

11-2022

## 'Flux+Mutability': A Conditional Generative Approach to One-class Classification and Anomaly Detection

Cristiano Fanelli

*William & Mary*, [cfanelli@wm.edu](mailto:cfanelli@wm.edu)

James Giroux

Z. Papandreou

Follow this and additional works at: <https://scholarworks.wm.edu/aspubs>



Part of the [Data Science Commons](#)

---

### Recommended Citation

Fanelli, Cristiano; Giroux, James; and Papandreou, Z., 'Flux+Mutability': A Conditional Generative Approach to One-class Classification and Anomaly Detection (2022). *Machine Learning: Science and Technology*, 3(4).

<https://doi.org/10.1088/2632-2153/ac9bcb>

This Article is brought to you for free and open access by the Arts and Sciences at W&M ScholarWorks. It has been accepted for inclusion in Arts & Sciences Articles by an authorized administrator of W&M ScholarWorks. For more information, please contact [scholarworks@wm.edu](mailto:scholarworks@wm.edu).



## PAPER

## OPEN ACCESS


RECEIVED  
3 May 2022REVISED  
29 September 2022ACCEPTED FOR PUBLICATION  
18 October 2022PUBLISHED  
4 November 2022

Original Content from  
this work may be used  
under the terms of the  
[Creative Commons  
Attribution 4.0 licence](#).

Any further distribution  
of this work must  
maintain attribution to  
the author(s) and the title  
of the work, journal  
citation and DOI.



# ‘Flux+Mutability’: a conditional generative approach to one-class classification and anomaly detection

C Fanelli<sup>1,2,3,4,\*</sup> , J Giroux<sup>5</sup> and Z Papandreou<sup>5</sup><sup>1</sup> Laboratory for Nuclear Science, Massachusetts Institute of Technology, Cambridge, MA 02139, United States of America<sup>2</sup> The NSF AI Institute for Artificial Intelligence and Fundamental Interactions<sup>3</sup> William & Mary, Data Science, Williamsburg, VA 23185, United States of America<sup>4</sup> Jefferson Lab, Newport News, VA 23606, United States of America<sup>5</sup> Department of Physics, University of Regina, Regina SK S4S 0A2, Canada

\* Author to whom any correspondence should be addressed.

E-mail: [cfanelli@wm.edu](mailto:cfanelli@wm.edu)**Keywords:** conditional masked autoregressive flow, clustering, anomaly detection, one-class classification, showers and jets

## Abstract

Anomaly Detection is becoming increasingly popular within the experimental physics community. At experiments such as the Large Hadron Collider, anomaly detection is growing in interest for finding new physics beyond the Standard Model. This paper details the implementation of a novel Machine Learning architecture, called Flux+Mutability, which combines cutting-edge conditional generative models with clustering algorithms. In the ‘flux’ stage we learn the distribution of a reference class. The ‘mutability’ stage at inference addresses if data significantly deviates from the reference class. We demonstrate the validity of our approach and its connection to multiple problems spanning from one-class classification to anomaly detection. In particular, we apply our method to the isolation of neutral showers in an electromagnetic calorimeter and show its performance in detecting anomalous dijets events from standard QCD background. This approach limits assumptions on the reference sample and remains agnostic to the complementary class of objects of a given problem. We describe the possibility of dynamically generating a reference population and defining selection criteria via quantile cuts. Remarkably this flexible architecture can be deployed for a wide range of problems, and applications like multi-class classification or data quality control are left for further exploration.

## 1. Introduction

Nuclear and particle physics are often characterized by problems where one needs to identify physical quantities that (a) *belong to* or (b) *deviate significantly from* a specific ‘reference’ class. In the first case we refer to one-class classification (OCC)—to identify objects of the reference class amongst all objects, in the latter to anomaly detection (AD)—which can leverage on OCC to detect abnormal data points compared to the reference class.

Examples of OCC can be found in an extensive literature review provided by [1]. As for AD, there is a growing number of applications that span from accelerator operations to physics analyses, the latter being of great interest for example at the Large Hadron Collider (LHC) since new physics Beyond Standard Model (BSM) remains elusive (as discussed, e.g. in [2–5]). In both cases, one typically deals with multiple features that vary as a function of the phase space of the final state particles reconstructed in the detector.

In this paper we introduce a novel approach to cope with OCC and AD that leverages two different stages that we call ‘flux’ and ‘mutability’ (F+M). In the first stage, flux, we learn the distributions of the reference class data, and in doing that we utilize a combination of conditional autoencoders (cAEs) [6] and flow-based models [7], particularly conditional masked autoregressive flows (cMAFs) [8], which are conditioned to the kinematics of the particles or events we are trying to distinguish. We refer the reader to [9, 10] for usage of AE’s, [4] for usage of MAF and [11] for the usage of clustering, all under the scope of AD in physics. As we

will describe herein, this allows us to augment the space of the features with a gain in classification performance. The second stage, mutability, consists of addressing if the data in the inference phase—which undergoes a forward pass in the cAE has significantly deviated from the reference class. In other words, at a given kinematics one can dynamically generate a reference cluster in the augmented space and measure if an object belongs to the reference cluster or not<sup>6</sup>. Hierarchical-based clustering (namely, Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [12]) is used to fit these data and come up with a probability cut that provides the confidence level for an object to belong or not to the reference class.

In this work we focus on two applications: (a) distinguishing neutrons from photons in the Barrel Calorimeter (BCAL) of the GLUEX experiment [13] where neutrons in certain kinematic regions are difficult to simulate or isolate from real data and photons are therefore used as reference class; and (b) identifying possible rare BSM dijet events from QCD dijet background events at LHC [3], the latter representing our reference class. In general, the reference could be background-only simulation or signal-depleted data. One of the advantages of our novel and flexible architecture described in the following sections, is that it relies only on the ‘reference’ class and remains agnostic to the class of objects complementary to the reference class during both the training and inference phases. The two classes can be thought of as ‘signal’ and ‘background’ in physics applications. When using strictly supervised methods instead, the model typically requires both signal and background as input in order to learn the feature space and produce a binary output. While these algorithms can be efficient and accurate, they are limited by the quality of data we inject. That is to say, they are prone to any bias we may introduce when constructing our training samples. This can be critical when our control over one of the two classes, (e.g. the background), is limited and we need to rely on the other class (e.g. the signal) or vice versa. Using an architecture that relies on the information of one class removes any assumptions we must make about the complementary class and can be extremely important for different applications: for example, it can be utilized for AD of rare events as well as a method to increase the purity of samples when the original set of real data is characterized by two classes only.

The remainder of the paper is structured in the following way: In section 2 we will describe the developed architecture and provide a detailed discussion of the training and inference phases. In section 3 we introduce the two problems that our architecture has been applied to: detection of neutrons within the GLUEX BCAL, and tagging of  $Z' \rightarrow \bar{t}t$  from QCD dijet background. In section 4 we present our analysis and results. Finally, in section 5 we conclude with a summary and perspectives on future work.

## 2. Flux + Mutability

The F+M approach can be broken into three components, namely: (a) a cAE [6], (b) a cMAF [8], and (c) a clustering algorithm [12].

The cAE provides access to the augmented space via the production of residuals; this space is then generated continuously as a function of the conditions via the cMAF; this in turn allows a comparison between the generated reference cluster (see footnote 6) with the observed objects via clustering. More details on each step can be found in what follows:

- (a) The cAE is trained to reconstruct features as a function of kinematic parameters. In this paper we will show two examples: 1. identifying single neutral showers that depend on 14 reconstructed observables which vary as a function of the shower energy and location in the BCAL calorimeter at GLUEX; and 2. analyzing topologies of events at LHC characterized by 2 jets, which are described by 15 reconstructed observables that depend on the transverse momentum of the jets. Note that in both problems the kinematic variables and the injected feature are continuous and they are scaled prior to injection to allow better convergence of the networks (cAE and cMAF) (e.g. features are scaled in the interval  $[-1,1]$ ). The reader can find more details about these datasets in section 3.

This model is trained first, independently of the cMAF, deploying Huber Loss (see equation (1)):

$$\mathcal{L}_{\text{cAE}} = \begin{cases} \frac{1}{2}(x - x')^2, & \text{for } |x - x'| \leq \delta \\ \delta(|x - x'| - \frac{1}{2}\delta), & \text{otherwise.} \end{cases} \quad (1)$$

We define our convergence criteria as a plateau in validation loss for ten consecutive training epochs. About 10 epochs allows the training to be robust to any fluctuations in validation loss and converge to a more optimal state. For the cAE’s we have explored a grid of points for the parameters—latent space

<sup>6</sup> The term *reference cluster* has been introduced in this work to indicate a large set of points in the augmented (residual plus features) space representative of a class of particles. Conceptually, this can be considered a probability density function (PDF) in that space.

dimensions ( $\mathcal{Z}$ ), number of layers and the Huber loss delta ( $\delta$ ), and selected those that provide optimal convergence. Using this trained model we forward pass all training samples and obtain both the reconstructed vectors ( $\mathbf{x}'$ ) and the residual vectors ( $\mathbf{x}' - \mathbf{x}$ ). The original features and corresponding residuals are then combined into an augmented space. Namely, the augmented space will consist of 28 dimensions for the GLUEX and 30 dimensions for the LHC problems, respectively.

- (b) This augmented dataset is then used to train the cMAF. We deploy similar convergence criteria as for the autoencoder. For the cMAF, the crucial parameter is the number of bijections used. Generations of the reference class were studied in both cases (GlueX and LHC), increasing bijection numbers until generation qualitatively improved compared to the actual distribution across the phase space. Let  $\mathbf{x} \in \mathbf{X}$  denote an element from the set of input vectors within the training dataset,  $\mathbf{k} \in \mathbf{K}$  the conditional vector for the kinematics, and  $\mathbf{z} \in \mathbf{Z}$  represent the Gaussian vector given by the invertible bijection  $f^7$ . A conditional flow with  $N$  layers can be described by:

$$\mathbf{x} = f(\mathbf{z}) = f_N \circ f_{N-1} \circ \dots \circ f_1(\mathbf{z}_0). \quad (2)$$

The logarithm of the transformed probability is then given by equation (3), where  $\pi$  denotes the probability under a Gaussian distribution:

$$\log p(\mathbf{x}|\mathbf{k}) = \log \pi(f^{-1}(\mathbf{x})|\mathbf{k}) - \sum_{i=1}^N \log \left| \det \left( \frac{\partial f_i^{-1}(\mathbf{x})}{\partial \mathbf{x}} \right) \right|. \quad (3)$$

The loss function is then given by the negative log-likelihood:

$$\mathcal{L}_{\text{cMAF}} = -\frac{1}{|\mathbf{X}|} \sum_{\mathbf{x} \in \mathbf{X}} \log p(\mathbf{x}|\mathbf{k}). \quad (4)$$

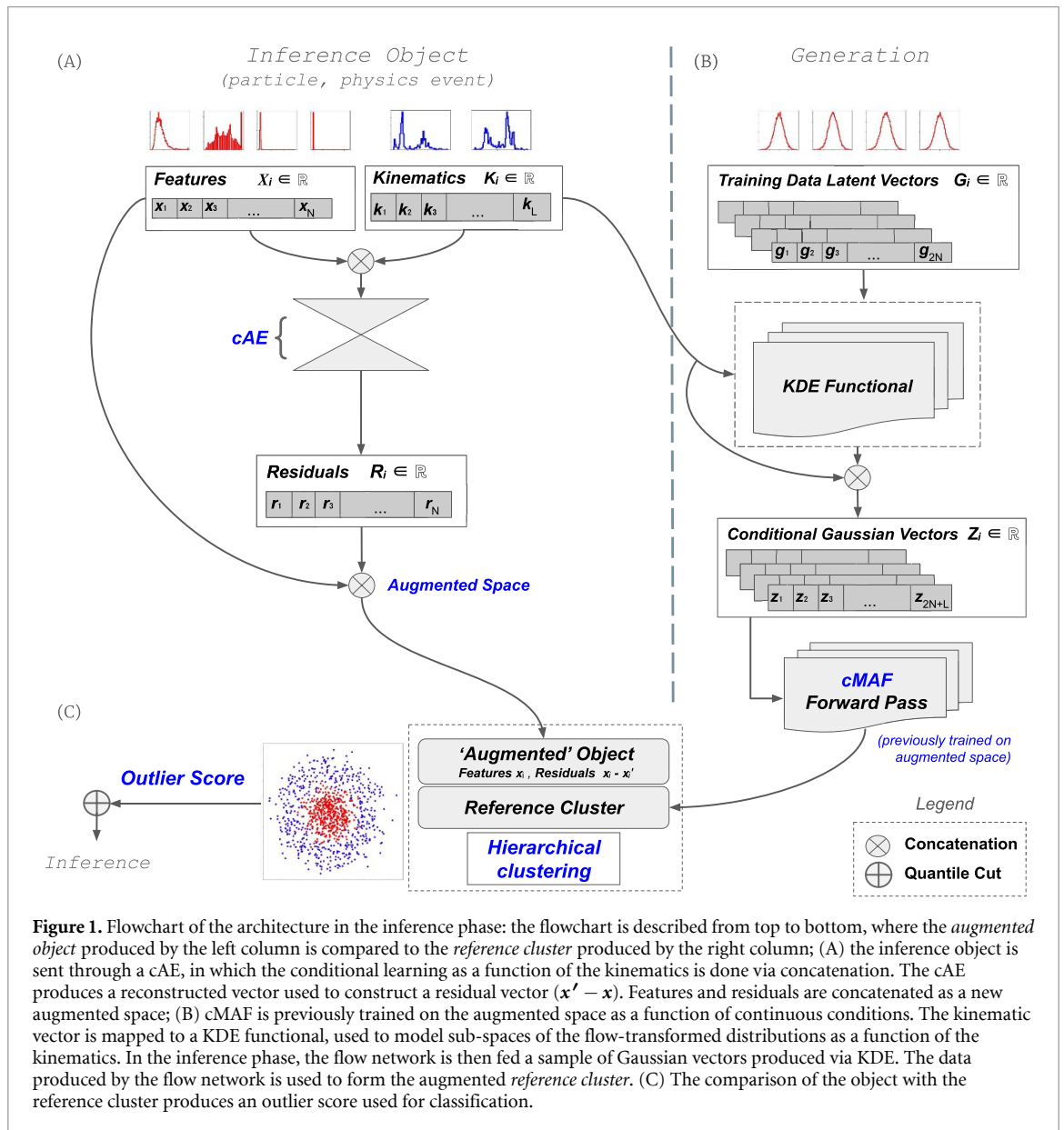
We observed that an augmented space made by the features and the residuals ( $\mathbf{x}' - \mathbf{x}$ ) increases the separation power. As it will be shown in section 4, our reference class data at each kinematics can be represented by a cluster in the feature space that is normalized on a hypersphere. It turns out that features provide localization in space, while residuals push events deemed ‘anomalous’ radially outward, in otherwords, the augmented space with residuals allows the extraction of clusters originally nested within the main population. Hence, cMAF is trained on the augmented space of residuals and features. The flow network will be used as the conditional generator to form the *reference cluster* in the augmented space as a function of the kinematics<sup>8</sup>.

- (c) The last part of the architecture consists in clustering based on HDBSCAN [12]. This allows us to fit the objects in the inference phase with respect to the reference cluster on an object-by-object basis, i.e. on a *particle-by-particle* basis for the neutral shower identification problem of GLUEX, and on an *event-by-event* basis for the LHC jet problem, as described in section 4. The clustering algorithm depends on two main hyperparameters (*minimum cluster size*, and *minimum samples*), and was found to be quite robust. Given a fixed sized reference cluster of  $N$  points to generate, we set the minimum cluster size slightly less than  $N$  and hold it fixed. We then explored points in the space of minimum samples, in which the parameter was gradually increased while retaining proper return rates across the phase space for a given quantile. This produces the most conservative clustering, utilizing only information of the reference class. The number of objects in the reference cluster ( $N$ ) has been minimized based on an accuracy (with respect to the quantile cut and true positive rate (TPR))—speed trade off. Details on hyperparameters and network structure for the cAE, cMAF and HDBSCAN can be found in tables 6 and 7, respectively.

The following provides more details on the approach taken to deal with certain aspects that characterize the F+M architecture. Section 2.1 describes the continuous conditional generation; section 2.2 covers the separation via clustering and the choice of the dynamic cuts that are applied; finally section 2.3 provides a global overview of the workflow during the inference phase with figure 1 depicting the connection of the components (a), (b), and (c) described in this section.

<sup>7</sup> A bijection is a mathematical function that is both surjective (onto) and injective (one-to-one). Exact details on the bijection  $f$  can be found in [8].

<sup>8</sup> As explained in section 4, for the GLUEX BCAL problem the reference cluster corresponds to the photon showers class, whereas for the LHC jet problem it corresponds to the QCD dijet sample.



**Figure 1.** Flowchart of the architecture in the inference phase: the flowchart is described from top to bottom, where the *augmented object* produced by the left column is compared to the *reference cluster* produced by the right column; (A) the inference object is sent through a cAE, in which the conditional learning as a function of the kinematics is done via concatenation. The cAE produces a reconstructed vector used to construct a residual vector ( $x' - x$ ). Features and residuals are concatenated as a new augmented space; (B) cMAF is previously trained on the augmented space as a function of continuous conditions. The kinematic vector is mapped to a KDE functional, used to model sub-spaces of the flow-transformed distributions as a function of the kinematics. In the inference phase, the flow network is then fed a sample of Gaussian vectors produced via KDE. The data produced by the flow network is used to form the augmented *reference cluster*. (C) The comparison of the object with the reference cluster produces an outlier score used for classification.

## 2.1. Continuous conditional generation

In a particle identification (PID) problem the particles are reconstructed as a function of their kinematics since the response of the detector depends on the kinematics of the incoming particle. A conditional approach allows for a comparison with other classical established methods in PID which depends on the kinematics. The response of the detector is learnt and represented by a reference cluster that is conditioned to the kinematics of the particle. This naturally extends to event classification problems, where high-level observables based on low-level observables from the detector response are generated by a reference cluster as a function of the kinematics of the particles in the event.

Continuous conditions give rise to the problem of sparsity within the dataset, meaning low numbers of events per kinematic condition or potentially unobserved kinematics during training. A method to circumvent this issue is pre-binning of conditions such that they become discrete, or performing a one-hot encoding, although this is generally not possible given that there are infinitely many conditions.

We instead choose to take a different approach, in which we allow conditions to remain continuous but enforce sampling from restricted domains. Our method is comparable to the hard and soft vicinal discriminator loss developed in [14], in which the authors use continuous conditionals (also termed regression labels). By deploying such a loss, the authors train the generative model to learn localized sub-spaces in the vicinity of a given condition  $y$  within the approximated density such that interpolation for low datum conditions is possible. In our work, this is achieved using kernel density estimation (KDE) to model the base Gaussian probability distribution of the training data in kinematic bins. That is to say, for each bin in the conditional space, we form a density estimation object in which we can call upon to sample

sets of latent vectors from restricted domains. This sampling method was found to be more reliable than allowing the cMAF to interpolate directly. In inference, we use the inference particle's kinematics to map to a KDE model, fit on the training samples base distribution. This model then generates a sample in the  $2N$  Gaussian space, where  $N$  is the dimensionality of the feature space. We then concatenate the original inference kinematics to each generated  $2N$  Gaussian vector, forming a vector of size  $2N + L$  and forward pass this vector through the cMAF to generate our augmented space.

## 2.2. Separation via clustering and dynamic cut

We introduce an *outlier score* which is obtained comparing each object candidate (e.g. either a particle or a physics event) to the reference cluster. This can be thought of as a probability of being an outlier with respect to the reference class. In other words, it corresponds to the complementary probability of being an inlier:

$$P_{\text{out}} = 1 - P_{\text{in}}. \quad (5)$$

This metric is used to make a decision if the object is more likely to belong to the reference class or not. Every point of the reference cluster is characterized by an inlier probability and therefore by an outlier score. In what follows we detail the process of generating the outlier distribution for the reference cluster. We will then compare the outlier score of the candidate object to the reference distribution.

The following is a brief and concise description of the HDBSCAN algorithm. This corresponds to a refined hierarchical version of a Density-based spatial clustering of applications with noise [15]. The reader can find more details about the algorithm in [16, 17]; for our implementation in particular we utilized the documentation in [12]. HDBSCAN utilizes the *mutual reachability* distance between the points to form a weighted graph, which is in turn used to build the clustering hierarchy<sup>9</sup>. Heuristically the density corresponds to the inverse of a distance,  $\lambda = \frac{1}{\text{distance}}$ ; the smaller the distance between points the higher the density. In order to get more information on the clustering, a *condensed tree* is formed. This tree contains clusters and each cluster contains underlying leaf clusters. This stems from the hierarchical nature of the algorithm. In the algorithm the user has control over hyperparameters such as 'min\_samples' which sets a lower limit on the number of points needed to be considered a core point and for the algorithm to perform any *mutual reachability* calculation. The algorithm prioritizes regions of high density, eventually merging less dense regions with the main cluster if they are reachable under some threshold. *Persistence* is introduced defining a notion of membership based on how long the point was retained in the cluster, i.e. position in the spanning tree, in order to compare the relative distance scales between a fixed cluster and a point in question. We also want to consider a density-based notion of membership. This is done via a modification of the Global-Local Outlier Score from Hierarchies algorithm [17] that allows us to perform the comparison of the points membership persistence with the maximum persistence of the cluster, in order to get a measure of how much of an outlier the point is relative to the fixed cluster.

Combining notions of both distance and density we can now obtain a membership distribution for the reference cluster at each kinematics. This is used to define an outlier metric when classifying new data points. This metric is dynamic in that requires generating a cluster representative of the reference population at any kinematics. Therefore one can define a quantile threshold, which can be some outlier score value corresponding to, e.g. keeping 95% of the population.

It should be stressed that the quantile cut defines the outlier score of the candidate, that is the probability that an object is an outlier with respect to the reference class. Thus, when we classify data via our quantile metric, we define an outlier score cut that corresponds directly to a certain confidence level in data. This metric allows us to remain completely agnostic with respect to the complimentary class, removing the need for semi-supervised methods—which require an example signal in mind during training, see, e.g. [18]—in defining the optimal selection threshold. Implementation details of HDBSCAN are reported in table 6.

## 2.3. Workflow at the inference phase

The workflow of F+M is depicted in figure 1 which describes the flow of an individual object (a shower in the GLUEX BCAL case or a dijet event in the LHC case) in which we wish to perform inference on. The object is initially fed through the cAE, producing both the feature and residual vectors to be augmented.

Subsequently, the kinematics of the input object are mapped to a KDE instance—pre-fit on the transformed distribution of training samples—which in turn produces a set of Gaussian vectors from a restricted domain. It is worth mentioning again that using KDE is necessary, given that we are using

<sup>9</sup> The *mutual reachability* between two points  $(x, y)$  is defined as the maximum value among the *core distance* of  $x$ , the *core distance* of  $y$ , and the distance (e.g. Euclidean) between  $x$  and  $y$ . The *core distance* (e.g. the  $k_{\text{th}}$  nearest neighbor) defines the density of neighbor points. Additional details can be found in [17].

continuous conditions. For a given condition vector  $\mathbf{k}$ , there are not enough data to reliably sample a full width Gaussian, so we must instead restrict the cMAF sampling domain at the generation stage, for which using KDE is an effective approach. The inference kinematics are concatenated to the vectors and fed to the cMAF for a forward pass. The model is then forced to interpolate over the restricted domain at the generation stage. The generated data is normalized either on a hypersphere via a MinMax Scaler (used in the GlueX problem) or by applying a Standard Scaler (used in the LHC problem) and fed to an HDBSCAN clustering instance, forming the reference cluster for the given kinematics<sup>10</sup>. The choice of normalization is driven by the agreement between TPR and selected quantile cut. The inference object is added to the cluster, and classification is performed via the dynamic quantile cut. We directly include the inference object in the initial reference cluster since this greatly improves the speed of the algorithm (saving a second clustering). In doing so we are careful to keep the true reference population large such that an individual point has no influence on the quantile metric.

### 3. Physics applications and corresponding datasets

Our approach can be applied to a plenitude of problems in different research areas. We selected two examples in particular, one related to the identification of neutral showers caused by neutrons in the GLUEX BCAL, the other one to BSM dijet events that significantly differ from SM background at LHC.

#### 3.1. Neutral showers in the GLUEX BCAL

The GLUEX experiment, located at Hall D Jefferson Lab, aims to confirm the predictions of Lattice Quantum Chromodynamics, searching for a class of particles known as exotic hybrid mesons [19]<sup>11,12</sup>. The theory predicts multiplets of exotic mesons with different quantum numbers, and the unambiguous establishment of exotic hybrids requires the full mapping of the hybrid multiplet spectrum. This mapping demands the identification of neutral and charged particles in the final state in several topologies and the validation of the results through consistency checks between different decay modes of the same hybrid meson. Production of charged exotic mesons implies a particle other than a proton must be produced in the reaction. This limits the resulting products to be either a  $\Delta$  baryon or a neutron. Charge exchange can also occur in which the proton provides its charge to a positively charged exotic meson, resulting in the production of a neutral  $\Delta$  and a neutron. In practice,  $\Delta$  baryons are difficult to work with. This is due to large underlying physics background, accompanied by difficulties describing their kinematics, which are necessary for analysis purposes. Ideally, we would like to detect and isolate the neutrons as they do not require detailed modeling of production and decays, and provide constraints to theoretical predictions.

We focus on the BCAL of GLUEX [20], a 400 cm long (about  $115^\circ$  in polar angle coverage) electromagnetic calorimeter designed primarily for photon detection. The detector consists of scintillating fibres compressed between thin layers of lead (see figure 2).

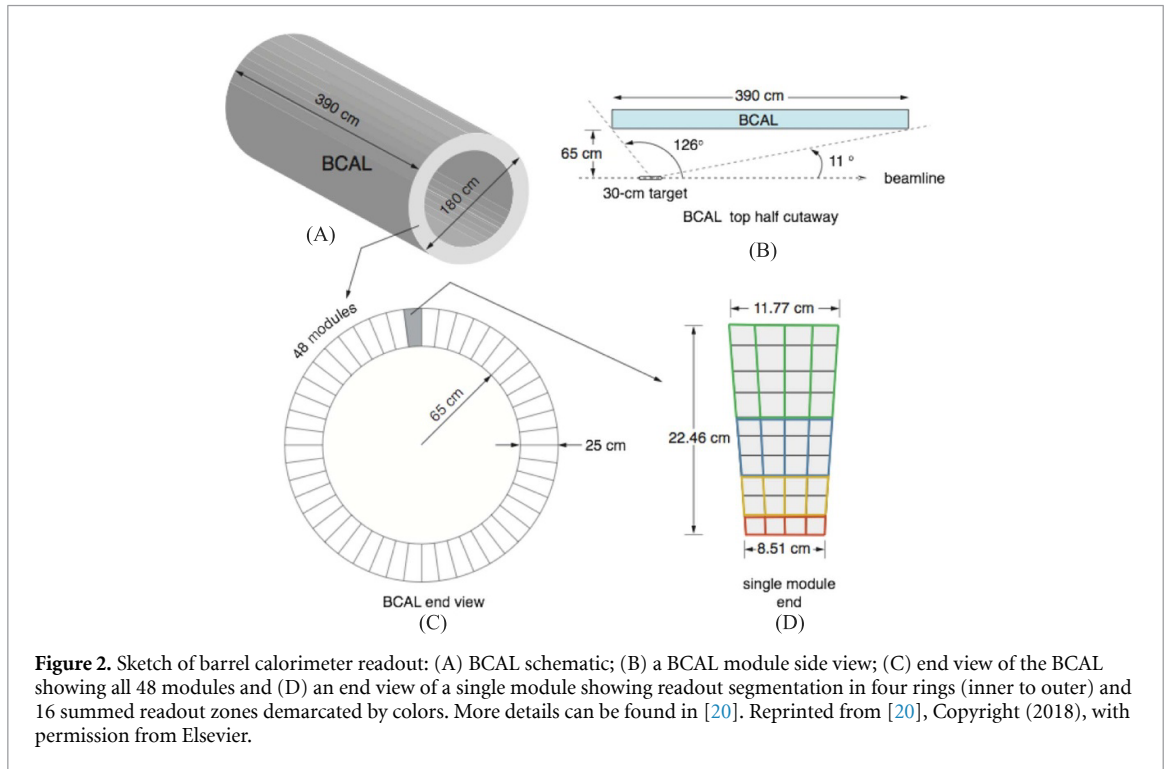
The detector is segmented into 48 azimuthal modules. Each module is partitioned into four readout channels, consisting of double-ended readout using Silicon Photo-Multipliers (SiPM). The GLUEX photon beam is incident on a liquid hydrogen target ( $\gamma + p$ ) which results in many different final states (termed ‘reaction channels’). Many of these particles leave the target and strike the BCAL, creating electromagnetic showers within the detector. It is from these showers we attempt to classify particles based on their profiles. Low-level and high-level observables reconstructed in BCAL are injected in the algorithm as input features (the reader can find a detailed description of the features obtained from the BCAL detector response in appendix A).

Dealing with neutrons is typically more complicated, in part due to the BCAL detector’s response being more difficult to extract compared to a photon. As a result, the extraction of reconstructed neutrons can be affected by sizable uncertainties (see section 4.2 for a detailed discussion). Calibrations from real data are also easier for photons, since one can rely on large samples of standard ‘candles’ like  $\pi^0$  decaying into photons which are abundantly produced at GLUEX and are detected in the BCAL. With this novel architecture, we are able to limit assumptions we impose on neutrons in the training phase of our algorithm as we rely on half the information of supervised algorithms, namely, the training will be based only on detected photons as they typically provide clear signatures of neutral showers. For the proof of principle of our architecture, we will focus on isolated neutral showers, so anything that is not recognized as a photon is

<sup>10</sup> See the [sklearn.preprocessing](#) module for details on MinMax and Standard Scalers.

<sup>11</sup> GLUEX is a fixed target photo-production experiment operating at intermediate energies using an electron beam with nominal energy of 12 GeV.

<sup>12</sup> An exotic hybrid meson possesses explicit gluonic degrees of freedom and has quantum numbers predicted by QCD but forbidden by the simple quark model, which includes only quarks and antiquarks.



then classified as a neutron<sup>13</sup>. The neutral shower reconstruction problem is characterized by 14 features. The dimensionality of this space will be augmented utilizing residuals. The detector response depends on the kinematics of the particles, that is with which energy and at which position the particle interacts in the calorimeter<sup>14</sup>. The dependence on the particle kinematics is encoded in our approach through conditional cAE and cMAF, as explained in section 2. In fact, as it should clear from figure 2, the BCAL design has cylindrical geometry, so the two main kinematic parameters that characterize the reconstruction of the neutral shower are the energy  $E$  of the particle and the  $z$  position at which it strikes the innermost layer.

Training and testing data consists of simulated samples, in which generation of only a specific particle type occurs via GEANT4 particle gun [21], allowing by construction high purity samples in both sets (neutrons and photons). These datasets are private and deploy the Hall D *recon-2019\_11-ver01.2* reconstruction version and 4.35.0 simulation version. Particle samples are simulated in such a way to be approximately flat in reconstructed energy  $E = 200\text{--}2200$  MeV and in  $z = 162\text{--}262$  cm within the BCAL. This corresponds to a region of the phase space that is highly active within the calorimeter.

We initially deploy fiducial cuts on reconstructed widths of showers, namely, width in  $z$ , radial width and width in time. Doing so removes any artifacts left over from the reconstruction process after simulation, in which are careful to apply loose cuts such that we do not impose restrictions on the data and lose sensitivity to the tails of data distributions. A strict requirement of one shower per event is also required to further eliminate any other interactions (for example split-offs in the photon sample)<sup>15</sup>. Since we are interested in only neutrons that mimic photons to a high degree, i.e. those not easily separated via rectangular cuts, we deploy a tight pre-selection on the radius of a shower (must be within the first 3 BCAL layers) and the amount of energy within the BCAL 4th layer (less than 0.1 GeV). The 14 features are comprised of detector response variables and their definitions can be found in appendix A. The entire training dataset consists of  $\approx 1.8\text{M}$  photons in which we reserve about 10% for validation. Testing samples of photons and neutrons are generated independently and each contain about the same number for validation. We condition the model on continuous values of  $E$  and  $z$ , modeling the cMAF latent representations in bins of 4 cm, 40 MeV, values that correspond to a coarse representation of the BCAL photon resolution.

The kinematics of the photons and neutrons detected by BCAL are displayed in appendix B. A comparison of the ‘original’ feature distributions injected in the cAE, the ‘reconstructed’ by the cAE and

<sup>13</sup> Isolated neutral showers do not match with a charged track reconstructed by the GLUEX tracking system, and thus can be clearly separated from charged particle hits.

<sup>14</sup> In the text we equivalently use kinematics or phase space to refer to the kinematic parameters energy and position ( $E, z$ ) of the neutral particle.

<sup>15</sup> Split-offs are defined as photon conversion to an  $e^+e^-$  pair.



those dynamically ‘generated’ by cMAF can be found in appendix C, along with a comparison of the residuals and their corresponding generations.

### 3.2. Dijet AD at the LHC

Our architecture can be utilized in other problems too. Despite the multiple searches for physics BSM conducted at the LHC, new physics remains elusive as of today. In the last few years many novel approaches have been developed for AD in order to detect signal events which would stand out as anomalous with respect to a reference background: these span from new unsupervised AD technique leveraging on neural density estimation [4] to tag-and-train techniques that can be applied to unlabeled data thus offering to be less sensitive to subtle features of jets which may not be well modeled in simulation [22]. In this context, our architecture can be utilized to characterize how anomalous an individual event is with respect to a background events by remaining agnostic with respect to the individual events being analyzed.

In this paper we consider QCD dijet events as background and we look for BSM dijet events from the decay of a  $Z'$ . We utilized a suite of jets for SM and BSM particle resonances which is available on Zenodo (test set version 2.1) [23], provided by the authors of [24]. Primarily, we isolate  $Z' \rightarrow t\bar{t}$  jets (anomalous signal) from SM QCD dijet (background) in order to remove the varying length feature vectors seen in other BSM datasets, such as  $W' \rightarrow W + jj$ . The datasets have been generated with MADGRAPH [25] and PYTHIA8 [26]. The DELPHES [27] framework has been used for fast detector simulation. For a detailed description of the dataset we refer the reader to [23]. The simulated QCD [28] and BSM dijets [23] are produced with the same selection criteria<sup>16</sup>.

Clustering of the jets was done using FASTJET [29], deploying the anti- $k_T$  algorithm [30] with a cone size of  $R = 1.0$ . As stated in [23, 28], events within the datasets must meet the requirement of the leading jet having transverse momenta  $p_T > 450$  GeV and the sub-leading jet having  $p_T > 200$  GeV.

In this case we use only a single conditional, namely the leading jet transverse momentum, and form a fixed length feature vector consisting of the remaining four vector properties of the leading jet, its  $n$ -subjettiness variables, the sub-leading jet four vector and its  $n$ -subjettiness variables<sup>17</sup>.  $N$ -subjettiness provides identification of  $N$ -pronged structures in jets, characterized by the jet’s transverse momenta, radius and distance between predefined axis’ in the azimuth-rapidity plane [31]. This feature vector then gets augmented with its residual vector from the cAE, resulting in a vector of 30 features at inference including residuals. We apply a further condition on the datasets, requiring the leading jet to have  $p_T < 800$  GeV in order to provide sufficient data as a function of the conditional parameter. We model the cMAF’s transformed space in bins of 1 GeV.

The architecture is trained on  $\approx 600$  k QCD dijet events and validated on  $\approx 50$  k, retaining around 50 k for testing. We use only a single top jet file from [23] ( $m_t = 174$  GeV), providing 50 k anomalous events. More details on the feature distributions of both classes can be found in appendix D which includes also a comparison of the ‘original’ feature distributions injected in the cAE, the ‘reconstructed’ by the cAE and those dynamically ‘generated’ by cMAF along with the residuals and their corresponding generations.

## 4. Analysis and results

In what follows, we deploy our architecture on the two different physics scenarios introduced in section 3.

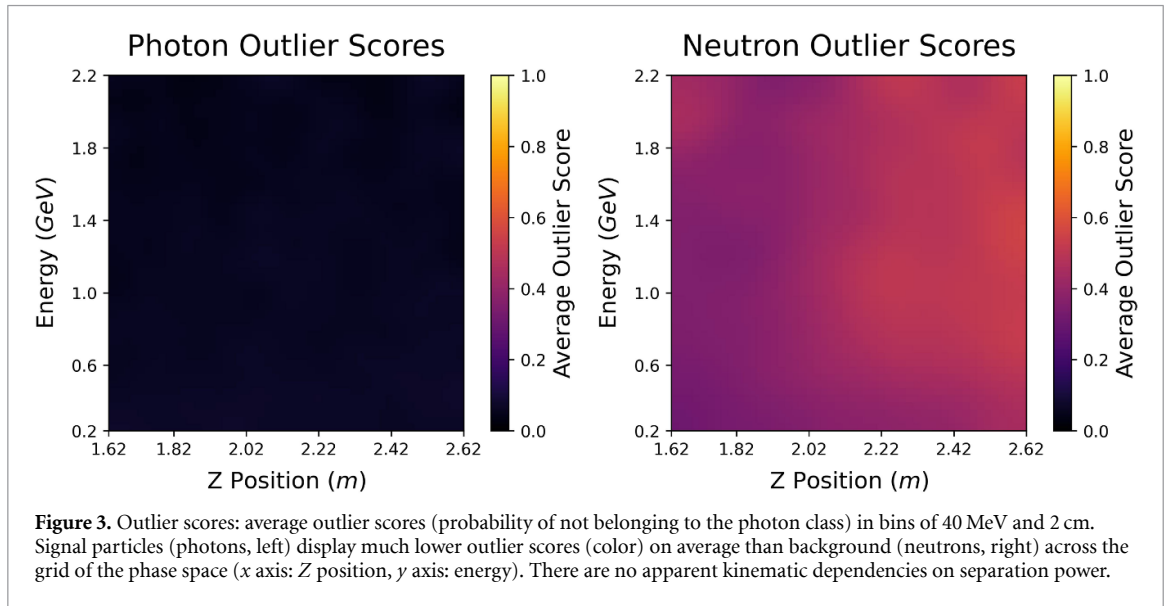
### 4.1. Neutral showers classification with the GLUEX BCAL

We demonstrate the potential of the model as an OCC method for GLUEX photons, which in turn allows to tag neutron candidates from the sample of isolated neutral showers described in section 3.1. As already explained, there are specific regions in the phase space of the BCAL where simulating the detector response to neutrons is challenging because it is characterized by large uncertainty.

Our strategy aims at isolating neutrons by applying cuts on the photon showers, the latter taken as the reference class. As described in section 2, our approach is unsupervised and agnostic to the neutrons; it allows us to dynamically generate a reference cluster in the augmented space of the features as a function of the particle kinematics. The reference cluster is used to establish if a new particle is more likely to belong to the photon class or the neutron class. A quantile cut is applied on an outlier score to determine the probability of a particle to be a member of the cluster or to be an outlier. This approach can be useful when the uncertainty on the distributions of the complementary class (neutrons) is expected to be large compared to the reference class (photons), and the distributions of neutrons and photons cannot be easily separated by

<sup>16</sup> Cheng [23] and Leissner-Martin *et al* [28] include the specific version number for the test datasets as well as information on the version numbers for programs used for generation.

<sup>17</sup> See [31] for mathematical definition of  $n$ -subjettiness.



standard rectangular selections. In such scenarios, fully supervised approaches become less reliable without a proper assessment of the uncertainty quantification. Our approach allows to select the TPR of the reference class which, by construction, is consistent with the quantile cut on the outlier score chosen for the selection. The outlier score of each particle corresponds to a probability of not belonging to the photon reference class, according to equation (5); given that we work with isolated neutral showers which can only be either a photon or a neutron, the outlier score is interpreted as how confident we are to have identified a neutron. In our analysis, we rely on ground-truth information from simulations to measure the performance of the architecture with respect to photons and neutrons. When deployed on real data, for the reference class we can use high purity samples of photons (leveraging standard ‘candles’ like  $\pi^0 \rightarrow \gamma\gamma$ ). A metric for neutron efficiency can be determined from physics reactions where neutrons (and photons) are involved, for example,  $\gamma p \rightarrow n\pi^0\pi^+$  or  $n\pi^+$ , after taking into account background factors which may mimic the above signatures.

Figure 3 depicts the average value of the outlier score in bins of  $E$  and  $z$ <sup>18</sup>. It is easily seen that the average outlier scores for neutrons are much higher across the entire phase space when comparing both plots. It is also apparent that the outlier score of photons is flat and close to zero as a function of the kinematics and for neutrons it is large in value and rather uniform in distribution too, despite the reconstructed features do largely depend on the kinematics of the particles (as displayed in figures B1 and B2). This means the architecture has provided an approximately uniform and good separation power in the phase space we are covering. Deploying a 95% quantile cut, we obtain a TPR for photons, and a true negative rate (TNR) for neutrons of **95.03%** and **52.70%**, respectively, as summarized in table 1. TPR and TNR are quite large and to our knowledge exceed by far results obtained with traditional rectangular selections [32].

We note that by assuming both photon and neutron training datasets are reliable, then deploying a fully supervised model like XGBoost [33] would result in a TPR and TNR at about 92% each, but again this is not the scenario we are tackling here<sup>19</sup>. XGBoost has been prototyped and studied with BCAL simulations, and a semi-optimal set of hyperparameters was obtained. Efforts have been made to improve the prediction head of the algorithm, in which we replaced the clusterer with a one-class support vector machine (OC-SVM). The conditional generations (see figure C3 for an example of generations produced via the cMAF) are then used to fit the OC-SVM and predict outliers (i.e. neutrons). In smaller scale tests it was found that the TPR of photons suffered drastically, at around 56%, while offering a slight increase in the TNR of neutrons at around 90%. These results are most comparable to the results obtained deploying a 68% quantile cut with clustering, yet pose no real performance increases. Using the OC-SVM also limits our ability to employ a quantile cut. The SVM-based method was deemed inferior to the clustering method used in our approach and not explored further.

<sup>18</sup> The binning in  $z$  is smaller than our KDE functionals in order to obtain a square grid for plotting.

<sup>19</sup> For completeness, we also utilized our F+M approach considering both classes as hypotheses to build a figure of merit of the form of a  $\Delta \log \mathcal{L}$  to apply a cut on. Results are outperformed by other methods, e.g. XGBoost, which seem more suitable for a fully supervised task.

**Table 1.** Summary of GLUEX BCAL results: photons are used as the reference class, and results are obtained using different quantile cuts on the outlier score. Fluctuations in TPR are within the error.

Quantile	TPR	TNR
68%	$68.15 \pm 0.18 \%$	$87.48 \pm 0.13\%$
95%	$95.03 \pm 0.08 \%$	$52.70 \pm 0.19\%$
99%	$98.96 \pm 0.04 \%$	$35.44 \pm 0.18\%$

#### 4.2. Uncertainty in the neutron sample

We further showcase the utility of our architecture when large uncertainty affects the class complementary to the reference class. This translates into uncertainties injected into machine learning models, specifically those who are fully supervised and rely on both photons and neutrons. Indeed neutron simulations in the phase space of interest described in this study can differ significantly from real data. F+M attempts to alleviate this problem by relying on knowledge of only the other neutral candidate, the photons, for which is possible to obtain high purity samples from real data.

We show how this affects a fully supervised approach such as XGBoost and make a comparison to our OCC approach which benefits from the usage of residuals. While the original injected neutron distributions overlap to a great extent with the photon distributions to begin with, we push this overlap to an extreme case: we artificially create an extra testing dataset of neutrons in which we ‘scale’ the neutron features in such a way to make them highly resemble photons. We pretend this new sample to be the *actual* data observed in an experiment representative of the true detector response, while we consider the original sample to be the *simulated* data in which we assume there exists  $\sim 10\%$ – $15\%$  difference from actual data. Figure C4 shows a comparison between photons and neutrons of the injected feature distributions. For the neutrons, we also show the case of the ‘scaled’ (i.e. actual) distributions. Similarly, figure C5 shows a comparison for the distributions of the features as reconstructed by the cAE; figure C6 shows the same comparison for the residuals. Finally, figure C7 shows a comparison for the distribution of the outlier scores.

XGBoost is then trained on the simulated neutron sample, and we compare performance on both *simulated* and actual (scaled) samples in the inference phase. In this study we use the TPR obtained with XGBoost to define the quantile cut for our architecture and make a comparison. Table 2 shows the results of this study. XGBoost TNR performance drops from about 92% to 79% when deployed to what we consider the actual data in this example. This is the result of a decision boundary obtained in the training phase using an inaccurate neutron sample.

On the other hand, F+M is an OCC approach that relies on photons only and is agnostic to the neutron sample which is ‘seen’ only during the inference phase. The TNR performance increases from 60% to 83% when deployed on the ‘actual’ neutron sample. The reason for the increase is likely due to learning correlations in the kinematics in the augmented space. In fact, while the new neutron features are artificially made more photon-like by scaling their distributions and hence look harder to separate from photons, their correlations with kinematics which cAE is able to pick up changed, and the resulting residuals are more easy to separate. This is further confirmed via the results obtained using the feature space only, in which performance drops in with respect the unperturbed neutron sample.

A result provided by XGBoost in this example would provide a large TNR of 92% but it is affected by a large systematic as indeed the true performance would be 79%. F+M training does not depend on the neutron sample, and therefore the TNR performance is not affected by the same large uncertainty and it actually provides in this particular case a larger value of 83%. We note that:

- The result of the TNR for F+M depends on how the actual data look like in the inference phase, i.e. the opposite result applies by switching labels in table 2;
- OCC is agnostic to the complementary class, which is unlabeled. External physical information can be used to label neutron data, e.g. this may depend on the event topology containing the neutron;
- The outlier score of each particle represents a proxy for the confidence level of its classification; it is worth reminding that the quantile cut is defined by the photon sample and is dynamical in that depends on the kinematics; we notice that the neutron’s outlier score increased on average for the new distributions, meaning that the uncertainty on each individual classified neutron is also decreased. More details can be found in figure C7.

#### 4.3. AD of dijet events at LHC

We deploy our algorithm for AD of BSM dijets with respect to a background of QCD dijet events. We consider the datasets described in section 3. The performance of our algorithm is compared to other works

**Table 2.** Neutron sample study: the table shows a comparison of the performance obtained using two neutron samples, one assumed to be simulated and the other one considered as the ‘actual’ detector response to neutrons. Differently from neutrons, photons simulations are more accurate and in agreement with real data. TPR is the true positive rate for photons and TNR the true negative rate for neutrons. XGBoost is trained on simulated photons and neutrons. F+M relies only on simulated photons. XGBoost TNR performance drops when deployed on actual data. The increased TNR of F+M depends on the residual space produced by the cAE which captures a different kinematic dependence of the neutron features compared to that of the photons in the actual case.

Algorithm	Simulation		‘Actual’ Detector Response	
	TPR	TNR	TPR	TNR
XGBoost	$92.15 \pm 0.10\%$	$91.93 \pm 0.10\%$	$92.15 \pm 0.10\%$	$78.82 \pm 0.15\%$
F + M (Augmented)	$92.01 \pm 0.18\%$	$60.34 \pm 0.10\%$	$92.45 \pm 0.10\%$	$82.86 \pm 0.14\%$
F + M (Features)	$92.44 \pm 0.10\%$	$56.24 \pm 0.19\%$	$92.45 \pm 0.10\%$	$49.31 \pm 0.19\%$

**Table 3.** AUC score comparison: AUC score comparison between our architecture and two methods, Fraser *et al* [5] and Cheng *et al* [24]. Our architecture performs on par with [24] within the uncertainty and slightly better than [5]. It should be noticed that our architecture can be further optimized by tuning its hyperparameters. This will possibly further improve the results.

	Ours	From [5]	From [24]
AUC	$0.891 \pm 0.005$	0.87	0.89

**Table 4.** Summary of dijet results at LHC: results are obtained using different quantile cuts on the outlier score. Fluctuations in TPR are within the error. For comparison, we also include a baseline rectangular selection based on loose fiducial cuts on each feature, defined in such a way to select when combined 99% of the SM QCD dijet events.

Quantile	TPR	TNR
68%	$68.48 \pm 0.22\%$	$93.05 \pm 0.06\%$
95%	$95.27 \pm 0.10\%$	$43.07 \pm 0.22\%$
99%	$99.04 \pm 0.05\%$	$12.74 \pm 0.15\%$
Fiducial cuts (99%)	$98.92 \pm 0.05\%$	$2.35 \pm 0.06\%$

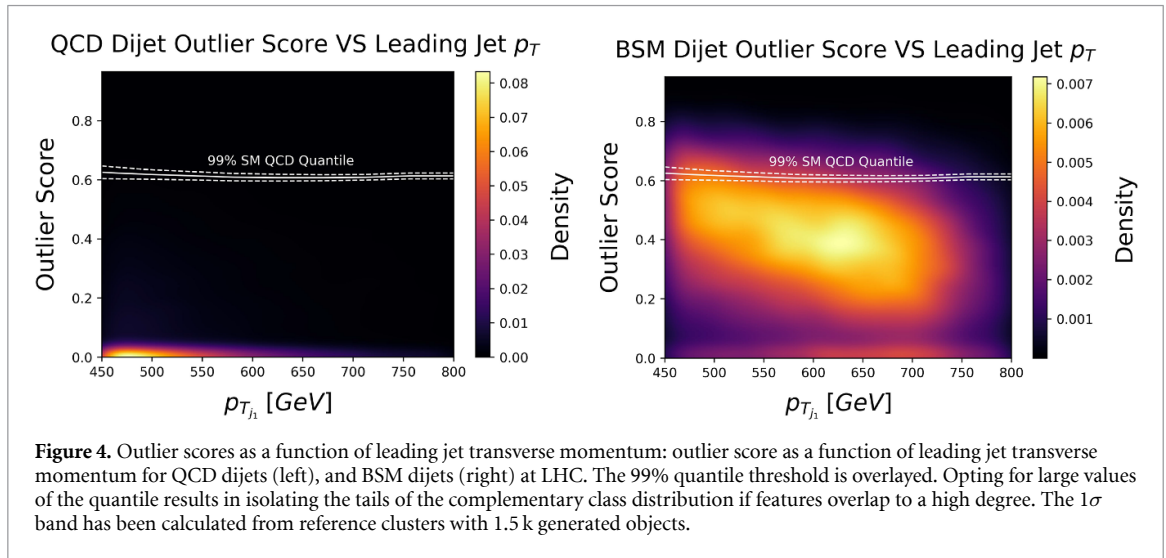
[5, 24] that used the same dataset and considered as a metric the area under the receiver operating characteristic curve (AUC).

However, it should be noted that the AUC by construction is not agnostic to the BSM signal in the AD problem: in fact it provides model performance as a function of threshold cuts. In order to pick the optimal threshold one must have prior knowledge of the anomalous class itself, that is not always possible and if it is, implies a bias towards the model which in turn becomes weakly supervised. There are of course other methods to identify a suitable cut remaining agnostic towards the anomalous sample, yet these values may be far from optimal. Thus AUC can be an inflation of true model performance. The AUC is computed with a ‘standard’ approach used by sklearn [34], namely numerical integration using the trapezoidal method. We propagate the uncertainty in both efficiencies via a bootstrapping method. We iterate this process 100k times, taking the mean AUC and standard deviation as our uncertainty.

We report the AUC values in table 3 and compare to the best values obtained by [5] and [24] using the same dataset. In [5, 24] different AUC scores are listed based on different settings, loss functions, etc. Some knowledge of the anomalous class is utilized in order to define the optimal threshold. The AUC score of our architecture is slightly larger than in [5] and consistent within the uncertainty with that of [24]. It should be noticed that our architecture can be further optimized by tuning its hyperparameters. This will possibly further improve the results shown in table 3.

In the following we calculate TPR and TNR. The idea is that of remaining agnostic to a potential BSM dijet signal, and changing the quantile cut on the QCD dijet background. For example, one may consider setting the threshold of our architecture to  $\approx 100\%$  for QCD dijets, very naively letting only largely anomalous BSM samples to stand out from the selected distributions. For completeness we show different scenarios (including also low quantile cuts) in table 4. Figure 4 depicts the outlier scores of both QCD and BSM dijets at LHC as a function of the leading jet transverse momentum, in which the isolation of the tail of the BSM distribution is visualizable with respect to the 99% quantile threshold. One may be tempted to instead opt for a global cut within the outlier space, however this explicitly demands prior knowledge of the complementary class, directly violating the conditions of AD frameworks.

As is clear in all our tables the TPR is consistent by construction with the quantile cut applied to the outlier score obtained using the generated reference cluster.



**Table 5.** Features vs augmented space for GLUEX BCAL and LHC problems. The benefits of residuals can be concluded via comparison of TNR at equal thresholds. Note the consistency between threshold and TPR by design in which fluctuations across differing spaces are within error.

Space	Quantile	(GLUEX)		(LHC)	
		TPR	TNR	TPR	TNR
Features	68%	68.15%	86.12%	68.49%	72.25%
Augmented	68%	68.15%	87.48%	68.48%	93.05%
Features	95%	95.05%	48.16%	95.18%	13.76%
Augmented	95%	95.03%	52.70%	95.27%	43.07%
Features	99%	99.01%	26.30%	98.98%	3.09%
Augmented	99%	98.96%	35.44%	99.04%	12.74%

#### 4.4. Benefits of the augmented space: residuals

We have demonstrated the good performance of our architecture for different physics problems. In order to illustrate the performance increase via residuals, we follow the same inference procedures discussed before except we remove the residuals as input to the cluster. Table 5 shows a comparison between results obtained using the feature space and the augmented space of features plus residuals. The comparison is done for both problems (neutron identification in GLUEX BCAL and dijet AD at LHC). In both cases, we observe a systematic improvement in the TNR by using the augmented space.

We deem the residuals to be highly valuable in terms of separation and provide further evidence that features localize the space, and the residuals push nested clusters radially outward to be more easily extracted and seen as outliers. This interpretation is also represented in figure 5 which shows a t-SNE representation of the feature and augmented space for the GLUEX problem. As shown in the figure, augmenting the space with residuals produces a better distinguishing power between photons and neutrons.

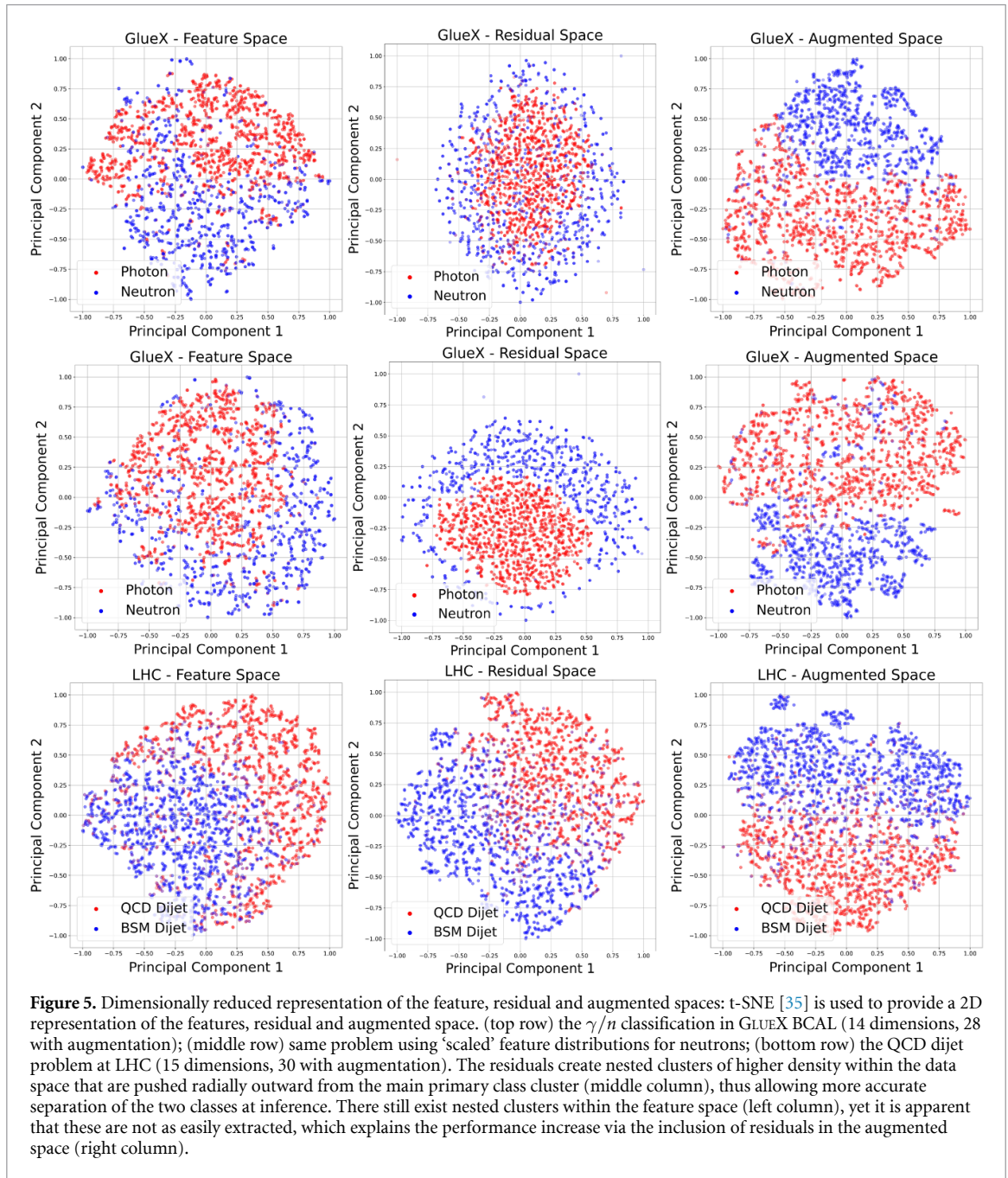
#### 4.5. Architecture specifications and computing resources

During the inference phase, in both problems, we are limited by the generation speed of the cMAF and the clustering. HDBSCAN is in fact not optimized to run on GPU and the fitting procedure dominates the computing time<sup>20</sup>. Considering these limitations, we generate only 1.5 k reference samples per particle in inference. We show in both cases the statistics are sufficiently high to run our analyses.

The architecture has not undergone a rigorous optimization and we expect better results could be obtained. Table 6 contains hyperparameter settings used throughout our experiments, and table 7 contains relevant network configurations. One can imagine optimizing the pipeline end-to-end under a Bayesian process, in which we rely on the signal class only. Key hyperparameters to tune are those of the clusterer. Training was done utilizing google services on Nvidia A100-SMX4-40GB and Nvidia V100-SMX2-16GB cards with TensorFlow 2.8.0 and TensorFlow-Probability 0.16.0 builds. Inference was supported via Compute Canada on Nvidia P100 PCIe 16 GB card<sup>21</sup>. The number of parameters correspond to the GLUEX BCAL problem. Additional technical details on the architecture can be found in table 8.

<sup>20</sup> Inference time on average is 3.3 s on a P100 PCIe 16 GB card.

<sup>21</sup> Cedar GPU clusters were used. See [www.computecanada.ca/](http://www.computecanada.ca/) for more details.



**Table 6.** Baseline hyperparameters of the F+M architecture: these values are not fine-tuned, but have shown to be reliable as an initial starting point.

Description	Symbol	Value
Huber Loss Delta	$\delta$	1.0
cAE Latent Space Dimension	$\mathcal{Z}$	6
cAE Learning Rate	$\lambda_{\text{cAE}}$	$5 \times 10^{-4}$
cMAF Learning Rate	$\lambda_{\text{cMAF}}$	$1 \times 10^{-4}$
GLUEX cMAF Bijections	$K_{\text{GlueX}}$	12
LHC cMAF Bijections	$K_{\text{LHC}}$	10
HDBSCAN Minimum Cluster Size	$M_{\text{CS}}$	1000
HDBSCAN Minimum Samples	$M_{\text{S}}$	100
Reference Cluster Size	$N$	1500

**Table 7.** Network structure of relevant F+M components: description of cAE sub-architecture, specified by the type of neural network. The cMAF bijections are predefined and we instead report the number used as a hyperparameter.

Architecture name	Type	Neurons	Activation function
GLUEX Encoder/Decoder	Dense	56/42/28/14/8	ReLU
		8/14/28/42/56	ReLU
LHC Encoder/Decoder	Dense	60/45/30/15/8	ReLU
		8/15/60/45/60	ReLU

**Table 8.** Specs of the F+M architecture: the table reports the average inference time per particle, the inference memory and the training memory, i.e. the GPU memory required by the network during inference and training phases. Training was done utilizing google services on Nvidia A100-SMX4-40GB and Nvidia V100-16GB cards with TensorFlow 2.8.0 and TensorFlow-Probability 0.16.0 builds. Inference was supported via Compute Canada on Nvidia P100 PCIe 16 GB card. The number of parameters correspond to the GLUEX BCAL problem. The GLUEX  $\gamma/n$  problem has slightly larger numbers due to extra layers in the cMAF needed to reproduce multi-modal distributions in the photon sample.

Specs	Value
inference time per particle	$\sim 3$ s
inference network memory	$\sim 1$ GB
training network memory	$\sim 6$ GB
network memory on local storage	$\sim 14$ MB
GLUEX cAE trainable parameters	10 172
GLUEX cMAF trainable parameters	3700 704
LHC cAE trainable parameters	11 545
LHC cMAF trainable parameters	3103 920

## 5. Summary and conclusions

We have developed a novel architecture that consists of two steps called ‘flux’ and ‘mutability’ (F+M). In the first stage we learn the distribution of a reference class. In the second stage we address if data significantly deviates from the reference class. The backbone of this architecture consists of: (a) a cAE that allows to reconstruct the injected features and calculate the residuals which combined to the first make an augmented space; (b) a cMAF, that combined with a KDE allows in the inference phase to dynamically generate the reference class in the augmented space; (c) a hierarchical-based clustering algorithm that allows to estimate if an object belongs to the reference class by producing an outlier score, which can be also seen as an estimate of how confident we are about the object belonging to the reference class.

We demonstrated its capability as a one-class-classification method when dealing with isolated neutral showers at GLUEX BCAL, providing good separation between photons (reference class) and neutrons (complementary class) while relying on only information related to the reference class. We then proved the advantage of this approach which is agnostic to the neutron sample, in particular when the latter is affected by large uncertainty.

We also showed the capability of the algorithm as an AD method, isolating possible BSM  $Z' \rightarrow t\bar{t}$  dijets topologies from SM QCD dijet background (reference class) at the LHC. We demonstrated that our model performs on par with other architectures using the same dataset, yet we are able to remain truly agnostic towards the complementary class using our final quantile metric. A possible extension of this problem that has been left for future exploration consists in conditioning on both  $p_T$  and the invariant mass of the leading jet. We also note in general that our architecture has not undergone rigorous optimization of the hyperparameters and we therefore expect further increase in performance in doing so.

In both cases, we have demonstrated an increased performance via inclusion of the residuals. We have concluded that an augmented space (features + residuals) is ideal for inference. The features localize the space for a given kinematics, and the residuals push the complementary class radially outward in the hypersphere used for clustering. This allows nested clusters existing in the data space to be more easily extracted and increase distinguishing power.

The inference time per particle is fairly slow. As such, the architecture is best suited for offline analysis purposes in its current state, in which it can be optimized via parallelization. Reduction in inference time can potentially be obtained via the use of a different flow network as MAF is generally known for its slower generation speeds. Other NF models have not yet been tested and are left as further exploration. A large bulk of the inference time remains at the clustering stage as HDBSCAN is not optimized to run on GPU. Other clustering methods, or an improved GPU-optimized HDBSCAN build could potentially solve this issue. In the future we plan to extend this work to an application for data quality monitoring in an experiment, in that

significant deviations from the expected quantiles of the reference distributions could determine if a new calibration/alignment is needed.

## Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: [10.5281/zenodo.4614656](https://zenodo.org/doi/10.5281/zenodo.4614656).

## Acknowledgment

The work of C F is supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under Grant No. DE-SC0019999. The work of J G and Z P is supported by the Natural Sciences and Engineering Research Council of Canada Grant SAPPJ-2018-00021. We would like to acknowledge the GLUEX collaboration for the simulations used in the neutron identification problem. We also want to thank E Cisbani and M Williams for useful comments.

## Appendix A. GLUEX BCAL feature definitions

All showers are **DBCALShower** objects in the GLUEX software package. We denote these showers as  $S$ , and label the quantity we use via a subscript ( $S_x$  denotes the  $x$  position of the shower object for example).  $R$  denotes the inner radius of the BCAL (64.3 cm radially outwards from the center line of the target) and  $T_z$  denotes the center  $z$  position of the hydrogen target (65 cm from the upstream edge of the BCAL). Showers are comprised of points, we define our features using the energy weighting of these points<sup>21</sup>.

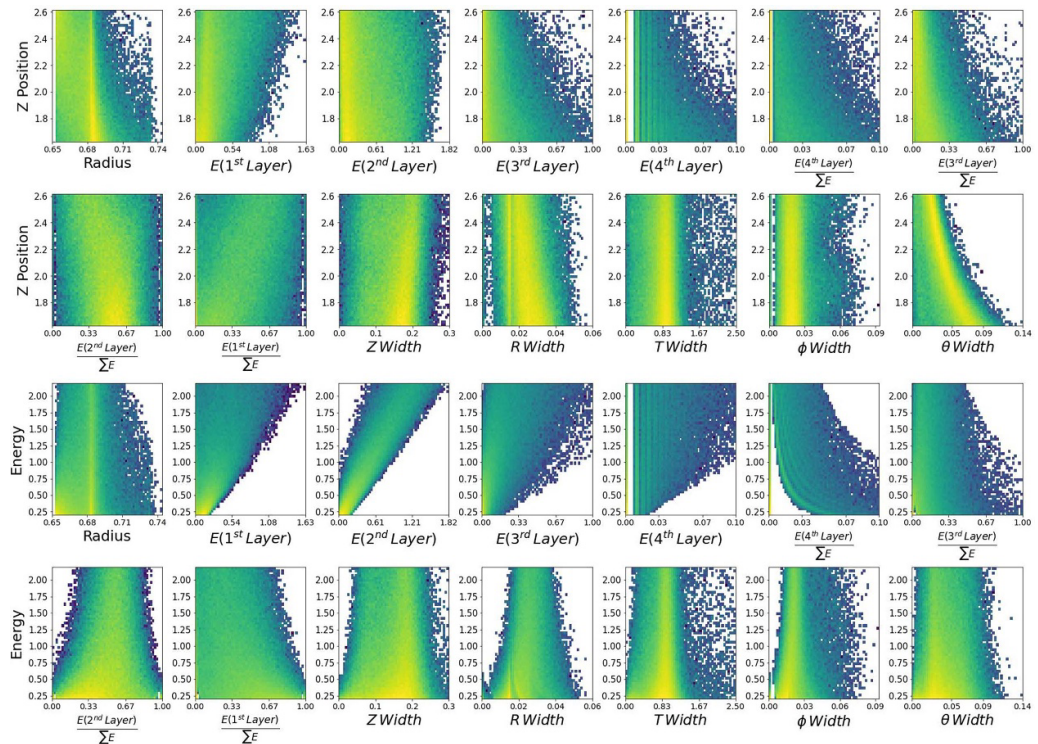
- **LayerM\_E** =  $\sum_i^N E_i M \in \{1, 2, 3, 4\}$  is the layer number and  $E_i$  is the energy of the  $i^{\text{th}}$  reconstructed point in the layer.
- **LayerMbySumLayers\_E** =  $\frac{1}{E_{\text{total}}} \sum_i^N E_i M \in \{1, 2, 3, 4\}$  is the layer number and  $E_i$  is the energy of the  $i^{\text{th}}$  reconstructed point in the layer.
- **Z Width** =  $\sqrt{\frac{1}{E_{\text{total}}} \sum_i^N E_i (\Delta z_i)^2}$ ,  $\Delta z_i = (z_i + T_z) - S_z$   
 $E_i$  and  $z_i$  are the energy and  $z$  position of the  $i^{\text{th}}$  point in the shower.
- **R Width** =  $\sqrt{\frac{1}{E_{\text{total}}} \sum_i^N E_i (\Delta r_i)^2}$ ,  $\Delta r_i = (R - r_i)$   
 $E_i$  and  $r_i$  are energy and radial position of the  $i^{\text{th}}$  point.
- **T Width** =  $\sqrt{\frac{1}{E_{\text{total}}} \sum_i^N E_i (\Delta t_i)^2}$ ,  $\Delta t_i = t_i - S_t$   
 $E_i$  and  $t_i$  are the energy and timing information of the  $i^{\text{th}}$  point.
- **$\theta$  Width** =  $\sqrt{\frac{1}{E_{\text{total}}} \sum_i^N E_i (\Delta \theta_i)^2}$ ,  $\Delta \theta_i = \theta_i - S_\theta$   
 $E_i$  and  $\theta_i$  are the energy and polar angle (from the target center) of the  $i^{\text{th}}$  point.
- **$\phi$  Width** =  $\sqrt{\frac{1}{E_{\text{total}}} \sum_i^N E_i (\Delta \phi_i)^2}$ ,  $\Delta \phi_i = \phi_i - S_\phi$   
 $E_i$  and  $\phi_i$  are the energy and azimuthal angle of the  $i^{\text{th}}$  point.
- **$z$  Entry** =  $(S_z - T_z) \frac{R}{S_r} + T_z$   
The position at which the particle hits the inner radius of the BCAL.

## Appendix B. GLUEX kinematic plots

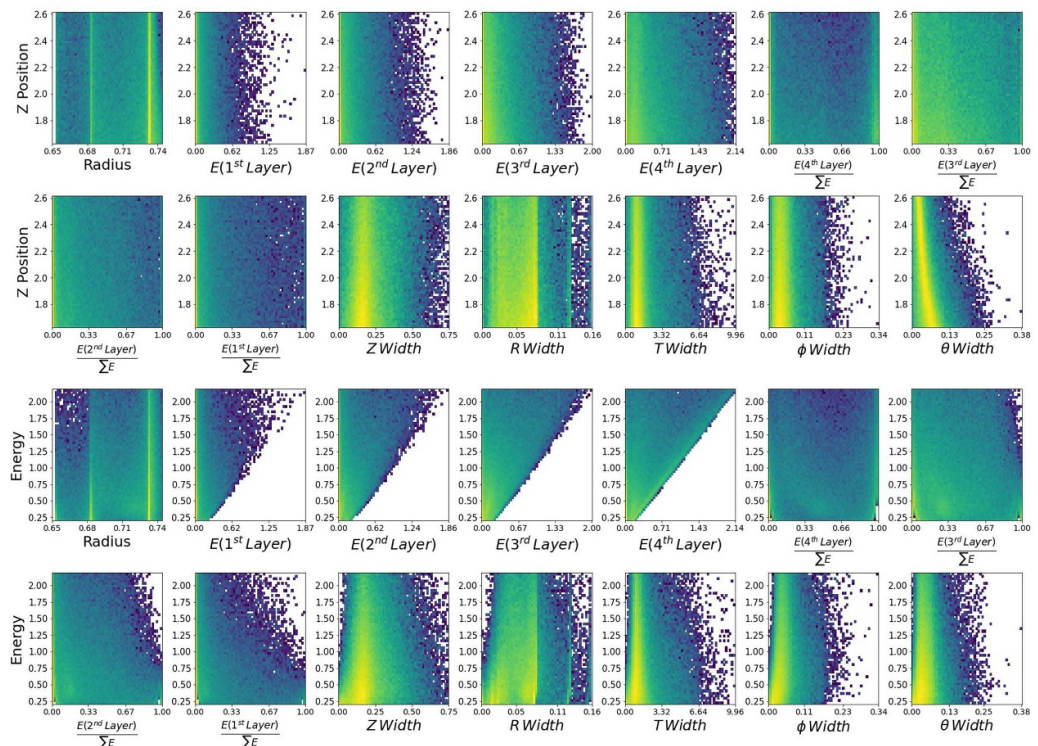
Plots contained in this section illustrate the 14 features (see appendix A) as a function of kinematics for both simulated photons and neutrons, as described in the captions. These plots are integrated over the entire phase space used in the analysis. The functional dependence of features on the phase space ( $E, z$ ) can be clearly seen in the simulated samples, more so for the photons as the simulation of a neutron interaction is more difficult. As one moves to training on real data from standard ‘candles’ such as  $\pi^0 \rightarrow \gamma\gamma$ , the dependencies become more pronounced. The distributions of the features with respect to the  $z$  position and energy in BCAL are shown in figures B1 and B2 for photons and neutrons, respectively.

<sup>21</sup> Hit—SiPM trigger at one end of the BCAL. Point—SiPM trigger at both ends of the BCAL. Points are used for: a) to find  $z$  position of a shower. b) noise reduction (two ended coincidence).



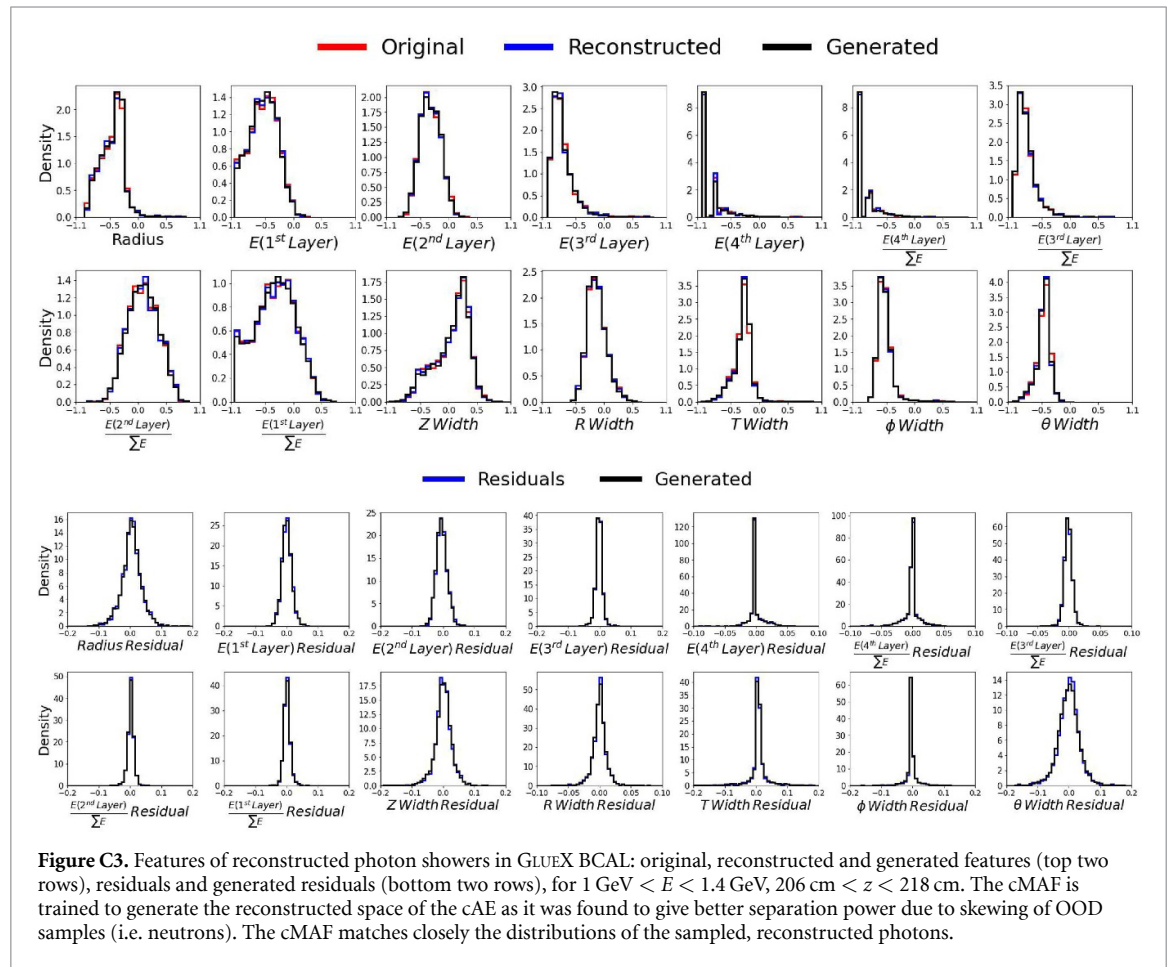


**Figure B1.** 2D histograms of photon features ( $x$ -axis) and  $z$  position (or energy) ( $y$ -axis): we see a clear dependence on  $z$  and energy for most of the features within the space. (top two rows) Features such as  $T$  Width, and  $\phi$  Width display less of a  $z$  dependence on average. (bottom two rows) Any feature corresponding to energy deposition within layers has a large dependence on  $z$  and  $E$ , along with width variables. By conditioning on  $z$  and  $E$ , we are able to capture the functional dependence of detector response at the generation stage.



**Figure B2.** 2D histograms of neutron features ( $x$ -axis) and  $z$  position (or energy) ( $y$ -axis): (top two rows) we see a lesser degree of  $z$  position dependence on neutron features in comparison to that of photons. Features with high dependence in the photon sample no longer exhibit the degree of functional relation in the neutron sample due to their differing interactions. (bottom two rows) We see a clear dependence on energy within the feature space: any feature corresponding to energy deposition within layers has a high dependence, along with width variables. The dependence differs from that of photons, but overlaps at certain regions in the phase space making separation of the two classes more difficult.

## Appendix C. GLUEX generations



**Figure C3.** Features of reconstructed photon showers in GLUEX BCAL: original, reconstructed and generated features (top two rows), residuals and generated residuals (bottom two rows), for  $1 \text{ GeV} < E < 1.4 \text{ GeV}$ ,  $206 \text{ cm} < z < 218 \text{ cm}$ . The cMAF is trained to generate the reconstructed space of the cAE as it was found to give better separation power due to skewing of OOD samples (i.e. neutrons). The cMAF matches closely the distributions of the sampled, reconstructed photons.

We conditionally generate data using the cMAF in a select kinematic region to demonstrate the quality of data produced. A central region of the phase space we are working with ( $1 \text{ GeV} < E < 1.4 \text{ GeV}$ ,  $206 \text{ cm} < z < 218 \text{ cm}$ ) is chosen, and used to compare three different quantities, namely the original injected features, reconstructions from the cAE and the generations of the cMAF. For each original point within the phase space, we generate ten artificial data points, as such, generated distributions an order of magnitude larger in terms of sample size but have been normalized. Being that we are generating the reconstructed space of the cAE, found to be beneficial due to skewing of out-of-distribution (OOD) samples, one may argue the use of a Conditional Variational Autoencoder (cVAE) may be appropriate. We have developed a similar algorithm using cVAE's although it is not optimal for a few reasons, namely, the quality of generations and also the problem with dead nodes (referring to vanishing gradients, in which outputs tend to zero) during training. Due to small values in the residual space, the training process can be very tricky and can lead to Dirac delta distributions at zero in some variables if dead nodes occur. Using an NF instead was found to be more reliable and it can be seen from figure C3 the distributions are consistent.

On a particle-by-particle basis we scaled the neutron features through this empirical formula:  $x + S * P * \frac{(x - x_{\min}) * (x_{\max} - x)}{(x_{\max} - x_{\min})^2}$ , where  $S$  being the sign that controls which direction to nudge,  $P$  is the scaling effect in percentage (we used a 10% effect, that is  $P = 90\%$  or  $110\%$  depending on the sign  $S$ ), and  $x_{\min}$ ,  $x_{\max}$  are the minimum and maximum in the feature range which are physically allowed on each feature. We applied this scaling to all features with the exception of the shower  $R$  for which it has been neglected. Figure C4 shows the injected distributions for photons, neutrons and scaled neutrons; figure C5 shows the corresponding reconstructed features, whereas figure C6 shows the residuals; figure C7 shows the corresponding outlier score distributions obtained with F+M.

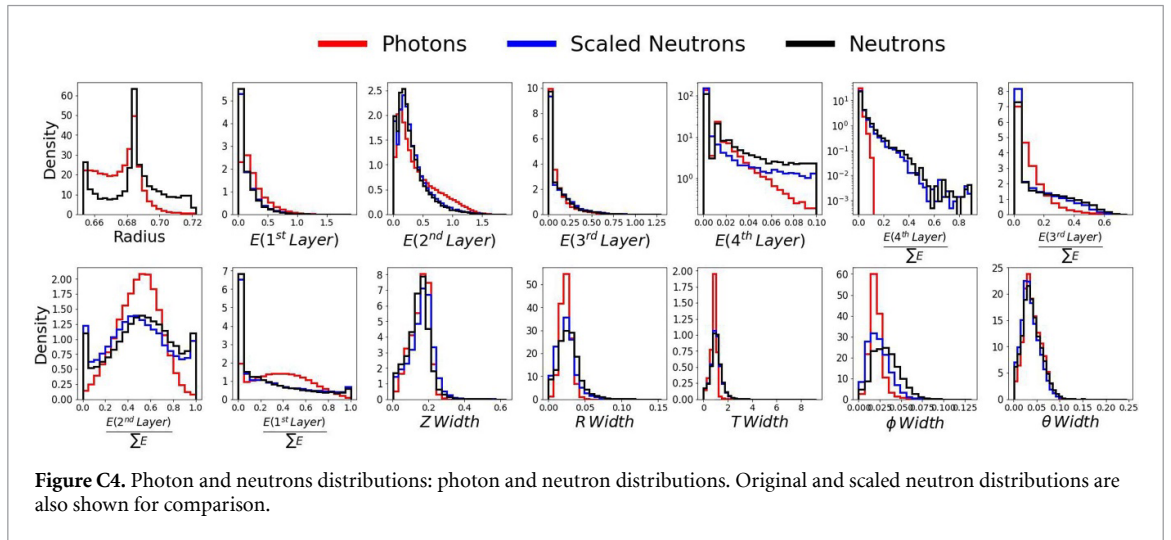


Figure C4. Photon and neutrons distributions: photon and neutron distributions. Original and scaled neutron distributions are also shown for comparison.

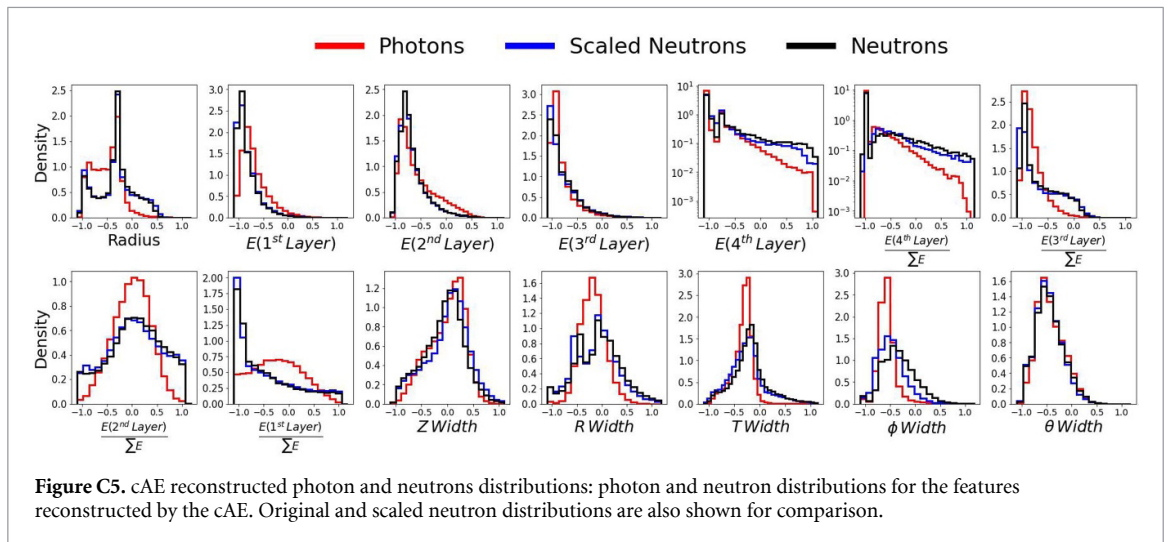


Figure C5. cAE reconstructed photon and neutrons distributions: photon and neutron distributions for the features reconstructed by the cAE. Original and scaled neutron distributions are also shown for comparison.

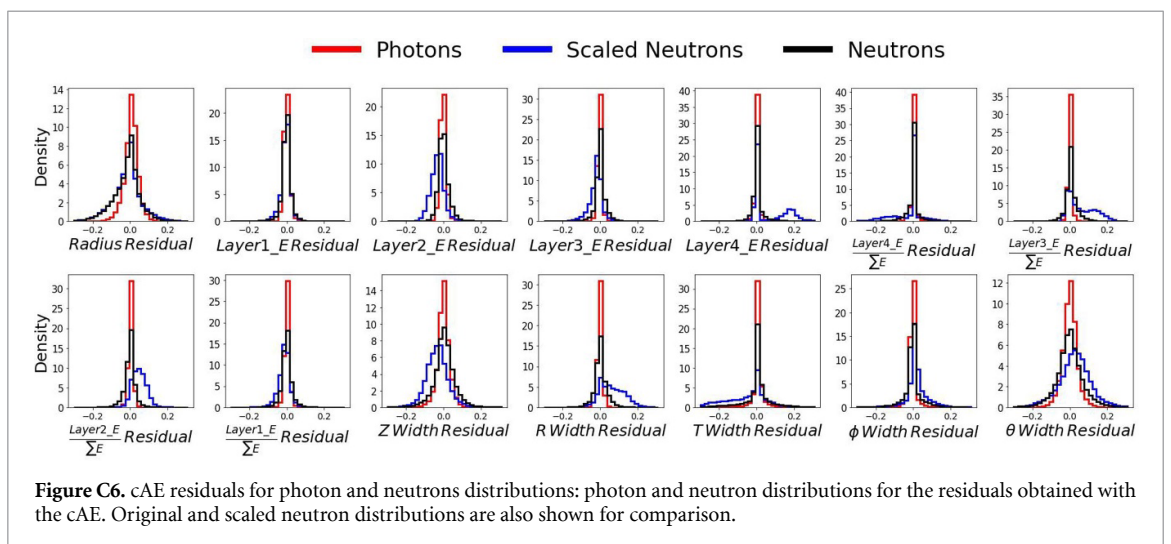
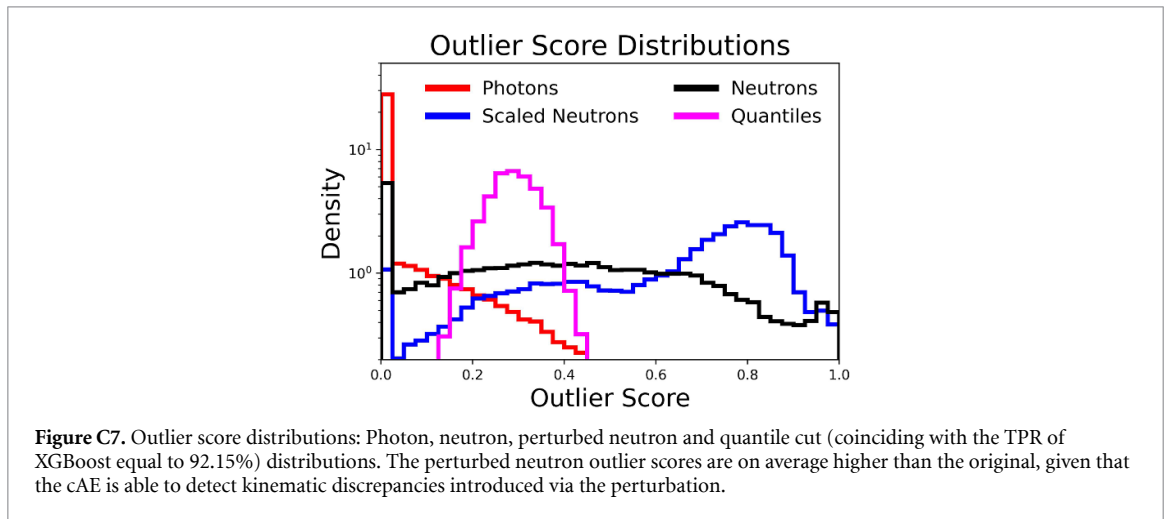


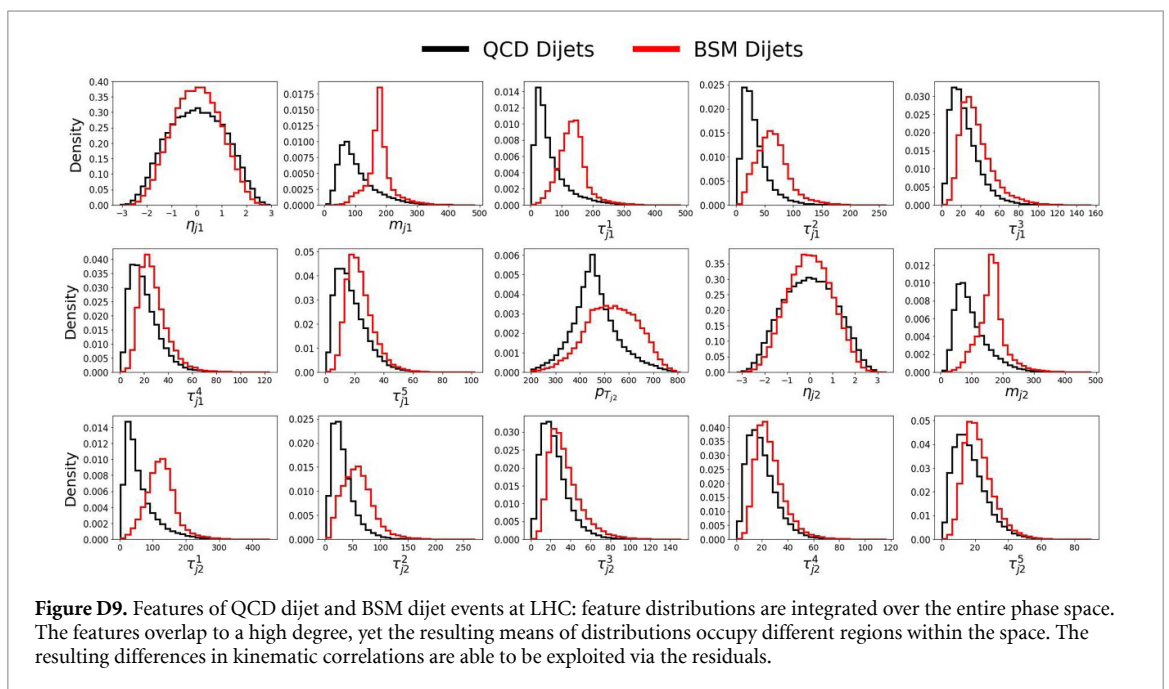
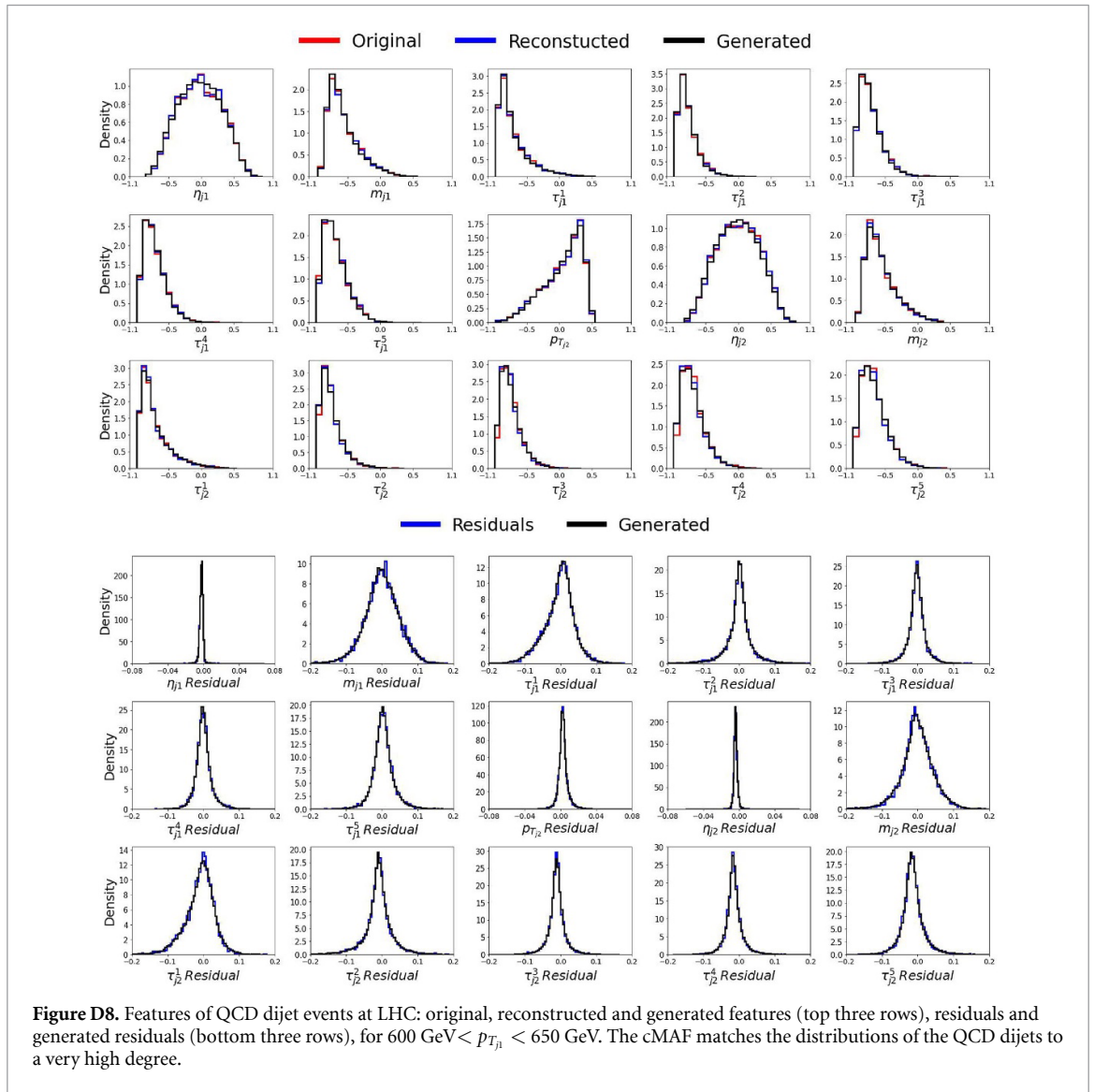
Figure C6. cAE residuals for photon and neutrons distributions: photon and neutron distributions for the residuals obtained with the cAE. Original and scaled neutron distributions are also shown for comparison.



## Appendix D. LHC generations

We conditionally generate data using the cMAF in a select kinematic region to demonstrate the quality of data produced. A central region of the of the phase space we are working with ( $600 \text{ GeV} < p_{T_{j1}} < 650 \text{ GeV}$ ) is chosen, and used to compare three different quantities, namely the original features, reconstructions from the AE and the generations of the cMAF. For each original point within the phase space, we generate ten artificial data points, as such, generated distributions an order of magnitude larger in terms of sample size but have been normalized. We notice in this dataset the generation quality is not as good as for *GLUEX* in appendix C, this is due to the relatively low training sample size and resulting large kinematic bins (1 GeV in transverse momenta) in the KDE modeling phase. The training data set should ideally be more dense (in terms of continuous conditionals) which would allow smaller modeling with KDE, and overall a more robust learning phase for the cMAF. We can see that when training data is sufficient, the generations are extremely accurate (see appendix C). The discrepancies in some variables undoubtedly affect performance although it can still be seen the clusterer is able to efficiently use injected data at the inference phase based off performance obtained.

Figure D8 shows the distributions of the SM QCD dijet events at LHC, for the injected, reconstructed and generated data; figure D9 shows a comparison between the SM QCD and the BSM feature distributions.



## ORCID iD

C Fanelli  <https://orcid.org/0000-0002-1985-1329>

## References

- [1] Seliya N, Abdollah Z A and Khoshgoftaar T M 2021 A literature review on one-class classification and its potential applications in big data *J. Big Data* **8** 1–31
- [2] Nachman B 2022 Anomaly detection for physics analysis and less than supervised learning *Artificial Intelligence for High Energy Physics* (Singapore: World Scientific) ch 4, pp 85–112
- [3] Kasieczka G et al 2021 The LHC Olympics 2020 a community challenge for anomaly detection in high energy physics *Rep. Prog. Phys.* **84** 124201
- [4] Nachman B and Shih D 2020 Anomaly detection with density estimation *Phys. Rev. D* **101** 075042
- [5] Fraser K, Homiller S, Mishra R K, Ostdiek B and Schwartz M D 2021 Challenges for unsupervised anomaly detection in particle physics (arXiv:2110.06948)
- [6] Sohn K, Lee H and Yan X 2015 *Advances in Neural Information Processing Systems 28 (NIPS 2015) (Montreal, Canada, 7–12 December 2015)*
- [7] Rezende D J and Mohamed S 2015 Variational inference with normalizing flows (arXiv:1505.05770)
- [8] Papamakarios G, Pavlakou T and Murray I 2018 Masked autoregressive flow for density estimation (arXiv:1705.07057)
- [9] Ostdiek B 2022 Deep set auto encoders for anomaly detection in particle physics *SciPost Phys.* **12** 45
- [10] Farina M, Nakai Y and Shih D 2020 Searching for new physics with deep autoencoders *Phys. Rev. D* **101** 075021
- [11] Mikuni V and Canelli F 2021 Unsupervised clustering for collider physics *Phys. Rev. D* **103**
- [12] McInnes L, Healy J and Astels S 2017 HDBSCAN: hierarchical density based clustering *J. Open Source Softw.* **2** 205
- [13] Adhikari S et al GlueX Collaboration 2021 The GlueX beamline and detector *Nucl. Instrum. Methods Phys. Res. A* **987** 164807
- [14] Ding X, Wang Y, Xu Z, Welch W J and Wang Z J 2021 CcGAN: continuous conditional generative adversarial networks for image generation *Int. Conf. on Learning Representations*
- [15] Ester M, Kriegl H P, Sander J and Xu X 1996 A density-based algorithm for discovering clusters in large spatial databases with noise *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD'96)* (AAAI Press) pp 226–31
- [16] McInnes L and Healy J 2017 Accelerated hierarchical density based clustering *IEEE Int. Conf. Data Min. Workshops (ICDMW)* (IEEE) pp 33–42
- [17] Campello R J G B, Moulavi D, Zimek A and Sander J 2015 Hierarchical density estimates for data clustering, visualization and outlier detection *ACM Trans. Knowl. Discovery Data* **10** 1–51
- [18] Park S E, Rankin D, Udrescu S M, Yunus M and Harris P 2021 Quasi anomalous knowledge: searching for new physics with embedded knowledge *J. High Energy Phys.* **JHEP06(2021)30**
- [19] Meyer C A and Swanson E 2015 Hybrid mesons *Prog. Part. Nucl. Phys.* **82** 21–58
- [20] Beattie T et al 2018 Construction and performance of the barrel electromagnetic calorimeter for the GlueX experiment *Nucl. Instrum. Methods Phys. Res. A* **896** 24–42
- [21] Agostinelli S et al 2003 Geant4—a simulation toolkit *Nucl. Instrum. Methods Phys. Res. A* **506** 250–303
- [22] Amram O and Suarez C M 2021 Tag N' Train: a technique to train improved classifiers on unlabeled data *J. High Energy Phys.* **JHEP01(2021)153**
- [23] Cheng T 2021 Test sets for jet anomaly detection at the LHC (<https://doi.org/10.5281/zenodo.4614656>)
- [24] Cheng T, Arguin J F, Leissner-Martin J, Pilette J and Golling T 2021 Variational autoencoders for anomalous jet tagging (arXiv:2007.01850)
- [25] Alwall J, Frederix R, Frixione S, Hirschi V and Maltoni F 2014 The automated computation of tree-level and next-to-leading order differential cross sections and their matching to parton shower simulations *J. High Energy Phys.* **JHEP07(2014)79**
- [26] Sjöstrand T, Ask S, Christiansen J R, Corke R, Desai N, Ilten P, Mrenna S, Prestel S, Rasmussen C O and Skands P Z 2015 An introduction to PYTHIA 8.2 *Comput. Phys. Commun.* **191** 159–77
- [27] de Favereau J, Delaere C, Demin P, Giammanco A and Lemaître V, Mertens A and Selvaggi M (The DELPHES 3 Collaboration) 2014 DELPHES 3: a modular framework for fast simulation of a generic collider experiment *J. High Energy Phys.* **JHEP02(2014)057**
- [28] Leissner-Martin J, Cheng T and Arguin J F 2020 QCD jet samples with particle flow constituents (<https://doi.org/10.5281/zenodo.4641460>)
- [29] Cacciari M, Salam G P and Soyez G 2012 FastJet user manual *Eur. Phys. J. C* **72**
- [30] Cacciari M and Salam G P 2006 Dispelling the  $N^3$  myth for the  $k_t$  jet-finder *Phys. Lett. B* **641** 57–61
- [31] Kasieczka G, Marzani S, Soyez G and Stagnitto G 2020 Towards machine learning analytics for jet substructure *J. High Energy Phys.* **JHEP09(2020)195**
- [32] Beattie T 2022 private communication
- [33] Chen T and Guestrin C 2016 XGBoost: a scalable tree boosting system *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (ACM)* pp 785–94
- [34] Pedregosa F et al 2011 Scikit-learn: machine learning in python *J. Mach. Learn. Res.* **12** 2825–30
- [35] Van der Maaten L and Hinton G 2009 Visualizing data using t-SNE *J. Mach. Learn. Res.* **9** 2579–605