ANALYSIS-SPECIFIC FAST SIMULATION WITH DEEP LEARNING

Cheng Chen[1], Maurizio Pierini[2], Olmo Cerri[3], Thong Nguyen[3]

[1]Peking University, [2]CERN, [3]California Institute of Technology

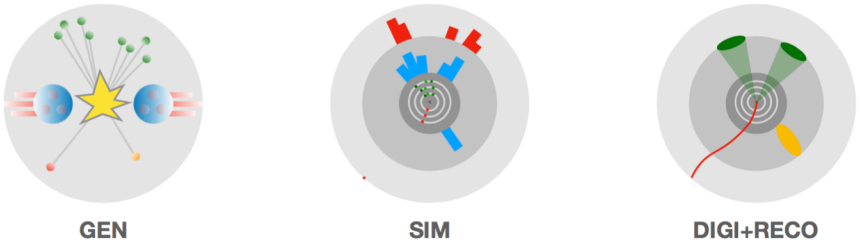
CLHCP meeting, Nov. 6th, 2020, Tsinghua University

Outline of this talk

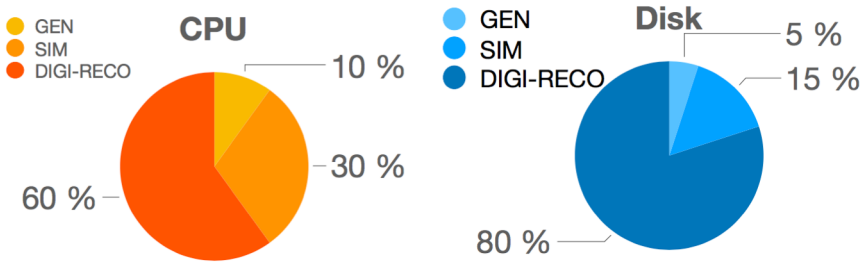
- Motivation of this study
- Deep learning model architecture
- Simulation of W+1jet
- Summary

Motivation

- *Recently, generative algorithms trained with Machine Learning techniques have been proposed as a possible solution to speed up GEANT4 simulations.*
- *When training both VAEs and GANs, the limited amount of data in the training dataset is ultimate precision-limiting factor, as discussed in arxiv:2002.06307.*
- *We propose to rephrase the problem of analysis-specific dataset generation into training a fast-and-accurate detector-response DL model.*
  - *Reduce computing time for about 90%; (Predict Reco with Gen)*
  - *One would also reduce the need for large storage elements. (O(1MB) for raw data and O(10KB) for analysis-ready object collections)*



Workflow of the CMS experiment



Computing resource for CMS experiment

Supervised Deep learning

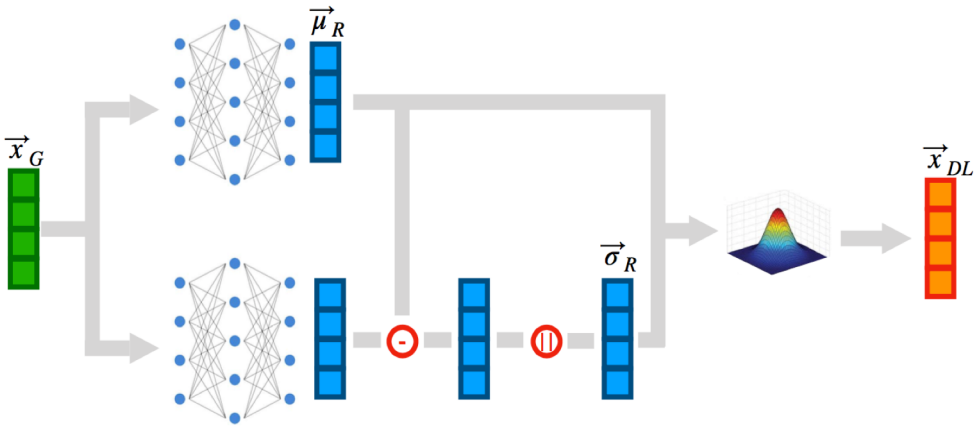
- *For a given data analysis, we assume that the interesting features of an LHC event can be represented by a limited set of high-level quantities (the feature vector $\vec{x}$).*
- *We assume that a training data set is provided. For each collision event in the data set, the feature vector is computed at three stages:*

(i) at generator level $\vec{x}_G$, i.e., before applying any detector simulation. This view of the collision event corresponds to the case of an ideal detector with perfect resolution;

(ii) at reconstruction level $\vec{x}_R$, i.e. after the simulation of the detector response, modelled with GEANT4;

(iii) model prediction $\vec{x}_{DL}$, i.e. the output of the Deep Learning model. We model function as a Normal function of the generator-level feature vector:

$$\vec{x}_{DL} = \prod_i \mathcal{N}(\mu_R^i(\vec{x}_G), \sigma_R^i(\vec{x}_G))$$



Model architecture

Simulation of W+1jet

Introduction of quantities to be regressed

- *The feature vector is built considering the following 9 quantities:*
  - The muon momentum in Cartesian coordinates: p_x^μ, p_y^μ, and p_z^μ.
  - The jet momentum in Cartesian coordinates: p_x^j, p_y^j, and p_z^j.
  - The logarithm of the jet mass $\log(M_j)$.
  - The missing transverse energy in Cartesian coordinates: E_x^{miss} and E_y^{miss}.
- *In addition, we consider a set of 12 auxiliary features, computed from the input vector features:*
  - The muon momentum in boost-invariant cylindrical coordinates: p_T^μ, η^μ, and ϕ^μ.
  - The jet momentum in boost-invariant cylindrical coordinates: p_T^j, η^j, and ϕ^j.
  - The missing transverse energy in polar coordinates: E_T^{miss} and ϕ_{miss}.
  - The transverse mass M_T, i.e., the mass of the four momentum obtained summing the the muon transverse momentum $(E_T^\mu, p_x^\mu, p_y^\mu, 0)$ to the missing transverse energy $(E_T^{\text{miss}}, E_x^{\text{miss}}, E_y^{\text{miss}}, 0)$.
  - The scalar sum of the jets, missing energy, and muon p_T: S_T.
  - The jet mass: M_j
- *In particular, we found that using of log(MJ) as part of the input feature vector and MJ in the list of auxiliary vector helped improving the description of MJ and its correlation to other quantities, as discussed in backup s29.*

Delphes conditions (before training) and Final-selection (after training)

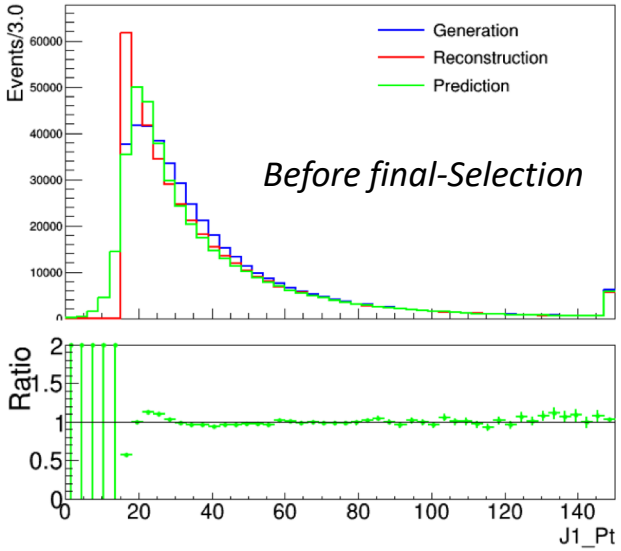
➢ *At reconstruction level, jets are clustered from the list of particles returned by the DELPHES particle-flow algorithm.*

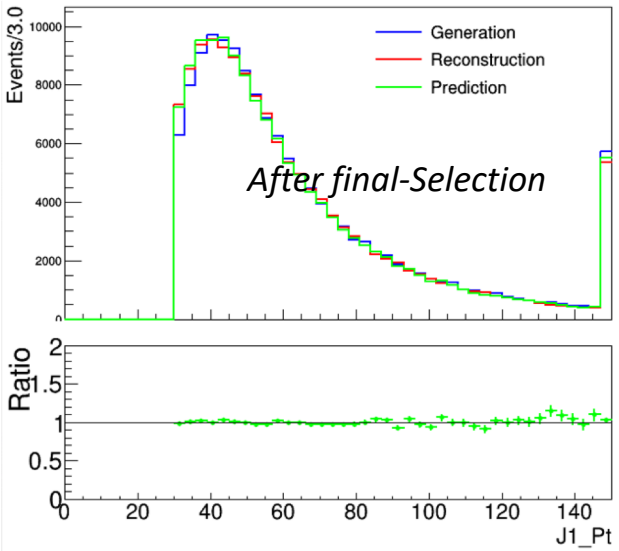
- *As for the GEN jets, we consider anti-kt jets with R = 0.5.*
- *In order to avoid the double counting of the muon as the jet, we require dR(Reco Muon, Reco Jet)>0.5.*

Gen: generation
Reco: reconstruction
Pre: prediction

➢ *Final-Selections after training (prediction):*

- *$P_T{}^{\mu}$>20GeV, $|\eta^{\mu}|$<2.4, $P_T{}^{J}$>30GeV, $|\eta^{J}|$<2.4, M^{J}>0 for Reco and Pre quantities.*





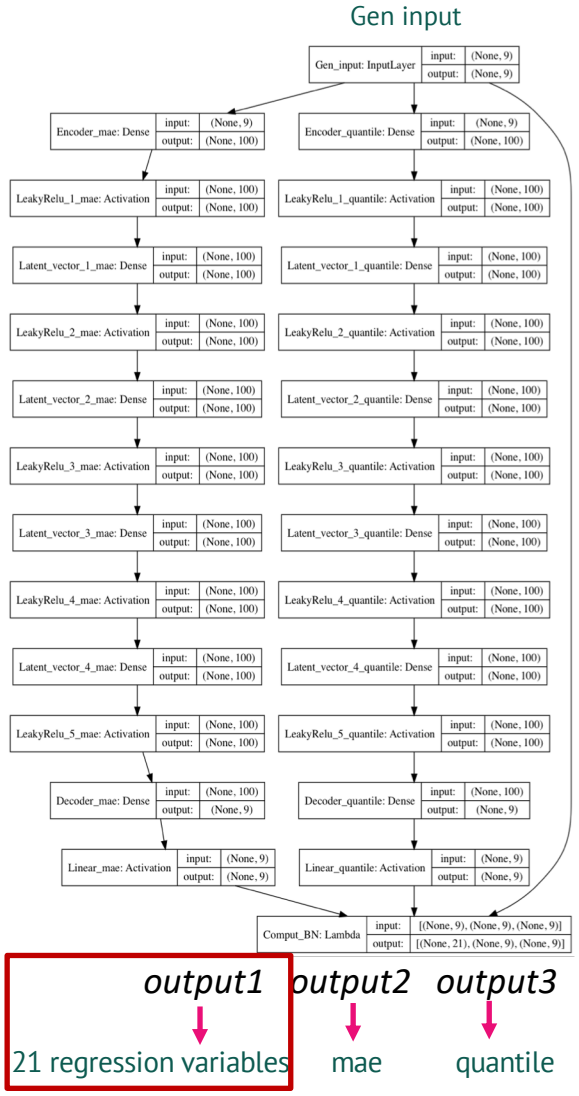
Model description

- *After batch normalization, input 9 kinematic Gen level variables. Return them back to true value, then compute 12 derived variables using Lambda layer.*
- *Network information (parameters):*

- *x_train(test):70%(30%)gen, y_train(test):70%(30%)reco.*
- *Batch size: 128, Latent dim: 100*
- *Loss: MAE and Quantile*
- *Alpha of LeakReLU: 0.05*
- *Optimizers: Adam*
- *Learning rate: 0.001/(1+epoch)*

~ 2M events

- *In addition, we get 10 additional models by training on x10 more data to draw the error bands.*
- *We apply the model on x5 validation data.*

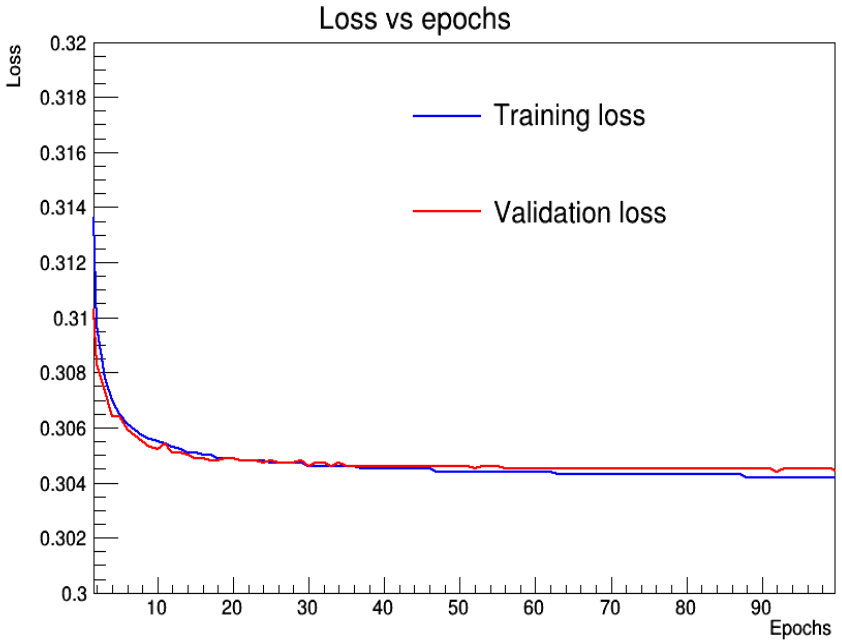


Loss result (use MAE and Quantile)

- *Since the distance between Gen and Rec for the first 9 variables are different, (some distances are quite small, we can not predict small sigma accurately) we do the following transformation before training:*

$$REC_T^i = GEN^i + \frac{\sigma_{REC-GEN}^{Log(MJ)}}{\sigma^{Log(MJ)}} * \left(REC^i - GEN^i\right) * \frac{\sigma^i}{\sigma_{REC-GEN}^i}$$

- *Where σ is the standard deviation, i=1~9.*
- *We have similar relative distance for 9 first variables after the transformation.*
- *We predict REC_T, then return to true REC.*
- *μ = output2(MAE)*
- *σ = |output3(Quantile)- output2(MAE)|,* for Quantile loss, *γ=0.5+0.341(1σ)*
- Prediction(output1) $= \mu + \sigma \odot \varepsilon$, where $\varepsilon = N(0,1)$
- *Save best model with minimal validation loss.*

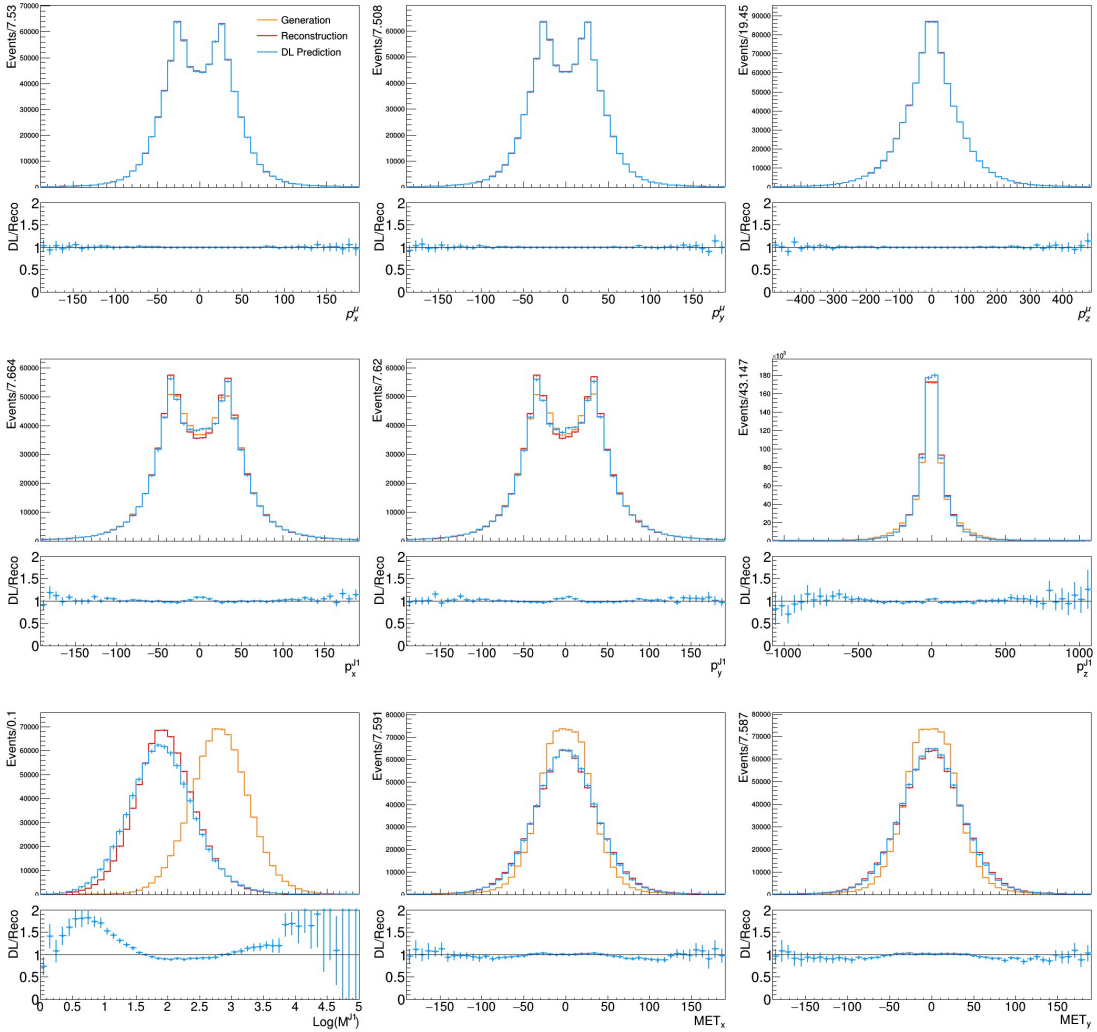


$$\mathcal{L}_{RECO} = \frac{1}{N}\left[\sum_{i=1}^{N}\sum_{j=1}^{k}|x_{DL}^j - x_R^j| + \sum_{i=1}^{N}\Theta(x_{DL}^j, x_R^j)|x_{DL}^j - x_R^j|\right]$$

where:

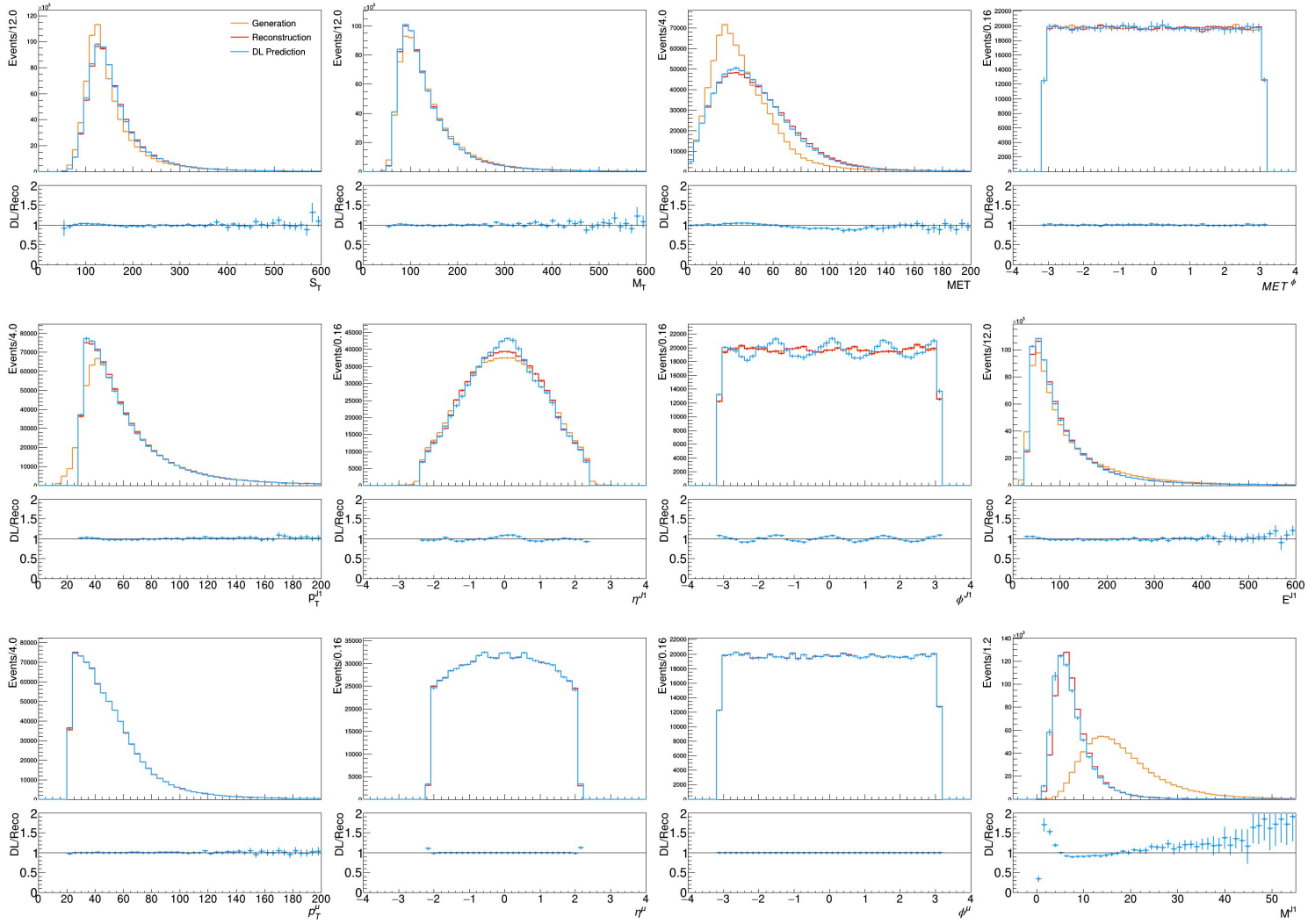
$$\Theta(x,y) = (1-\gamma)\theta(x-y) + \gamma\theta(y-x)$$

Comparison plots for basic quantities (x5 validation data)



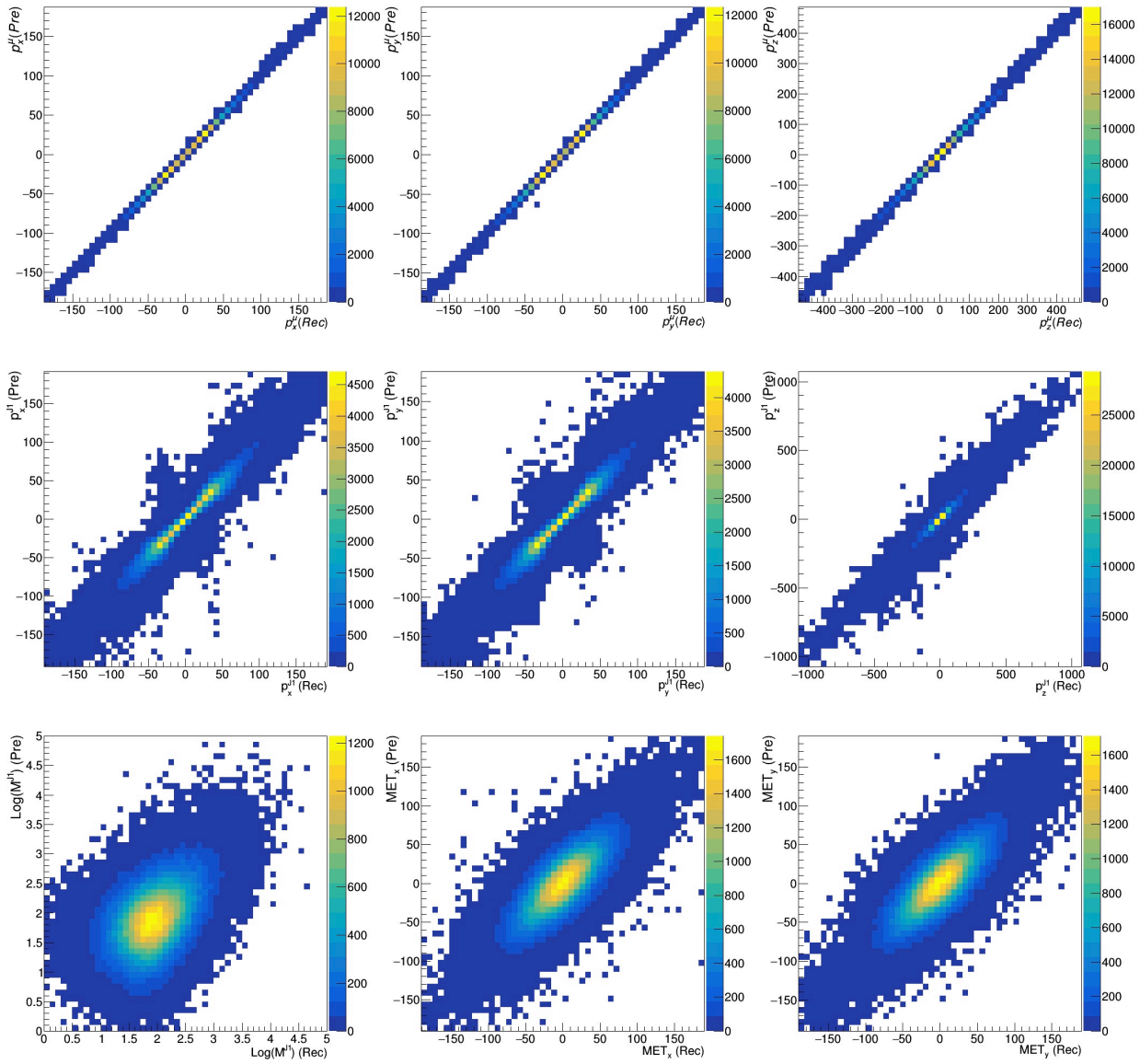
- *Error bands of DL prediction include both systematic uncertainty provided by rms of 10 additional models and statistic uncertainty. (Gen and Reco have only statistic uncertainty)*

Comparison plots for derived quantities

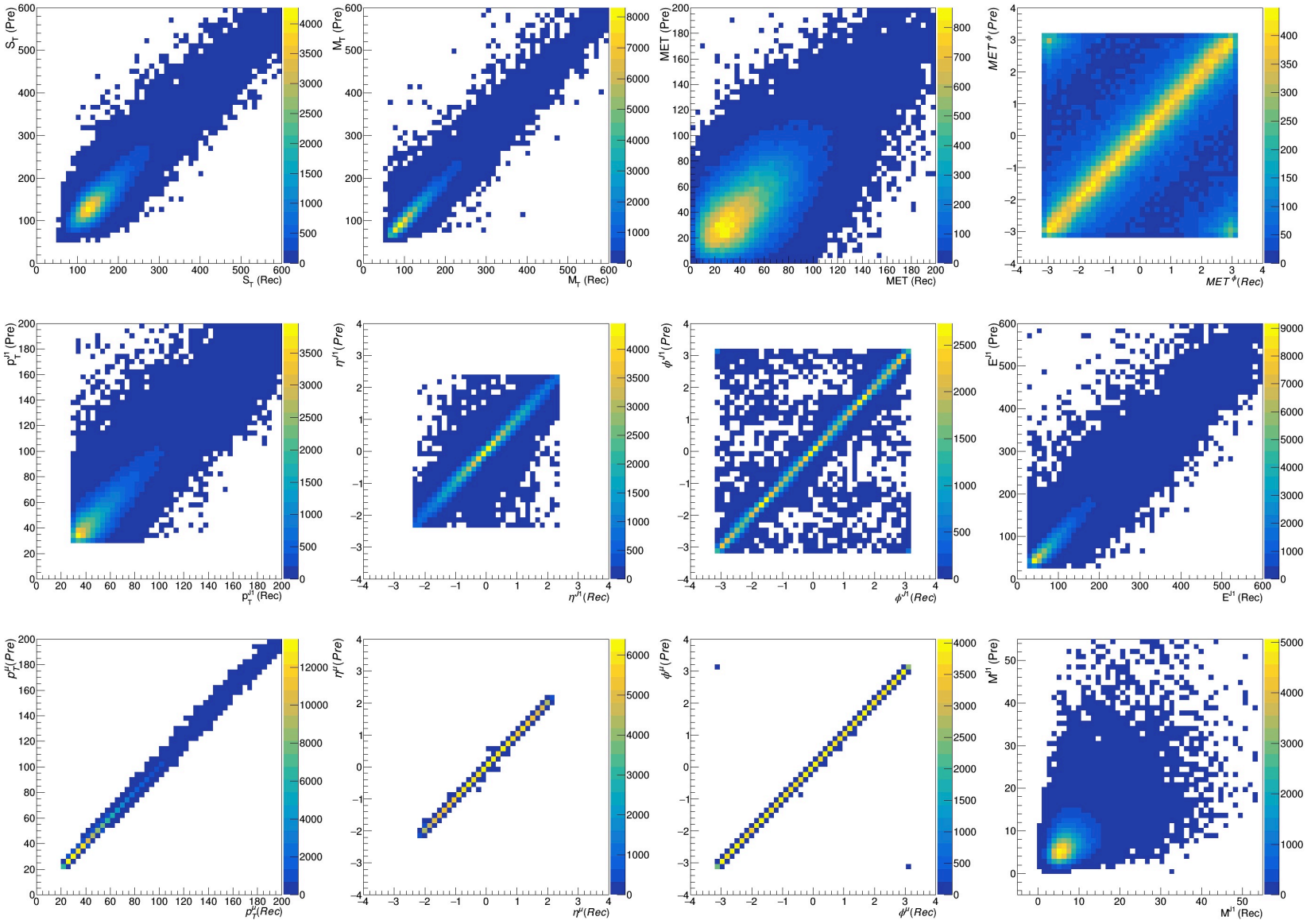


- *Jet 4 momentum are hard to predict, this is due to some irregular correlation between Reco and Gen of jet 4 momentum.*

2D plots for basic quantities (Pre vs Rec)



2D plots for derived quantities (Pre vs Rec)



Paper submitted in arXiv

DATA AUGMENTATION AT THE LHC THROUGH ANALYSIS-SPECIFIC FAST SIMULATION WITH DEEP LEARNING

C. Chen
Peking University
Haidian, China, 100871

O. Cerri, T. Q. Nguyen, J.R. Vlimant
California Institute of Technology
Pasadena, CA 91125, USA

M. Pierini
European Organization for Nuclear Research (CERN)
CH-1211 Geneva 23, Switzerland

arXiv:2010.01835

October 6, 2020

ABSTRACT

We present a fast simulation application based on a Deep Neural Network, designed to create large analysis-specific datasets. Taking as an example the generation of W+jet events produced in $\sqrt{s} = 13$ TeV proton-proton collisions, we train a neural network to model detector resolution effects as a transfer function acting on an analysis-specific set of relevant features, computed at generation level, i.e., in absence of detector effects. Based on this model, we propose a novel fast-simulation workflow that starts from a large amount of generator-level events to deliver large analysis-specific samples. The adoption of this approach would result in about an order-of-magnitude reduction in computing and storage requirements for the collision simulation workflow. This strategy could help the high energy physics community to face the computing challenges of the future High-Luminosity LHC.

Summary

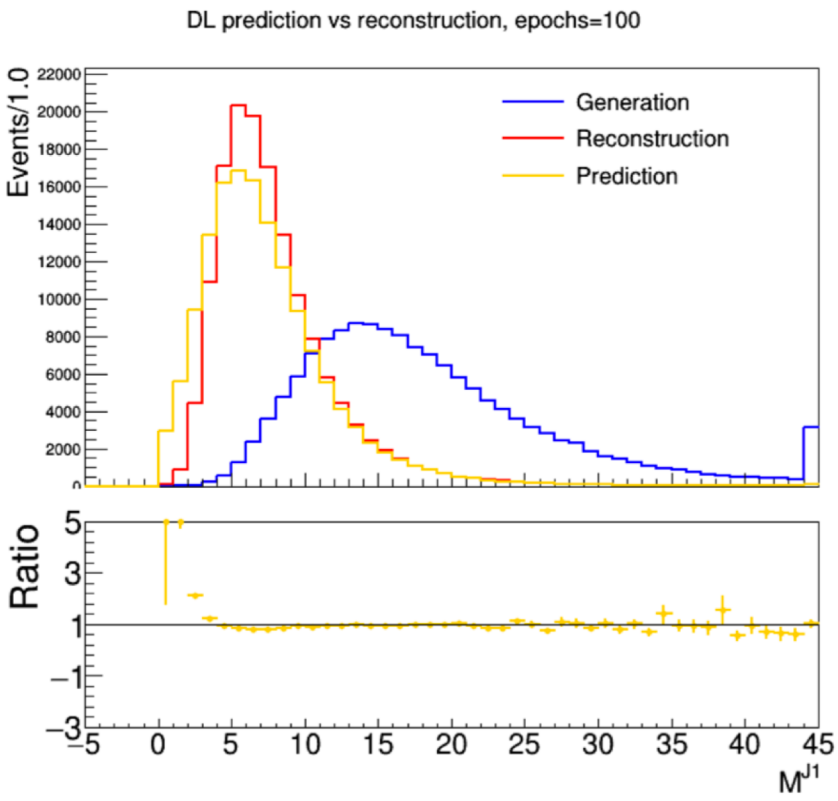
- *We investigate analysis-specific fast-and-accurate detector-response DL model for Reco quantities regression.*
- *Varying degrees of uncorrelation between Pre and Reco results from the stochasticity of prediction.*
- *Some irregular correlation between Reco and Gen of jet 4 momentum affect the accuracy of prediction.*
- *In general, we have good agreement between DL prediction and Delphes simulation.*

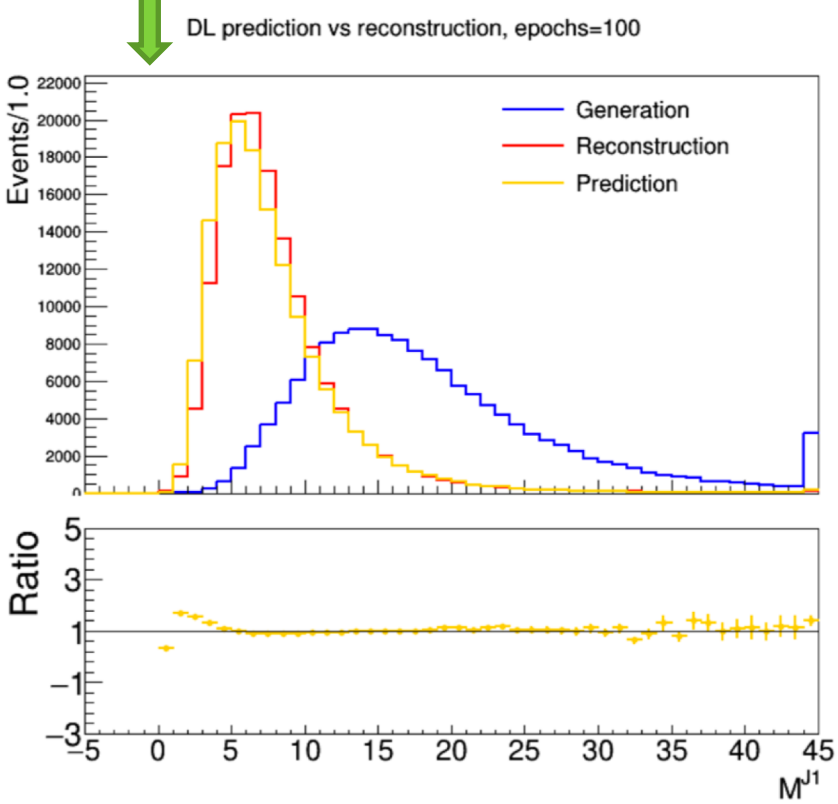
Backup

Comparison between:

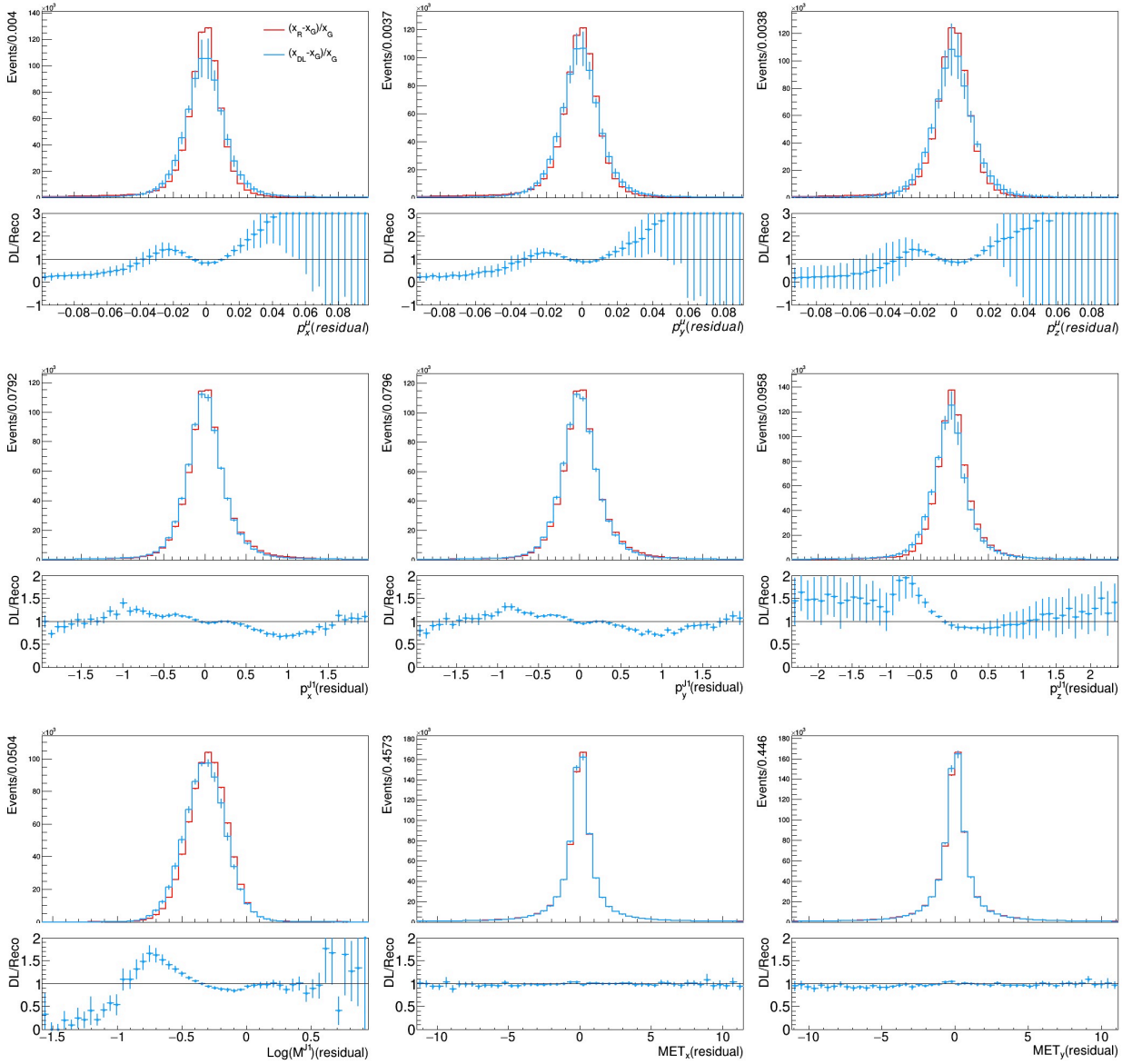
1, Train M_J, derive $Log(M_J)$

2, Train $Log(M_J)$, derive M_J

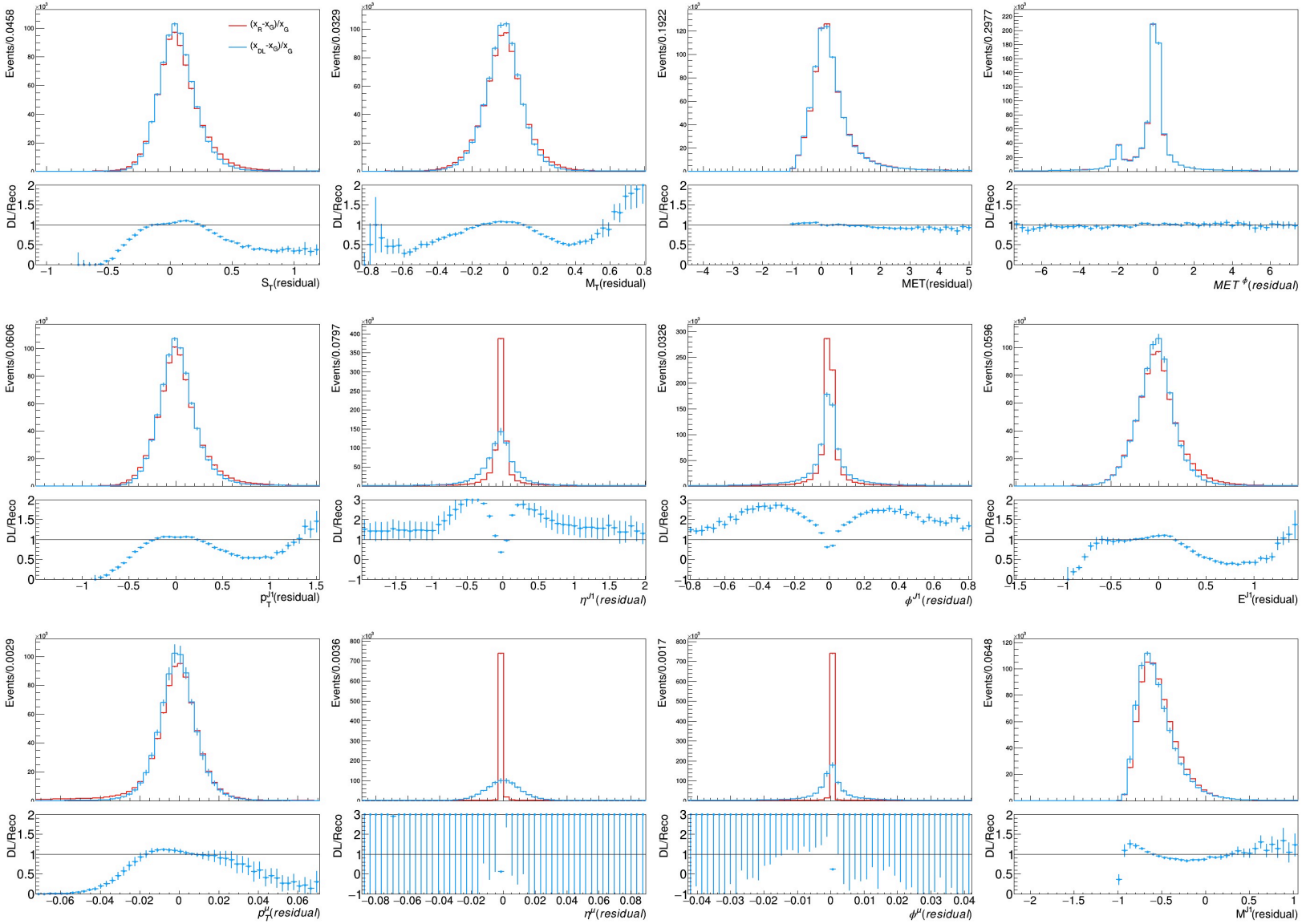




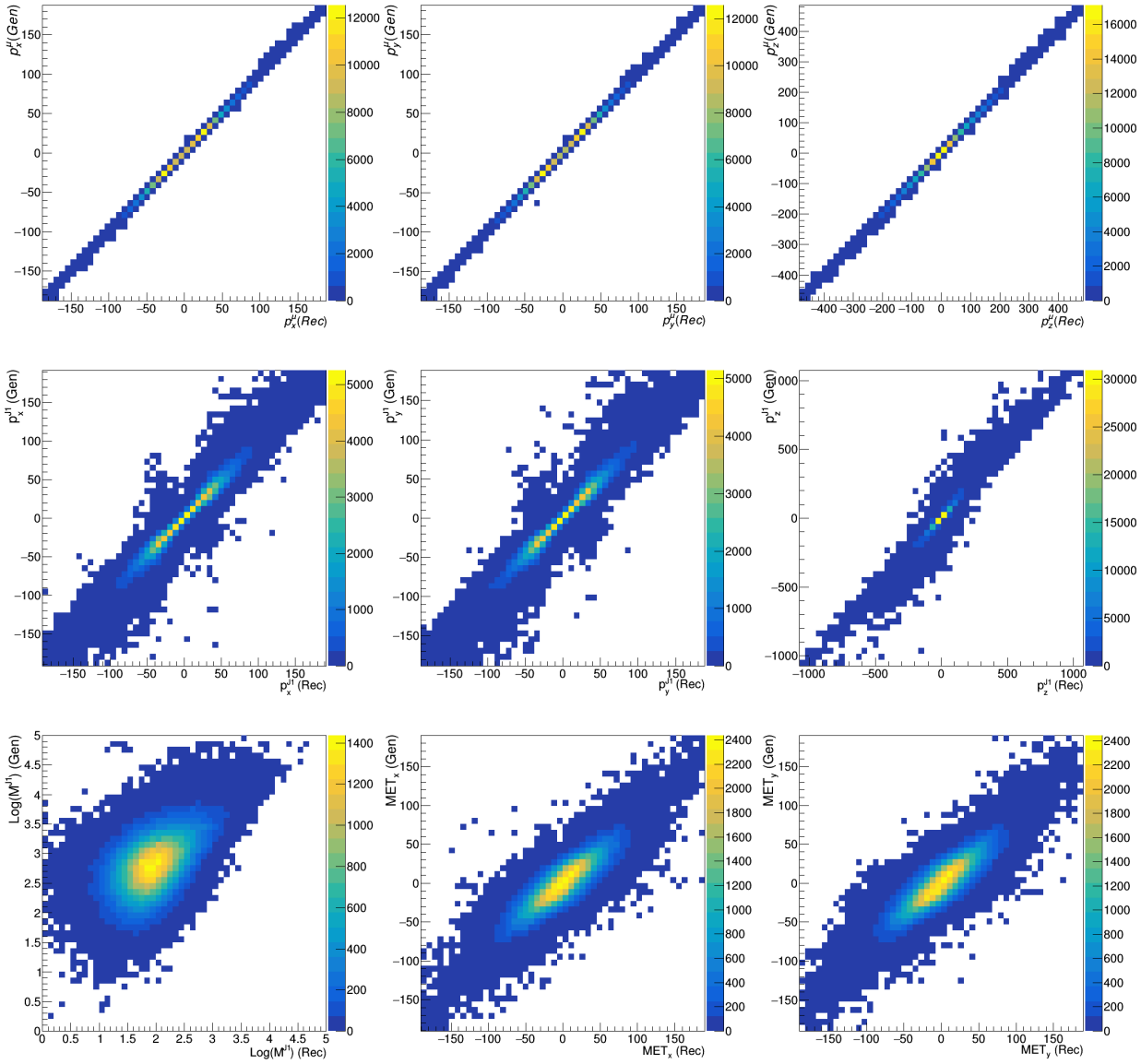
Comparison plots for basic quantities (relative residual: [Rec-Gen]/Gen)



Comparison plots for derived quantities (relative residual: [Rec-Gen]/Gen)



2D plots for basic quantities (Gen vs Rec)



2D plots for derived quantities (Gen vs Rec)