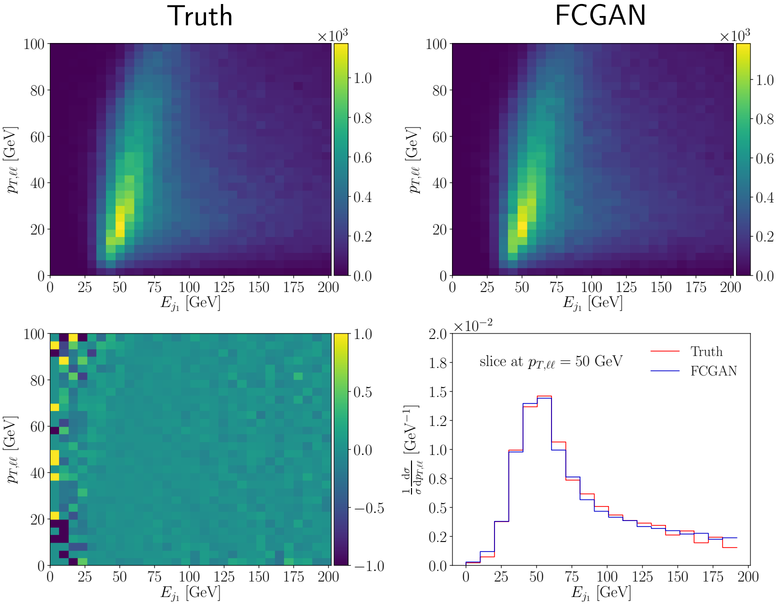
- $p_{T,j_1} = 30 \ldots 100$ GeV
- $p_{T,j_1} = 30 \ldots 60$ GeV, $p_{T,j_2} = 30 \ldots 50$ GeV

Fully Conditional GAN (FCGAN): a generative model using detector-level information as conditional input

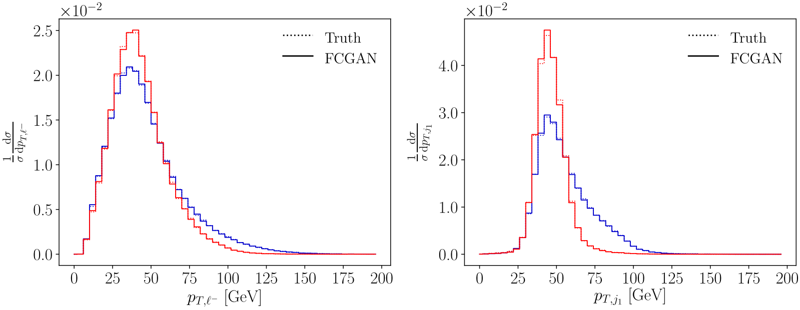


Full detector dataset mapping with no systematic error



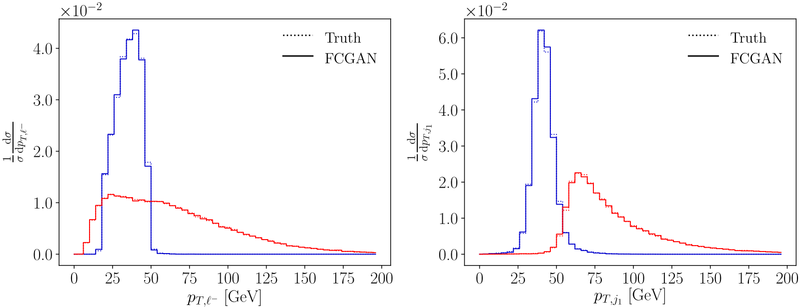

Truth
FCGAN
$\times 10^3$
$p_{T,\ell\ell}$ [GeV]
E_{j_1} [GeV]
$\times 10^{-2}$
slice at $p_{T,\ell\ell} = 50$ GeV
Truth
FCGAN
$\frac{1}{\sigma}\frac{d\sigma}{dp_{T,\ell\ell}}$ [GeV^{-1}]

Inversion works with sliced input



- $p_{T,j_1} = 30 \ldots 100$ GeV
- $p_{T,j_1} = 30 \ldots 60$ GeV, $p_{T,j_2} = 30 \ldots 50$ GeV

Robust against multiple, harder cuts



- $p_{T,j_1} = 30 \ldots 50$ GeV, $p_{T,j_2} = 30 \ldots 40$ GeV, $p_{T,\ell^-} = 20 \ldots 50$ GeV
- $p_{T,j_1} > 60$ GeV

What our FCGAN can do:

- invert a Monte Carlo simulation, like a fast detector simulation
- exploit initial state structures/correlations
- exploit data structure

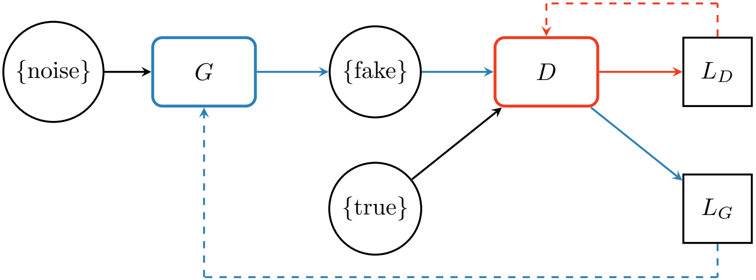
What it (still) cannot do:

- (N resolved jets $\rightarrow$ 2 hard partons inversion)
- use of a proper conditional MMD loss
- prepare coffee

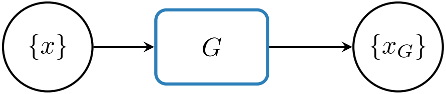
Thank you!

M. B., A. Butter, G. Kasieczka, T. Plehn, R. Winterhalder, arXiv:1912.00477

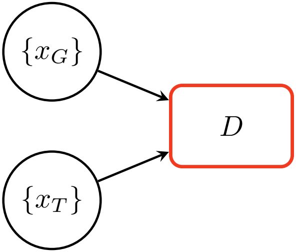
GAN models:
a counterfeit game between a Generator (G) and a Discriminator (D)



- Sample noise $\{x\}$ and true data $\{x_T\} \sim P_T$, compute $\{x_G\} = G(\{x\})$



- feed $\{x_T\}$ and $\{x_G\}$ to the discriminator



The discriminator and generator objectives are then:

$$L_D = \langle -\log D(x) \rangle_{x \in \{x_T\}} + \langle -\log(1 - D(x)) \rangle_{x \in \{x_G\}}$$
$$L_G = \langle -\log D(x) \rangle_{x \in \{x_G\}}$$

plus regularizations, additional losses, networks, ...

The model has a global minimum for $G(x) = P_T$

Parameter	Value	Parameter	Value
Layers	12	Batch size	512
Units per layer	512	Epochs	1200
Trainable weights G	3M	Iterations per epoch	500
Trainable weights D	3M	Number of training events	3×10^5
λ_G	1		
λ_D	10^{-3}		

Table: FCGAN setup.

Our actual objective:

given the logit function

$$\phi(x) = \log \frac{D(x)}{1 - D(x)}$$

the regularized Jensen-Shannon GAN objective is

$$L_D^{(\mathrm{reg})} = L_D + \lambda_D \langle (1 - D(x))^2 \, |\nabla \phi| \rangle_{x \sim P_T} + \lambda_D \langle D(x)^2 \, |\nabla \phi|^2 \rangle_{x \sim P_d}$$

The FCGAN basic objective is

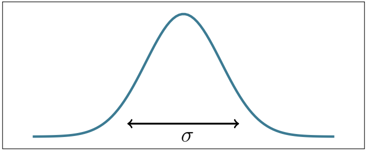
$$L_D^{(\mathrm{FC})} = \langle - \log D(x,y) \rangle_{x \sim P_T, y \sim P_d} + \langle - \log(1 - D(x,y)) \rangle_{x \sim P_G, y \sim P_d};$$

$$L_G^{(\mathrm{FC})} = \langle - \log D(x,y) \rangle_{x \sim P_G, y \sim P_d}$$

and the corresponding regularized FCGAN objective is

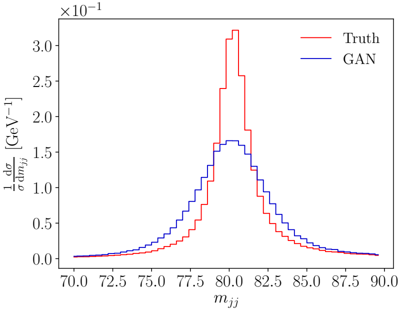
$$\begin{aligned} L_D^{(\mathrm{reg,\ FC})} = L_D^{(\mathrm{FC})} &+ \lambda_D \langle (1 - D(x,y))^2 \, |\nabla \phi| \rangle_{x \sim P_T, y \sim P_d} \\ &+ \lambda_D \langle D(x,y)^2 \, |\nabla \phi|^2 \rangle_{x \sim P_G, y \sim P_d} \end{aligned}$$

We employ a Maximum Minimum Discrepancy loss (MMD) to learn more easily the invariant mass of intermediate resonances

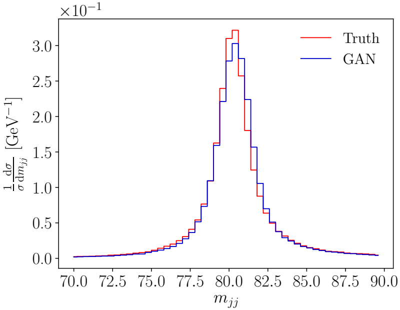


- choose one (or many) kernels $k\left(x,y\right)=k\left(|x-y|^2\right)$
- take samples $x,x'\sim p$ and $y,y'\sim q$
- minimize $\text{MMD}^2=\langle k\left(x,x'\right)\rangle+\langle k\left(y,y'\right)\rangle-2\langle k\left(x,y\right)\rangle$

MMD loss effect on intermediate resonances mapping



no MMD



MMD

The (square rooted) MMD Loss is defined as:

$$\text{MMD} = \left[\langle k\left(x, x'\right)\rangle_{x,x'\sim P_G} + \langle k\left(y, y'\right)\rangle_{y,y'\sim P_p} - 2\langle k\left(x, y\right)\rangle_{x\sim P_G, y\sim P_p}\right]^{1/2}.$$

Common choices of the kernel k are parametrized by a resolution parameter σ (e.g. $k_\sigma\left(x, y\right) \sim \exp\frac{|x-y|^2}{\sigma^2}$).

Natural choice: $\sigma \sim \Gamma$ of the intermediate resonance

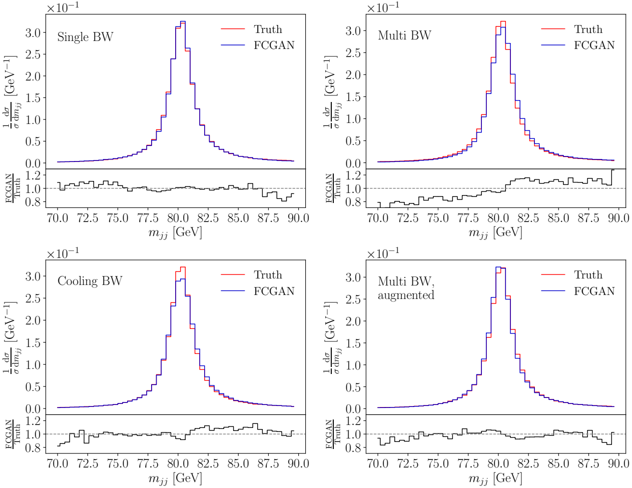
Pro:

- precise at the end of the training

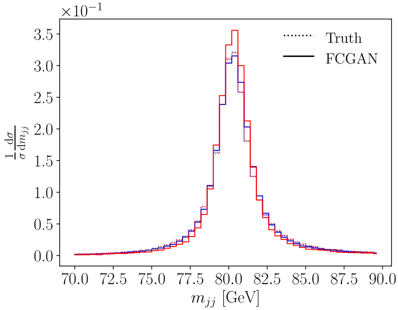
Con:

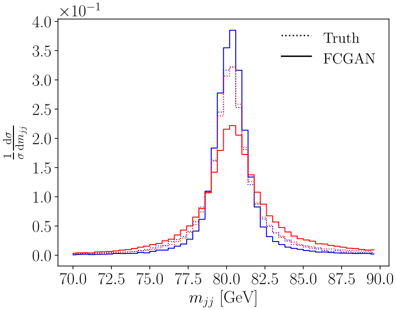
- gradient vanishes at beginning of the training
- not feasible for extremely narrow resonances (e.g. Higgs)

Several workaround to the vanishing gradient problem



In our current set-up the MMD loss for resonances is not using the detector conditioning → less precise if we only invert a particular phase space region

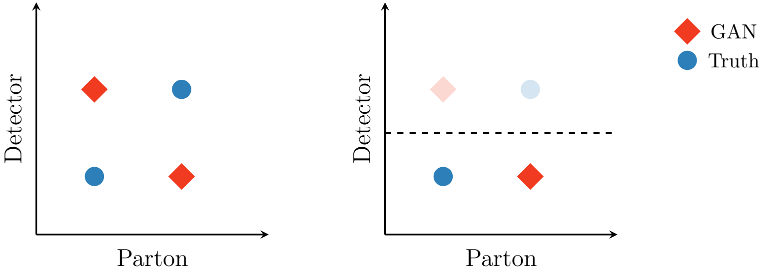




Fraction of events per cut:

- cut I:
  $p_{T,j_1} = 30 \ldots 100$ GeV,
  88%,
- cut II:
  $p_{T,j_1} = 30 \ldots 60$ GeV, $p_{T,j_2} = 30 \ldots 50$ GeV,
  38%
- cut III:
  $p_{T,j_1} = 30 \ldots 50$ GeV, $p_{T,j_2} = 30 \ldots 40$ GeV, $p_{T,\ell^-} = 20 \ldots 50$ GeV,
  14%
- cut IV:
  $p_{T,j_1} > 60$ GeV,
  39%,

The failure is not surprising, in the GAN model the Discriminator only sees the projection on the Parton axis



and is therefore not able to punish a wrong input - output relation