JHEP

Published for SISSA by Springer

Received: March 22, 2019
Revised: May 6, 2019
Accepted: May 15, 2019
Published: May 28, 2019

Machine learning templates for QCD factorization in the search for physics beyond the standard model

Joshua Lin,[a] **Wahid Bhimji**[b] **and Benjamin Nachman**[c]

[a] *Department of Physics, University of California, Berkeley, Berkeley, CA 94720, U.S.A.*
[b] *National Energy Research Scientific Computing Center, Berkeley, CA 94720, U.S.A.*
[c] *Physics Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, U.S.A.*

E-mail: joshua.z.lin@berkeley.edu, wbhimji@lbl.gov, bpnachman@lbl.gov

Abstract: High-multiplicity all-hadronic final states are an important, but difficult final state for searching for physics beyond the Standard Model. A powerful search method is to look for large jets with accidental substructure due to multiple hard partons falling within a single jet. One way for estimating the background in this search is to exploit an approximate factorization in quantum chromodynamics whereby the jet mass distribution is determined only by its kinematic properties. Traditionally, this approach has been executed using histograms constructed in a background-rich region. We propose a new approach based on Generative Adversarial Networks (GANs). These neural network approaches are naturally unbinned and can be readily conditioned on multiple jet properties. In addition to using vanilla GANs for this purpose, a modification to the traditional WGAN approach has been investigated where weight clipping is replaced by drawing weights from a naturally compact set (in this case, the circle). Both the vanilla and modified WGAN approaches significantly outperform the histogram method, especially when modeling the dependence on features not used in the histogram construction. These results can be useful for enhancing the sensitivity of LHC searches to high-multiplicity final states involving many quarks and gluons and serve as a useful benchmark where GANs may have immediate benefit to the HEP community.

Keywords: Jets

ArXiv ePrint: 1903.02556

https://doi.org/10.1007/JHEP05(2019)181

JHEP05(2019)181

Contents

1 Introduction **1**

2 Machine learning architectures **2**
- 2.1 Overview of GAN setup 2
- 2.2 Modified WGAN 3

3 Templates for RPV-SUSY **5**
- 3.1 Overview of RPV-SUSY and the template method 5
- 3.2 Simulation setup 6
- 3.3 Machine learning results 7

4 Conclusions **12**

JHEP05(2019)181

1 Introduction

Even though collimated sprays of particles (jets) produced from high energy quarks and gluons are ubiquitous at the Large Hadron Collider (LHC), analyzing their substructure has proven to be a powerful tool in the search for physics beyond the Standard Model (SM) [1, 2]. Many theories of physics beyond the SM predict new particles with cascade decays that can result in large multiplicity final states. When many quarks and gluons are produced in these cascades, multiple large-radius jets with non-trivial substructure can be created [3–5]. As a result, one powerful method for searching for new particles in the all-hadronic channel is to look for events with a large $\sum_{j\in J} m_j$, where m_j is the jet mass and J is a set of jets in an event. The key challenge for such an analysis is to estimate the SM background, as high multiplicity multi-jet final states are difficult to accurately predict with current simulation tools.

Based on the approximate factorization of quantum chromodynamic (QCD) jet production at the LHC [6], the authors of ref. [7] proposed an innovative background estimation technique. The idea of the procedure is to estimate the conditional probability $p(m_j|\text{jet kinematics})$ with an event selection suppressed in signal and then to convolve it with the jet kinematic spectrum in the signal region (from data). A comparison between the predicted m_j and observed m_j is then sensitive to the presence of new particles. The ATLAS collaboration has successfully applied this method in both Run 1 and Run 2 to set strong limits on potential gluino and squark production [8, 9].

The background estimation procedure described above has two major limitations.[1] First, $p(m_j|\text{jet kinematics})$ is represented as a histogram and each bin is uncorrelated so many events are needed for a precise determination. Second, there are physics and

[1]The extensive smoothing studies in ref. [7] help to mitigate binning effects, but do not have an impact on the feature conditioning challenge.

detector effects which change the distribution of the jet mass between the region it is constructed ('trained') and the region where it is applied ('tested'). For example, the quark/gluon composition of the background may be different between the two regions. One way to mitigate this source of method bias is to condition on more features of the jet when constructing the conditional probability. To reduce the impact of changes in quark/gluon composition, one could add the number of charged-particle tracks inside the jet. Gluon jets tend to have more particles than quark jets due to their larger color factor. Since the templates $p(m_j|\text{jet kinematics})$ are binned, it is not simple to condition on more features as one needs many more bins and thus larger samples of events for training.

This paper proposes a solution to both of these limitations using modern machine learning. Deep neural networks are becoming popular tools for classification and regression tasks in high energy physics (HEP) data analysis, but there is a growing machine learning literature on neural network-based generative models as well. Training a generative model can be viewed as a regression task that maps noise to structure, mimicking the Jacobian from a pre-defined probability distribution to a target probability distribution. One of the most well-studied paradigms for such models is the Generative Adversarial Network (GAN) [10] (details in section 2). GANs have also been studied in HEP and show great promise for accelerating simulations [11–21] and may also be useful for other tasks such as sampling from the space of effective field theory models [22]. In the context of QCD factorization studied in this paper, the GAN will learn the probability distribution of the jet mass given the jet kinematics and any other useful information.

This paper is organized as follows. Section 2 introduces GANs and how they can be used to exploit QCD factorization. The application of GANs to the phase space relevant to the Supersymmetry (SUSY) search from refs. [7–9] is presented in section 3. The paper ends with conclusions and outlook in section 4.

2 Machine learning architectures

2.1 Overview of GAN setup

The goal of this section is to introduce an approach to learn the conditional distribution of the jet mass given various kinematic features. Neural networks are chosen due to their flexibility and the resulting algorithms are naturally unbinned. There are multiple neural network-based approaches to generative modeling such as Variational Autoencoders (VAE) [23, 24], Mixture Density Networks (MDN) [25], and Generative Adversarial Networks (GAN) [10]. GANs are selected because they can readily model asymmetric distributions and accommodate conditional features.

Generative Adversarial Network training uses a pair of neural networks: one to map noise into structure (generator) and one to classify (discriminator) physics-based examples from the generator examples. The generator is structured as a densely connected feed-forward neural network that takes as input both jet kinematic features and noise and outputs a jet mass. The generated masses are then input to the discriminator network where they are compared against masses from a physics-based generator that match the kinematic quantities. The two networks 'compete' until the discriminator network is as

JHEP05(2019)181

bad as possible, which means that the generator is proficient at modeling the conditional jet mass distribution. This minimax structure uses the loss function described in ref. [10], constructed to minimize the Jensen-Shannon divergence between the distribution of the real data (in this case, a physics-based simulator) and the distribution of the generated data. A schematic of this GAN is shown in figure 1.

The GAN networks for the jet mass are relatively small compared to others in the literature, which are mostly used for modeling image data. For the vanilla GAN implementation, both the Generator and Discriminator networks are composed of three hidden layers between input (kinetic variables and noise/mass for generator/discriminator respectively) and output (generated mass/likelihood for generator/discriminator respectively). The first layer has 160 neurons, the second has 80 neurons, and the last hidden layer has 40 neurons. Additionally, the generator network is made more robust by adding 50% dropout [26]. Network weights were chosen using the Adam optimizer [27] with early stopping. These settings were chosen after a modest hyper-parameter scan.

All of the neural networks are built using TensorFlow [28] on Nvidia GeForce 1080 Ti GPUs. Since the jet mass distribution is skewed toward the high mass tail, the input noise distribution was varied according to a skew-normal distribution [29]: $f(x|\alpha) = 2\phi(x)\Phi(\alpha x)$, where α is a hyper-parameter and ϕ and Φ are the probability density and cumulative distribution of the standard normal distribution, respectively. The best value of the skew parameter was identified to be $\alpha = 5$ (labeled *skew* in the results plots). This will be compared to the standard no-skew case ($\alpha = 0$; labeled *noskew*).

2.2 Modified WGAN

As an alternative to the vanilla GAN described in section 2.1, a popular variant called the Wasserstein-GAN (WGAN) [30] was also studied (for another HEP application, see ref. [17]). The WGAN differs from the vanilla GAN in that it minimizes the Earth-Mover distance (also known as the Wasserstein Distance):

$$W(\mathbb{P}_{\text{real}}, \mathbb{P}_{\text{generated}}) = \inf_{\gamma \in \Pi(\mathbb{P}_{\text{real}}, \mathbb{P}_{\text{generated}})} \mathbb{E}_{(x,y)\sim\gamma}[\| x - y \|], \tag{2.1}$$

where $\gamma \in \Pi(\mathbb{P}_{\text{real}}, \mathbb{P}_{\text{generated}})$ is a joint distribution with marginal distributions $\mathbb{P}_{\text{real}}, \mathbb{P}_{\text{generated}}$ respectively, and $(x, y) \sim \gamma$ means that the random variable (x, y) is drawn from the distribution γ. This 'softer' metric was introduced as a way to combat the vanishing gradients that often occured when training regular GANs [30, 31]. In the algorithm suggested by the WGAN paper to minimise the Earth-Mover distance, the discriminator is a function f that is taken from a space of trial functions that are all K-Lipshitz for some K. To enforce such conditions, the functions f are constructed as feed-forward neural networks with weights w that are clipped (after every update) to a compact space $[-\alpha, \alpha]$ for some fixed α which enforces K-lipshitz for some K.

An exact implementation of the WGAN approach resulted in weights that would often aggregate around the specific limit values α chosen, leading to vanishing gradients and generated mass distributions that did not match the physical mass distributions. The challenges surrounding the clipping operation to enforce the Lipshitz condition are discussed in the original WGAN paper, and explored in variations of WGAN where the clipping is replaced by gradient penalty [32] or asymmetric clipping [33].

JHEP05(2019)181

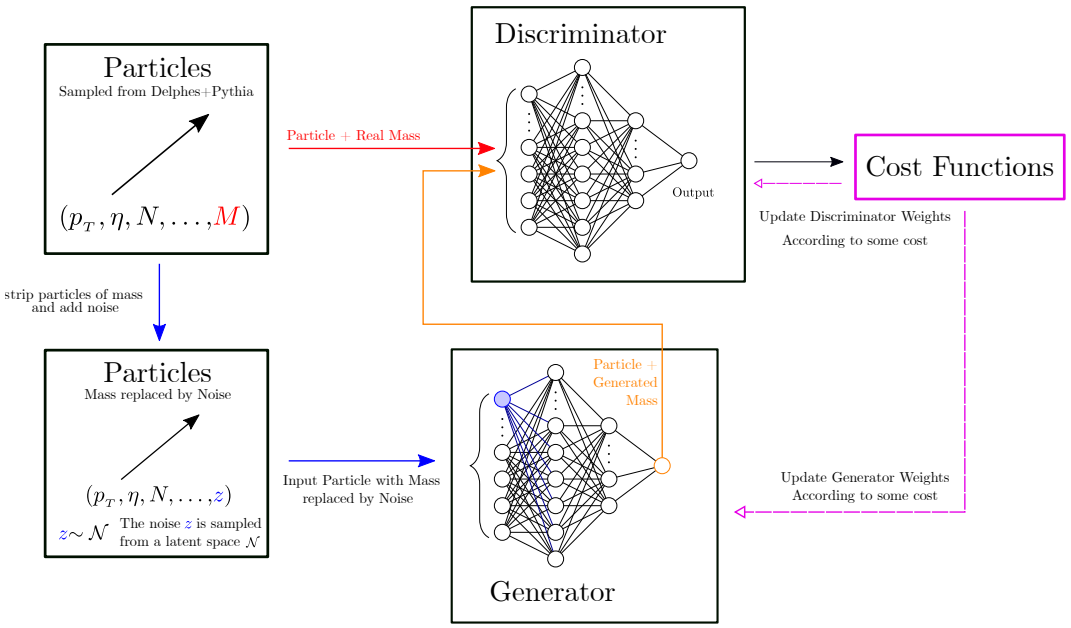

Figure 1. Flowchart describing how GANs are used to learn templates (shown here mass templates for the RPV-SUSY search) given kinematic variables. The generator network is a feed-forward network that takes as input particles with their mass replaced by noise, and generates mass according to a learnt distribution. These fake particles are bundled with real particles and passed to the discriminator, which learns to discriminate between the real and fake distributions.

In this paper, a modification to weight clipping is introduced that changes how the weights act on the outputs of neurons to enforce the Lipshitz condition. A schematic for this modification to the WGAN is shown in figure 2. The original weight-clipping operation enforces a Lipshitz function because the composition of Lipshitz functions is still a Lipshitz function; in particular, there is the weak assumption that all the activation functions used in the neural networks are themselves Lipshitz (true for the most popular activation functions including sigmoid, tanh, and ReLU). Then, the fact that there is a single constant K for which all the functions f that we are considering are K-Lipshitz is due to the weight-clipping — this restricts the function space to be compact.

Another natural way to enforce the Lipshitz constraint that eliminates boundary effects is to draw weights from a compact space with no boundaries. One way to achieve this is to draw the weights and biases from the unit circle:[2]

$$x_n^{(k+1)} = \psi\left(\left|\sum_m x_m^{(k)} e^{i\theta_{nm}^{(k)}} + e^{i\phi_n^{(k+1)}}\right|\right), \tag{2.2}$$

[2]In principle, one could generalize this idea to modify the WGAN where all weights and biases are drawn from a generic compact Lie Group. The outputs of the neurons exist in a vector space with the Lie Group acting by a chosen representation. In this framework, a normal linear neural network uses the Lie Group $\mathbb{R}$ with canonical action on itself, and the modified WGAN shown in eq. 2.2 uses $\mathrm{U}(1) \approx S^1$. The point of this construction is that the compactness of the weight-space assures the convergence of the WGAN and is well-formed because Lie Group actions can be differentiated. Such applications of Lie Groups to Machine Learning have been explored elsewhere in the literature, for example in applications to 3D-classification problems [34].

JHEP05(2019)181

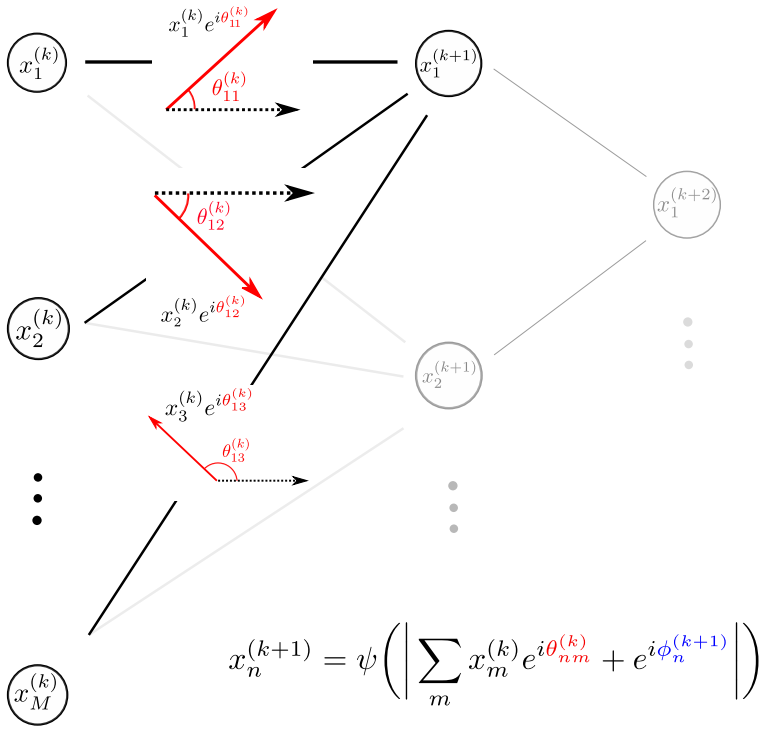

Figure 2. A schematic diagram to illstrate the modified WGAN, where all the weights and biases are angles. The compactness of the unit circle then guarantees that the trained functions are all K-lipschitz for sufficiently large K. The inputs to the layer are $x_1, \ldots, x_n$ and there is a hidden layer with two nodes y_1 and y_2 (in general, there can be many more than two) followed by a single output layer z_1. The equation shows the form of the activation that maps $x_1, \ldots, x_n$ to y_1.

where $x_m^{(k)}$ represents the output of the m-th neuron in the k-th layer of the neural network and ψ is the activation function. The bias is replaced by $b_i = e^{i\phi_n}$ and the weights are replaced by $w_{nm} = e^{i\theta_{nm}^{(k)}}$.

3 Templates for RPV-SUSY

3.1 Overview of RPV-SUSY and the template method

Supersymmetry (SUSY) [35–40] is a well-studied extension of the Standard Model in which there is a new symmetry relating fermions and bosons. In models of SUSY that conserve R-parity [41–45] — an additional symmetry that requires SUSY particles to be produced in pairs — collider signatures often feature large missing transverse momentum (MET) carried by the lightest supersymmetric particle (LSP) which must be neutral under the electromagnetic and strong forces. However, R-parity is not present in all SUSY models and collider-based limits relying on MET are typically insensitive to such models. One R-parity violating (RPV) coupling (often denoted λ'') results in gluino/neutralino decay into three quarks, as illustrated in figure 3. When this coupling is present, traditional searches for SUSY are largely ineffective because there can be little MET and no charged

JHEP05(2019)181

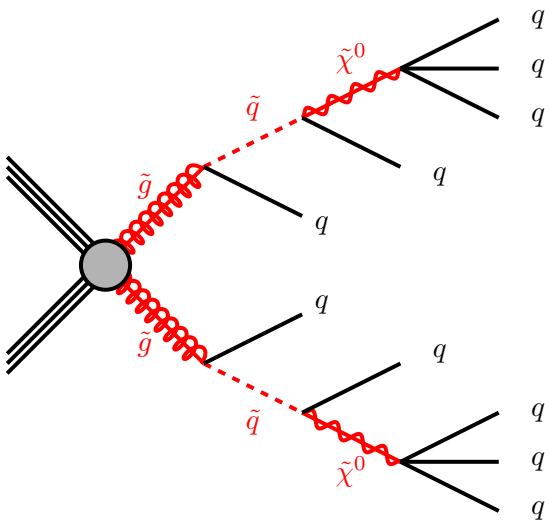

Figure 3. Schematic Feynman-like diagram for RPV SUSY.

leptons. Despite lacking standard handles for separating potential SUSY events from SM background processes, RPV signatures like those in figure 3, all-hadronic SUSY searches have been able to set strong limits on gluino and squark production in models with a large λ'' (see references within refs. [8, 9] for other constraints on such models).

One approach to search for all-hadronic SUSY decays like those in figure 3 is to exploit the high-multiplicity of hard, well-separated partons in the final state. In the recent ATLAS analysis [9], the main kinematic observable used in the search for λ'' RPV decays is the total jet mass M_Σ^J defined as the sum of the masses of the four leading large $R = 1.0$ jets. Multiple jets with a large mass are generated from well-separated high-energy partons that happen to be clustered within a single jet. The SM multijet background is estimated by a data-driven method [7], whereby mass templates are constructed from control regions containing a low purity of potential signal. These templates are histograms that model the dependence of QCD mass on jet kinematic properties (p_T and η). If the jets in the signal region are only due to QCD, then the mass distribution can be estimated by convolving the jet kinematics with the mass templates. This estimate is compared with the actual mass distribution and deviations would be an indication of BSM physics.

The goal of this study is to demonstrate that GANs may be a useful alternative method to simple histogram templates for learning the dependence of the jet mass on jet properties.

3.2 Simulation setup

To demonstrate the potential of GAN-based templates, $pp \to$ jets at $\sqrt{s} = 13$ TeV are generated with Pythia 8.223 [46] using the fast detector simulation Delphes 3.4.0 [47] with the detector card `delphes_card_ATLAS.tcl`. Following the ATLAS analysis strategy [9], events are clustered [48] into jets using the anti-k_t algorithm [49] with radius parameter $R = 1$. These jets are groomed according to the trimming procedure [50] where subjets with radius $R = 0.2$ are created from the large-radius jet constituents and removed if their

JHEP05(2019)181

Region	$N_{\text{jet}}(p_T > 200$ GeV$)$	$p_{T,1}$	$\lvert\Delta\eta_{1,2}\rvert$	M_J^{Σ}
Control	$= 3$	—	–	—
Validation	$= 4$	> 400 GeV	> 1.4	—
	≥ 5	—	> 1.4	—
Signal	$= 4$	> 400 GeV	< 1.4	> 1.0 TeV
	≥ 5	—	< 1.4	> 0.8 TeV

Table 1. Phasespace requirements for the different regions considered.

transverse momentum is below 5% of the parent jet's p_T. The remaining large-radius jets are only considered if $p_T > 200\,\text{GeV}$ and $|\eta| < 2.0$ and are divided into control, validation and signal regions according to table 1. The control region is used to construct the templates (train the GAN) and the signal region is where they are applied.[3] The validation region is expected to be sufficiently devoid of potential signals that it can be used to study the fidelity of the templates. Even though the Pythia is only leading order in the strong coupling constant, ref. [9] used the same setup and found a good agreement with data so this setup is sensible for testing new methods.

As a baseline, mass templates (histograms) are constructed following the ATLAS study: jets in the control region divided into 4 $|\eta|$ bins defined uniformly between 0 and 2, and 15 p_T bins defined uniformly in $\log_{10}(p_T)$. This results in 60 mass histograms in total. No b-tagging is applied, though future extensions to include flavor tagging information are possible. Using the mass histograms, jets in the validation and signal regions can be dressed with random masses given their p_T and η.

3.3 Machine learning results

After the selections described in the previous section, there were 1.1 million jets in the control region and 30k in the validation region. The validation region is used to test the efficacy of the neural network training, filling the role of the 'test set' and by construction is independent from the training set. The jets in the control region are split 50%-50% for the purpose of training the neural network and 'validating' the network to enforce the early stopping condition.

The accuracy of the generated mass templates was quantified[4] using the separation power metric [51, 52]:

$$S(\mathbb{P}_1, \mathbb{P}_2) = \frac{1}{2}\int_X \frac{(\mathbb{P}_1(x) - \mathbb{P}_2(x))^2}{\mathbb{P}_1(x) + \mathbb{P}_2(x)}\mathrm{d}x, \tag{3.1}$$

[3] We note that the original template proposal from ref. [7] called for a smoothing procedure that was used in the earlier ATLAS result [8] but not the later one [9]. Our analysis here more closely resembles the later approach. The smoothing should in principle help mitigate binning effects, though would not change the qualitative conclusions about feature dependence.

[4] There is no unique way to monitor the GAN performance during training. For a typical GAN trained with images, this is qualitatively different than classifier training because the entire multi-dimensional probability is being modeled, not just the likelihood ratio. The 1D case here is not as extreme, but still different than classification or regression monitoring. One can use the full GAN loss, the discriminator loss, or any divergence that gives a 'distance' between probability distributions. This particular divergence is popular in HEP and is therefore used as a diagnostic here.

JHEP05(2019)181

where $\mathbb{P}_1, \mathbb{P}_2$ are probability distributions over a space X and eq. 3.1 is normalized to be between 0 and 1. For the RPV SUSY case, $X \simeq \mathbb{R}_{p_T} \times \mathbb{R}_\eta \times \mathbb{R}_N \times \mathbb{R}_m$, one real line for each of the jet properties p_T, η, constituent track multiplicity[5] (N), and m. The $\mathbb{P}_i$ are the real and generated distributions:

$$\mathbb{P}_{\text{real}}(p_T, \eta, N, m) \tag{3.2}$$

$$\mathbb{P}_{\text{generated}}(p_T, \eta, N, m) = \mathbb{P}_{\text{real}}(p_T, \eta, N) G(m|p_T, \eta, N), \tag{3.3}$$

where $G(m|p_T, \eta, N)$ is the learned mass distribution by the GAN for a given p_T, η, N value, and $\mathbb{P}_{\text{real}}(p_T, \eta, N) = \int \mathbb{P}_{\text{real}}(p_T, \eta, N, m) \mathrm{d}m$. Equation 3.3 explicitly encodes the QCD factorization of the jet kinematics and the mass given those kinematic properties. Neither the GAN or physics-based simulator provide $\mathbb{P}$ directly; instead, only examples are drawn from the distributions. Empirical distributions are constructed from samples and the separation power is approximated by first binning the jets into eight regions of equal statistics in the kinematic variables p_T, η, N, by splitting the events into two collections based on the jet p_T, then splitting each of these collections evenly in η, and then in N. In each of these eight bins, the jet mass distribution is used to calculate the (binned) separation power for each dataset and then all eight sets[6] are combined.

Figure 4 shows the separation power for various generative models as a function of the number of epochs used to train. GAN models that were initialized with a high separation error (above 0.6) training much slower due to vanishing gradients, so only those GANs with an initialized value below 0.6 are considered for figure 4. Furthermore, to reduce the impact of fluctuations in the initialization, the average value over ten random initializations are used. By construction, the default template method does not involve neural networks and thus is constant. As desired, all of the GAN approaches converge to a separation power that is smaller than the template method, as they have access to more and unbinned information. For both the vanilla GAN and the WGAN, using a skew-normal distribution for the noise accelerates the training time. The Vanilla GAN also converges faster than the WGAN, though all GAN approaches have a similar final separation power.

The mass distributions in the validation region are presented in figure 5, in bins of jet p_T, η, and N. The average jet mass scales approximately as $\alpha_s \times p_T \times R$ and the width of the distribution also grows with p_T. Given the jet p_T, the jet mass should be approximately independent of η, aside from detector effects. Gluon jets have a large jet mass than quark jets and also a higher constituent multiplicity so there is a positive correlation between N and mass.

Overall, the level of agreement between the GAN and the real mass distributions is better than for the template method and the real distributions. This is particularly true for N, where the real mass is shifted to lower values for low N (more quark-like) and to higher values for high N (more gluon-like). The GAN is typically well within 50% of the real distribution, while the template method can be much more than a factor of 2 off of

[5] Due to their robustness to pileup and excellent angular resolution, charged-particle tracks are associated to jets and used as proxy for the number of particles inside the jet.

[6] The estimated separation power, for our dataset size, is not sensitive to increasing bin number beyond 8.

JHEP05(2019)181

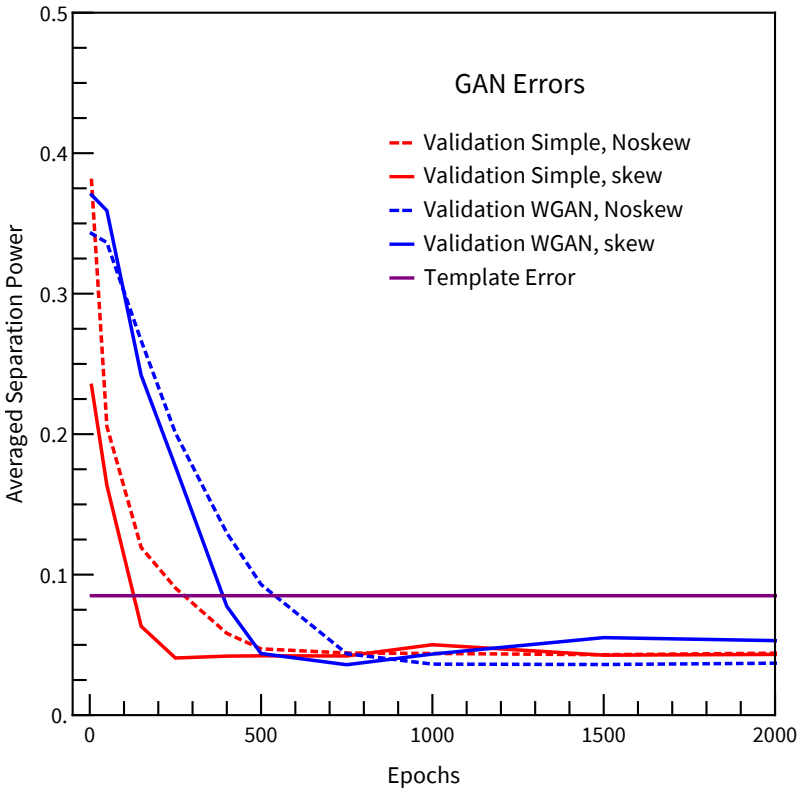

Figure 4. Estimated separation power between the generated jet-kinematic distribution and the real jet-kinematic distribution for various GAN architectures. The template error corresponds to the baseline approach with no neural networks and is thus independent of the number of training epochs.

the real distribution for low and high N. This is particularly important if the quark/gluon composition is different between the control, validation, and signal regions, either by chance or because some quark-tagging is applied to suppress the QCD background in such high multiplicity final states [53].

The modeling of the jet mass distribution in the validation region is used to determine systematic uncertainties on the templates. Figure 6 explicitly constructs the systematic uncertainty as the sum in quadrature of the non-closure from the validation region and the control region statistical uncertainty. These uncertainties are computed for the $\sum_{j\in J} m_j$ distribution (sum of the masses of the top four jets in the event), which is the main observable used in the RPV SUSY search [8, 9]. In blue we have the deviation from the exact ratio 1 for the template mass distributions and the GAN mass distributions. The GAN outperforms the template method in both the low mass and high mass limits; also note that when placing a cut on N_{track}, the performance of the GAN becomes even more pronounced. This is particularly encouraging because we expect such RPV-SUSY signals to be quark jet dominated, with a lower multiplicity on average than a gluon jet dominated background. For a 50% quark jet efficiency requirement on all four jets ($N_{\text{track}} < 26$), the uncertainty for the GAN approach is about 20% in the high $\sum_{i\in J} m_j$ tail while it is well over 100% for the template appraoch.

JHEP05(2019)181

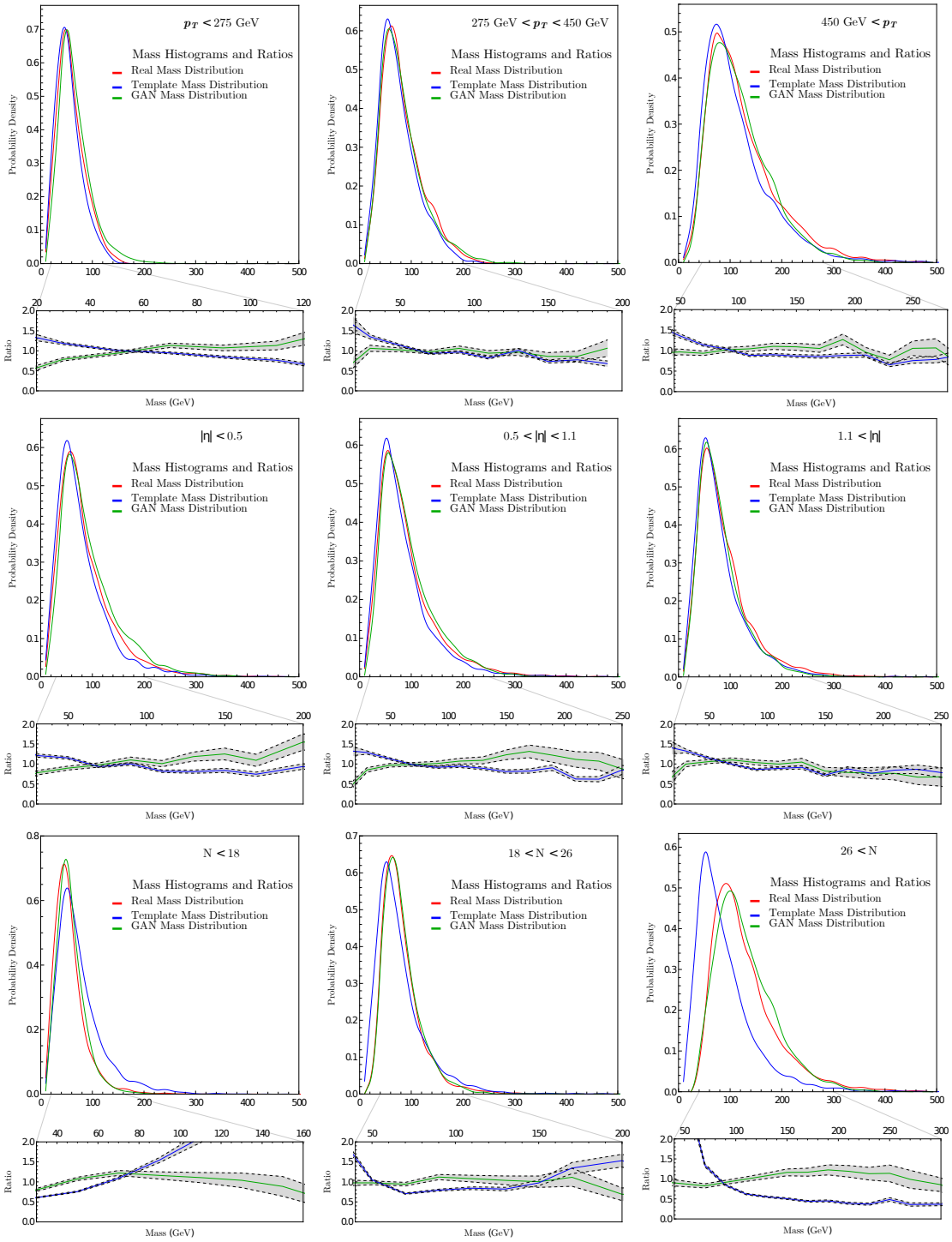

Figure 5. The physics-based ('real') mass distributions compared with distributions from the template method and the vanilla GAN in bins of jet p_T (top row), η (middle row), and N (bottom row). The uncertainty in the ratio was calculated as the 1-sigma error assuming poisson distributions of events in each bin. The error shown in the plots is the calculated statistical error. The corresponding plot in the control region is qualitatively similar, but converges quicker.

JHEP05(2019)181

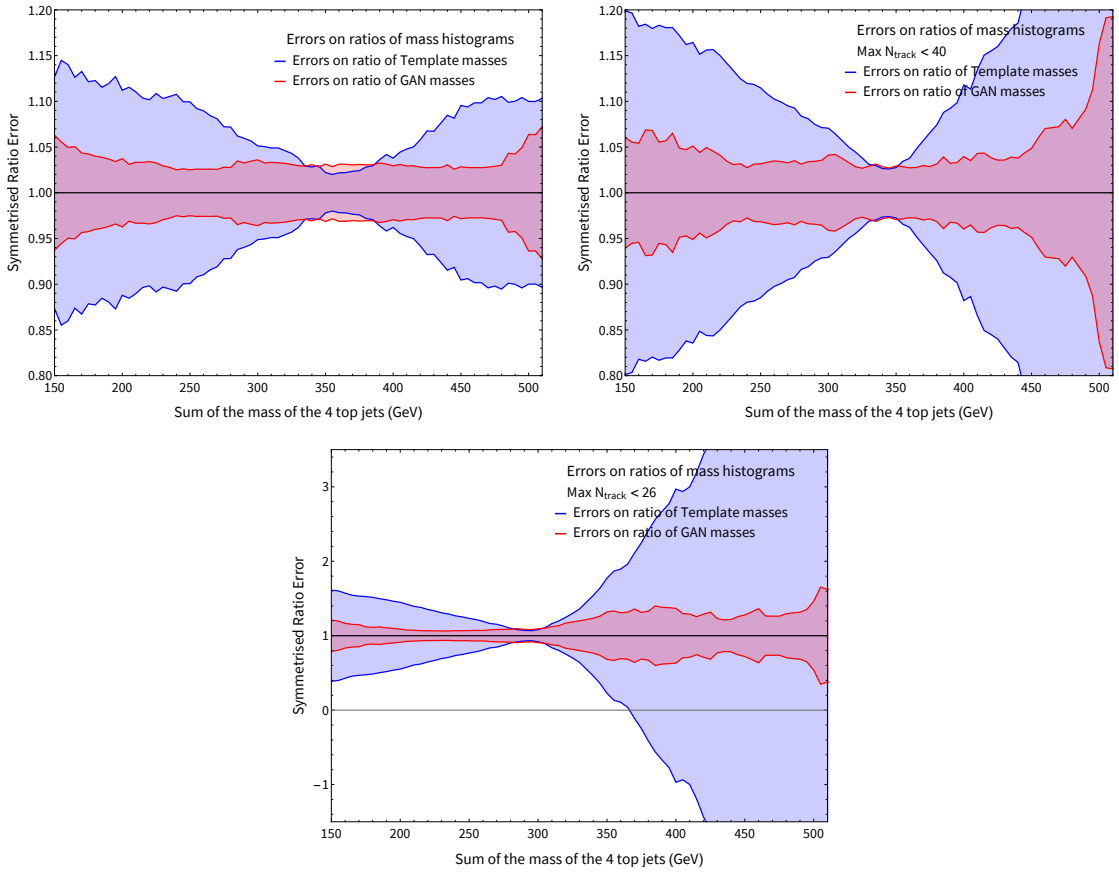

Figure 6. Computing the ratio error of generated mass distributions ($\sum m_J$ of the masses of the top 4 jets) to the real distribution, symmetrized by reflecting across the ideal ratio = 1, with statistical uncertainties included. Inclusive error measurements are shown on the top left, for the other plots a cut is placed on N_{track}: where for $N_{\text{track}} < X$ all four jets of the events considered are required to have $N_{\text{track}} < X$. As we reduce the efficiency of this cut, we see that the GAN's relative performance to the template distributions becomes better-expected because the GAN learns how mass changes in N_{track}. For the top right plot, the efficiency of the cut is 50% overall while for the bottom plot, the efficiency is about 50% per jet (about 6% overall).

An important part of any background estimation technique is the associated systematic uncertainty. One of the main sources of uncertainty here is the limited size of the training set in the control region. The authors of ref. [7] suggested a bootstrapping technique to estimate the uncertainty by rerunning the template procedure on bootstrapped datasets. In principle, one could do the same procedure for the GAN training, with one GAN per bootstrap dataset. More sophisticated methods include modeling GAN weights and biases as nuisance parameters to be profiled by the data with prior distributions.

A challenge for assessing previous GAN applications in HEP is that they have been designed to model high-dimensional feature spaces that are difficult to visualize and study [11–21]. The mass distribution example presented here provides a concrete testing ground to study GAN approaches where quantitative agreement can be studied and achieved using

JHEP05(2019)181

existing techniques. While this study used only leading-order simulations of jet production, the methods are applicable more generally and can be applied to collision data from the LHC experiments.

4 Conclusions

Generative Adversarial Networks have been proposed as an alternative to histogram-based mass templates for the background estimation in LHC searches for RPV SUSY. These methods rely on the approximate QCD factorization whereby a jets type and kinematic properties are sufficient for determining the distribution of the jet mass. The neural network approaches are naturally unbinned and can be readily conditioned on multiple jet properties. In addition to using vanilla GANs for this purpose, a modification to the traditional WGAN approach has been investigated where weight clipping is replaced with drawing weights from a naturally compact set (in this case, the circle). Both the vanilla and modified WGAN approaches were able to outperform the histogram method, especially when modeling the dependence on features not used in the histogram construction. When training such generative models for physical applications, the usual limitations of the method apply such as the potential for overfitting, sensitivity to hyperparameters, and vanishing gradients slowing training down — though methods of circumnavigating these limitations have been studied in the last few years, such as using 'softer metrics' such as the Wasserstein metric. These results can be useful for enhancing the sensitivity of LHC searches to high-multiplicity final states involving many quarks and gluons and serve as a useful benchmark where GANs may have immediate benefit to the HEP community.

Acknowledgments

We would like to thank Tim Cohen, Luke de Oliveira, Mustafa Mustafa, Michela Paganini, Max Swiatloski, and Jesse Thaler for their helpful feedback on the manuscript. This work was supported by the U.S. Department of Energy, Office of Science under contract DE-AC02-05CH11231.

References

[1] A.J. Larkoski, I. Moult and B. Nachman, *Jet substructure at the Large Hadron Collider: a review of recent advances in theory and machine learning*, arXiv:1709.04464 [INSPIRE].

[2] L. Asquith et al., *Jet substructure at the Large Hadron Collider: experimental review*, arXiv:1803.06991 [INSPIRE].

[3] T. Cohen, E. Izaguirre, M. Lisanti and H.K. Lou, *Jet substructure by accident*, *JHEP* **03** (2013) 161 [arXiv:1212.1456] [INSPIRE].

JHEP05(2019)181

[4] S. El Hedri, A. Hook, M. Jankowiak and J.G. Wacker, *Learning how to count: a high multiplicity search for the LHC*, *JHEP* **08** (2013) 136 [arXiv:1302.1870] [INSPIRE].

[5] A. Hook, E. Izaguirre, M. Lisanti and J.G. Wacker, *High multiplicity searches at the LHC using jet masses*, *Phys. Rev.* **D 85** (2012) 055029 [arXiv:1202.0558] [INSPIRE].

[6] J.C. Collins, D.E. Soper and G.F. Sterman, *Factorization of hard processes in QCD*, *Adv. Ser. Direct. High Energy Phys.* **5** (1989) 1 [hep-ph/0409313] [INSPIRE].

[7] T. Cohen et al., *Jet substructure templates: data-driven QCD backgrounds for fat jet searches*, *JHEP* **05** (2014) 005 [arXiv:1402.0516] [INSPIRE].

[8] ATLAS collaboration, *Search for massive supersymmetric particles decaying to many jets using the ATLAS detector in pp collisions at* $\sqrt{s} = 8$ *TeV*, *Phys. Rev.* **D 91** (2015) 112016 [*Erratum ibid.* **D 93** (2016) 039901] [arXiv:1502.05686] [INSPIRE].

[9] ATLAS collaboration, *Search for R-parity-violating supersymmetric particles in multi-jet final states produced in p-p collisions at* $\sqrt{s} = 13$ *TeV using the ATLAS detector at the LHC*, *Phys. Lett.* **B 785** (2018) 136 [arXiv:1804.03568] [INSPIRE].

[10] I.J. Goodfellow et al., *Generative adversarial networks*, arXiv:1406.2661 [INSPIRE].

[11] M. Paganini, L. de Oliveira and B. Nachman, *Accelerating science with generative adversarial networks: an application to 3D particle showers in multilayer calorimeters*, *Phys. Rev. Lett.* **120** (2018) 042003 [arXiv:1705.02355] [INSPIRE].

[12] M. Paganini, L. de Oliveira and B. Nachman, *CaloGAN: simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks*, *Phys. Rev.* **D 97** (2018) 014021 [arXiv:1712.10321] [INSPIRE].

[13] L. de Oliveira, M. Paganini and B. Nachman, *Controlling physical attributes in gan-accelerated simulation of electromagnetic calorimeters*, *J. Phys. Conf. Ser.* **1085** (2018) 042017 [arXiv:1711.08813] [INSPIRE].

[14] V. Chekalina et al., *Generative models for fast calorimeter simulation*, in the proceedings of the 23rd *International Conference on Computing in High Energy and Nuclear Physics (CHEP 2018)*, July 9–13, Sofia, Bulgaria (2018), arXiv:1812.01319 [INSPIRE].

[15] F. Carminati et al., *Three dimensional generative adversarial networks for fast simulation*, *J. Phys. Conf. Ser.* **1085** (2018) 032016.

[16] S. Vallecorsa, *Generative models for fast simulation*, *J. Phys. Conf. Ser.* **1085** (2018) 022005.

[17] M. Erdmann, J. Glombitza and T. Quast, *Precise simulation of electromagnetic calorimeter showers using a Wasserstein generative adversarial network*, *Comput. Softw. Big Sci.* **3** (2019) 4 [arXiv:1807.01954] [INSPIRE].

[18] P. Musella and F. Pandolfi, *Fast and accurate simulation of particle detectors using generative adversarial networks*, *Comput. Softw. Big Sci.* **2** (2018) 8 [arXiv:1805.00850] [INSPIRE].

[19] M. Erdmann, L. Geiger, J. Glombitza and D. Schmidt, *Generating and refining particle detector simulations using the Wasserstein distance in adversarial networks*, *Comput. Softw. Big Sci.* **2** (2018) 4 [arXiv:1802.03325] [INSPIRE].

[20] ATLAS collaboration, *Deep generative models for fast shower simulation in ATLAS*, ATL-SOFT-PUB-2018-001 (2018).

JHEP05(2019)181

[21] L. de Oliveira, M. Paganini and B. Nachman, *Learning particle physics by example: location-aware generative adversarial networks for physics synthesis*, *Comput. Softw. Big Sci.* **1** (2017) 4 [arXiv:1701.05927] [INSPIRE].

[22] H. Erbin and S. Krippendorf, *GANs for generating EFT models*, arXiv:1809.02612 [INSPIRE].

[23] D.P. Kingma and M. Welling, *Auto-encoding variational bayes*, arXiv:1312.6114 [INSPIRE].

[24] D.J. Rezende, S. Mohamed, and D. Wierstra, *Stochastic backpropagation and approximate inference in deep generative models*, in the proceedings of the 31st *International Conference on International Conference on Machine Learning (ICML'14)*, June 21–26, Beijing, China (2014).

[25] C. Bishop, *Mixture density networks*, Neural Computing Research Group Report NCRG/94/004 (1994).

[26] N. Srivastava et al., *Dropout: a simple way to prevent neural networks from overfitting*, *J. Mach. Learn. Res.* **15** (2014) 1929.

[27] D.P. Kingma and J. Ba, *Adam: a method for stochastic optimization*, arXiv:1412.6980 [INSPIRE].

[28] M. Abadi et al., *Tensorflow: a system for large-scale machine learning*, *OSDI* **16** (2016) 265.

[29] A. O'Hagan and T. Leonard, *Bayes estimation subject to uncertainty about parameter constraints*, *Biometrika* **63** (1976) 201.

[30] M. Arjovsky, S. Chintala and L. Bottou, *Wasserstein GAN*, arXiv:1701.07875.

[31] M. Arjovsky and L. Bottou, *Towards principled methods for training generative adversarial networks*, arXiv:1701.04862.

[32] I. Gulrajani et al., *Improved training of wasserstein gans*, arXiv:1704.00028.

[33] X. Guo, J. Hong, T. Lin and N. Yang, *Relaxed Wasserstein with Applications to GANs*, arXiv:1705.07164.

[34] Z. Huang, C. Wan, T. Probst and L.V. Gool, *Deep learning on Lie groups for skeleton-based action recognition*, arXiv:1612.05877.

[35] Yu. A. Golfand and E.P. Likhtman, *Extension of the algebra of Poincaré group generators and violation of p invariance*, *JETP Lett.* **13** (1971) 323 [INSPIRE].

[36] D.V. Volkov and V.P. Akulov, *Is the neutrino a Goldstone particle?*, *Phys. Lett.* **46B** (1973) 109 [INSPIRE].

[37] J. Wess and B. Zumino, *Supergauge transformations in four-dimensions*, *Nucl. Phys.* **B 70** (1974) 39 [INSPIRE].

[38] J. Wess and B. Zumino, *Supergauge invariant extension of quantum electrodynamics*, *Nucl. Phys.* **B 78** (1974) 1 [INSPIRE].

[39] S. Ferrara and B. Zumino, *Supergauge invariant Yang-Mills theories*, *Nucl. Phys.* **B 79** (1974) 413 [INSPIRE].

[40] A. Salam and J.A. Strathdee, *Supersymmetry and Nonabelian Gauges*, *Phys. Lett.* **51B** (1974) 353 [INSPIRE].

[41] G.R. Farrar and P. Fayet, *Phenomenology of the production, decay, and detection of new hadronic states associated with supersymmetry*, *Phys. Lett.* **B 76** (1978) 5575.

JHEP05(2019)181

[42] S. Dimopoulos and H. Georgi, *Softly broken supersymmetry and* SU(5), *Nucl. Phys.* **B 193** (1981) 150 [INSPIRE].

[43] S. Weinberg, *Supersymmetry at ordinary energies. 1. Masses and conservation laws*, *Phys. Rev.* **D 26** (1982) 287 [INSPIRE].

[44] N. Sakai and T. Yanagida, *Proton decay in a class of supersymmetric grand unified models*, *Nucl. Phys.* **B 197** (1982) 3533.

[45] S. Dimopoulos, S. Raby and F. Wilczek, *Proton decay in supersymmetric models*, *Phys. Lett.* **B 112** (1982) 2133.

[46] T. Sjöstrand, S. Mrenna and P.Z. Skands, *PYTHIA 6.4 physics and manual*, *JHEP* **05** (2006) 026 [hep-ph/0603175] [INSPIRE].

[47] DELPHES 3 collaboration, *DELPHES 3, a modular framework for fast simulation of a generic collider experiment*, *JHEP* **02** (2014) 057 [arXiv:1307.6346] [INSPIRE].

[48] M. Cacciari, G.P. Salam and G. Soyez, *FastJet user manual*, *Eur. Phys. J.* **C 72** (2012) 1896 [arXiv:1111.6097] [INSPIRE].

[49] M. Cacciari, G.P. Salam and G. Soyez, *The anti-k_t jet clustering algorithm*, *JHEP* **04** (2008) 063 [arXiv:0802.1189] [INSPIRE].

[50] D. Krohn, J. Thaler and L.-T. Wang, *Jet trimming*, *JHEP* **02** (2010) 084 [arXiv:0912.1342] [INSPIRE].

[51] **BaBar** collaboration, *The BABAR physics book: physics at an asymmetric B factory*, talk given at the the *Workshop on Physics at an Asymmetric B Factory*, September 22–24, Pasadena, U.S.A. (1998).

[52] A. Hocker et al., *TMVA — Toolkit for Multivariate Data Analysis*, physics/0703039 [INSPIRE].

[53] Y. Sakaki, *Quark jet rates and quark/gluon discrimination in multi-jet final states*, arXiv:1807.01421 [INSPIRE].

JHEP05(2019)181