Performance versus Resilience in Modern Quark–Gluon Tagging

Anja Butter[1,2], Barry M. Dillon[1], Tilman Plehn[1] and Lorenz Vogel[1]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 LPNHE, Sorbonne Université, Université Paris Cité, CNRS/IN2P3, Paris, France

February 9, 2023

Abstract

Discriminating quark-like from gluon-like jets is, in many ways, a key challenge for many LHC analyses. First, we use a known difference in PYTHIA and HERWIG simulations to show how decorrelated taggers would break down when the most distinctive feature is aligned with theory uncertainties. We propose conditional training on interpolated samples, combined with a controlled Bayesian network, as a more resilient framework. The interpolation parameter can be used to optimize the training evaluated on a calibration dataset, and to test the stability of this optimization. The interpolated training might also be useful to track generalization errors when training networks on simulations.

1 Introduction

Jets are the main analysis objects at the LHC, and the success of the LHC program is, to a large degree, being driven by an improved understanding of jets experimentally and theoretically. In practice, the main task in jet physics is to predict their features precisely, and to use them to tag the parton initiating the jet. The improved understanding of subjet physics at the LHC has allowed us to skip high-level observables and instead analyze jets using low-level detector output employing modern machine learning [1]. ML-tools can incorporate all available information in a jet and significantly improve the performance of classic, multivariate jet taggers. While, in the interest of optimality, taggers should be trained on data, ATLAS and CMS follow a more conservative approach and train taggers on simulations. This attitude reflects a flattering trust in theoretical simulations, but it also creates new sources of uncertainties.

In this paper we add uncertainty-aware features to the CMS ParticleNet tagger [2], using a Bayesian classification network setup [3], and propose an interpolated training method using conditional networks. This combination allows us to capture different sources of uncertainties [4] while protecting the performance of the tagger. The Bayesian setup would raise a flag when the training datasets are too inconsistent to be combined. The conflict between optimal performance and uncertainty control is the key weakness of adversarial training approaches and makes it preferable to instead use nuisance parameters to describe systematics [5]. For theory uncertainties, adversarial approaches are not even likely to cover the full uncertainty range [6].

ML-methods for jet tagging [7] can be applied to the whole range of top jets, Higgs jets, W/Z-jets, τ-jets, bottom or charm jets, all the way to quark vs. gluon jets. From a theoretical and an experimental perspective it is easiest to tag partons of which only the decay products hadronize. For instance top taggers then look for distinctive features like the jet mass or the multiplicity of subjet constituents [8–10]. Being a well-defined problem, top tagging has played a key role in developing and establishing a wide range of network architectures [11], including uncertainty-aware extensions.

In contrast, quark vs. gluon tagging is not actually defined theoretically, beyond leading-order in QCD and including parton splittings. Still, because of its great analysis potential, quark–gluon tagging has a long history [12–19], including early applications at the LHC [20–25]. In spite of the serious theoretical challenges [26–35], efficient ML-approaches have been devised to separate "quark jets" from "gluon jets" [36–41]. They include study of hadronization and detector effects [42] and modern network architecture like transformers [43], Lorentz-equivariant networks [44], and normalizing flows [45]. One way to overcome the fundamental problem of defining quark and gluon jets is to instead use well-defined hypotheses in terms of LHC signatures, for instance mostly quarks in LHC signals like weak boson fusion vs. gluons in QCD backgrounds [46–50]. An alternative method is to train a classifier without labels, just on samples with an enhanced partonic quark or gluon fraction [51].

In this paper we first look at a known issue, namely the differences between HERWIG and PYTHIA jets and the effect of these differences on ML-taggers introduced in Sec. 2.1. To control the cutting-edge ParticleNet tagger and understand its output better, we present its Bayesian variant in Sec. 2.2. It allows us to understand the problem of quark–gluon taggers trained on HERWIG and PYTHIA and makes it obvious that a naive resilience improvement through decorrelation will massively hurt the performance of the tagger, as discussed in Sec. 3. In Sec. 4 we target this problem through a new, interpolated training of the conditional ParticleNet tagger on two distinct samples. We realize this interpolation with the same ParticleNet classifier. After discussing this method in detail, we extend it to a fresh look at a more interpretable, continuous calibration of jet taggers.

2 Dataset and classification network

One of the most exciting goals of subjet tagging is the discrimination of quarks versus gluons. The precise task is not well-defined beyond leading order in QCD, but it approximates important questions like how to identify electroweak decay jets or how to separate weak boson fusion from QCD backgrounds. In both cases, the signals are quark-enriched, while most QCD jets at the LHC come from gluon emission. Another aspect which makes quark–gluon tagging especially interesting is that there exists a study which raises questions about the behavior of ML-taggers in this application.

2.1 Quark–gluon datasets

The starting point of our study are two datasets of simulated quark and gluon jets [52–54], each with 2M jets, one generated with PYTHIA and one generated with HERWIG. The two samples are generated using the partonic processes

$$q\bar{q} \to Z(\to \nu\bar{\nu}) + g \qquad \text{and} \qquad qg \to Z(\to \nu\bar{\nu}) + (uds) \qquad (1)$$

at the 14 TeV LHC, simulated with PYTHIA 8.226 [55, 56] and with HERWIG 7.1.4 [57]. Both setups use default tunes and shower parameters. Hadronization and multi-parton interactions (MPI) are turned on, and we do not consider jets with charm or bottom quark content. The jets are defined through anti-k_T algorithm [58, 59] in FASTJET 3.3.0 [60] with a radius of $R = 0.4$. No detector simulation is included, which cuts into the realism of the analysis, but allows us to extract the underlying question and issues and solve them before adding detector simulations to the problem. For each event the dataset keeps the leading jet, provided

$$p_{T,\text{jet}} = 500 \dots 550 \, \text{GeV} \qquad \text{and} \qquad |\eta_{\text{jet}}| < 1.7 \,. \qquad (2)$$

If we assume that all light-flavor jet constituents are approximately massless, each jet x_i is defined by

$$x_i = \left\{ (p_T, \eta, \phi)_k \right\} \qquad \text{with} \qquad k = 1, \dots, n_C \,. \qquad (3)$$

For our analysis we allow for up to $n_C = 100$ constituents per jet. The jets are zero-padded with constituents, and all constituents have azimuthal angles ϕ within π of the jet.

We refer to the final-state jets from the two partonic processes in Eq. (1) as quark and gluon jets, even through it is clear that this statement is scale dependent and only defined at leading order in perturbation theory. A more appropriate way of referring to these jets would be in the sense of semi-supervised learning and quark-enhanced vs. gluon-enhanced samples. A standard way of realizing this setup would be jets reconstructed as coming from a two-body Z-decay vs. jets produced in association with a Higgs boson.

We supplement the PYTHIA and HERWIG datasets with a third simulation of the two processes in Eq. (1) using SHERPA 2.2.10 [61], again with the default tune and shower parameters. Using PYHEPMC [62, 63], a PYTHON wrapper for the HEPMC2 [64] library, we select the constituent coordinates of those final-state particles not labelled as neutrinos. The SHERPA jets are defined through the PYJET [62, 65] interface to FASTJET. From a physics perspective, the SHERPA jet resembles the HERWIG jets through the common use of cluster fragmentation, but we will see that the numerical results differ.

Each of our three jet datasets consists of 20 files with 100k jets each, equally split between quark and gluon jets. For each generator we divide the dataset into training/validation/test subsets with 200k/50k/50k jets for quarks and gluons, each, unless mentioned otherwise.

For state-of-the-art jet tagging we need to include particle identification (PID) information. Our PYTHIA and HERWIG datasets include two forms of PIDs [52], (i) the full particle-ID

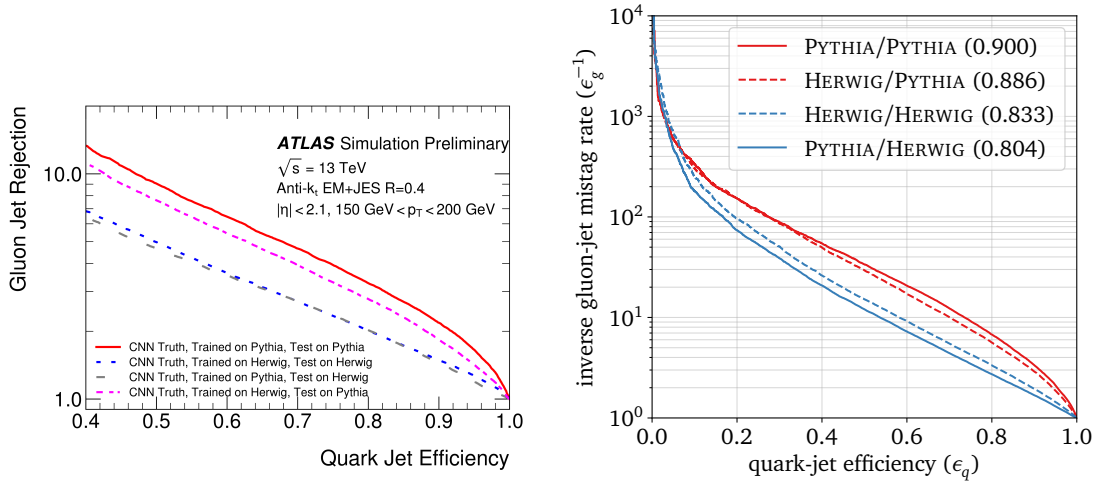



Figure 1: *Left:* preliminary result of the ATLAS study on quark–gluon tagging, raising questions on the best way to train such a tagger. Figure from Ref. [22]. The same pattern has been observed in Fig. 8 of Ref. [37]. *Right:* our results on the same task, using a new, Bayesian version of ParticleNet-Lite [2], trained on 400k PYTHIA and HERWIG jets each, with the parameters given in Tab. 2.

information from PYTHIA or HERWIG, and (ii) experimentally realistic particle IDs. We follow the ParticleNet approach [2], using the five particle types electron, muon, charged hadron, neutral hadron, and photon, plus the electric charge as input to the network. The standard encoding by the Particle Data Group in terms of large and irregular integer values is not an ideal ML-input. Instead, we use a one-hot encoding of our experimentally realistic PIDs.

High-level observables

There exist standard kinematic observables for subjet physics, specifically quark–gluon discrimination [48], for instance the multiplicity of constituents or particle flow objects (n_{PF}), the radiation distribution or girth (w_{PF}) [66, 67], the width of the p_T-distribution of the constituents ($p_T D$) [23], or the weighted angular correlator ($C_{0.2}$) [68]. They are defined in terms of the jet constituents as

$$n_{PF} = \sum_i 1 \qquad\qquad w_{PF} = \frac{\sum_i p_{T,i} \Delta R_{i,\text{jet}}}{\sum_i p_{T,i}}$$
$$p_T D = \frac{\sqrt{\sum_i p_{T,i}^2}}{\sum_i p_{T,i}} \qquad\qquad C_{0.2} = \frac{\sum_{i,j} p_{T,i} p_{T,j} (\Delta R_{ij})^{0.2}}{\left(\sum_i p_{T,i}\right)^2} \ . \tag{4}$$

Distinguishing quark jets from gluon jets exploits two features encoded in these observables [42, 69]. First, the QCD color factors for quarks are smaller than for gluons, which means radiating a gluon off a hard gluon versus off a hard quark comes with the ratio $C_A/C_F = 9/4$. This leads for a higher multiplicity and broader girth for hard gluons. Second, the quark and gluon splitting functions differ in the soft limit. The harder fragmentation for quarks leads to quark jet constituents carrying a larger average fraction of the jet energy, tracked by $p_T D$.

In Fig. 2 we show these four distributions for the quark and gluon jets simulated by PYTHIA, SHERPA, and HERWIG. The biggest difference appears in n_{PF}, where the quark distributions from the three generators are similar, but the gluon distributions vary significantly. The maximum of the broad peak is the smallest, $n_{PF} \sim 40$ for the HERWIG gluons and the largest,

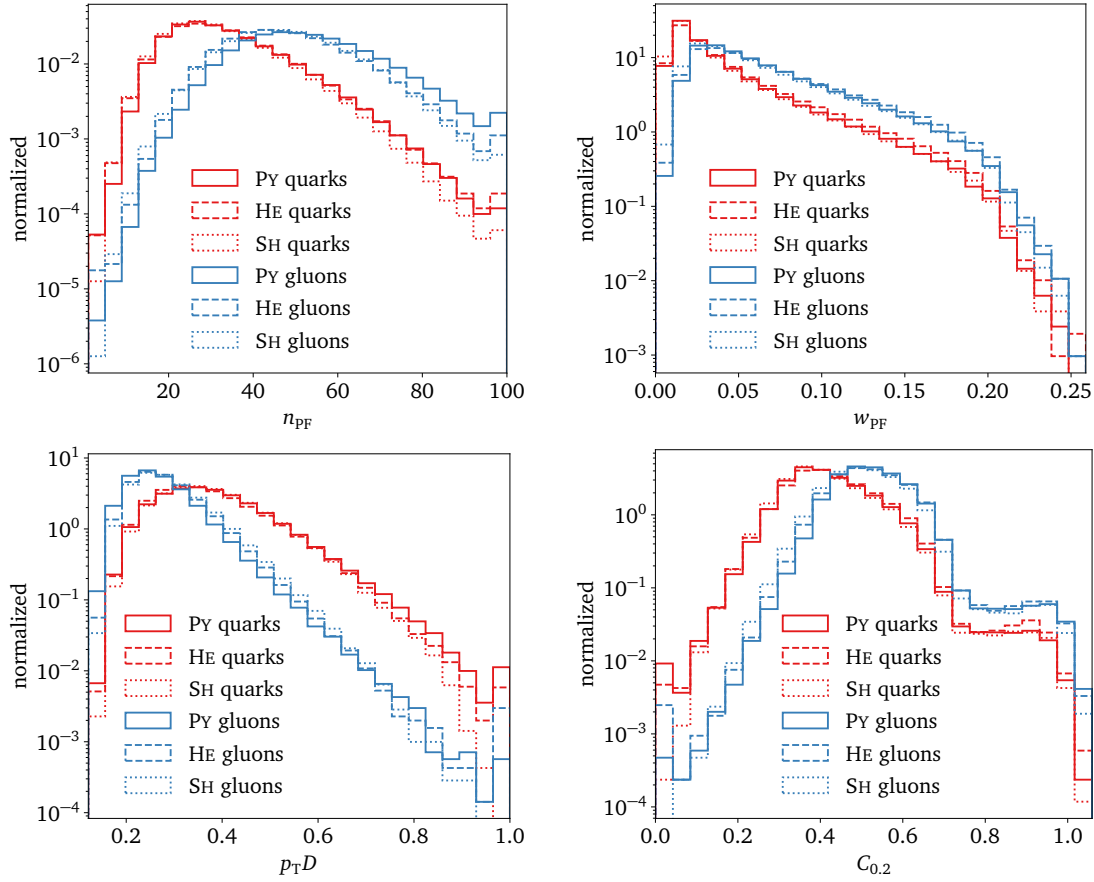



Figure 2: High-level observables, as defined in Eq. (4), for the three different generators, split into quark and gluon jets.

$n_{PF} \sim 45 \ldots 50$ for PYTHIA gluons. This difference between HERWIG and PYTHIA jets is not cause by noise or a broader distribution, but by an actually different prediction for n_{PF}.

This difference in n_{PF} vanishes for w_{PF}, indicating that it comes from infrared and collinear unsafe regions of phase space, and might become less relevant once we include detector effects. We emphasize that this does not mean we should expect the shower algorithms to fail, but that these difference are not easily computable in perturbative QCD. Similarly, the $p_T D$ distributions are significantly different for quarks and gluons, combined with a small shift in the position of the comparably sharp gluon peaks from the different generators. Finally, the only actual two-constituent correlation $C_{0.2}$ is also different for quarks and gluons, but consistent for the different generators. We have studied a range of additional high-level operators and traced significant deviations between the gluon jets from the different generators to a strong correlation with n_{PF}.

generator	n_{PF}	w_{PF}	$p_T D$	$C_{0.2}$
PYTHIA	2050	3207	4000	1316
SHERPA	1711	3149	3217	1112
HERWIG	1326	2910	3406	1128

Table 1: First Wasserstein distance, or earth mover's distance, between quark and gluon distributions for the observables defined in Eq. (4). We show 200k quark jets and 200k gluon jets for each generator.

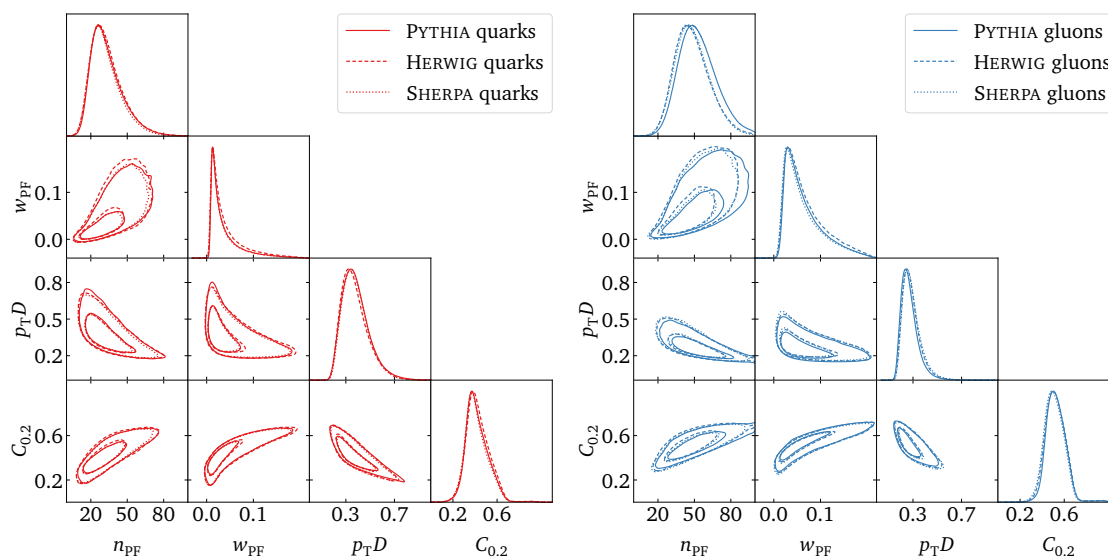



Figure 3: Correlations between the high-level observables from Eq. (4). We show results for the three different generators, split into quark jets (*left*) and gluon jets (*right*).

In Fig. 3 we also show the correlations between the same observables, for each of the three generators and separated into true quark and gluon jets. All observables are correlated with the most powerful n_{PF}, but this correlation is not very different for quarks and for gluons, suggesting that a multi-dimensional analysis will be dominated by the completely understood shifts in n_{PF}.

To judge the relevance of the difference in n_{PF} from the different generators for quark–gluon tagging we can separate quarks from gluons based on the individual observables given in Eq. (4). We can estimate the power of the individual distributions using the Wasserstein distances between 200k quark and gluon jet histograms, as given in Tab. 1. For all observables the PYTHIA jets are most easily separated, followed by SHERPA for n_{PF} and w_{PF}, whereas HER-WIG predicts a stronger discrimination power for $p_T D$ than SHERPA. The actual values of the Wasserstein distance for the different kinematic observables depends on the detailed shape and does not correlate with the separating power of a kinematic cut. We show the corresponding ROC curves in Fig. 4, generated by choosing such a cut value for each observable. The n_{PF}-based and $p_T D$-based tagging shows a significant degradation when tagging HERWIG jets as compared to the easier-to-separate PYTHIA quarks and gluons. This confirms the observation from Fig. 2, where both distributions for HERWIG gluons are further from the common quark distributions than they are for the PYTHIA gluons. In contrast, the tagging performance from w_{PF} and $C_{0.2}$ is unaffected by the choice of simulation and in general also much weaker.

To summarize the key result from this simple study — the most powerful observables for quark–gluon tagging show a significant shift in the gluon predictions between HERWIG and PYTHIA. This shift brings the HERWIG gluons closer to quarks.

2.2 Bayesian ParticleNet

To work with a controlled cutting-edge ML-tagger we develop a Bayesian version of the Particle-Net(-Lite) graph convolutional network architecture [2] adapted from TENSORFLOW to PY-TORCH, to be able to use our standard Bayesian network. For a detailed discussion of Bayesian networks we refer to some original Bayesian network papers [70–72] and the didactic intro-duction in Ref. [1]. The ADAMW optimizer [73, 74], with a weight decay of 10^{-4}, minimizes

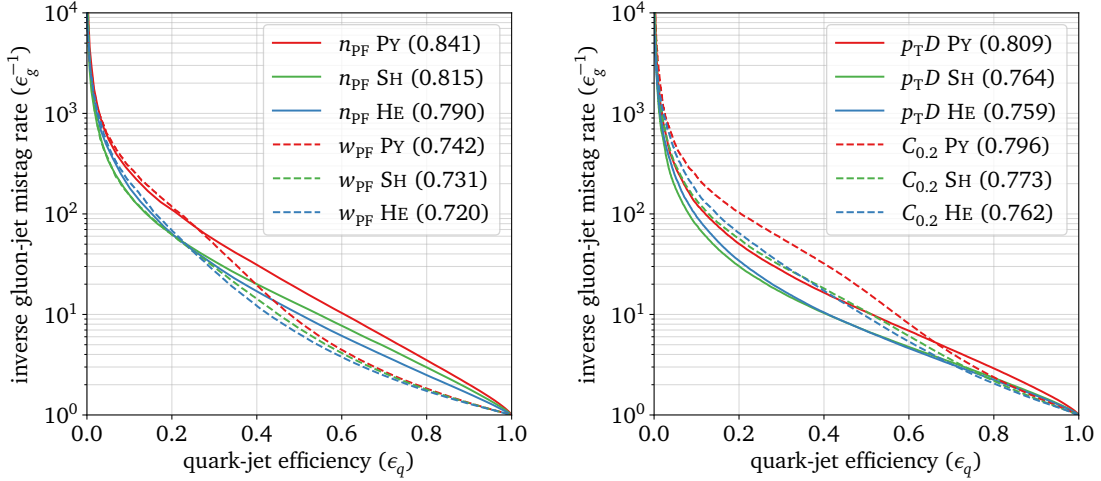



Figure 4: AUCs for quark–gluon discrimination based on individual high-level observables of Eq. (4), simulated with PYTHIA, SHERPA and HERWIG.

the usual binary cross-entropy loss combined with a sigmoid activation function for the classification task,

$$\mathcal{L}_{\mathrm{PN}} = -\frac{1}{M}\sum_{i=1}^{M} y_i \log f(x_i) + (1-y_i)\log\big(1-f(x_i)\big), \tag{5}$$

where M is the mini-batch size, $f(x_i) \in [0,1]$ the model prediction for jet i, and $y_i \in \{0,1\}$ the jet truth-label. The two term in the loss lead to a classification of

$$\begin{aligned} f(x_i) \;\rightarrow\; y_i &= 1 && \text{quark signal} \\ f(x_i) \;\rightarrow\; y_i &= 0 && \text{gluon background}\,. \end{aligned} \tag{6}$$

We adopt the learning-rate scheduling from Ref. [2]. The feature input to the ParticleNet are the hardest 100 jet-constituent particles, specifically

$$\left\{\Delta\eta_k\,,\; \Delta\phi_k\,,\; \Delta R_k\,,\; \log p_{\mathrm{T},k}\,,\; \log\frac{p_{\mathrm{T},k}}{p_{\mathrm{T,jet}}}\,,\; \log E_k\,,\; \log\frac{E_k}{E_{\mathrm{jet}}}\,,\; \mathrm{PID}_k\right\}, \tag{7}$$

where the first coordinates are computed relative to the jet axis. The distance in $\Delta\eta_k$ and $\Delta\phi_k$ are used to compute the distances between particles in the first edge convolution (EdgeConv) block (coordinate input). The PID information includes the particle charge [2, 52].

While deterministic neural networks adapt a large number of weights to approximate a training function, Bayesian neural networks (BNNs) learn distributions of these weights [72].* We can then sample over the weight distributions to produce a central value and an uncertainty distribution for the network output. In LHC physics, Bayesian networks can be applied to classification [3], regression [75,76], and generative networks [4,77,78]. While it is in general possible to separate these uncertainties into statistical and systematic (stochasticity [75] or model limitations [76]), we know that our number of training jets is sufficiently large to only leave us with systematic uncertainties from the training process.

The Bayesian loss follows from a variational approximation of the conditional probability for the network parameters. It combines a likelihood loss with a regularization through a prior

---

*In that sense there is nothing Bayesian about BNNs, they can just be viewed as an extremely efficient way to train network ensembles.

hyperparameter	BPN-Lite architecture
number of EdgeConv blocks	2
number of nearest neighbors	7
number of channels for each EdgeConv block	$(32, 32, 32)$, $(64, 64, 64)$
channel-wise pooling	average
fully-connected layer	128 and ReLU
dropout probability	0.1
number of epochs	100
batch size	128
number of constituents	100, with highest p_T
training/validation/testing	400k/100k/100k
signal-to-background ratio	1.0
re-sampling for testing	80 times

Table 2: Bayesian ParticleNet-Lite (BPN-Lite) architecture and hyperparameters [2].

for the weight distributions,

$$\mathcal{L}_{\text{BPN}} = -\frac{1}{M} \sum_{i=1}^{M} \log p(y_i|x_i, \omega) + \frac{1}{N} \text{KL}\big[q_{\mu,\sigma}(\omega), p_{\mu,\sigma}(\omega)\big] \tag{8a}$$

$$\approx -\frac{1}{M} \sum_{i=1}^{M} \log p(y_i|x_i, \omega) + \frac{1}{2N} \sum_{\text{weights } \omega_j} \big(\mu_j^2 + \sigma_j^2 - \log \sigma_j^2 - 1\big), \tag{8b}$$

where we choose the prior as a Gaussian with mean zero and width one and use the fact that the resulting weight distributions will become approximately Gaussian as well, described by μ_j and σ_j. A change of prior has been shown to not affect the network output [3]. As in Eq. (5) M denotes the mini-batch size, and N is the number of training jets.

The parameters μ_j and σ_j define the model parameters ω_j of the Bayesian network and need to be trained. In our case, only the weights in the linear and 2D-convolutional layers are extended to Gaussian distributions. The hyperparameters of the original ParticleNet(-Lite) network and its Bayesian counterparts are given in Tab. 2. We use the same BPN-Lite network for quark vs. gluon discrimination and for the generator reweighting which we will introduce in Sec. 4.

The performance of the BPN-Lite quark–gluon classifier is illustrated in the right panel of Fig. 1. Independent of the competitive AUC values we see that, as before, the network trained and tested on PYTHIA performs best, closely followed by the network trained on HERWIG and tested on PYTHIA. This suggests that the choice of training sample only has a small effect. In contrast, when we test networks on HERWIG the performance drops significantly, with the consistent training on HERWIG superseding the training on the alternative PYTHIA dataset. This hierarchy indicates that, indeed, PYTHIA quarks and gluons are easier to separate than the HERWIG quarks and gluons, and that the key features for this classification are similar for the two generators. We will study this aspect more closely in the following section.

3 Where have all the gluons gone?

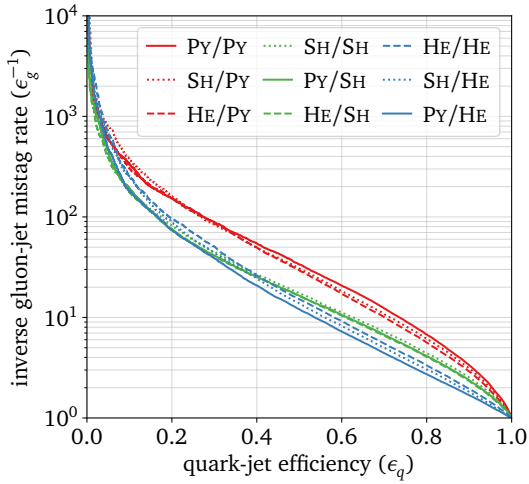
Trying to solve the puzzle of quark–gluon taggers trained and tested on different generators will lead us to the more general question, namely how to control classification networks trained on one dataset and tested on another. All combinations of training and testing the BPN-Lite tagger are illustrated in Fig. 5, with some of the main results collected in the two tables.

AUC		training		
		PYTHIA	SHERPA	HERWIG
testing	PYTHIA	0.900	0.893	0.886
	SHERPA	0.856	0.863	0.853
	HERWIG	0.804	0.820	0.833
$\epsilon_g^{-1}(\epsilon_q=0.50)$		training		
		PYTHIA	SHERPA	HERWIG
testing	PYTHIA	33.8	30.8	29.4
	SHERPA	16.5	17.4	16.4
	HERWIG	12.1	13.8	15.2

Figure 5: *Left:* ROC curves for training and testing on PYTHIA, SHERPA, and HERWIG in different combinations. *Right:* AUC and background rejection performance of the BPN-Lite for quark–gluon tagging, trained and tested on the three different generators.

Since the original PYTHIA and HERWIG results form the two extreme poles, we assign the three datasets to the real-world problem of training a tagger on two independent training datasets and testing it on independent data as

- labelled training dataset 1: PYTHIA,
- labelled training dataset 2: HERWIG,
- independent test dataset: SHERPA.

We will start by comparing different trainings on the labelled PYTHIA and HERWIG datasets, as motivated by Fig. 1 and eventually add SHERPA results as an independent test, in the sense of actual data analyzed by the tagger.

Performance comparison

The Bayesian nature of the BPN-Lite tagger comes with two pieces of information, which allow us to understand the network training. First, the Bayesian tagger provides a per-jet uncertainty $\sigma_{\text{pred}}(x_i)$ on the classification output $\mu_{\text{pred}}(x_i) \in [0, 1]$. This means we can separate jets for which the network training leads to a confident classification from jets where the training provides less information. Second, the final sigmoid layer of the classification network leads to a correlation of μ_{pred} and σ_{pred}, namely

$$\sigma_{\text{pred}}(x_i) \propto \mu_{\text{pred}}(x_i)\big[1 - \mu_{\text{pred}}(x_i)\big]. \tag{9}$$

This inverse parabola correlation is a feature of the network structure and has to be present in the Bayesian tagging output, its absence points to a stability issue in the networks training. In Fig. 6 we show the μ_{pred}- and σ_{pred}-distributions for PYTHIA and HERWIG test datasets, after consistently training the networks on PYTHIA and HERWIG. Already the μ_{pred}-distributions shows three major issues:

1. While the tagging of quarks vs. gluons is never symmetric, training and testing on PYTHIA indicates some gluons confidently identified as gluons $\mu_{\text{pred}}(x_i) \rightarrow 0$.
2. Training and testing on HERWIG hardly ever allows the network to confidently identify gluons with $\mu_{\text{pred}} \lesssim 0.1$.

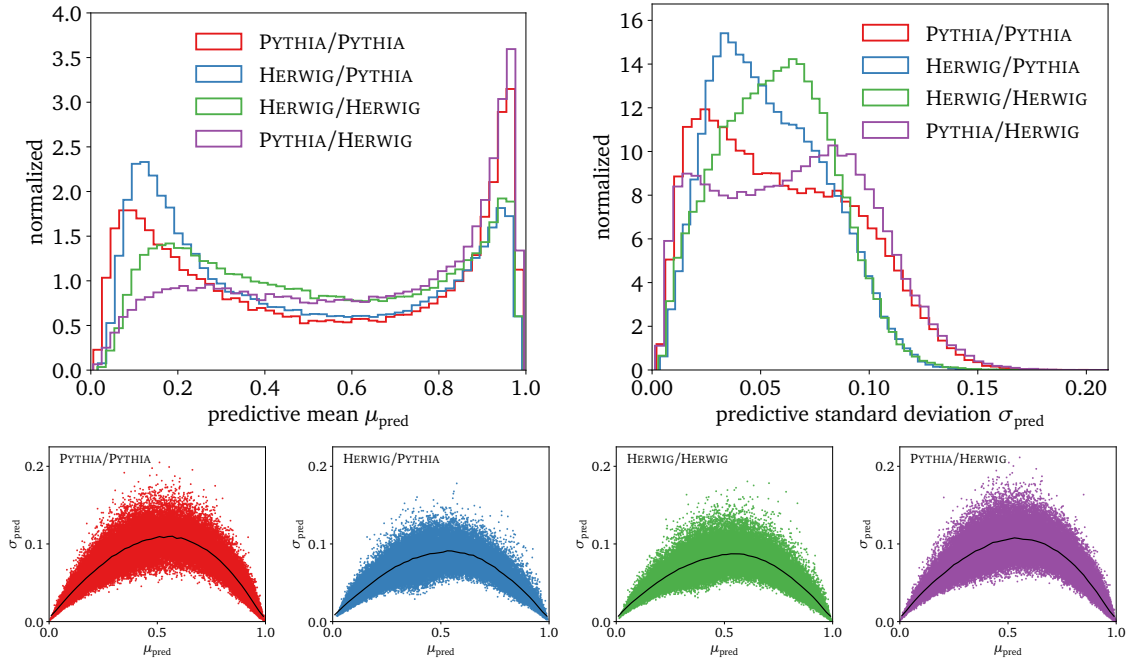



Figure 6: Predictive means ($\mu_{\mathrm{pred}} = 0$ for gluons, $\mu_{\mathrm{pred}} = 1$ for quarks) and standard deviations from the BPN-Lite tagger trained and tested on PYTHIA and HERWIG in different combinations. The lower panels illustrate a stochastic pattern around the correlation of Eq. (9).

3. Training on PYTHIA and testing on HERWIG identifies at least some gluons with as small σ_{pred} as training and testing on PYTHIA.

Looking at the σ_{pred}-distribution, the results from training on PYTHIA look as expected as long as we test on PYTHIA jets, but tested on HERWIG a slight shoulder around $\sigma_{\mathrm{pred}} \sim 0.07$ develops into a second peak. This peak corresponds to jets or phase-space configurations where the PYTHIA training does not allow for a confident application to HERWIG jets. Second, the general uncertainty after training on HERWIG jets peaks at larger σ_{pred}, indicating that the network faces difficulties to extract the relevant features for the tagging, but also drops off at smaller σ_{pred} values than the PYTHIA trainings. This reflects the problem with the single main feature n_{PF}, as expected from our discussion in Sec. 2.1.

Finally, the four lower panels in Fig. 6 show the per-jet correlation of the predictive means and standard deviations. Again confirming our suspicions from Sec. 2.1 that training on HERWIG jets is not completely stable, leading to slight irregularities of the scattering pattern around the inverse parabola predicted by Eq. (9).

High-level observables

We can trace back the problems with the performance and stability of the HERWIG training to the high-level observables of Eq. (4). In Fig. 7 we show two of the most interesting kinematic variables in slices of μ_{pred}, the probabilistic output of BPN-Lite. We know already that n_{PF} is the leading discriminating feature separating quarks from gluons, while $C_{0.2}$ is the only actual correlator amongst the standard high-level observables. In the upper panels we show PYTHIA jets, in the lower panels HERWIG jets. The slices are bases on consistent training and testing on the two samples. For $\mu_{\mathrm{pred}} > 0.6$ the two distributions agree, as expected for correctly identified quarks.

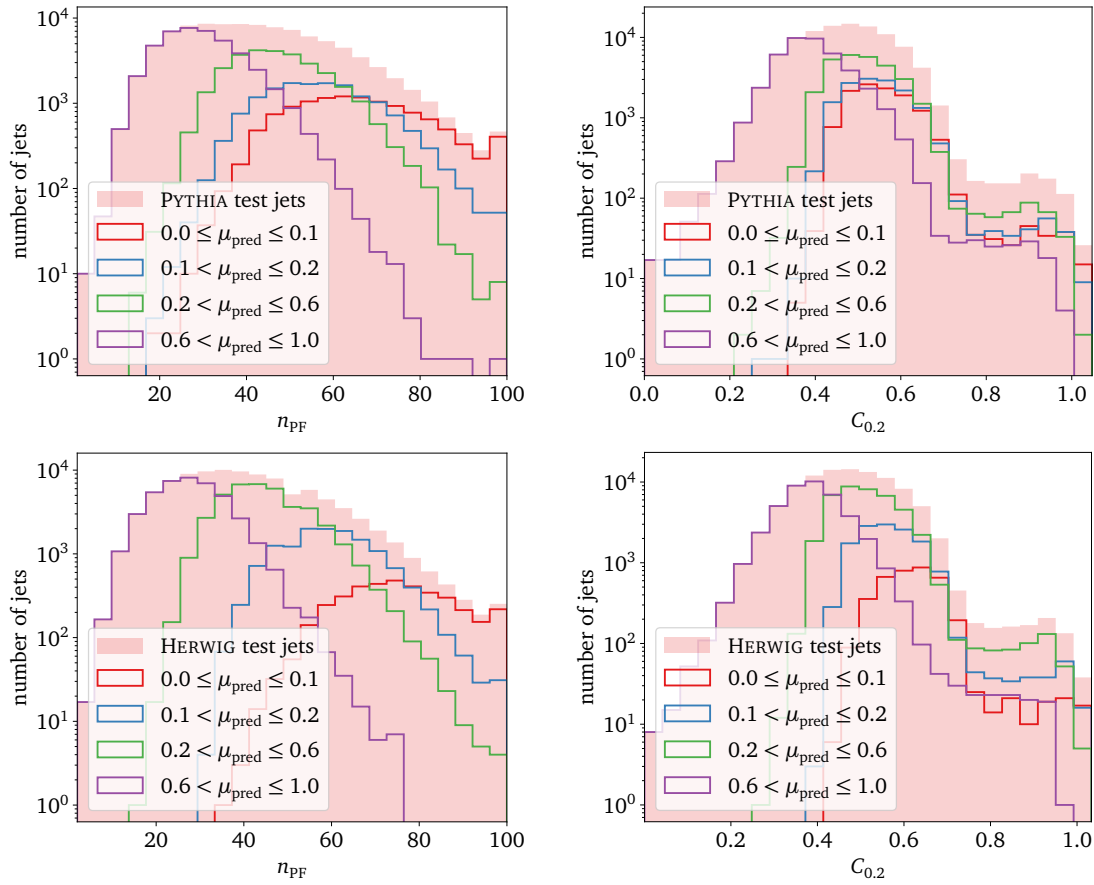



Figure 7: Kinematic distributions defined in Eq. (4). We train the BPN-Lite tagger consistently trained and tested on PYTHIA (*upper*) and on HERWIG (*lower*). The histograms are normalized such that they reflect the fractions of jets in the respective slices in μ_{pred}, extracted from consistent testing.

While the two n_{PF}-distributions are very similar for correctly identified quark-like jets with $\mu_{\text{pred}} > 0.6$, differences appear towards the gluon regime and become quite dramatic for correctly identified gluons with $\mu_{\text{pred}} < 0.1$. Requiring increasingly small μ_{pred} values for more and more confidently identified gluons, the fraction of jets remaining in these slices from the PYTHIA sample is much larger than it is for the HERWIG sample. While for PYTHIA jets values $n_{\text{PF}} > 60$ indicate confidently identified gluons, HERWIG gluons are harder to identify and typically require $n_{\text{PF}} > 70$ to lead to the rare occurrence of $\mu_{\text{pred}} < 0.1$. In the right panels we show the correlator $C_{0.2}$. While the main difference is the number of jets in the individual slices, we also see that the secondary maximum around $C_{0.2} > 0.8$ is predominantly, but not exclusively populated by gluon jets.

Predictive uncertainties

Finally, we can see what the predictive uncertainties tell us in addition to this information from the network performance. For a given tagger the predictive mean μ_{pred} and the predictive standard deviation σ_{pred} are strongly correlated through Eq. (9), but this argument does not hold for different training datasets. In Fig. 8 we show the predictive uncertainties the BPN-Lite tagger extracts when training and testing on all possible combinations of PYTHIA, HERWIG, and SHERPA. In the ranges $\mu_{\text{pred}} \sim 0.1 \dots 0.9$ the different training samples define the size of the predictive uncertainty.

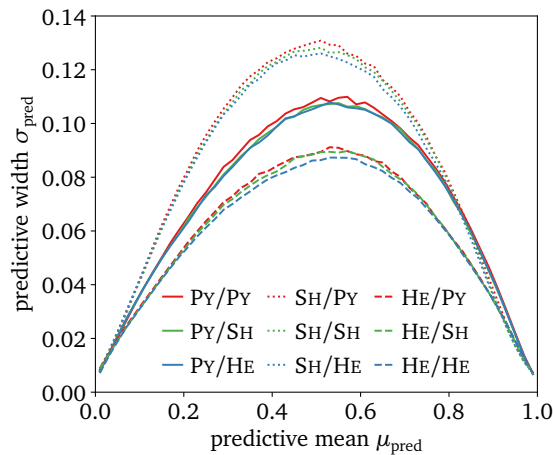



Figure 8: Correlation between the predictive mean and average uncertainty from the BPN-Lite tagger for different combinations of training and testing data.

The ranking of the three generators providing the training dataset is independent of the test sample. This confirms that the predictive uncertainty of the Bayesian network reflects almost entirely limitations in the training data. While μ_{pred} and σ_{pred} are correlated for a given training dataset, the σ_{pred} values in a given range of μ_{pred} are not correlated with the respective μ_{pred} values for different generators. For instance, the poorly performing HERWIG training might not exploit features optimally, but it is less affected for instance by the stochasticity of the training data. We also see that any kind of training on PYTHIA and HERWIG will provide smaller uncertainties on the independent SHERPA data than a Bayesian network trained on SHERPA and tested on SHERPA. We again emphasize that this kind of behavior should not appear for μ_{pred}, because consistent training should provide better performance than inconsistent training, but it can happen for σ_{pred}, as it reflects limitations of the training dataset only.

4 Resilient interpolated training

Once we have understood what the physics issues and the ML-implications with the PYTHIA and HERWIG training datasets are, we can follow the setup from the beginning of Sec. 3 and see how to best deal with two significantly different training datasets, when the tasks is to identify quarks in a third, independent dataset (SHERPA). This corresponds to the standard ATLAS and CMS strategy, which is to train ML-classifiers on Monte Carlo simulations, understand their behavior, and then apply them to data. The major drawback of this strategy is a generalization error whenever simulations do not reproduce data perfectly. Such a generalization error can introduce a bias, but at the very least it is leading to non-optimal performance. A re-calibration should remove biases, but it will not improve poorly trained taggers. We propose a flexible choice of training data, defining an optimal training dataset by evaluating the tagger performance on an independent calibration dataset.

A related question is how to estimate systematic uncertainties related to the choice of training data. In general, whenever uncertainties can be described reliably, it is preferable to include the corresponding nuisance parameters in the analysis, instead of removing a model dependence through adversarial training [5]. Decorrelating theory uncertainties induced by different datasets is especially tricky, since it enforces an insensitive direction in feature space and does not allow us to claim that a general dependence on different training datasets is significantly reduced [6].

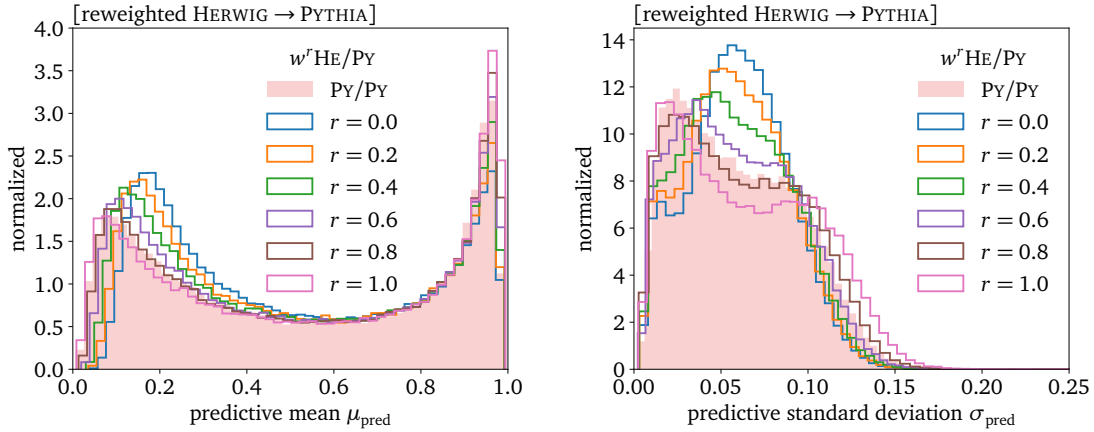



Figure 9: Bayesian ParticleNet-Lite, trained on reweighted HERWIG → PYTHIA jets and tested on PYTHIA jets. The curves should be compared to those in Fig. 6.

In the case of HERWIG vs. PYTHIA training for quark–gluon tagging the situation would be even worse, because the two datasets are systematically different in a way that is fully correlated with the features used for tagging. Decorrelating the difference of the two datasets would effectively remove n_{PF} from the available features and render the tagger useless. Instead, we need to find a way to best train the tagger and assign an uncertainty to this choice of training data.

Interpolated training samples

To add some resilience to the otherwise extreme choice of training either on HERWIG or on PYTHIA, we would like to use a combination of the two datasets for a stable training, benchmarked on the independent SHERPA data. There are, at least, two ways to interpolate between the two training datasets. First, we simply train the network on mixtures of quarks from PYTHIA and HERWIG vs. mixtures of gluons from PYTHIA and HERWIG in the same proportions,

$$\text{HERWIG training} \quad \xleftrightarrow{0 \leq r \leq 1} \quad \text{PYTHIA training} . \tag{10}$$

The interpolation parameter r for the mixed sample is the fraction of PYTHIA jets in the training dataset.

An alternative method to achieve the same interpolated training is to train a discriminator on PYTHIA vs. HERWIG quark and gluon jets and to re-weight the HERWIG jets to their PYTHIA counterparts. Since each jet now comes with a weight, this method is also only defined on jet samples. This method has the advantage that we can train the network conditional on the interpolation parameter $r = 0 \dots 1$, to stabilize the training. In our case, the discriminator between HERWIG and PYTHIA jets is the same BPN-Lite network used to tag quarks vs. gluons. We use the same settings as in Tab. 2 and the same loss function as in Eq. (8). The only difference it that for the HERWIG vs. PYTHIA case we use generator truth-labels instead of jet truth-labels. We train the HERWIG vs. PYTHIA discriminator for quarks and gluons separately.

Using the per-jet reweighting factors from the classification network,

$$w(x_i) = \frac{p_{PY}(x_i)}{p_{HE}(x_i)} , \tag{11}$$

we can train a quark–gluon classifier on w^r-reweighted HERWIG jets. The weights enter the

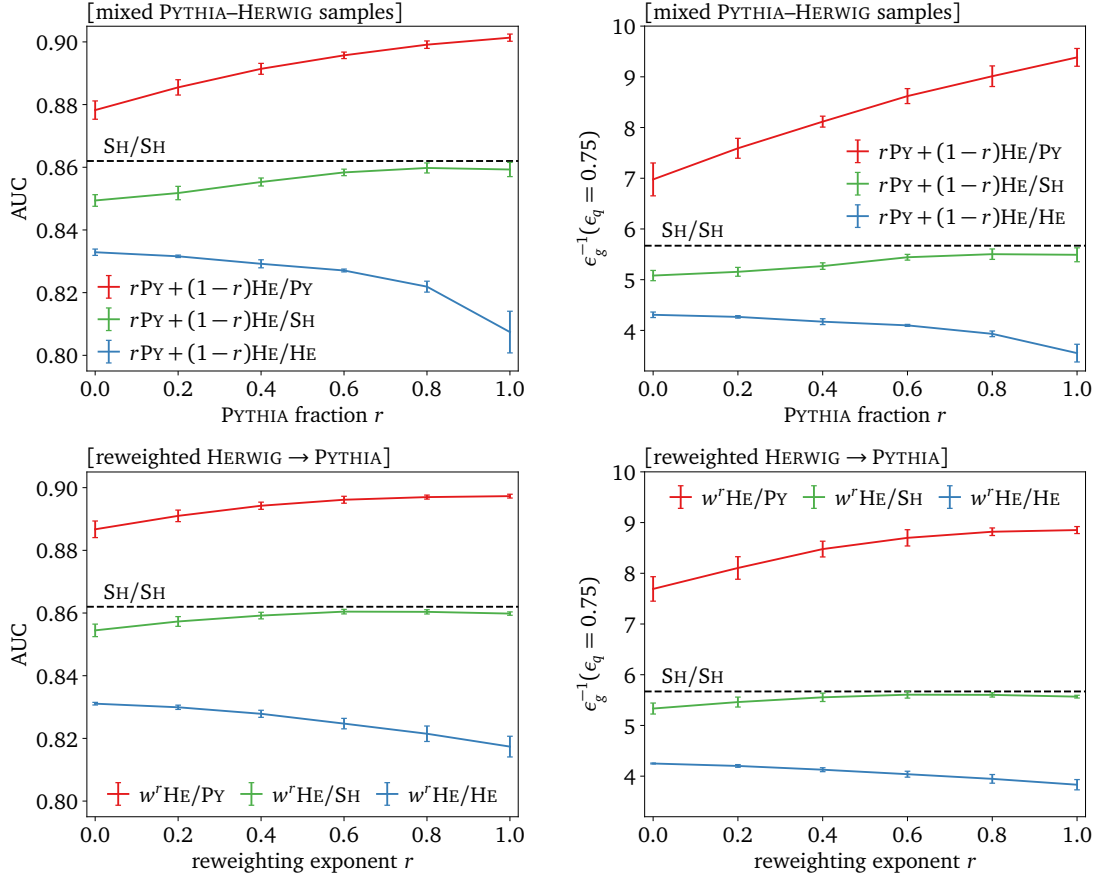



Figure 10: Performance of the interpolated training on HERWIG → PYTHIA, using mixed samples (*upper*) and conditional reweighting (*lower*). The performance is tested on pure HERWIG, PYTHIA, and independent SHERPA data. The error bars reflect six independent network trainings.

BPN-Lite loss function of Eq. (8) as

$$\mathcal{L}_{\text{BPN}} = -\frac{1}{M} \sum_{i=1}^{M} w(x_i)^r \log p(y_i|x_i, \omega) + \frac{1}{2N} \sum_{\text{weights } \omega_j} \left(\mu_j^2 + \sigma_j^2 - \log \sigma_j^2 - 1 \right), \quad (12)$$

and the reweighting exponent r is used as an additional feature input, uniformly sampled from $[0, 1]$ during training. In Fig. 9 we illustrate how the conditional reweighting network works on HERWIG jets. We show the distributions of the predictive mean μ_{pred} and the predictive uncertainty σ_{pred} for a tagger trained conditionally on the weighted samples and tested on PYTHIA jets. In the limit $r \to 1$ the results approach the consistent PYTHIA training and testing shown in Fig. 6.

Optimized training data and uncertainties

The new aspect in this section is the performance of the interpolated training on the independent SHERPA data. Now, r can be understood as a hyperparameter of the network training, so we can choose an optimal value from the independent calibration sample, in our case SHERPA. The actual tagging performance of the two methods of interpolated training is shown in Fig. 10, with mixed samples in the upper panels and reweighting in a conditional network setup in the lower panels. First, the results of the two methods are completely consistent with each other.

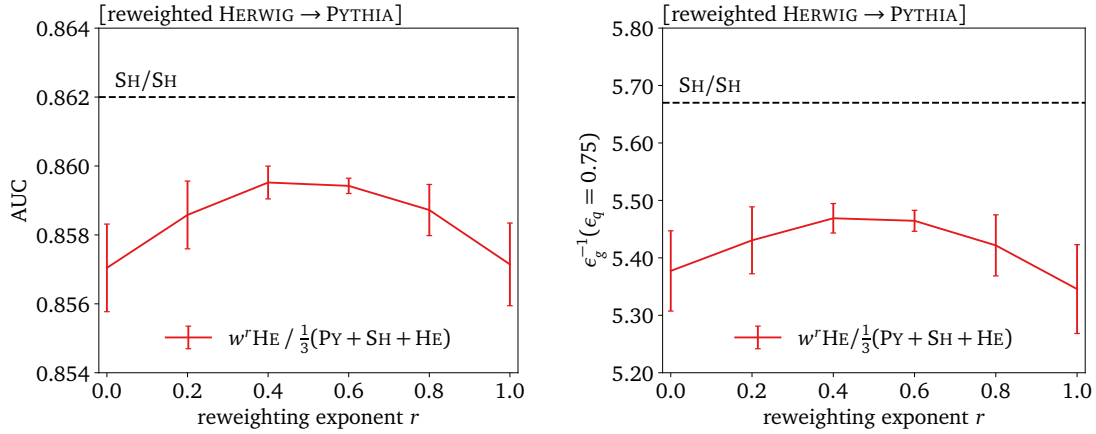



Figure 11: Performance of the interpolated training on HERWIG → PYTHIA, using conditional reweighting. The performance is tested on equal parts of HERWIG, PYTHIA, and SHERPA jets. The error bars reflect six independent network trainings.

They are also consistent with our previous results; for $r = 1$ the performance on all three test datasets approaches the results from proper PYTHIA training in Fig. 5. Similarly, in the limit $r = 0$ we reproduce the performance of HERWIG-based training. In between, we observe a continuous and featureless performance drop from PYTHIA to SHERPA training. Comparing the two methods, the conditional reweighting setup is smoother than the mixed sample training. Finally, after optimizing the interpolated training sample, the achieves performance comes very close to the consistent SHERPA training and testing.

As a side remark, while testing on SHERPA jets leads us to conclude that a choice $r \to 1$ provides the optimal tagging performance, we can also test the interpolated training on combination of HERWIG, PYTHIA, and SHERPA jets. Because the power of the main tagging features in the SHERPA dataset tends to lie in between HERWIG and PYTHIA, shown in Tab. 1, an interpolated training with $r \approx 0$ now gives the best tagging performance.

After optimizing the performance on a calibration dataset, we can also vary the interpolation parameter r around its optimal value to estimate the uncertainty from our parameter choice. In the lower panels of Fig. 10 we see that for our setup the uncertainty from optimizing in the range $r \approx 0.5 \ldots 1.0$ are significantly smaller than the variation from different network trainings. Strictly speaking, the performance gap even of the best training on the combined PYTHIA and HERWIG sample is significant, gauged by the uncertainty from the choice of r and from different trainings. While our example interpolates between two samples, this kind of uncertainty estimate can easily be generalized to many training setups with a conditional reweighting network.

Training-related, predictive uncertainties

We can make use of the uncertainty-aware BPN-Lite tagger to provide the uncertainties σ_{pred} for the interpolated training shown in Fig. 12. In analogy to the performance test in Fig. 10 we now show σ_{pred} as a function of the interpolation parameter r. We know from Fig. 8 that the predictive uncertainties are given by the training data, and we can confirm that the interpolated training reproduces the small HERWIG uncertainties for $r = 0$ and the slightly larger PYTHIA uncertainties for $r = 1$. The reweighted and less consistent sample does not pose a challenge to the training, and the induced generalization errors are not large enough to affect the results for the different test datasets. As alluded to before, the interpolated training on PYTHIA and HERWIG comes with smaller uncertainties than consistent training on SHERPA,

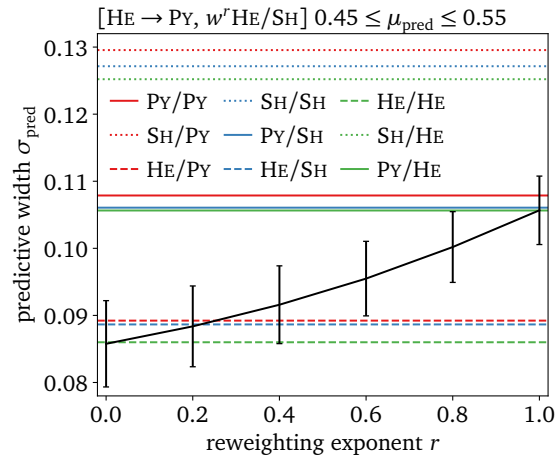



Figure 12: Predictive width for interpolated training on HERWIG → PYTHIA, using conditional reweighting. The error bars indicate the ranges from six independent trainings.

even when tested on SHERPA data. This can make sense, if the predictive uncertainties just reflect limitations in the training, for instance noise or stochasticity.

Calibration and uncertainties

One measurement where we expect the generalization error to appear is the calibration of the different taggers. In principle, the Bayesian PN-Lite tagger should be calibrated, but of course the calibration is only guaranteed when we train and test on consistent data. Any deviation from this consistency is expected to lead to a poorer calibration. In the left panel of Fig. 13 we first confirm that the consistent training and testing leads to well-calibrated taggers over the entire tagging score.

The picture changes when we train the tagger conditionally on the HERWIG–PYTHIA interpolation and evaluate the calibration on the independent SHERPA sample. In the right panel of Fig. 13 we see that HERWIG training leads to a well-calibrated tagger on the SHERPA dataset, reflecting the fact that the physics properties behind the two samples are similar. On the other hand, training on PYTHIA data leads to a poorly calibrated tagger on SHERPA data. Here, the fraction of correctly identified quark jets is lower than the score, which means the tagger is overconfident. This is consistent with PYTHIA being the dataset where it is easiest to separate quarks from gluons.

Because the change in the calibration curve reflects a more dramatic r-dependence than the network performance in Fig. 10 and the predictive uncertainty in Fig. 12, it provides the best handle on the generalization error which arises when we train a tagger flexibly on different generated samples and apply it to actual (calibration) data.

To summarize our findings from the interpolated training between HERWIG and PYTHIA and testing on SHERPA: if we are interested in the tagging performance only, we need to optimize $r \to 1$, corresponding to training on pure PYTHIA jets. When we want to minimize the Bayesian uncertainties from the training data, training with $r \to 0$, or on HERWIG, will give the smallest predictive uncertainties. Finally, when we want to maintain the tagger calibration, we again need to train on $r \to 0$ (HERWIG). Even in ML-applications there is no one size that fits all.

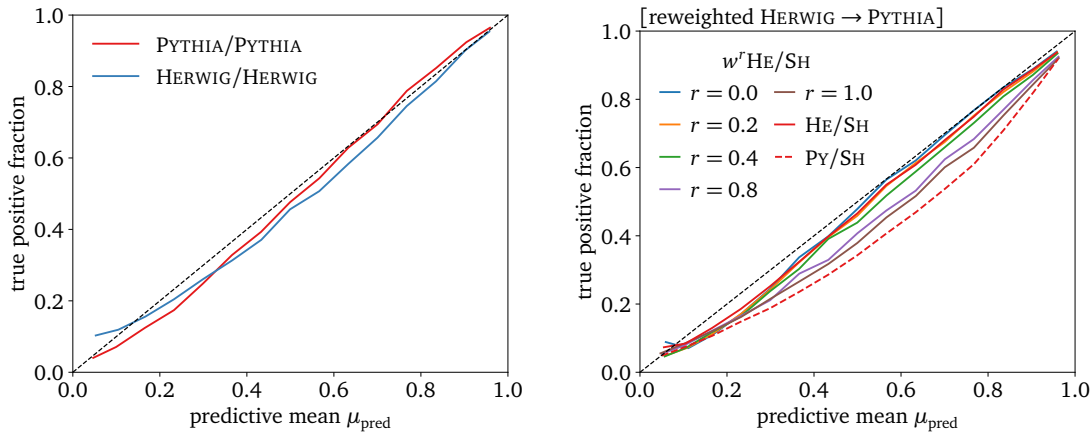



Figure 13: *Left:* calibration curves for consistent PYTHIA and HERWIG training and testing. *Right:* the same calibration curves using conditional HERWIG → PYTHIA reweighting on the training data, tested on SHERPA. We also show the benchmark results from training on pure HERWIG and PYTHIA jets, corresponding to $r \in \{0, 1\}$.

Continuous calibration

If it possible to train a tagger on a continuous interpolation between different datasets, the same kind interpolation should be possible between a simulated training dataset and an approximately labeled calibration dataset. A reliable training dataset should then be transferable into the actual data continuously, and without major changes in the performance and the behavior of the ML-tagger.

Instead of the interpolated training of Eq. (10), we now look at a triangle, defined by flexible training on an interpolated HERWIG–PYTHIA dataset and an additional interpolation between the training data and the independent SHERPA data,

$$\text{HERWIG/PYTHIA training} \quad \xleftrightarrow{0 \leq r \leq 1} \quad \text{SHERPA training} . \tag{13}$$

In this scheme, we can interpolate several ways, within the training data, as described before, and from any kind of training data to the calibration data. The calibration data will not be properly labeled, so we can either rely on an approximate labeling or apply classification without labels [51]. The only tool required to implement this complex interpolation program is the same ParticleNet classification network that is used for the actual tagging task.

In Fig. 14 we illustrate the network training on a continuous interpolation between the simulated data and the calibration data. The benchmark performance is defined by the consistent SHERPA training and testing. As expected, there is hardly any change in performance when we train on PYTHIA jets, which means that the calibration of the tagger can focus on the correct calibration. The situation is different for HERWIG training data, where the performance indicates a generalization gap, but the calibration of the tagger is stable across the interpolation parameter. A situation like the one shown in Fig. 14 indicates that training the tagger on simulations (PYTHIA) and applying it to data is, essentially, optimal, with a very small uncertainty due to the generalization.

5 Conclusions

Now that it has become clear that ML-jet taggers will provide a transformative performance boost to a huge number of LHC analyses, we should turn our attention to aspects like resilience,

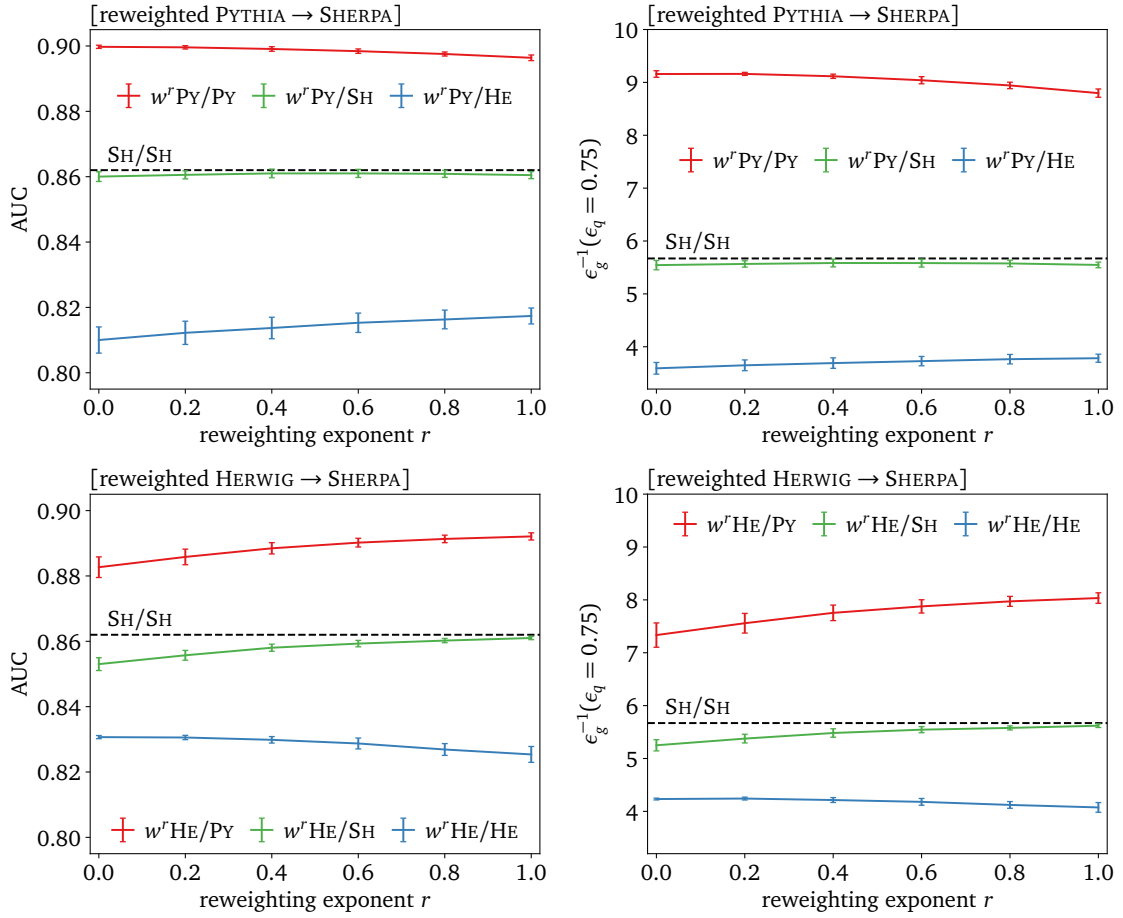



Figure 14: Performance of the interpolation between training and test data, using conditional reweighting PYTHIA → SHERPA (*upper*) and HERWIG → SHERPA (*lower*). The performance is benchmarked on pure HERWIG, PYTHIA, and SHERPA data. The error bars reflect six independent trainings.

uncertainties, control, or explainability. None of these are particularly strong points for classic multivariate taggers, so we again expect ML-taggers to further outperform traditional methods.

As long as we train taggers on simulations and test them on, or apply them to, an independent dataset, generalization errors will limit their performance, even if we remove biases through calibration. These generalization errors contribute to the theory uncertainty, specifically the dependence of the analysis outcome on the Monte Carlo simulation.

First, we have shown for quark–gluon tagging based on HERWIG and PYTHIA training data that improving the resilience through adversarial training is bound to fail, because the number of constituents is not only the leading tagging feature, it is also the main difference between the two simulations.

Just relying on two discrete datasets makes it hard to properly evaluate the corresponding theory uncertainty. We proposed a conditional training on a continuous interpolation between two training datasets, where the interpolation is best implemented using re-weighting through a classification network. The continuous interpolation parameter allows us to optimize the tagging performance and to estimate the related uncertainty. Our method can be generalized to larger numbers of training datasets and to continuous parameters describing the training datasets.

A Bayesian version of the ParticleNet(-Lite) subjet tagger allows us to track the stability of

the conditional training and identify training-related uncertainties or even a breakdown of the interpolated training. For our application to quark–gluon tagging, trained on an interpolation between HERWIG and PYTHIA jets and tested on SHERPA jets, we find that from a pure performance perspective, training on PYTHIA gives the best results. They are very close to training on SHERPA and indicate a very small generalization gap. In contrast, if we are interested in small predictive uncertainties from the Bayesian network, we best train on HERWIG data. Similarly, for in a stable calibration HERWIG training also outperforms PYTHIA training, reflecting a common physics picture between HERWIG and SHERPA. For a test dataset combining the three generators, an interpolated training dataset right in between HERWIG and PYTHIA performs best. This indicates that different objectives require a flexible approach to simulation-based training.

Finally, we have speculated that our continuous interpolation between training samples can be generalized to an interpolation between training and calibration data, turning the actual calibration into a continuous procedure, where stability issues should be easily detectable.

Acknowledgements

We would like to thank Michel Luchmann for inspiring discussions on Bayesian networks, Huilin Qu for expert advice on jet tagging, Theo Heimel for help with the SHERPA dataset, and Mathias Trabs for very helpful advice. A coordinated publication by Markus Klute and his group, reflecting inspiring discussions, will appear early in 2023. Furthermore, we would like to thank Fabrizio Klassen for his collaboration during an early phase of this project. AB and TP would like to thank the Baden-Württemberg-Stiftung for financing through the program *Internationale Spitzenforschung*, project *Uncertainties – Teaching AI its Limits* (BWST_IF2020-010). BMD acknowledges funding from the Alexander von Humboldt Foundation. This research is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under grant 396021762 – TRR 257: *Particle Physics Phenomenology after the Higgs Discovery* and through Germany's Excellence Strategy EXC 2181/1 – 390900948 (the *Heidelberg STRUCTURES Excellence Cluster*).

References

[1] T. Plehn, A. Butter, B. Dillon and C. Krause, *Modern Machine Learning for LHC Physicists*, arXiv:2211.01421.

[2] H. Qu and L. Gouskos, *ParticleNet: Jet Tagging via Particle Clouds*, Phys. Rev. D **101**, 056019 (2020), doi:10.1103/PhysRevD.101.056019, arXiv:1902.08570.

[3] S. Bollweg, M. Haußmann, G. Kasieczka, M. Luchmann, T. Plehn and J. Thompson, *Deep-Learning Jets with Uncertainties and More*, SciPost Phys. **8**, 006 (2020), doi:10.21468/SciPostPhys.8.1.006, arXiv:1904.10004.

[4] A. Butter, T. Heimel, S. Hummerich, T. Krebs, T. Plehn, A. Rousselot and S. Vent, *Generative Networks for Precision Enthusiasts*, arXiv:2110.13632.

[5] A. Ghosh, B. Nachman and D. Whiteson, *Uncertainty-aware machine learning for high energy physics*, Phys. Rev. D **104**, 056026 (2021), doi:10.1103/PhysRevD.104.056026, arXiv:2105.08742.

[6] A. Ghosh and B. Nachman, *A cautionary tale of decorrelating theory uncertainties*, Eur. Phys. J. C **82**, 46 (2022), doi:10.1140/epjc/s10052-022-10012-w, arXiv:2109.08159.

[7] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman and A. Schwartzman, *Jet-images — deep learning edition*, J. High Energy Phys. **07**, 069 (2016), doi:10.1007/JHEP07(2016)069, arXiv:1511.05190.

[8] G. Kasieczka, T. Plehn, M. Russell and T. Schell, *Deep-learning Top Taggers or The End of QCD?*, J. High Energy Phys. **05**, 006 (2017), doi:10.1007/JHEP05(2017)006, arXiv:1701.08784.

[9] A. Butter, G. Kasieczka, T. Plehn and M. Russell, *Deep-learned Top Tagging with a Lorentz Layer*, SciPost Phys. **5**, 028 (2018), doi:10.21468/SciPostPhys.5.3.028, arXiv:1707.08966.

[10] S. Macaluso and D. Shih, *Pulling Out All the Tops with Computer Vision and Deep Learning*, J. High Energy Phys. **10**, 121 (2018), doi:10.1007/JHEP10(2018)121, arXiv:1803.00107.

[11] G. Kasieczka, T. Plehn, A. Butter, K. Cranmer, D. Debnath, B. M. Dillon, M. Fairbairn, D. A. Faroughy, W. Fedorko, C. Gay, L. Gouskos, J. F. Kamenik *et al.*, *The Machine Learning Landscape of Top Taggers*, SciPost Phys. **7**, 014 (2019), doi:10.21468/SciPostPhys.7.1.014, arXiv:1902.09914.

[12] R. D. Field and R. P. Feynman, *A Parametrization of the Properties of Quark Jets*, Nucl. Phys. B **136**, 1 (1978), doi:10.1016/0550-3213(78)90015-9.

[13] H. P. Nilles and K. H. Streng, *Quark-Gluon Separation in Three Jet Events*, Phys. Rev. D **23**, 1944 (1981), doi:10.1103/PhysRevD.23.1944.

[14] L. M. Jones, *Tests for Determining the Parton Ancestor of a Hadron Jet*, Phys. Rev. D **39**, 2550 (1989), doi:10.1103/PhysRevD.39.2550.

[15] Z. Fodor, *How to See the Differences Between Quark and Gluon Jets*, Phys. Rev. D **41**, 1726 (1990), doi:10.1103/PhysRevD.41.1726.

[16] L. Jones, *Towards a Systematic Jet Classification*, Phys. Rev. D **42**, 811 (1990), doi:10.1103/PhysRevD.42.811.

[17] L. Lönnblad, C. Peterson and T. Rögnvaldsson, *Using neural networks to identify jets*, Nucl. Phys. B **349**, 675 (1991), doi:10.1016/0550-3213(91)90392-B.

[18] CLEO Collaboration, *Shape studies of quark jets versus gluon jets at $\sqrt{s} = 10$ GeV*, Phys. Rev. D **46**, 4822 (1992), doi:10.1103/PhysRevD.46.4822.

[19] OPAL Collaboration, *Test of QCD analytic predictions for the multiplicity ratio between gluon and quark jets*, Phys. Lett. B **388**, 659 (1996), doi:10.1016/S0370-2693(96)01319-6.

[20] ATLAS Collaboration, *Light-quark and gluon jet discrimination in pp collisions at $\sqrt{s} = 7$ TeV with the ATLAS detector*, Eur. Phys. J. C **74**, 3023 (2014), doi:10.1140/epjc/s10052-014-3023-z, arXiv:1405.6583.

[21] ATLAS Collaboration, *Quark versus Gluon Jet Tagging Using Charged Particle Multiplicity with the ATLAS Detector*, ATLAS PUB Note (2017), https://cds.cern.ch/record/2263679.

[22] ATLAS Collaboration, *Quark versus Gluon Jet Tagging Using Jet Images with the ATLAS Detector*, ATLAS PUB Note (2017), http://cds.cern.ch/record/2275641.

[23] CMS Collaboration, *Performance of quark/gluon discrimination in* 8 TeV *pp data*, CMS Physics Analysis Summary (2013), http://cds.cern.ch/record/1599732.

[24] CMS Collaboration, *Search for $t\bar{t}H$ production in the all-jet final state in proton-proton collisions at $\sqrt{s}$* = 13 TeV, J. High Energy Phys. **06**, 101 (2018), doi:10.1007/JHEP06(2018)101, arXiv:1803.06986.

[25] M. Andrews, J. Alison, S. An, P. Bryant, B. Burkle, S. Gleyzer, M. Narain, M. Paulini, B. Poczos and E. Usai, *End-to-end jet classification of quarks and gluons with the CMS Open Data*, Nucl. Instrum. Meth. A **977**, 164304 (2020), doi:10.1016/j.nima.2020.164304, arXiv:1902.08276.

[26] G. F. Sterman and S. Weinberg, *Jets from Quantum Chromodynamics*, Phys. Rev. Lett. **39**, 1436 (1977), doi:10.1103/PhysRevLett.39.1436.

[27] S. D. Ellis and D. E. Soper, *Successive combination jet algorithm for hadron collisions*, Phys. Rev. D **48**, 3160 (1993), doi:10.1103/PhysRevD.48.3160, arXiv:hep-ph/9305266.

[28] M. H. Seymour, *Jet shapes in hadron collisions: Higher orders, resummation and hadronization*, Nucl. Phys. B **513**, 269 (1998), doi:10.1016/S0550-3213(97)00711-6, arXiv:hep-ph/9707338.

[29] S. Catani and B. R. Webber, *Infrared safe but infinite: Soft gluon divergences inside the physical region*, J. High Energy Phys. **10**, 005 (1997), doi:10.1088/1126-6708/1997/10/005, arXiv:hep-ph/9710333.

[30] A. Banfi, G. P. Salam and G. Zanderighi, *Infrared safe definition of jet flavor*, Eur. Phys. J. C **47**, 113 (2006), doi:10.1140/epjc/s2006-02552-4, arXiv:hep-ph/0601139.

[31] G. P. Salam, *Towards Jetography*, Eur. Phys. J. C **67**, 637 (2010), doi:10.1140/epjc/s10052-010-1314-6, arXiv:0906.1833.

[32] P. Gras, S. Höche, D. Kar, A. Larkoski, L. Lönnblad, S. Plätzer, A. Siódmok, P. Skands, G. Soyez and J. Thaler, *Systematics of quark/gluon tagging*, J. High Energy Phys. **07**, 091 (2017), doi:10.1007/JHEP07(2017)091, arXiv:1704.03878.

[33] P. T. Komiske, E. M. Metodiev and J. Thaler, *An operational definition of quark and gluon jets*, J. High Energy Phys. **11**, 059 (2018), doi:10.1007/JHEP11(2018)059, arXiv:1809.01140.

[34] A. J. Larkoski and E. M. Metodiev, *A Theory of Quark vs. Gluon Discrimination*, J. High Energy Phys. **10**, 014 (2019), doi:10.1007/JHEP10(2019)014, arXiv:1906.01639.

[35] A. Romero, D. Whiteson, M. Fenton, J. Collado and P. Baldi, *Safety of Quark/Gluon Jet Classification*, arXiv:2103.09103.

[36] J. Gallicchio and M. D. Schwartz, *Quark and Gluon Jet Substructure*, J. High Energy Phys. **04**, 090 (2013), doi:10.1007/JHEP04(2013)090, arXiv:1211.7038.

[37] P. T. Komiske, E. M. Metodiev and M. D. Schwartz, *Deep learning in color: towards automated quark/gluon jet discrimination*, J. High Energy Phys. **01**, 110 (2017), doi:10.1007/JHEP01(2017)110, arXiv:1612.01551.

[38] T. Cheng, *Recursive Neural Networks in Quark/Gluon Tagging*, Comput. Softw. Big Sci. **2**, 3 (2018), doi:10.1007/s41781-018-0007-y, arXiv:1711.02633.

[39] H. Luo, M.-x. Luo, K. Wang, T. Xu and G. Zhu, *Quark jet versus gluon jet: fully-connected neural networks with high-level features*, Sci. China Phys. Mech. Astron. **62**, 991011 (2019), doi:10.1007/s11433-019-9390-8, arXiv:1712.03634.

[40] K. Fraser and M. D. Schwartz, *Jet Charge and Machine Learning*, J. High Energy Phys. **10**, 093 (2018), doi:10.1007/JHEP10(2018)093, arXiv:1803.08066.

[41] E. Alvarez, M. Spannowsky and M. Szewc, *Unsupervised Quark/Gluon Jet Tagging With Poissonian Mixture Models*, Front. Artif. Intell. **5**, 852970 (2022), doi:10.3389/frai.2022.852970, arXiv:2112.11352.

[42] G. Kasieczka, N. Kiefer, T. Plehn and J. M. Thompson, *Quark-Gluon Tagging: Machine Learning vs Detector*, SciPost Phys. **6**, 069 (2019), doi:10.21468/SciPostPhys.6.6.069, arXiv:1812.09223.

[43] V. Mikuni and F. Canelli, *Point Cloud Transformers applied to Collider Physics*, Mach. Learn.: Sci. Technol. **2**, 035027 (2021), doi:10.1088/2632-2153/ac07f6, arXiv:2102.05073.

[44] S. Gong, Q. Meng, J. Zhang, H. Qu, C. Li, S. Qian, W. Du, Z.-M. Ma and T.-Y. Liu, *An efficient Lorentz equivariant graph neural network for jet tagging*, J. High Energy Phys. **07**, 030 (2022), doi:10.1007/JHEP07(2022)030, arXiv:2201.08187.

[45] M. J. Dolan and A. Ore, *TopicFlow: Disentangling quark and gluon jets with normalizing flows*, arXiv:2211.16053.

[46] J. Gallicchio and M. D. Schwartz, *Pure Samples of Quark and Gluon Jets at the LHC*, J. High Energy Phys. **10**, 103 (2011), doi:10.1007/JHEP10(2011)103, arXiv:1104.1175.

[47] S. Bright-Thonney and B. Nachman, *Investigating the Topology Dependence of Quark and Gluon Jets*, J. High Energy Phys. **03**, 098 (2019), doi:10.1007/JHEP03(2019)098, arXiv:1810.05653.

[48] A. Biekötter, F. Keilbach, R. Moutafis, T. Plehn and J. Thompson, *Tagging Jets in Invisible Higgs Searches*, SciPost Phys. **4**, 035 (2018), doi:10.21468/SciPostPhys.4.6.035, arXiv:1712.03973.

[49] Y. Sakaki, *Quark jet rates and quark-gluon discrimination in multijet final states*, Phys. Rev. D **99**, 114012 (2019), doi:10.1103/PhysRevD.99.114012, arXiv:1807.01421.

[50] C.-W. Chiang, D. Shih and S.-F. Wei, *VBF vs. GGF Higgs with Full-Event Deep Learning: Towards a Decay-Agnostic Tagger*, arXiv:2209.05518.

[51] E. M. Metodiev, B. Nachman and J. Thaler, *Classification without labels: Learning from mixed samples in high energy physics*, J. High Energy Phys. **10**, 174 (2017), doi:10.1007/JHEP10(2017)174, arXiv:1708.02949.

[52] P. T. Komiske, E. M. Metodiev and J. Thaler, *Energy Flow Networks: Deep Sets for Particle Jets*, J. High Energy Phys. **01**, 121 (2019), doi:10.1007/JHEP01(2019)121, arXiv:1810.05165.

[53] P. Komiske, E. Metodiev and J. Thaler, *Pythia8 Quark and Gluon Jets for Energy Flow*, Zenodo (2019), doi:10.5281/zenodo.3164691.

[54] A. Pathak, P. Komiske, E. Metodiev and M. Schwartz, *Herwig7.1 Quark and Gluon Jets*, Zenodo (2019), doi:10.5281/zenodo.3066475.

[55] T. Sjöstrand, S. Mrenna and P. Z. Skands, *PYTHIA 6.4 Physics and Manual*, J. High Energy Phys. **05**, 026 (2006), doi:10.1088/1126-6708/2006/05/026, arXiv:hep-ph/0603175.

[56] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen and P. Z. Skands, *An Introduction to PYTHIA 8.2*, Comput. Phys. Commun. **191**, 159 (2015), doi:10.1016/j.cpc.2015.01.024, arXiv:1410.3012.

[57] J. Bellm, S. Gieseke, D. Grellscheid, P. Kirchgaeßer, F. Loshaj, G. Nail, A. Papaefstathiou, S. Plätzer, R. Podskubka, M. Rauch, C. Reuschle, P. Richardson *et al.*, *Herwig 7.1 Release Note*, arXiv:1705.06919.

[58] M. Cacciari, G. P. Salam and G. Soyez, *The anti-k_t jet clustering algorithm*, J. High Energy Phys. **04**, 063 (2008), doi:10.1088/1126-6708/2008/04/063, arXiv:0802.1189.

[59] M. Cacciari and G. P. Salam, *Dispelling the N^3 myth for the k_t jet-finder*, Phys. Lett. B **641**, 57 (2006), doi:10.1016/j.physletb.2006.08.037, arXiv:hep-ph/0512210.

[60] M. Cacciari, G. P. Salam and G. Soyez, *FastJet user manual*, Eur. Phys. J. C **72**, 1896 (2012), doi:10.1140/epjc/s10052-012-1896-2, arXiv:1111.6097.

[61] E. Bothmann, G. S. Chahal, S. Höche, J. Krause, F. Krauss, S. Kuttimalai, S. Liebschner, D. Napoletano, M. Schönherr, H. Schulz, S. Schumann and F. Siegert, *Event Generation with Sherpa 2.2*, SciPost Phys. **7**, 034 (2019), doi:10.21468/SciPostPhys.7.3.034, arXiv:1905.09127.

[62] E. Rodrigues, B. Krikler, C. Burr, D. Smirnov, H. Dembinski, H. Schreiner, J. Nandi, J. Pivarski, M. Feickert, M. Marinangeli, N. Smith and P. Das, *The Scikit HEP Project – overview and prospects*, EPJ Web Conf. **245**, 06028 (2020), doi:10.1051/epjconf/202024506028, arXiv:2007.03577.

[63] H. Dembinski, E. Rodrigues and A. Fedynitch, *scikit-hep/pyhepmc: v2.9.0*, Zenodo (2022), doi:10.5281/zenodo.7334812.

[64] A. Buckley, P. Ilten, D. Konstantinov, L. Lönnblad, J. Monk, W. Pokorski, T. Przedzinski and A. Verbytskyi, *The HepMC3 event record library for Monte Carlo event generators*, Comput. Phys. Commun. **260**, 107310 (2021), doi:10.1016/j.cpc.2020.107310, arXiv:1912.08005.

[65] N. Dawe, E. Rodrigues, H. Schreiner, B. Ostdiek, D. Kalinkin, J. Nielsen, R. Marcel, S. Meehan, A. Roy and Domen13, *scikit-hep/pyjet: Legacy version 1.9.0*, Zenodo (2022), doi:10.5281/zenodo.6787002.

[66] J. Gallicchio, J. Huth, M. Kagan, M. D. Schwartz, K. Black and B. Tweedie, *Multivariate discrimination and the Higgs + W/Z search*, J. High Energy Phys. **04**, 069 (2011), doi:10.1007/JHEP04(2011)069, arXiv:1010.3698.

[67] J. Gallicchio and M. D. Schwartz, *Quark and Gluon Tagging at the LHC*, Phys. Rev. Lett. **107**, 172001 (2011), doi:10.1103/PhysRevLett.107.172001, arXiv:1106.3076.

[68] A. J. Larkoski, G. P. Salam and J. Thaler, *Energy Correlation Functions for Jet Substructure*, J. High Energy Phys. **06**, 108 (2013), doi:10.1007/JHEP06(2013)108, arXiv:1305.0007.

[69] C. Frye, A. J. Larkoski, J. Thaler and K. Zhou, *Casimir Meets Poisson: Improved Quark/Gluon Discrimination with Counting Observables*, J. High Energy Phys. **09**, 083 (2017), doi:10.1007/JHEP09(2017)083, arXiv:1704.06266.

[70] D. J. C. MacKay, *Probable networks and plausible predictions — a review of practical Bayesian methods for supervised neural networks*, Netw.: Comput. Neural Syst. **6**, 469 (1995), doi:10.1088/0954-898X_6_3_011.

[71] R. M. Neal, *Bayesian Learning for Neural Networks*, Springer Lecture Notes in Statistics **118** (1996), doi:10.1007/978-1-4612-0745-0.

[72] Y. Gal, *Uncertainty in Deep Learning*, Ph.D. thesis, University of Cambridge (2016), https://www.cs.ox.ac.uk/people/yarin.gal/website/thesis/thesis.pdf.

[73] D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv:1412.6980.

[74] I. Loshchilov and F. Hutter, *Decoupled Weight Decay Regularization*, arXiv:1711.05101.

[75] G. Kasieczka, M. Luchmann, F. Otterpohl and T. Plehn, *Per-Object Systematics using Deep-Learned Calibration*, SciPost Phys. **9**, 089 (2020), doi:10.21468/SciPostPhys.9.6.089, arXiv:2003.11099.

[76] S. Badger, A. Butter, M. Luchmann, S. Pitz and T. Plehn, *Loop Amplitudes from Precision Networks*, arXiv:2206.14831.

[77] M. Bellagente, M. Haußmann, M. Luchmann and T. Plehn, *Understanding Event-Generation Networks via Uncertainties*, SciPost Phys. **13**, 003 (2022), doi:10.21468/SciPostPhys.13.1.003, arXiv:2104.04543.

[78] A. Butter, T. Heimel, T. Martini, S. Peitzsch and T. Plehn, *Two Invertible Networks for the Matrix Element Method*, arXiv:2210.00019.