Two-point energy correlation spectra analysis for top tagging and beyond

Sung Hak Lim
Theory Center, KEK

Particle Physics in Computing Frontier

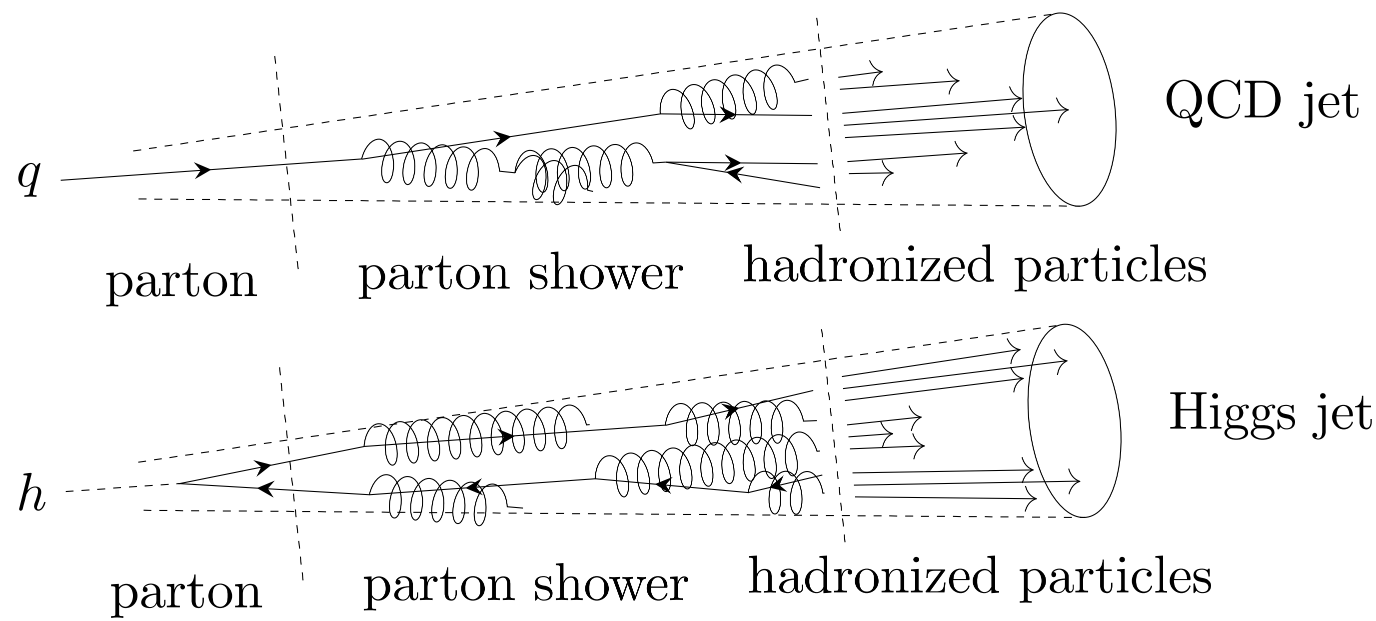
CTPU, IBS, Daejeon, Korea

Dec. 2019

__S. H. Lim__, M. M. Nojiri, arXiv:1807.03312, JHEP10(2018)181.
A. Chakraborty, __S. H. Lim__, M. M. Nojiri, arXiv:1904.02092, JHEP07(2019)135.
A. Chakraborty, __S. H. Lim__, M. M. Nojiri, M. Takeuchi, will appear in arXiv soon

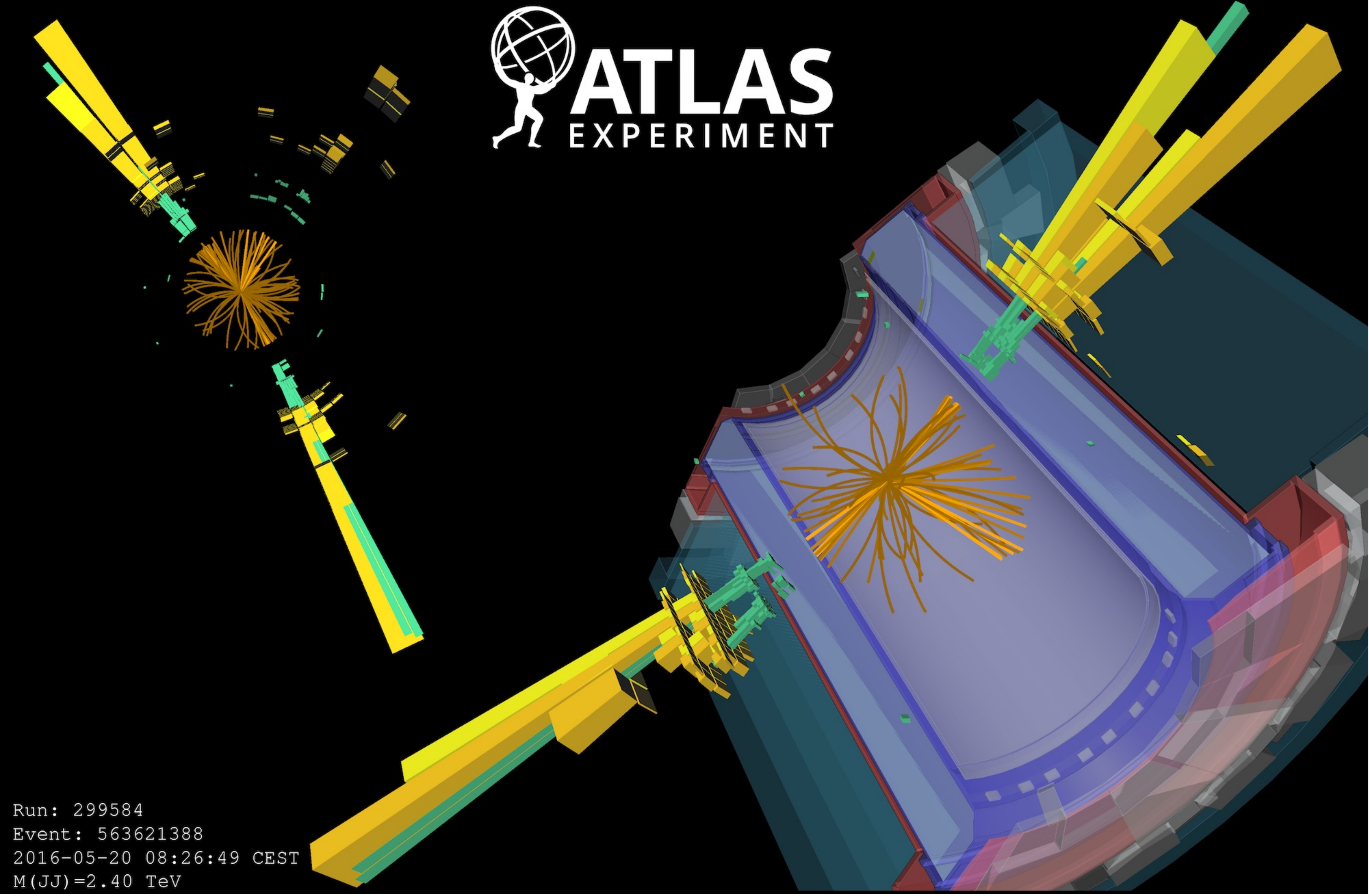
Boosted Jets:
Jets have substructure!

- As LHC stacking up multi TeV center-of-mass energy events, boosted heavy particles is produced and forms a single collimated cluster of particles similar to the QCD jets. ($m_{\rm EW}/\sqrt{\hat{s}} \approx \mathcal{O}[0.1]$)



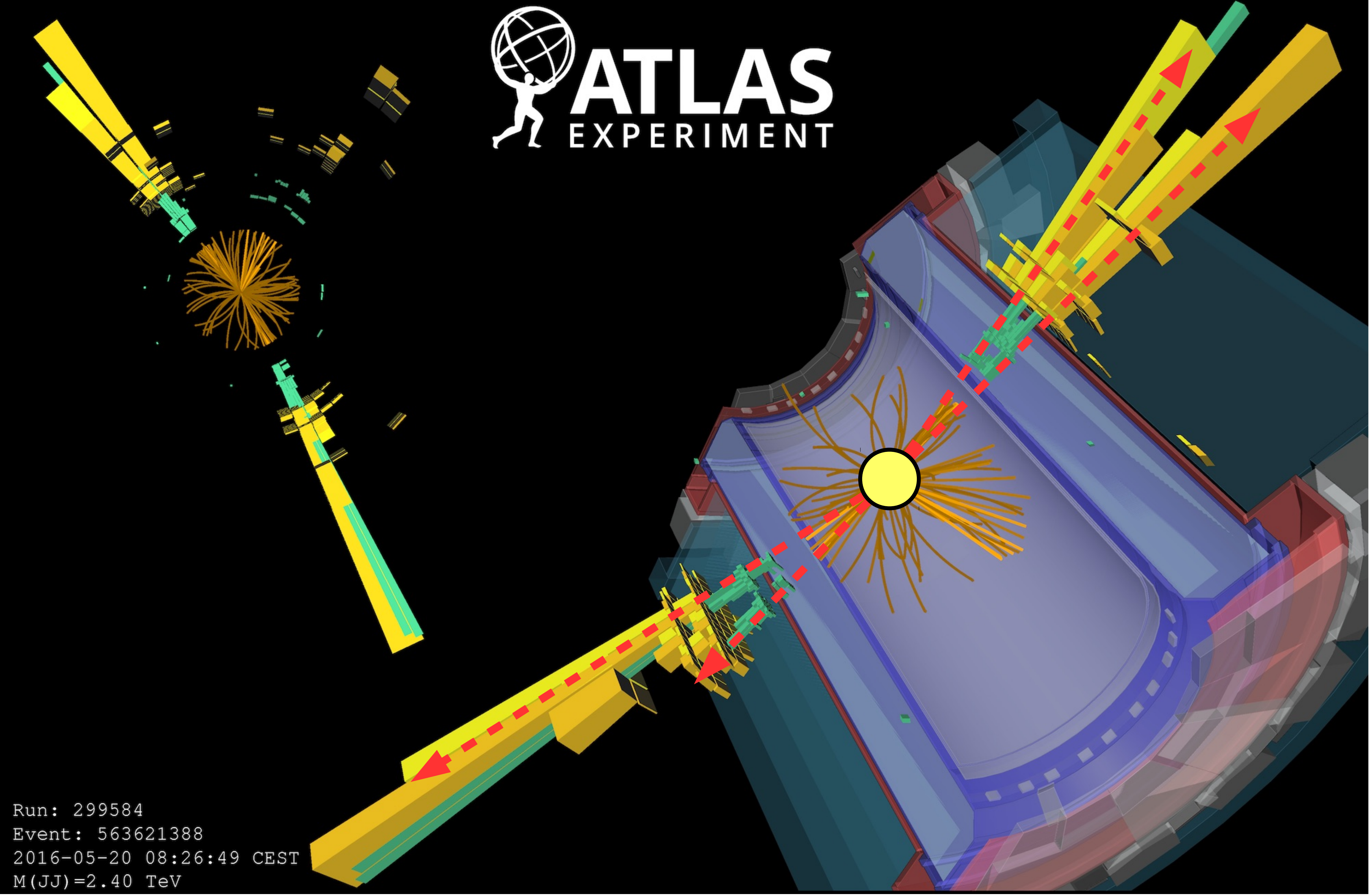
- We will see more and more these boosted jets!

LHC	HE-LHC	FCC
14 TeV →	27 TeV →	100 TeV



Two boosted jets from the old 2 TeV resonance searches…

https://atlas.cern/updates/physics-briefing/hunting-new-physics-boosted-bosons



We may require **deeper** understanding on these objects...

https://atlas.cern/updates/physics-briefing/hunting-new-physics-boosted-bosons

Deep Learning and Jet Physics

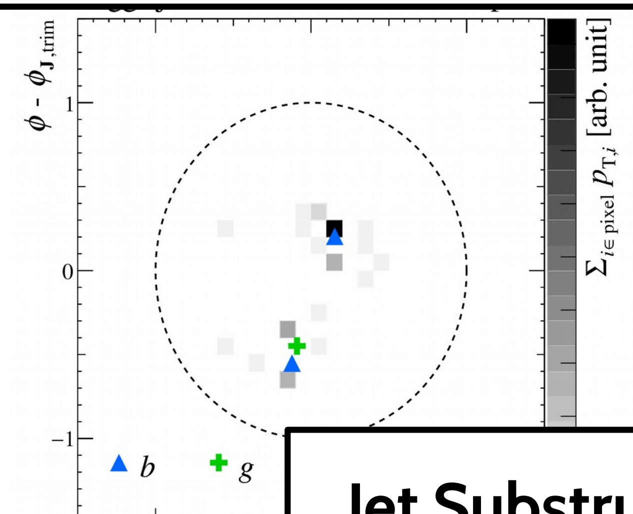
- We want a quick and reliable method for classifying those jets.
- Thanks to the development in physics and computer science…

Theoretical Understanding of QCD

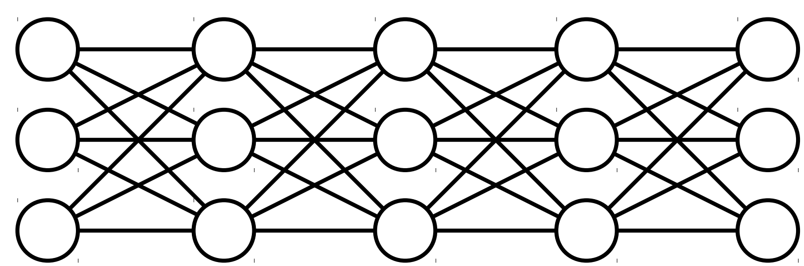
Development of Machine Learning

Huge Experimental Data

More Computational Power



Jet Substructure



Deep Learning

Classification Problem with Images

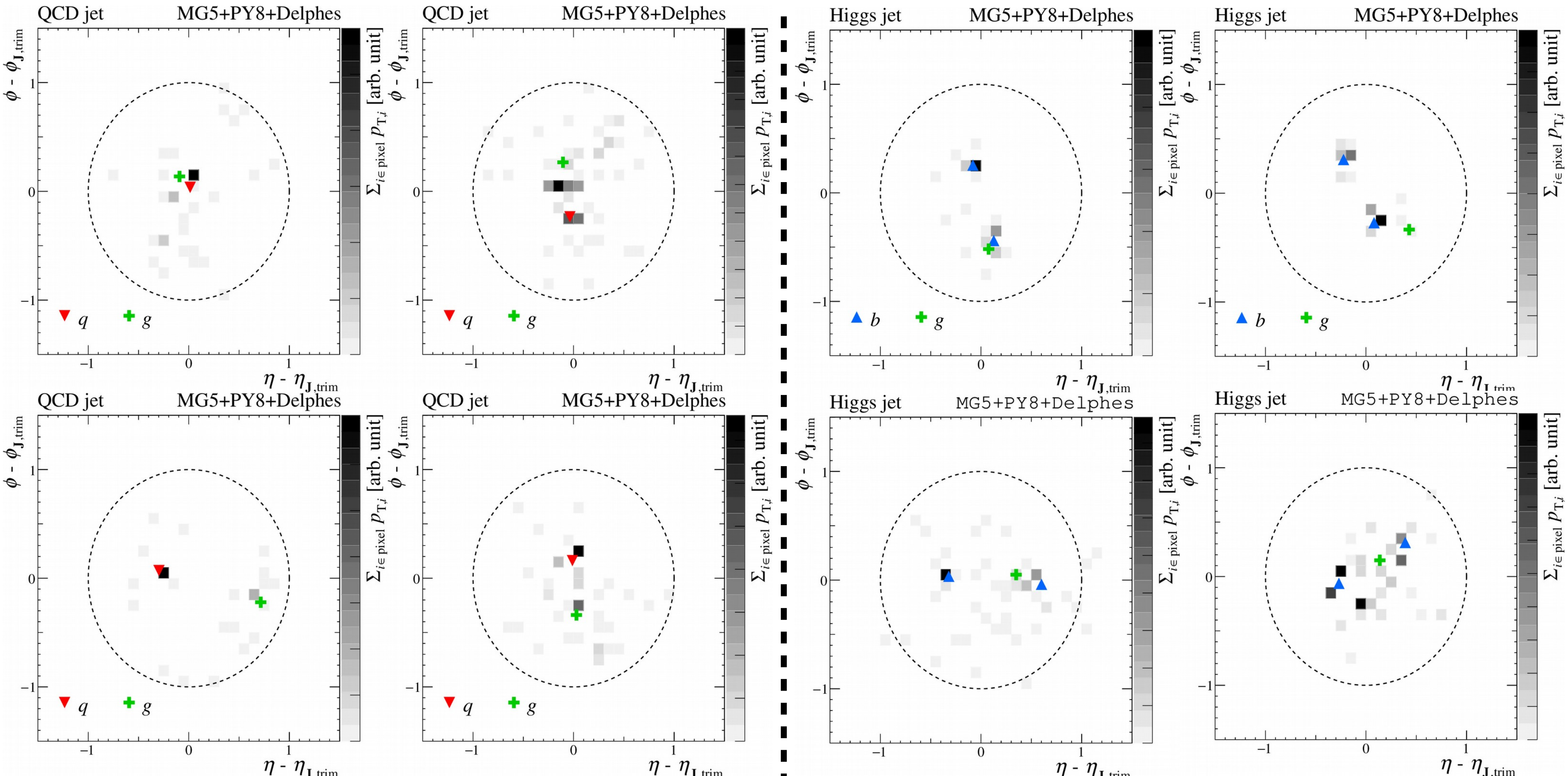
Can you distinguish cats and dogs from **pictures**?

Cats (?)

Dogs (?)

Classification Problem with Jet Images

Can you distinguish <u>QCD jets</u> and <u>Higgs jets</u> from **reconstructed particles**?

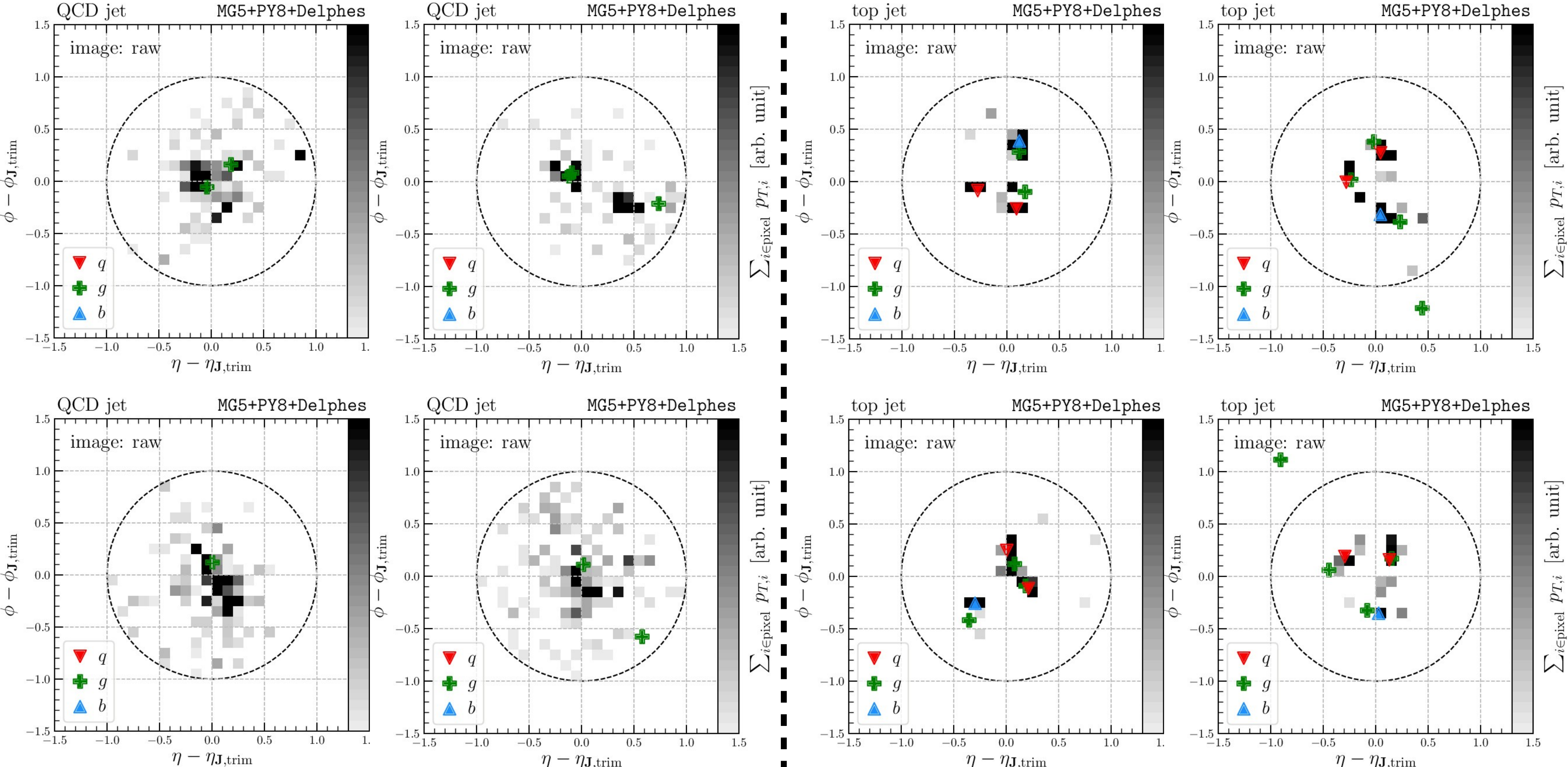


QCD jets
(quark jets from pp → Zj)

Higgs jets

Classification Problem with Jet Images

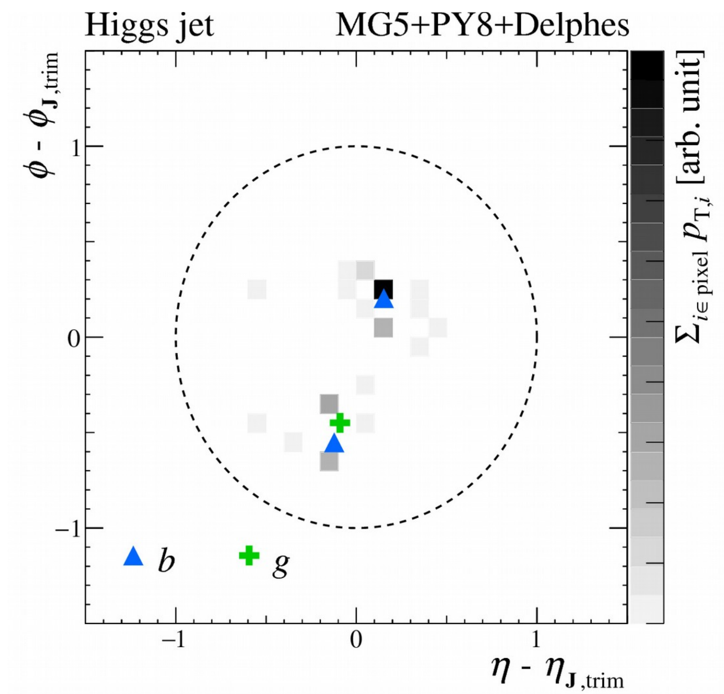
Can you distinguish QCD jets and Top jets from **reconstructed particles**?

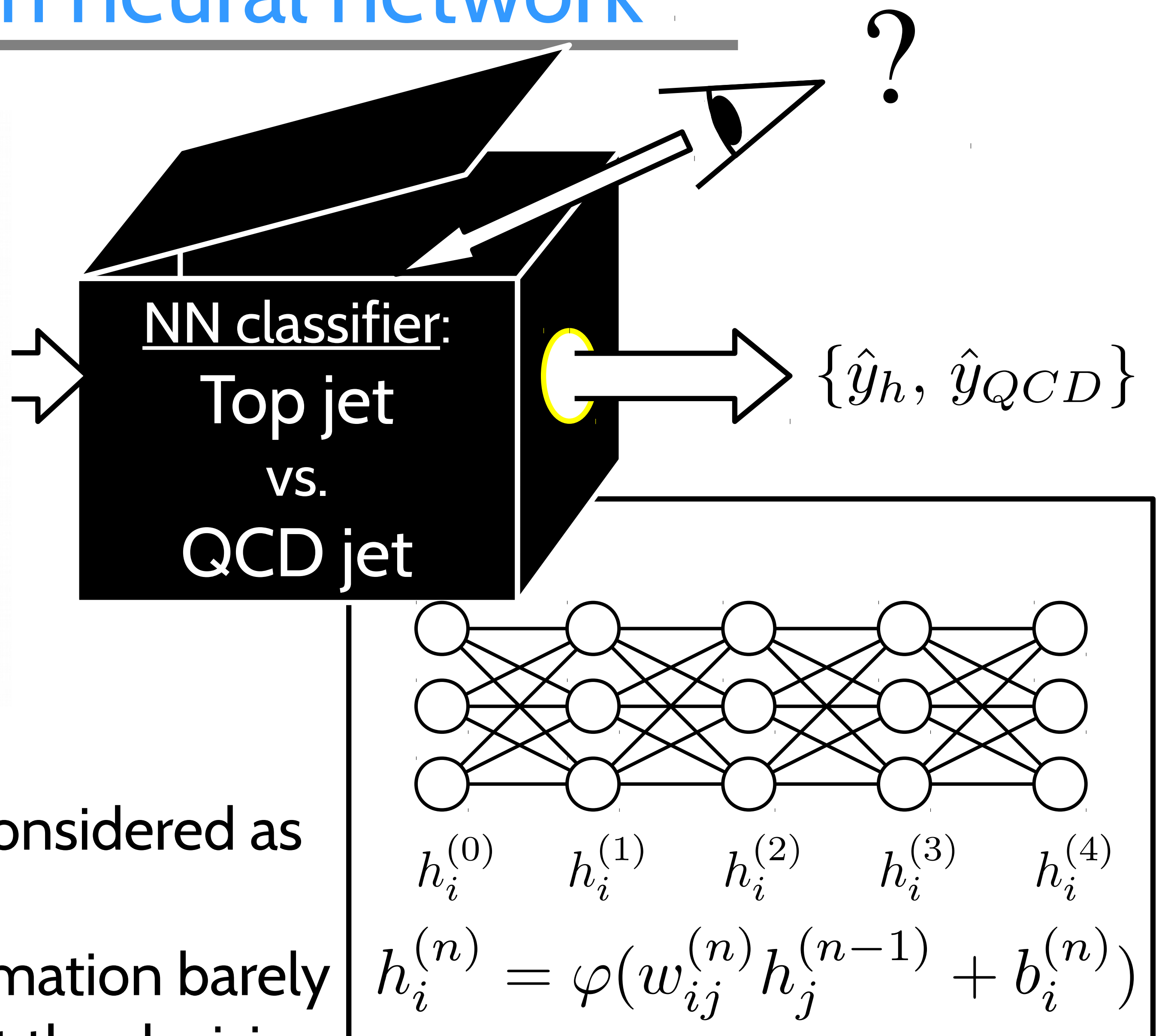


QCD jets
(gluon jets from pp → jj)

Top jets

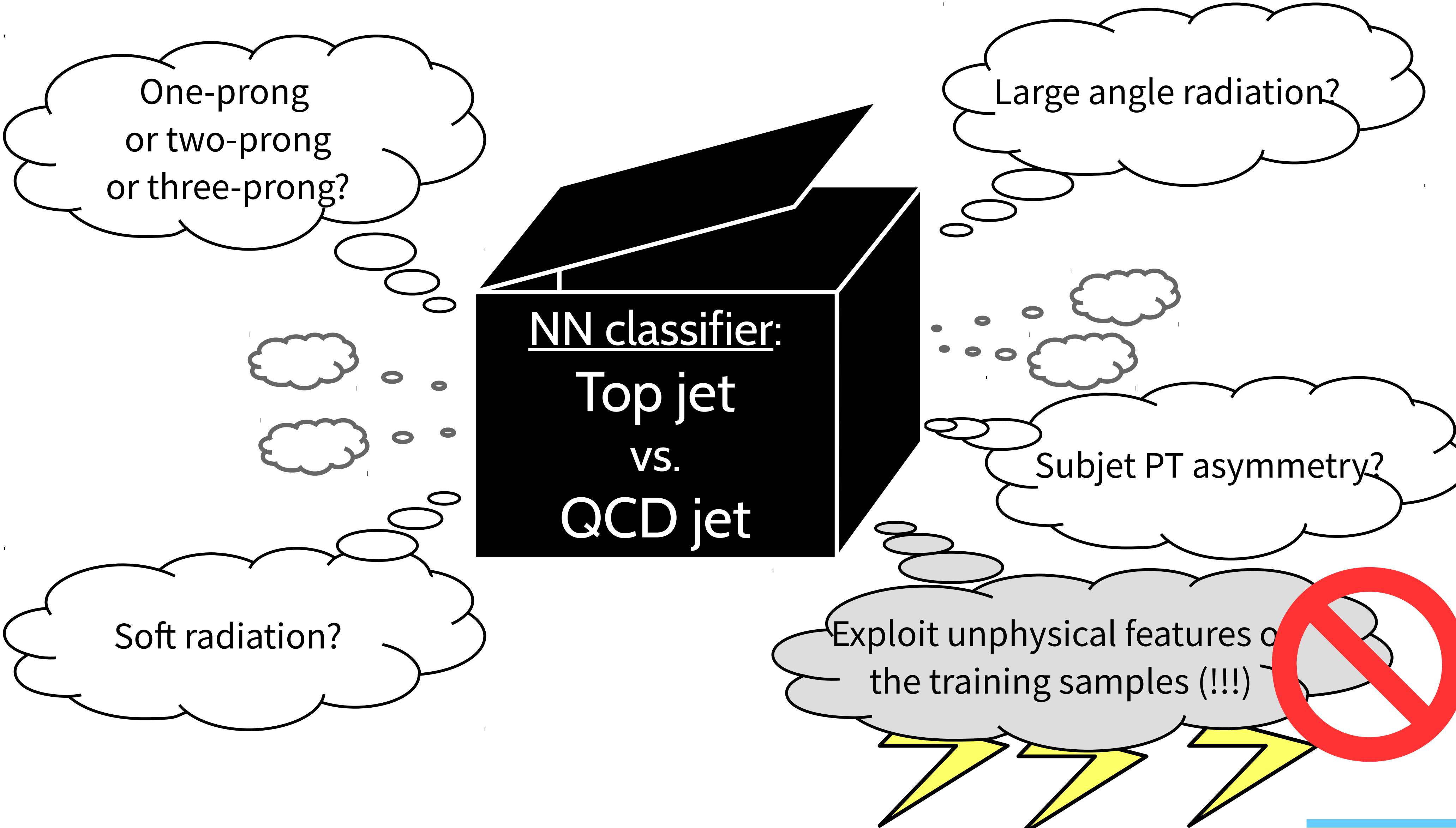
Difficulties on understanding the results from neural network





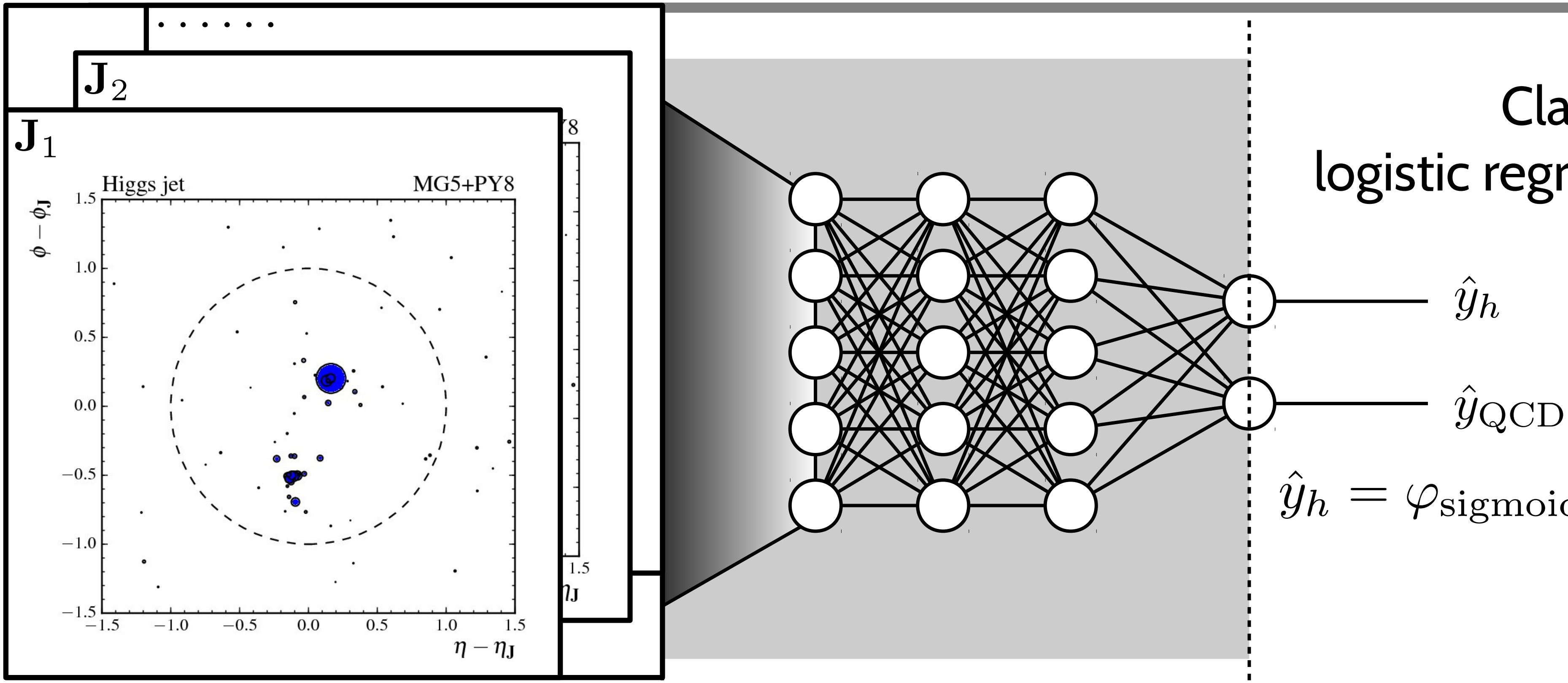
Neural network is often considered as a **black box** because studying its internal information barely gives you an insight about the decision...

Difficulties on understanding the results from neural network



We also want to know **the reasoning** behind the decision!

Basic Structure of a Neural Network Classifier



Classifier:
logistic regression

$$\hat{y}_h = \varphi_{\mathrm{sigmoid}}(\Psi[P_T])$$

$$P_{T,a}(\vec{R}) = \sum_{i \in \mathbf{J}_a} p_{T,i}\, \delta(\vec{R} - \vec{R}_i)$$

Inputs: energy flow

<u>Lots of inputs</u>

$$\Psi[P_T] = \sum_k w_k \Phi_k[P_T]$$

Feature map

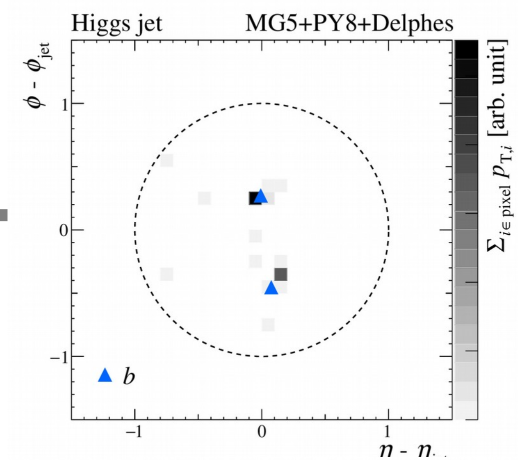
<u>Too many parameters</u>

We simplify inputs and the feature map to **<u>linearize</u>** this classifier for the interpretability.

The decision boundary is <u>highly nonlinear</u>.
It is hard to <u>interpret</u> the NN itself...

Our road map: Higgs jet

CNN with Jet Images

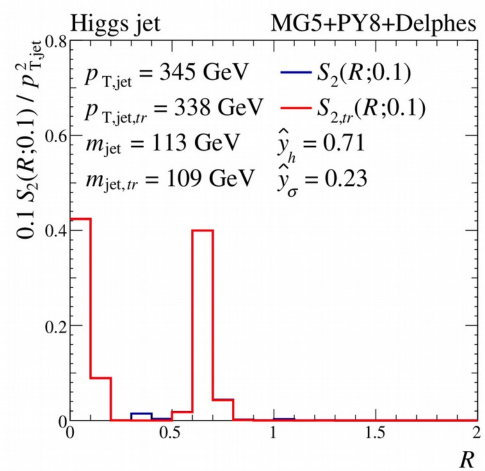


2D inputs

Physics-motivated IRC safe inputs

Model simplification

1D inputs



Linear model

Model complexity

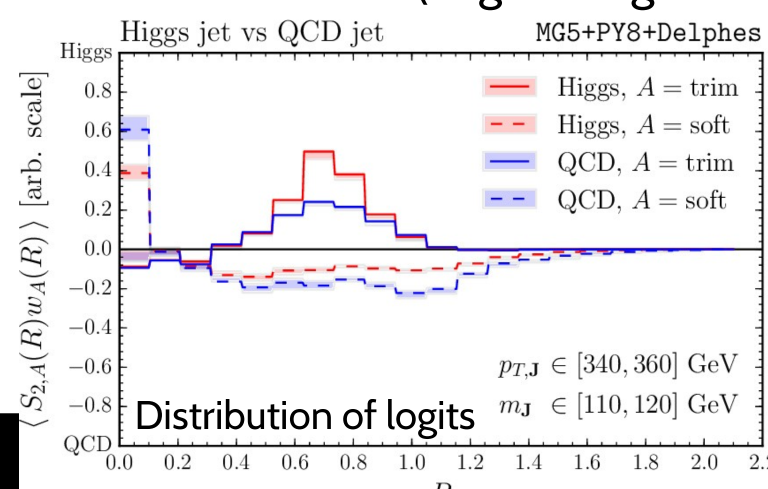
arXiv:1807.03312

MLP with two-point energy correlators

Deep neural net

Shallow neural net (Logistic regression)

Model interpretability



arXiv:1904.02092

Logistic Regression with two-point energy correlators

Lower bound

Our road map: Higgs jet

Model complexity

CNN with Jet Images

Model simplification

Physics-motivated
IRC safe inputs

arXiv:1807.03312

MLP with
two-point energy correlators

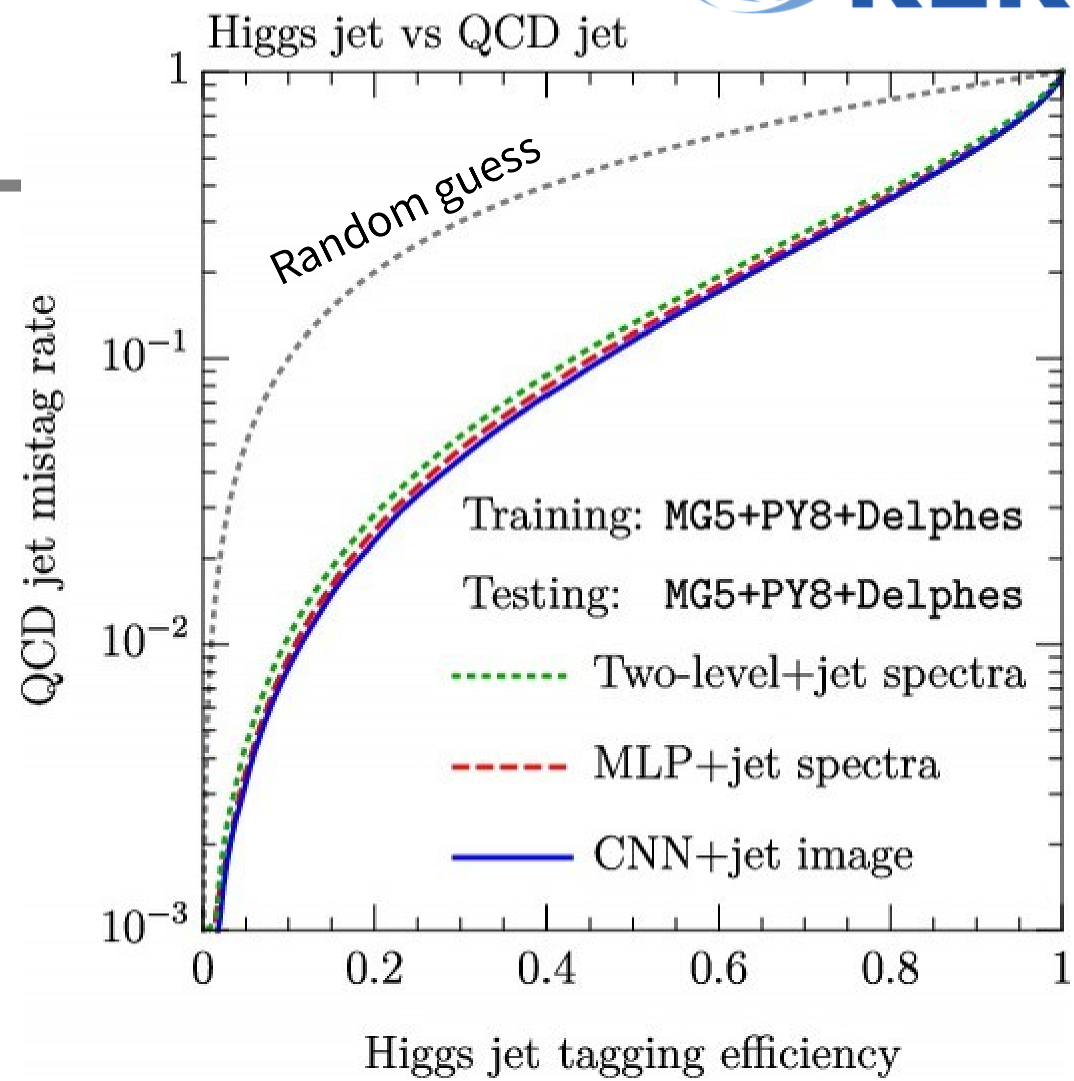
Model interpretability

Linear model

arXiv:1904.02092

Logistic Regression with
two-point energy correlators

Lower bound



Similar performance

Two-point energy correlation spectrum

See also **energy flow polyomials** arXiv:1712.07124

Let us consider the “functional Taylor expansion” of the logit.

$$\Phi[P_{T,a}] = w^{(0)} + \int d\vec{R}\; P_{T,a}(\vec{R}) w_a^{(1)}(\vec{R})$$

$$+ \boxed{\frac{1}{2!}\int d\vec{R}_1 d\vec{R}_2\; P_{T,a}(\vec{R}_1) P_{T,b}(\vec{R}_2) w_{ab}^{(2)}(\vec{R}_1, \vec{R}_2)} + \cdots$$

If we only use relative distance between constituents, the first nontrivial term is

$$w^{(0)} + p_{T,\mathbf{J}_a}\, w_a^{(1)}$$

$$\Phi[P_{T,a}] = \int dR\; S_{2,ab}(R) w_{ab}^{(2)}(R) + \cdots$$

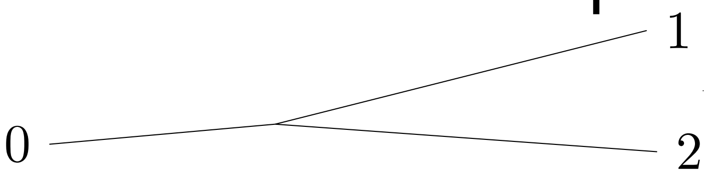
$$S_{2,ab}(R) = \int d\vec{R}_1\, d\vec{R}_2\; P_{T,a}(\vec{R}_1) P_{T,b}(\vec{R}_2) \delta(R - R_{12})$$

Two-point correlation between constituents at distance *R*

Reduce the dimension of inputs
[Length/bin width]^2 → [Length/bin width]

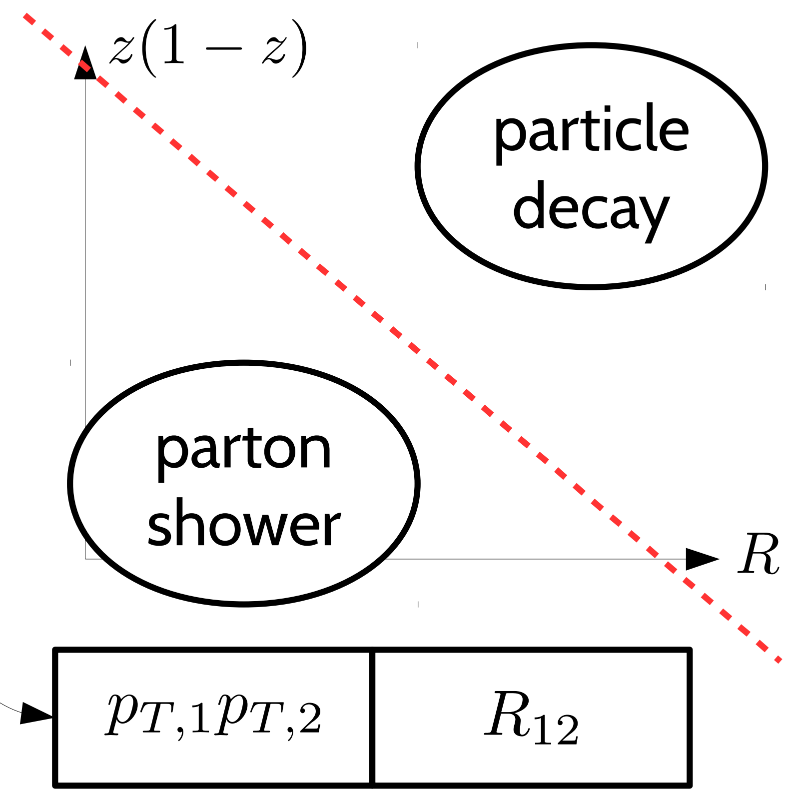
Kinematics inside Jet

The parameter set $(p_{T,0}, z, R)$ is a set of charateristic variables for the kinematics of parton evolution.

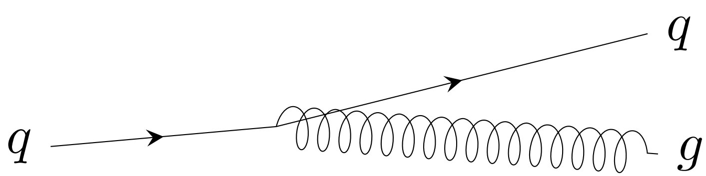


$$p_{T,1} = p_{T,0}z \qquad p_{T,2} = p_{T,0}(1-z)$$

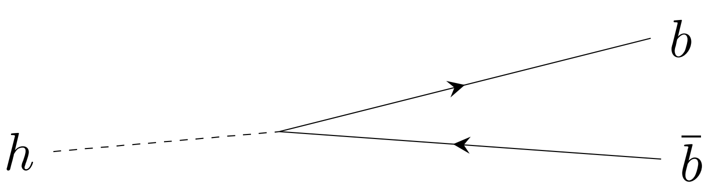
$$R^2 = (y_1 - y_2)^2 + (\phi_1 - \phi_2)^2$$



- parton shower



  - Collinear: small R
  - Soft: z ~ 0 or 1
- particle decay



  - Transverse decay:
    R ~ 2m/pt and z ~ 0.5

The second order term encodes parton shower and particle decay efficiently.

Two-Point Correlation Spectrum: Trimmed Spectrum

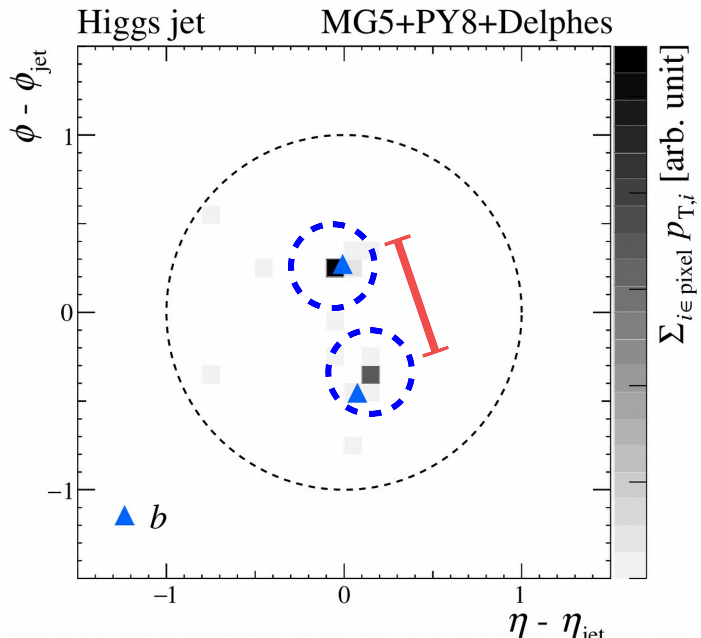
First, let us focus on correlation between hard constituents.
We may consider the two-point correlation spectrum of **trimmed jet**.

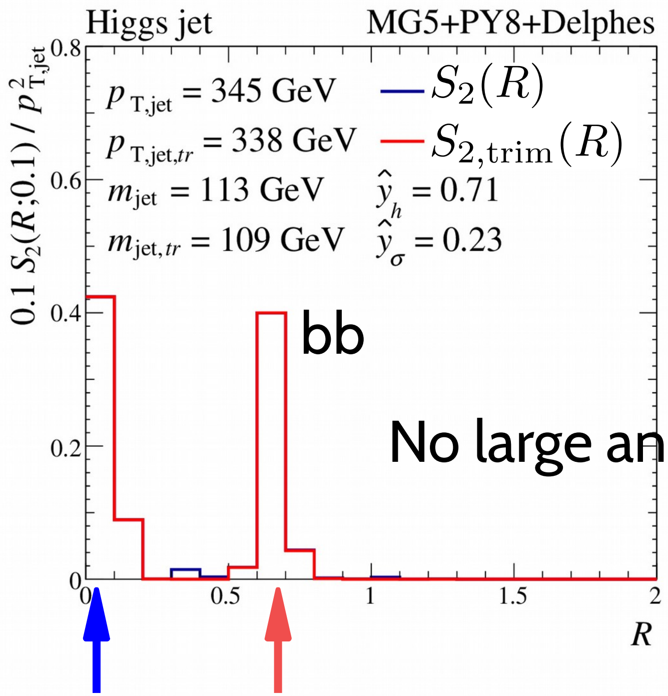
$$S_{2,\mathrm{trim}}(R) = \int d\vec{R}_1\, d\vec{R}_2\, P_{T,\mathbf{J}_{\mathrm{trim}}}(\vec{R}_1) P_{T,\mathbf{J}_{\mathrm{trim}}}(\vec{R}_2) \delta(R - R_{12})$$

only sensitive to **hard-hard correlations**

For Higgs jet:

$$S_{2,\mathrm{trim}}(R) = (p_{T,b}^2 + p_{T,\bar{b}}^2)\delta(R) + 2p_{T,b}p_{T,\bar{b}}\delta(R - R_{b\bar{b}})$$





Calorimeter jet, anti-kt, R=1.0

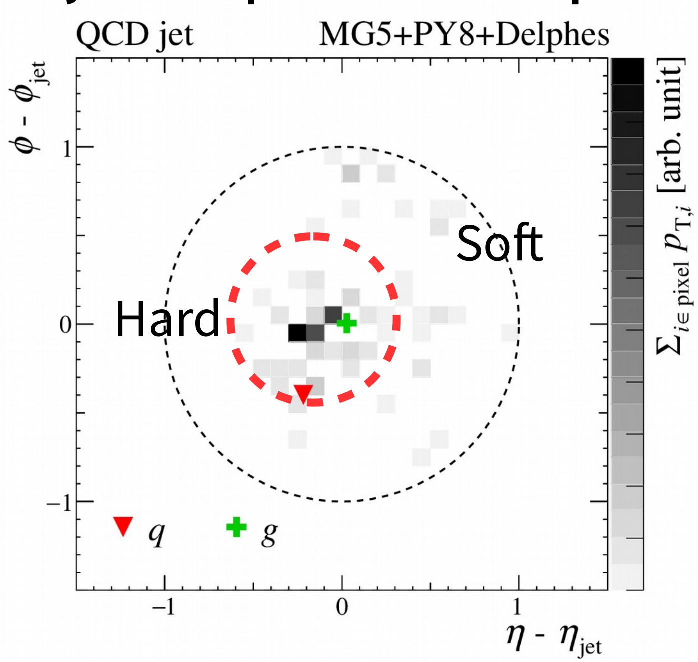
Two-Point Correlation Spectrum: Hard-Soft Correlation

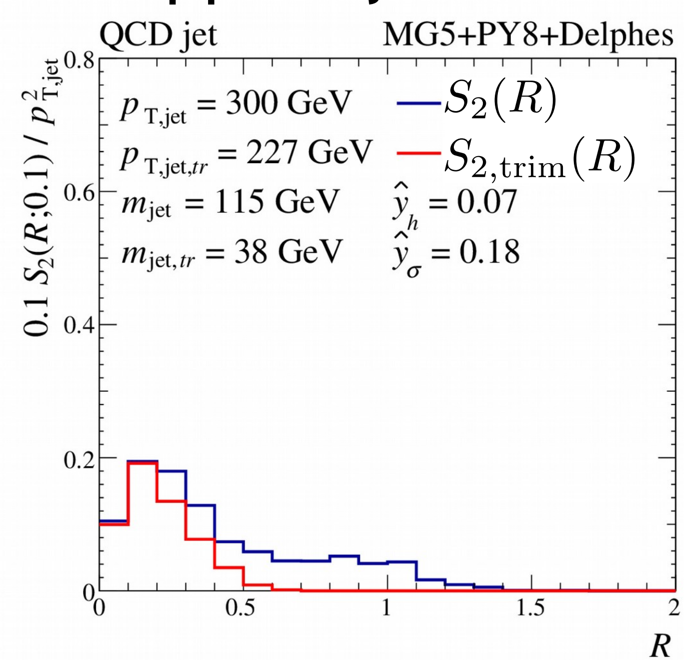
QCD jets have significant soft radiations. We may consider correlation between the soft parts and the hard parts.

$$S_{2,\mathrm{soft}}(R) = S_2(R) - S_{2,\mathrm{trim}}(R)$$

sensitive to hard-soft correlations
subleading soft-soft correlations

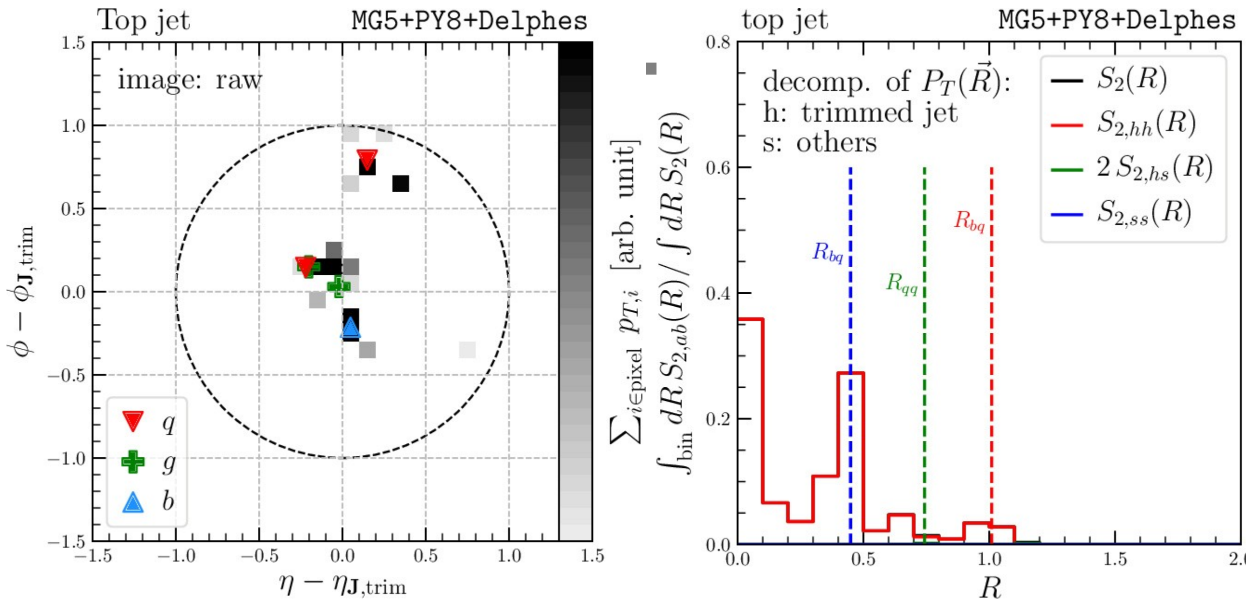
For QCD jet (in particular, quark jet from pp → Zj):



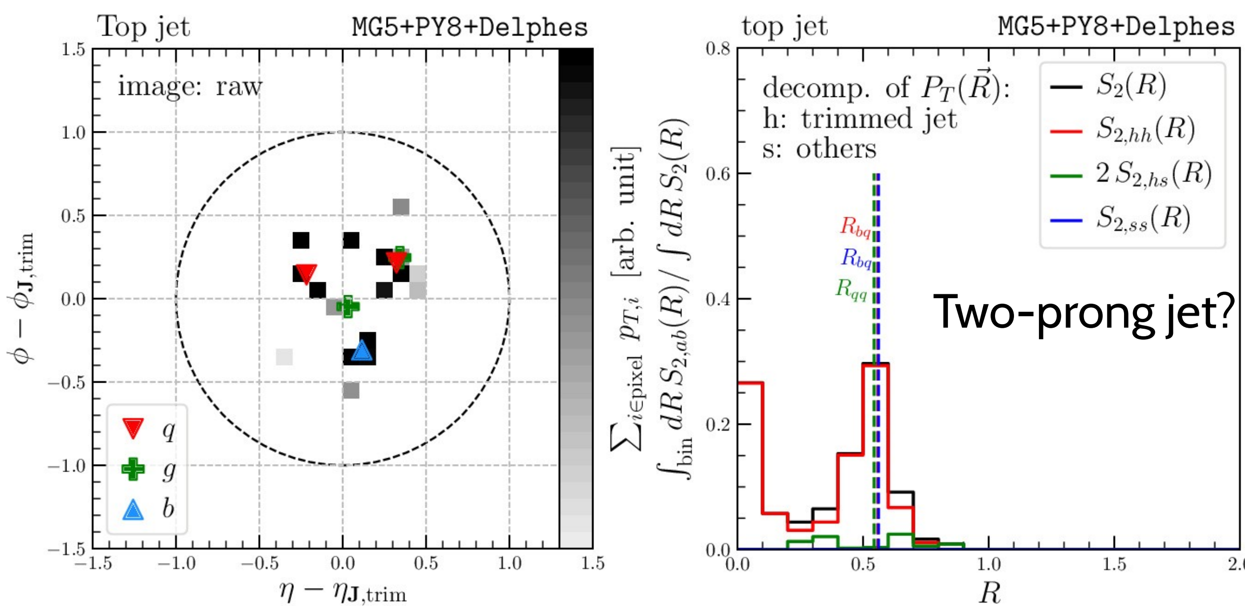


Calorimeter jet, anti-kt, R=1.0

Top Jets



Two point correlations are enough.

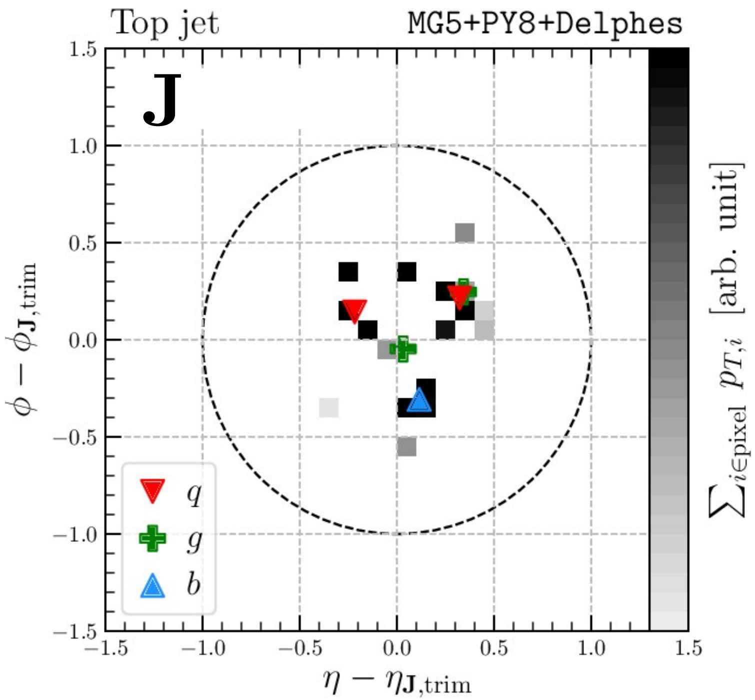


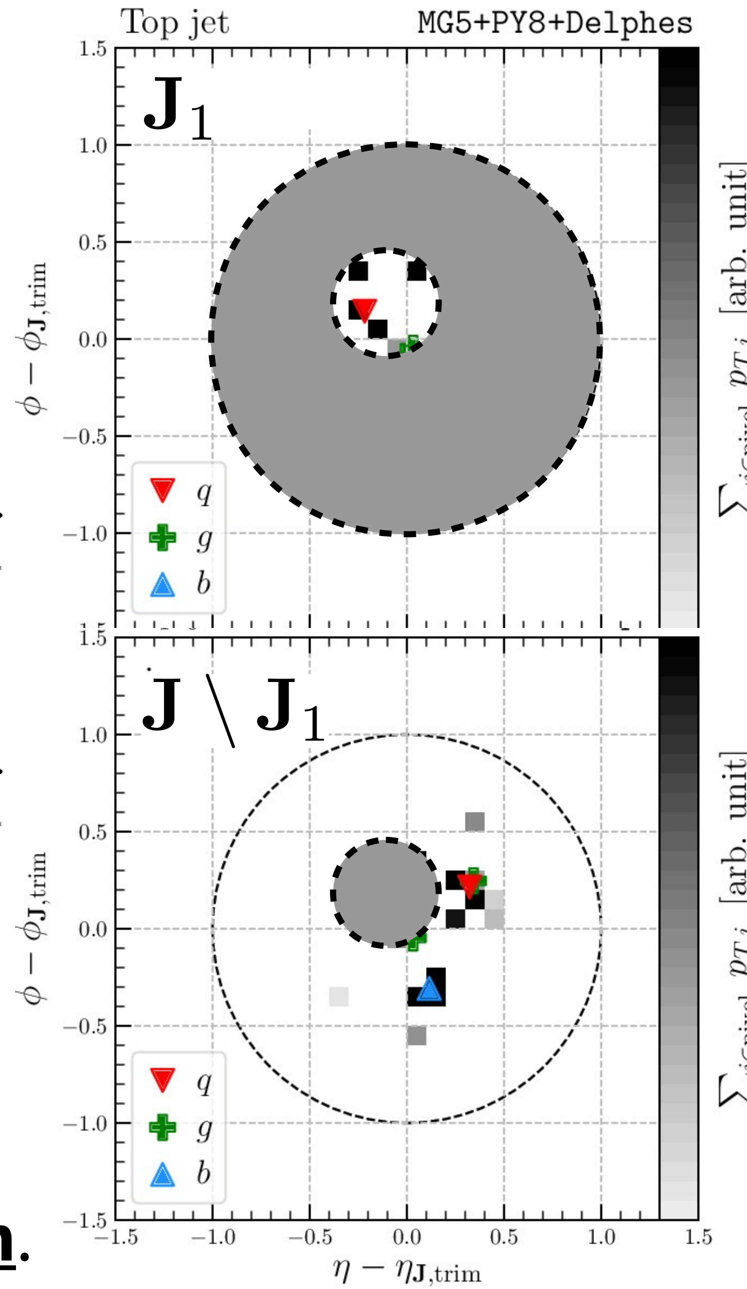
Need more information to resolve overlapping peaks...

Calorimeter jet, anti-kt, R=1.0

Decomposition of Problem

Preliminary





After this separation, the problem is decomposed to analysis of **one-prong** jet, **two-prong** jet, and their **cross-correlation**.

New set of spectra

Leading subjet autocorrelation:

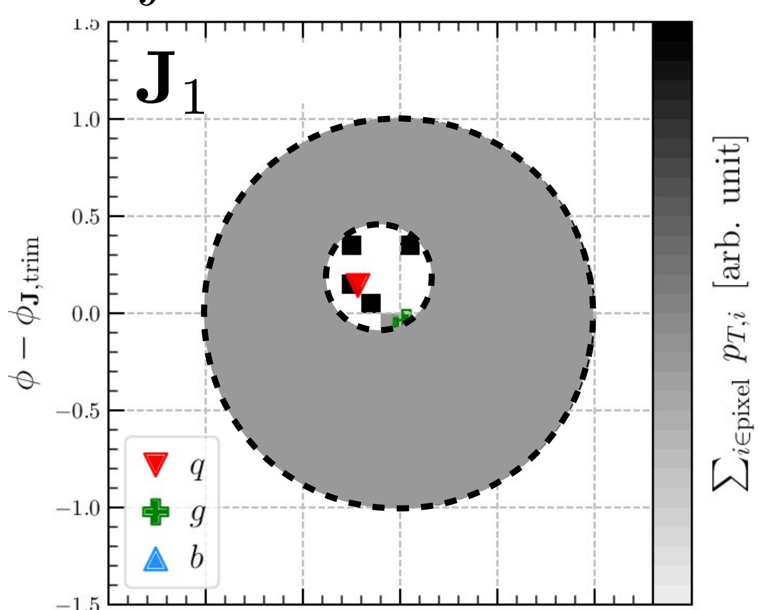
$$S_{2,11}(R) = \int d\vec{R}_1\, d\vec{R}_2\, P_{T,\mathbf{J}_1}(\vec{R}_1) P_{T,\mathbf{J}_1}(\vec{R}_2) \delta(R - R_{12})$$

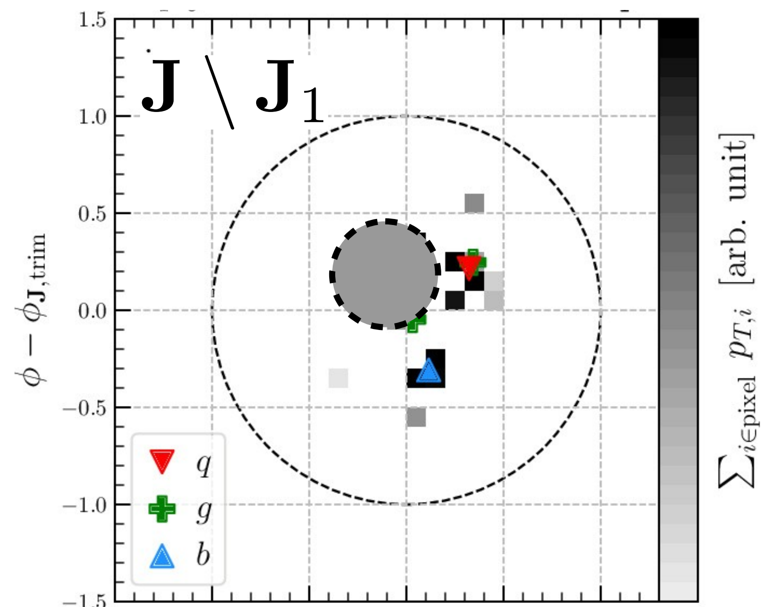
Complement autocorrelation:

$$S_{2,cc}(R) = \int d\vec{R}_1\, d\vec{R}_2\, P_{T,\mathbf{J}\setminus\mathbf{J}_1}(\vec{R}_1) P_{T,\mathbf{J}\setminus\mathbf{J}_1}(\vec{R}_2) \delta(R - R_{12})$$

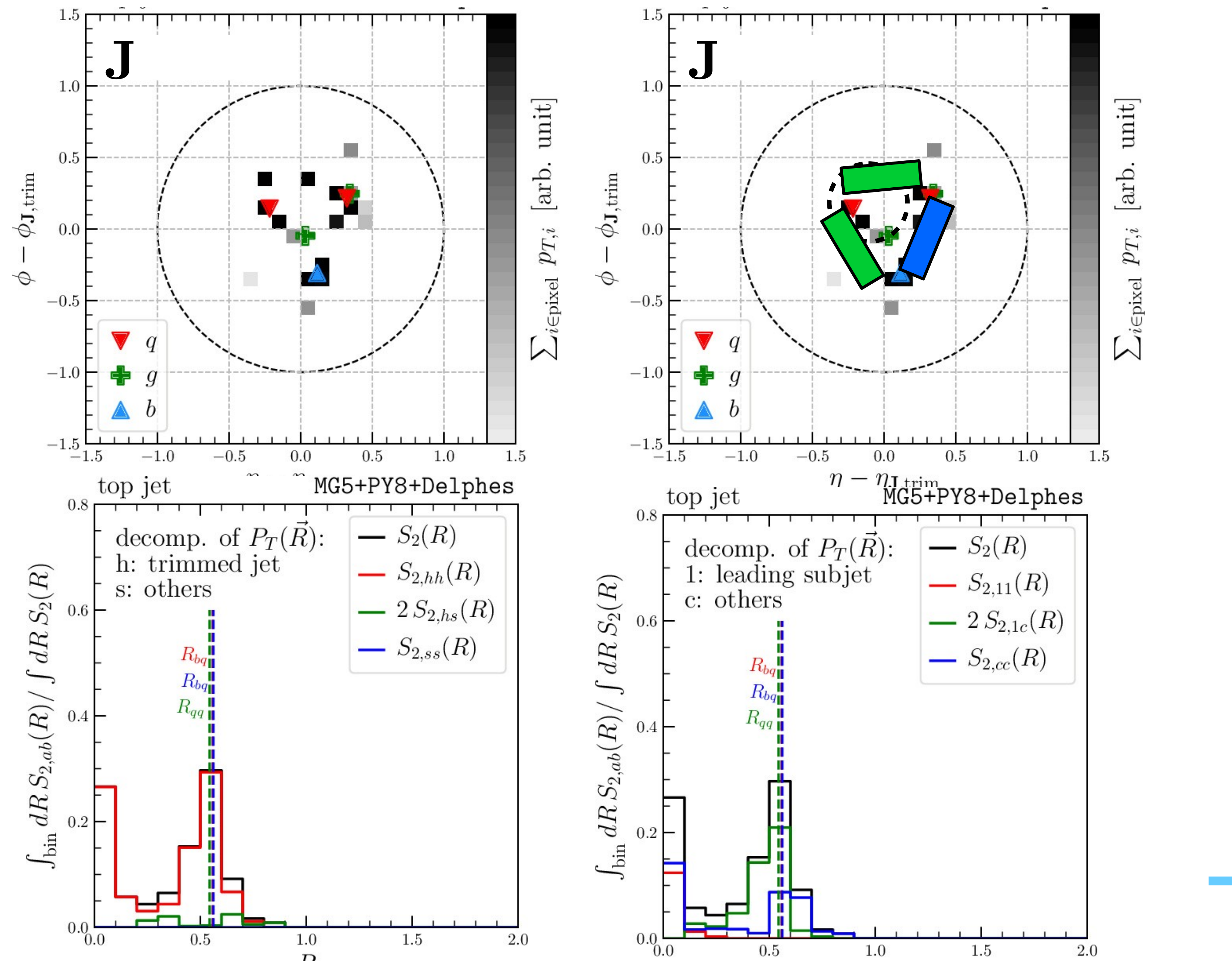
cross-correlation:

$$S_{2,1c}(R) = \int d\vec{R}_1\, d\vec{R}_2\, P_{T,\mathbf{J}_1}(\vec{R}_1) P_{T,\mathbf{J}\setminus\mathbf{J}_1}(\vec{R}_2) \delta(R - R_{12})$$





Two-Point Correlation Spectrum: Leading jet and its complementary



Relationship to other variables

See also
Yang-Ting Chien, et al.
1711.11041

The cross-correlation is similar to the **telescoping**:

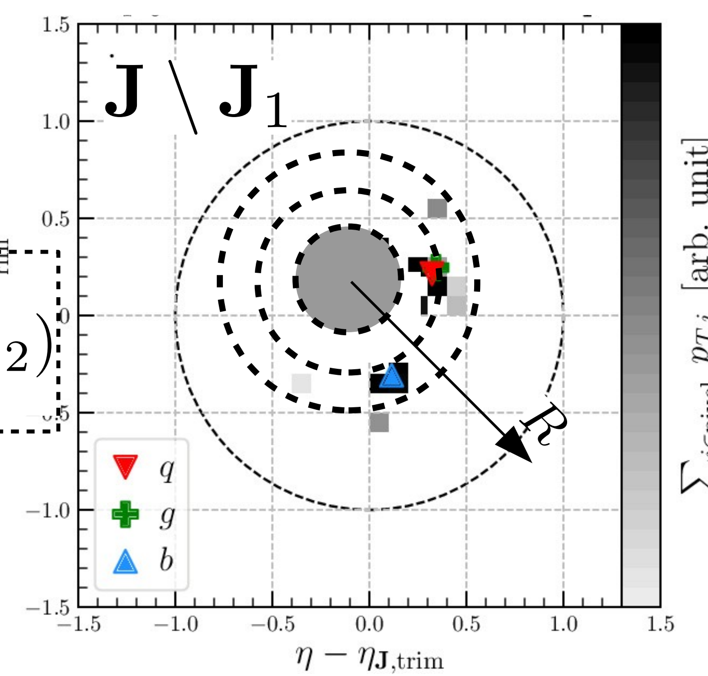
$$S_{2,1c}(R) = \int d\vec{R}_1\, d\vec{R}_2\; P_{T,\mathbf{J}_1}(\vec{R}_1) P_{T,\mathbf{J}\backslash\mathbf{J}_1}(\vec{R}_2) \delta(R - R_{12})$$

If we approximate the leading subjet energy flow to a delta function,

$$P_{T,\mathbf{J}_1}(\vec{R}) = \sum_{i\in\mathbf{J}_1} p_{T,i}\, \delta(\vec{R} - \vec{R}_i) \approx p_{T,\mathbf{J}_1}\, \delta(\vec{R} - \vec{R}_{\mathbf{J}_1})$$

The spectrum is a differential distribution of energy flow with respect to angular distance from the subjet axis of J_1

$$S_{2,1c}(R) = p_{T,\mathbf{J}_1} \boxed{\int d\vec{R}_2\; P_{T,\mathbf{J}\backslash\mathbf{J}_1}(\vec{R}_2)\, \delta(R - R_{\mathbf{J}_1 2})}$$



This will improve prong substructure identification.

A top tagger architecture with two-point energy correlation spectra

Hard and soft substructure analyzer

$S_{2,\mathrm{trim}}(R)$

$S_{2,\mathrm{soft}}(R)$

Leading subjet and its complmentary substructure analyzer

$S_{2,11}(R)$ $S_{2,cc}(R)$

$S_{2,1c}(R)$

kinematics

$p_{T,\mathbf{J}}$, $m_{\mathbf{J}}$

$p_{T,\mathbf{J}_{\mathrm{trim}}}$, $m_{\mathbf{J}_{\mathrm{trim}}}$

$p_{T,\mathbf{J}\setminus\mathbf{J}_1}$, $m_{\mathbf{J}\setminus\mathbf{J}_1}$

* some inputs might not be necessary.

MLP

MLP

MLP

$\hat{y}(x) = q(\mathrm{top}|x)$

Train the network output with the cross-entropy.

$$\mathcal{L} = \sum_{i=0}^{N_{\mathrm{event}}} w^{(i)}(x)\, y^{(i)}\ \log \hat{y}^{(i)}(x)$$

Training setup

- The model is implemented with Keras with backend tensorflow.
- Optimizer: ADAM, mimimize the weighted cross-entropy.

$$\mathcal{L} = \sum_{i=0}^{N_{\mathrm{event}}} w^{(i)}(x)\, y^{(i)}\ \log \hat{y}^{(i)}(x)$$

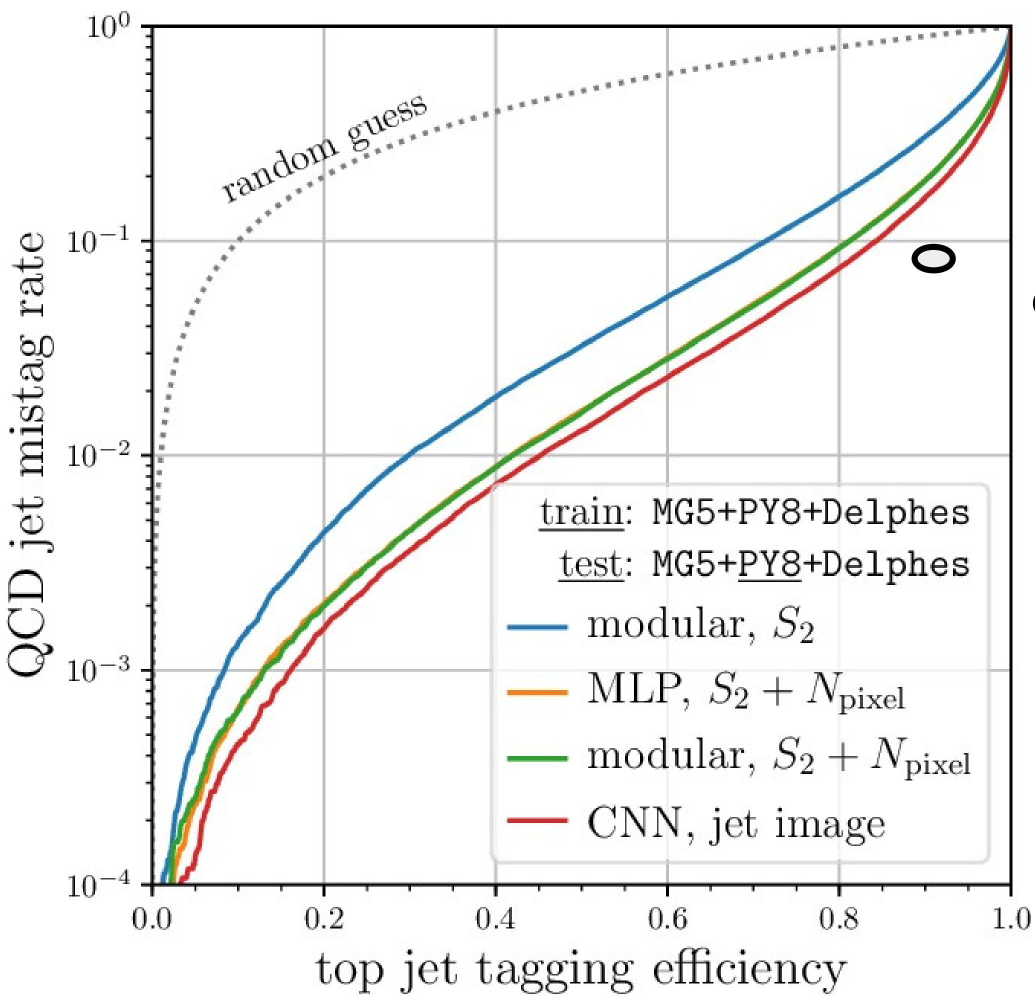
- $p_{T,J}$ distribution is reweighted to be flat.
  The marginal distribution is approximated by the kernel density estimation.

$$w(x) = \frac{1}{f_{p_{T,\mathbf{J}}}(p_{T,\mathbf{J}})}$$

- Weight initialization: He uniform
- L2 regularization: weight decay constant: 0.001
- Early stopping: patience = 50
- Use moving average of weights and bias for the validation and test.
  Ignore early t_0=50 epochs.

- Batch size: modular NN: 20, 50, 100, CNN: 100, 200, 500
- Tested two random seeds
- Select a network with the smallest validation AUC
- Validate the trained model with the model with a focal loss.

Performance comparison

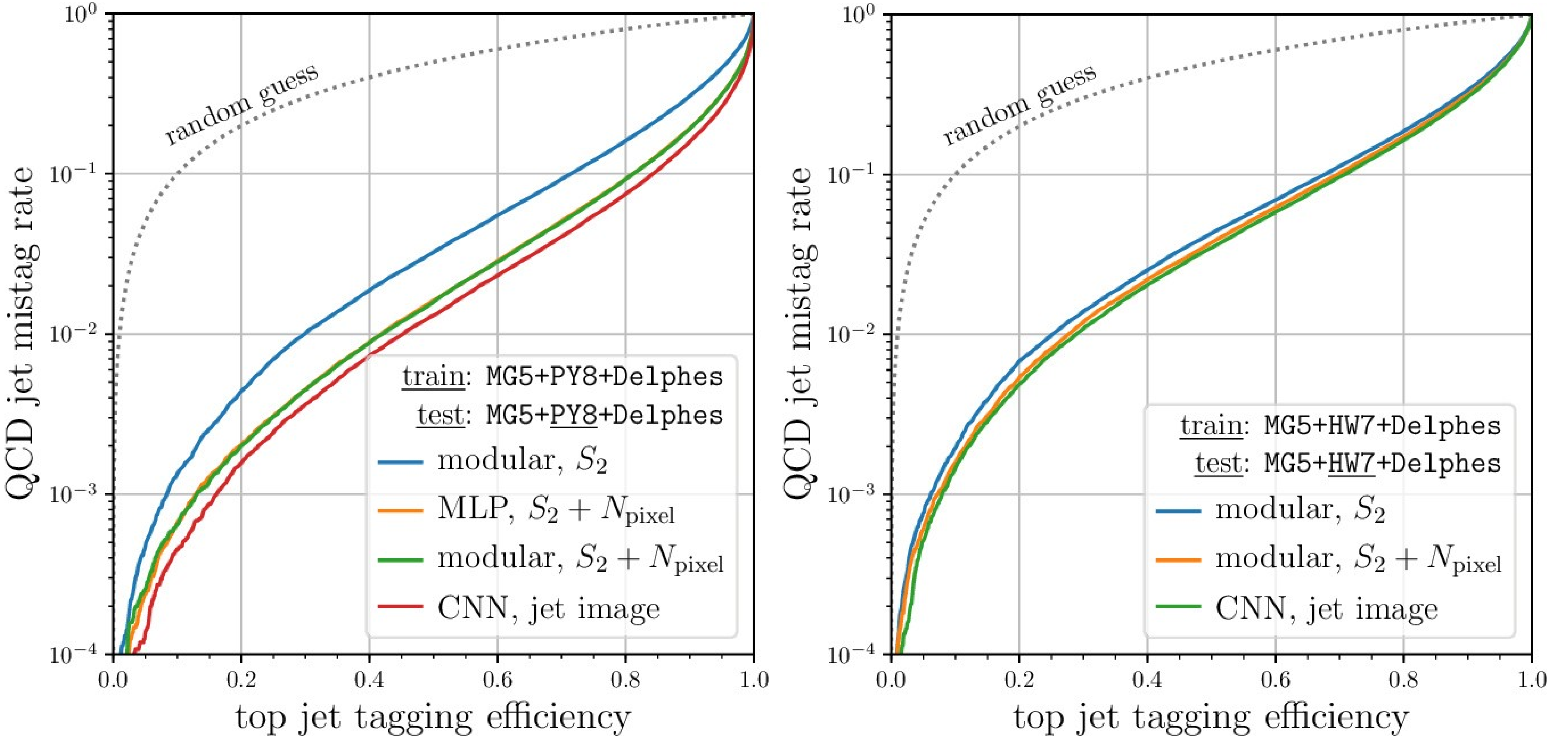


Is CNN better in performance?
- Yes, because we are only using IRC safe two-point energy correlators.

Gap is there.

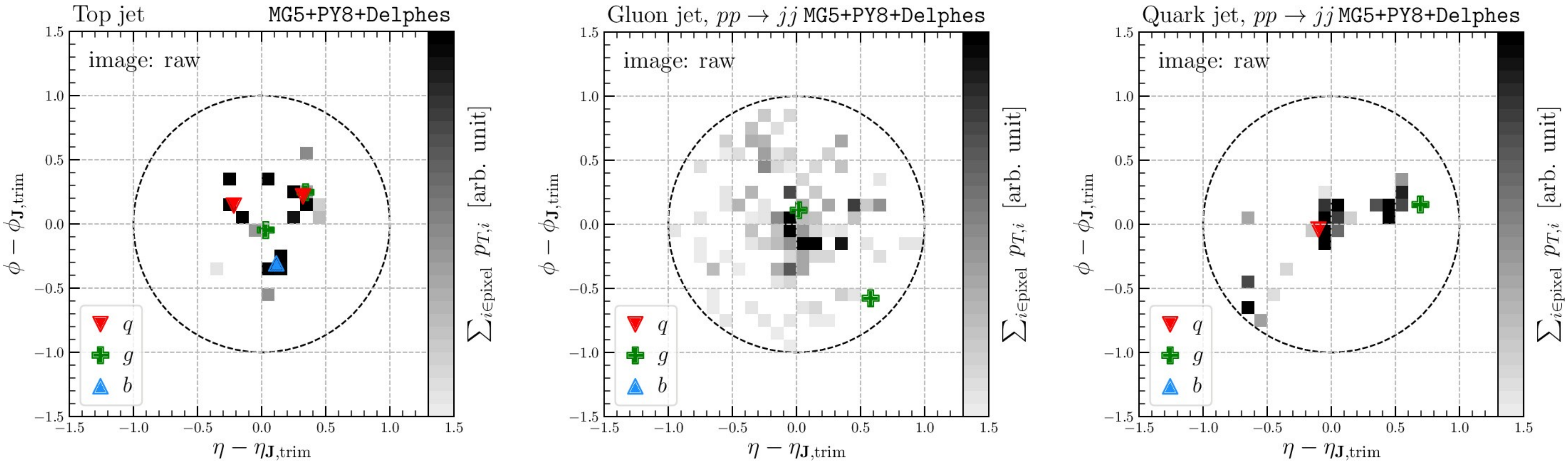
S2's are not enough, but adding one parameter N_{pixel} will fill the gap.

Pythia8 vs. Herwig7 cross-check



ROC of modular NN + S2 setup does not differ much between PY8 and HW7.
However, CNN is not.
There might be some difference between PY8 samples and HW7 samples, which only CNN can find. One of that is the N_{pixel}.

Number of pixels

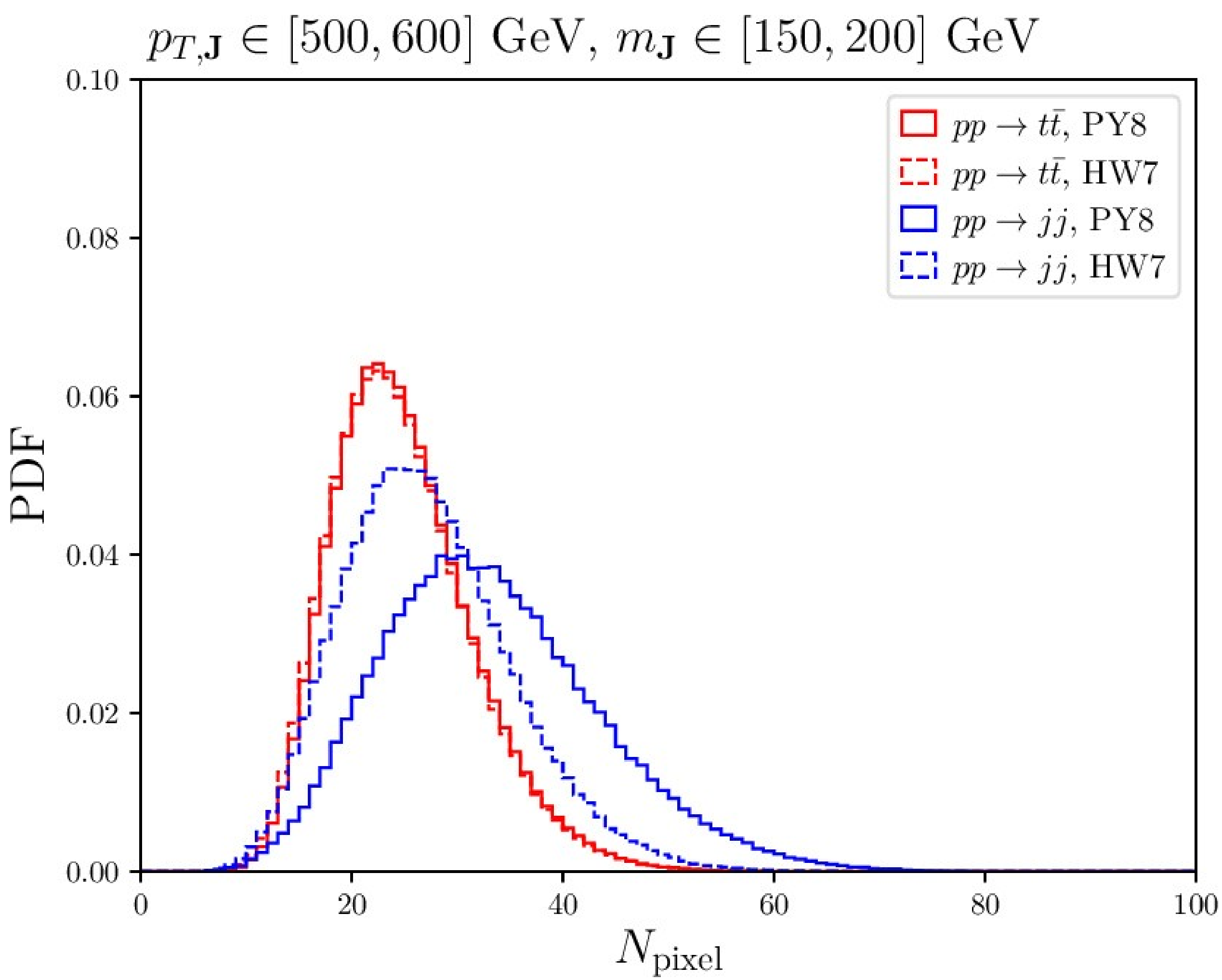


- Top: **color triplet**
- Gluon: **color octet**

It is well known that counting variables helps the classification.

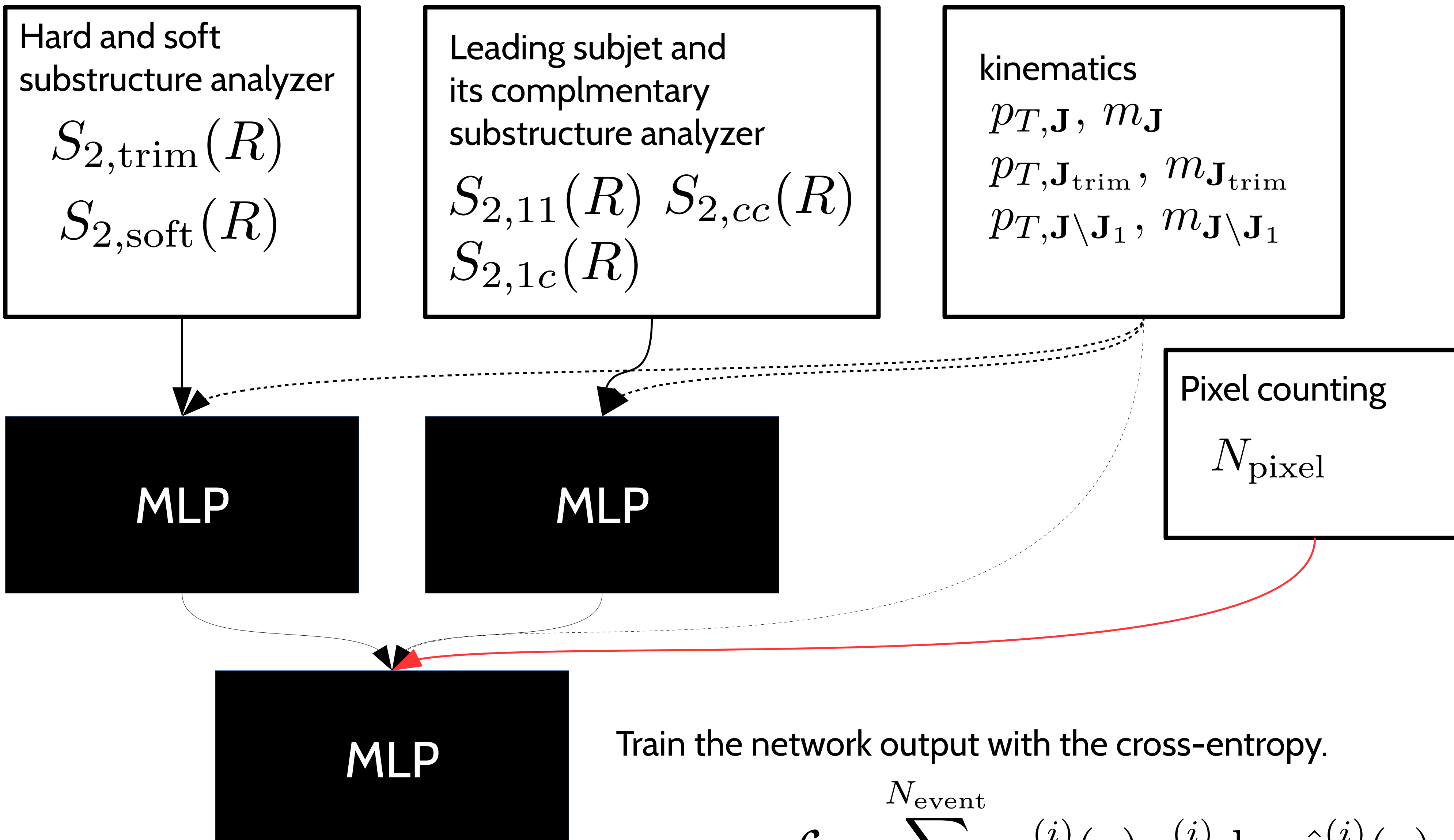
The number of pixels is calculable from the jet image.

N_{pixel} distribution: top jet and QCD jet



pp → jj samples are gluon jet rich, so that the deviation is large.

A top tagger architecture with two-point energy correlation spectra + N_{pixel}

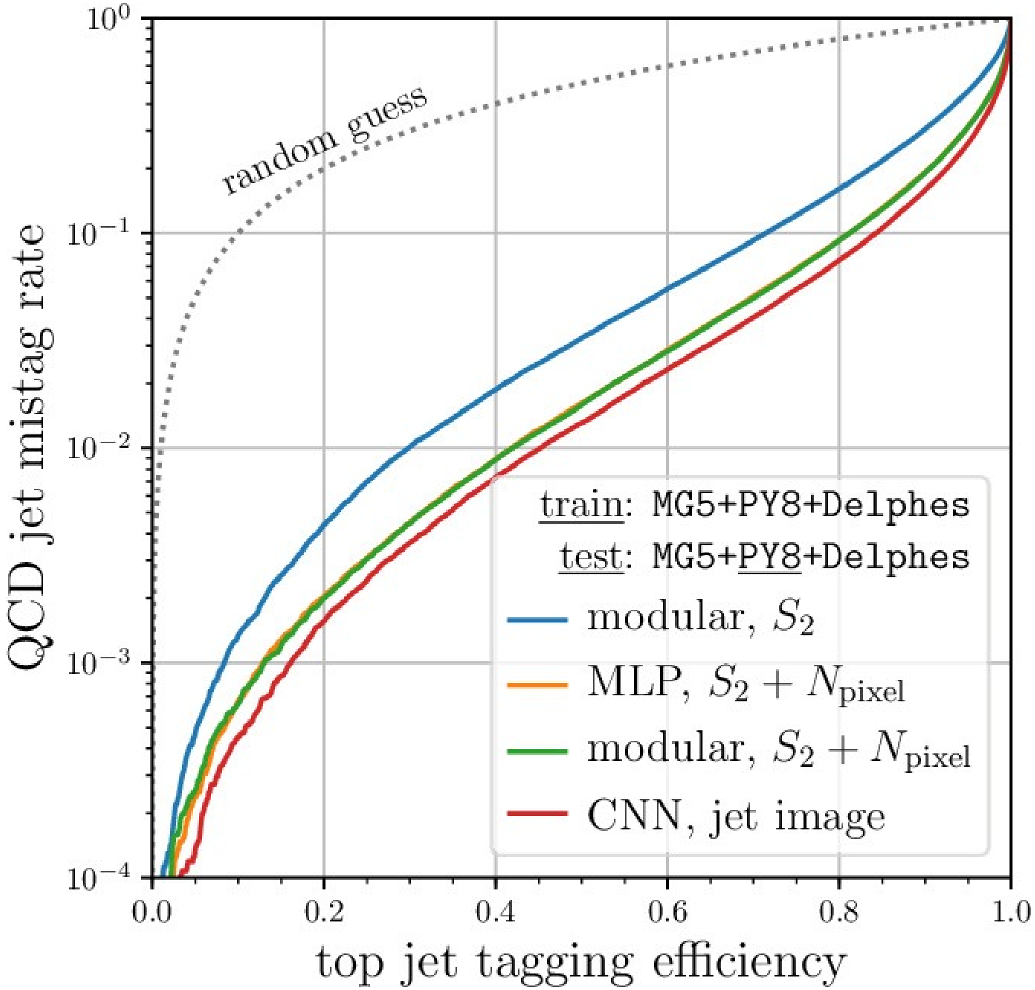


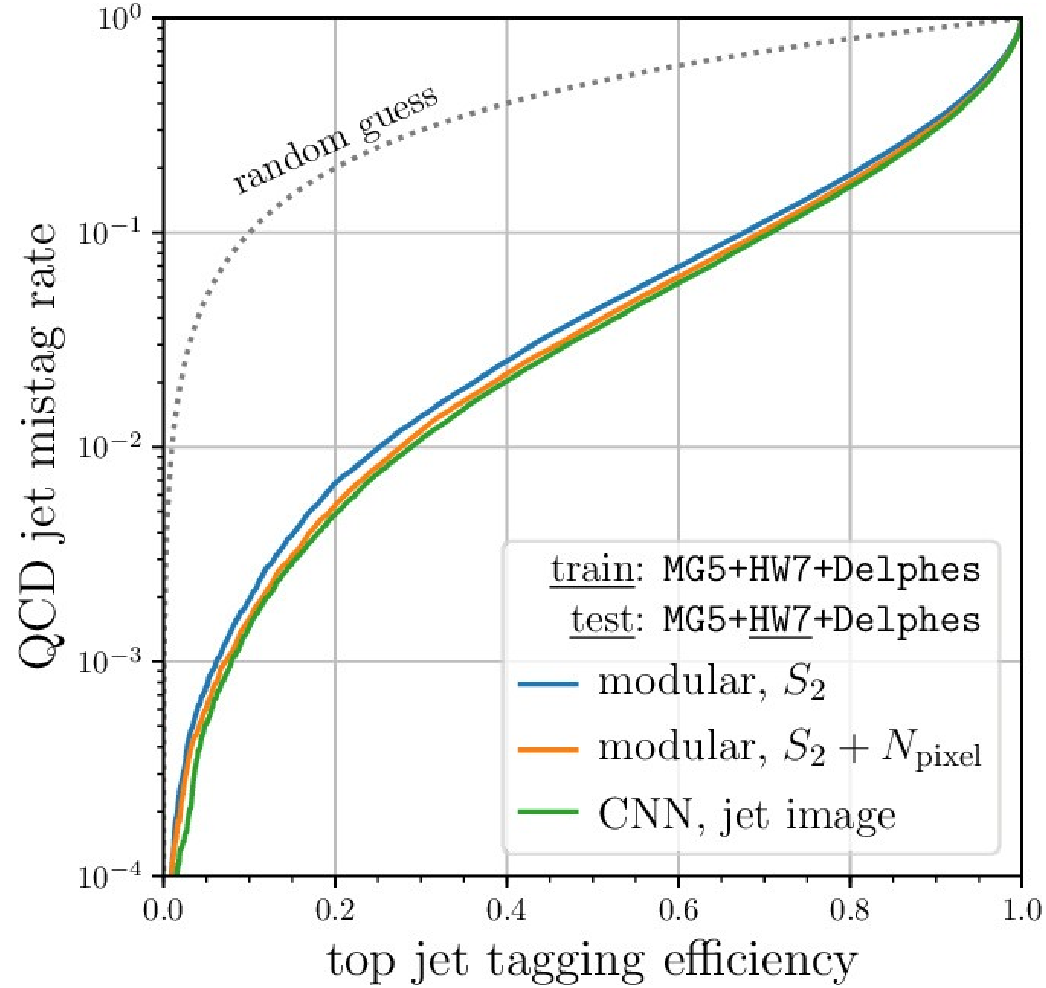
$$\hat{y}(x) = q(\text{top}|x)$$

Train the network output with the cross-entropy.

$$\mathcal{L} = \sum_{i=0}^{N_{\text{event}}} w^{(i)}(x)\, y^{(i)} \, \log \hat{y}^{(i)}(x)$$

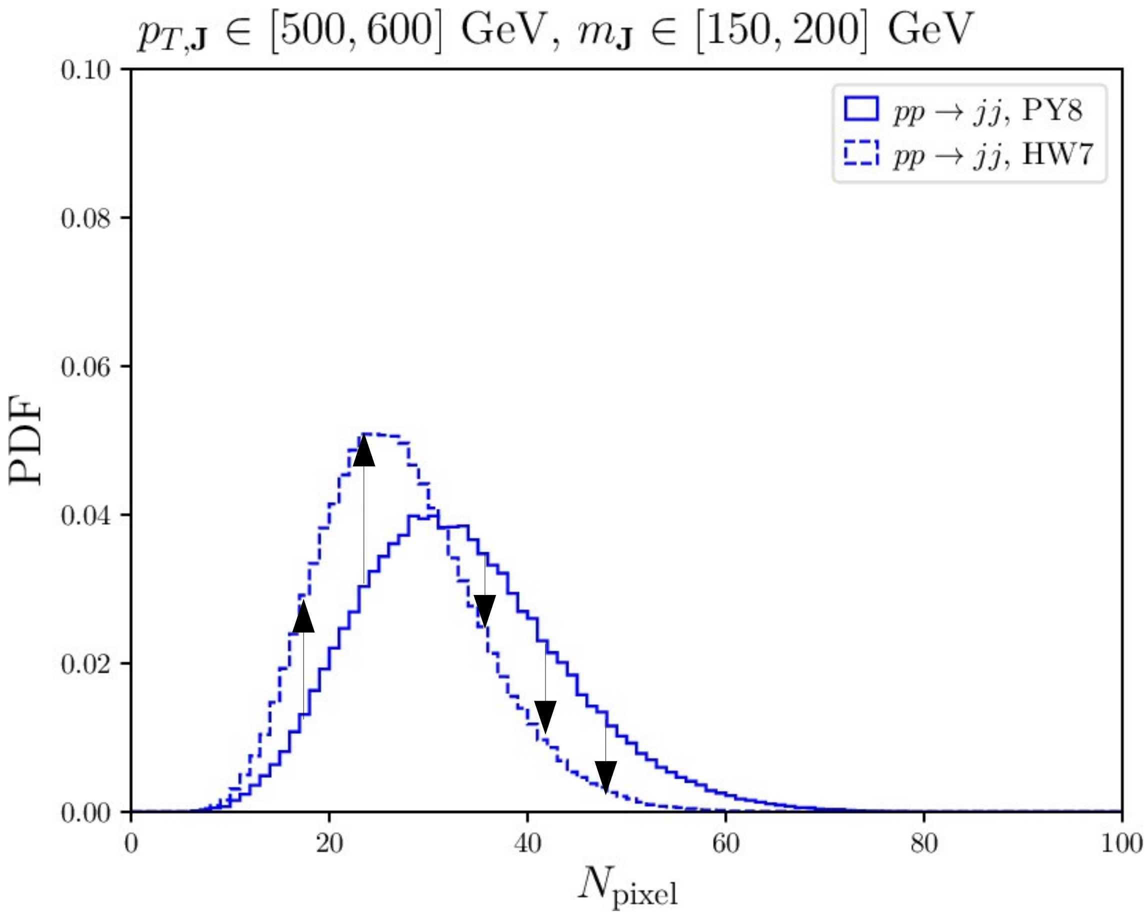
ROCs

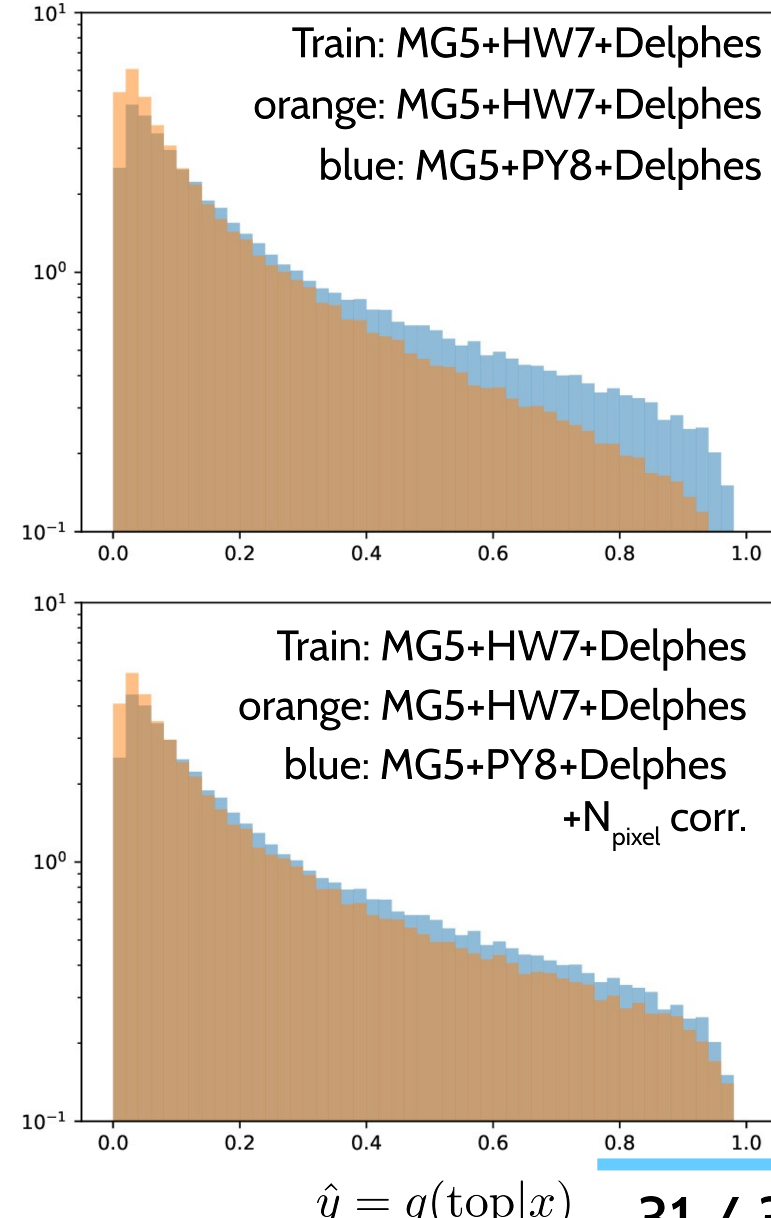




The gap between ROC of modular NN + S2 setup and ROC of CNN is filled.

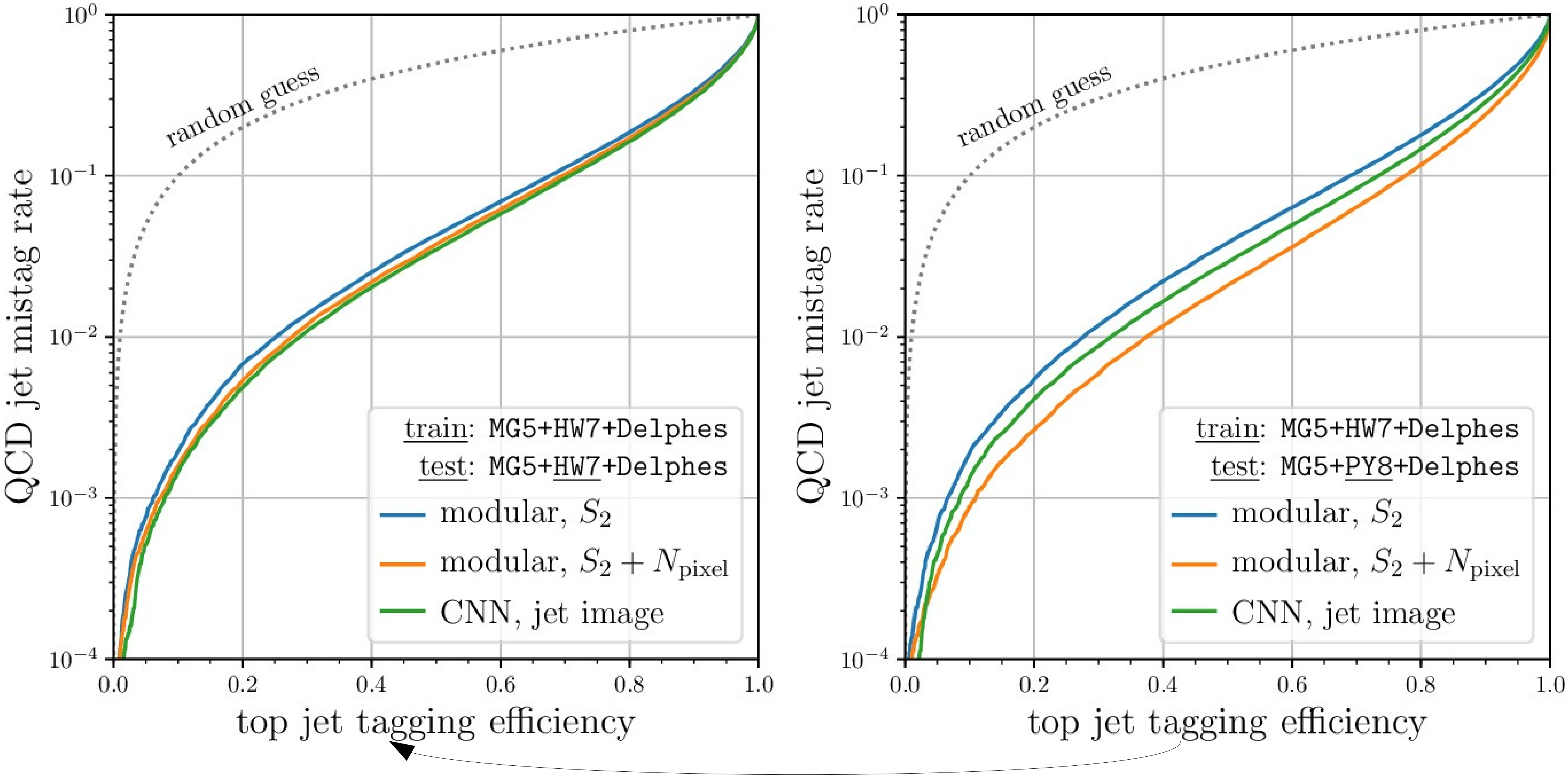
Correcting MC: reweighting PY8 to HW7





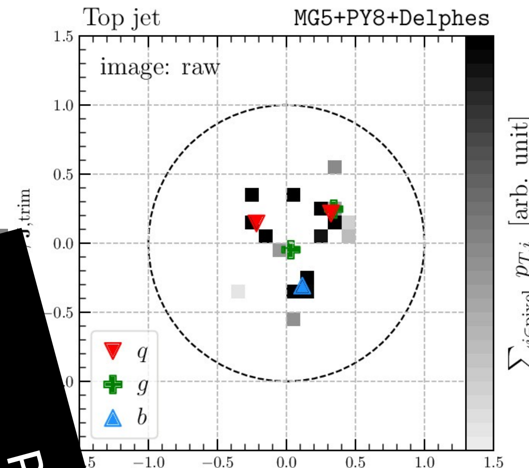
We rescale N_{pixel} distribution of PY8 dijet samples to that of HW7 dijet samples.
The NN output distributions between PY8 and HW7 are more close.

ROCs



ROC of the corrected MC will be close to that with the same train and test sample.

Our road map: Top jet



CNN with Jet Images

Model simplification

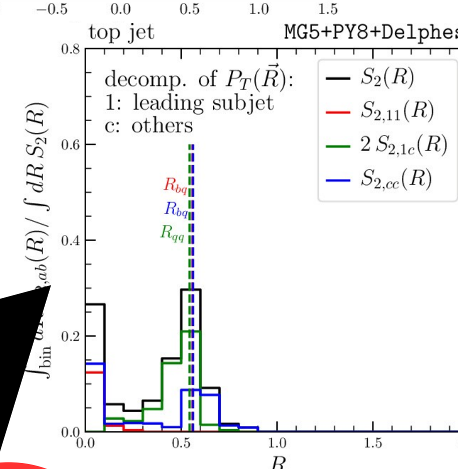
Physics-motivated inputs

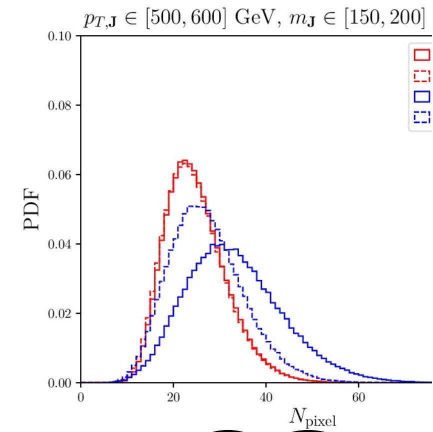
To be appeared on arXiv soon

MLP with two-point energy correlators

Progress...

Model complexity





N_{pixel}?

Model interpretability

Linear model

Work ongoing: generalization of pixel variables

Logistic Regression with two-point energy correlators + ...

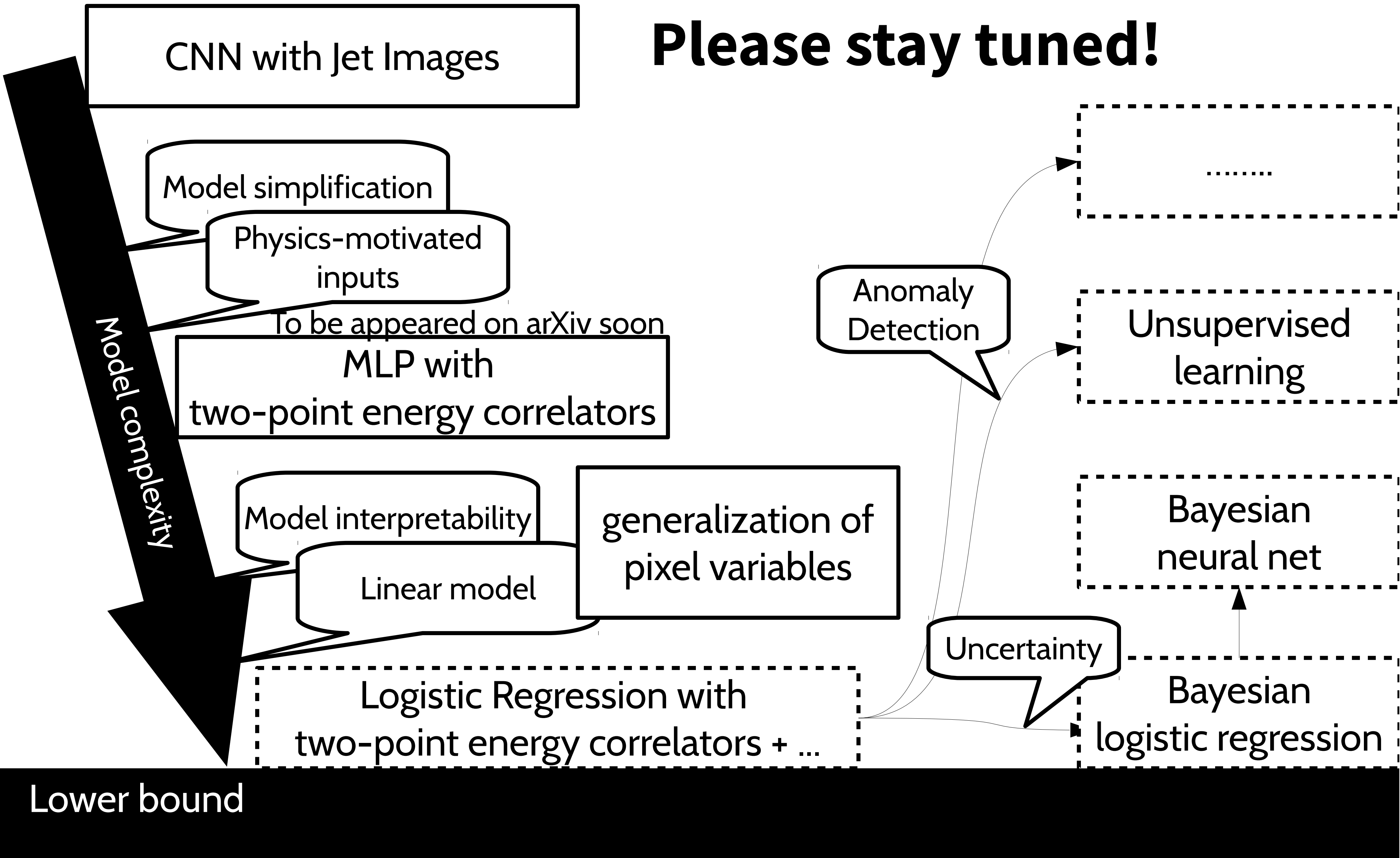
Lower bound

Summary

- For the next run of LHC and future colliders, we need a quick and reliable jet substructure analysis framework.
- We developed a machine learning framework using **two-point correlation spectrum** for analyzing jet substructures.
- The modular neural network with the spectra is useful for classifying top jets from QCD jets with less number of inputs.
- Variables other than energy correlators are also important in the jet classification.

KEK

A quick brainstorming



Backup

Training setup

- The model is implemented with Keras with backend tensorflow.
- Optimizer: ADAM, mimimize the weighted cross-entropy.

$$\mathcal{L} = \sum_{i=0}^{N_{\mathrm{event}}} w^{(i)}(x)\, y^{(i)} \log \hat{y}^{(i)}(x)$$

- $p_{T,J}$ distribution is reweighted to be flat.

$$w(x) = \frac{1}{f_{p_{T,\mathbf{J}}}(p_{T,\mathbf{J}})}$$

  The marginal distribution is approximated by the kernel density estimation.
- Weight initialization: He uniform
- L2 regularization: weight decay constant: 0.001
- Early stopping: patience = 50
- Use moving average of weights and bias for the validation and test. Ignore early t_0=50 epochs.

- Batch size: modular NN: 20, 50, 100, CNN: 100, 200, 500
- Tested two random seeds
- Select a network with the smallest validation AUC
- Cross-validate the trained model with the model with a focal loss.

Training setup

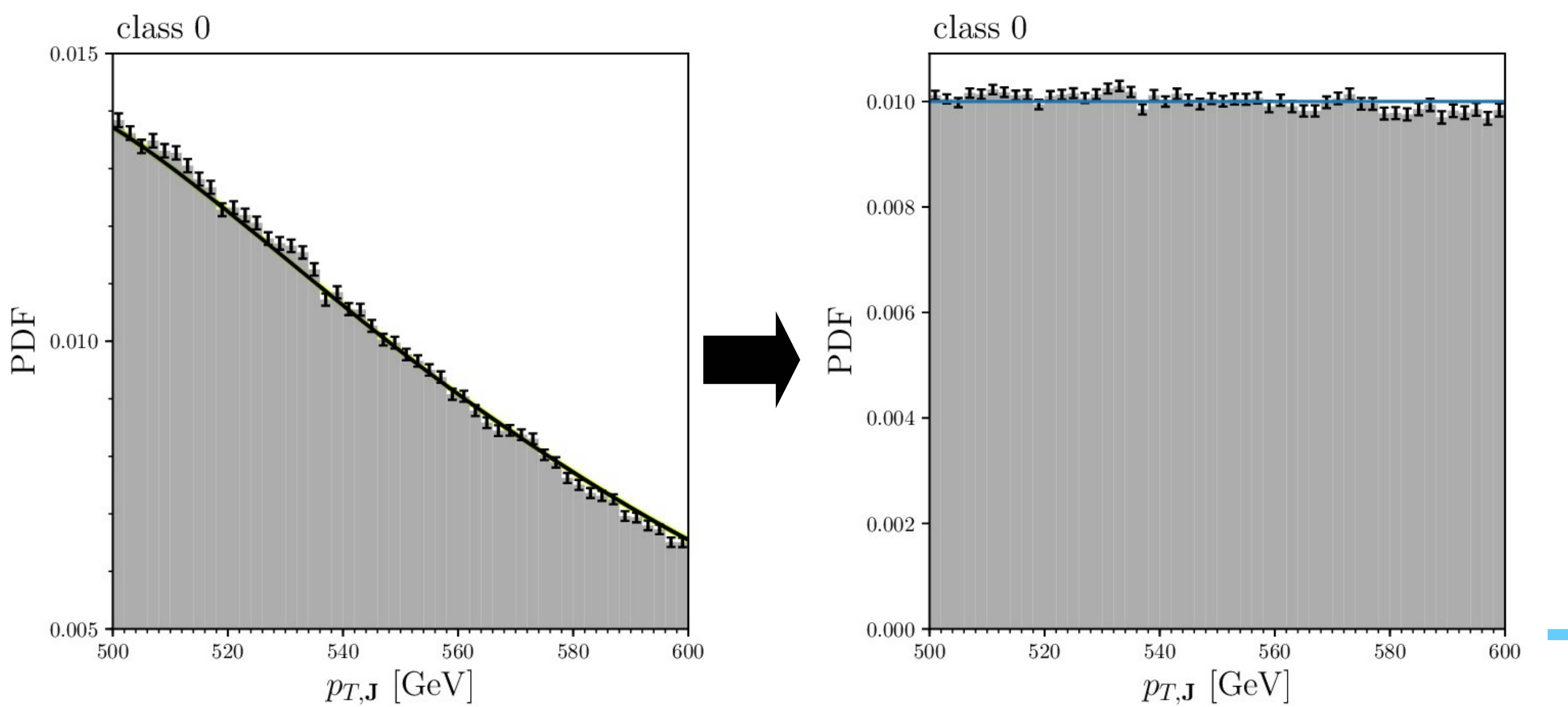
- The model is implemented with Keras with backend tensorflow.
- Optimizer: ADAM, mimimize the weighted cross-entropy.

$$\mathcal{L} = \sum_{i=0}^{N_{\rm event}} w^{(i)}(x)\, y^{(i)} \, \log \hat{y}^{(i)}(x)$$

- $p_{T,J}$ distribution is reweighted to be flat.

$$w(x) = \frac{1}{f_{p_{T,\mathbf{J}}}(p_{T,\mathbf{J}})}$$

  The marginal distribution is approximated by the kernel density estimation.



Training setup

- Weight initialization: He uniform
- L2 regularization: weight decay constant: 0.001
- Early stopping: patience = 50
- Use moving average of weights and bias for the validation and test. Ignore early t_0=50 epochs.

$$\bar{\theta}^{(t)} = \alpha\bar{\theta}^{(t-1)} + (1-\alpha)\theta^{(t)}$$

$$\hat{\theta}^{(t)} = \frac{1}{1-\alpha^{t-t_0+1}}\bar{\theta}^{(t)}$$

For training: $q(\mathrm{top}|x;\theta^{(t)})$ For validation and test: $q(\mathrm{top}|x;\hat{\theta}^{(t)})$

- Batch size: modular NN: 20, 50, 100, CNN: 100, 200, 500
- Tested two random seeds
- Select a network with the smallest validation AUC

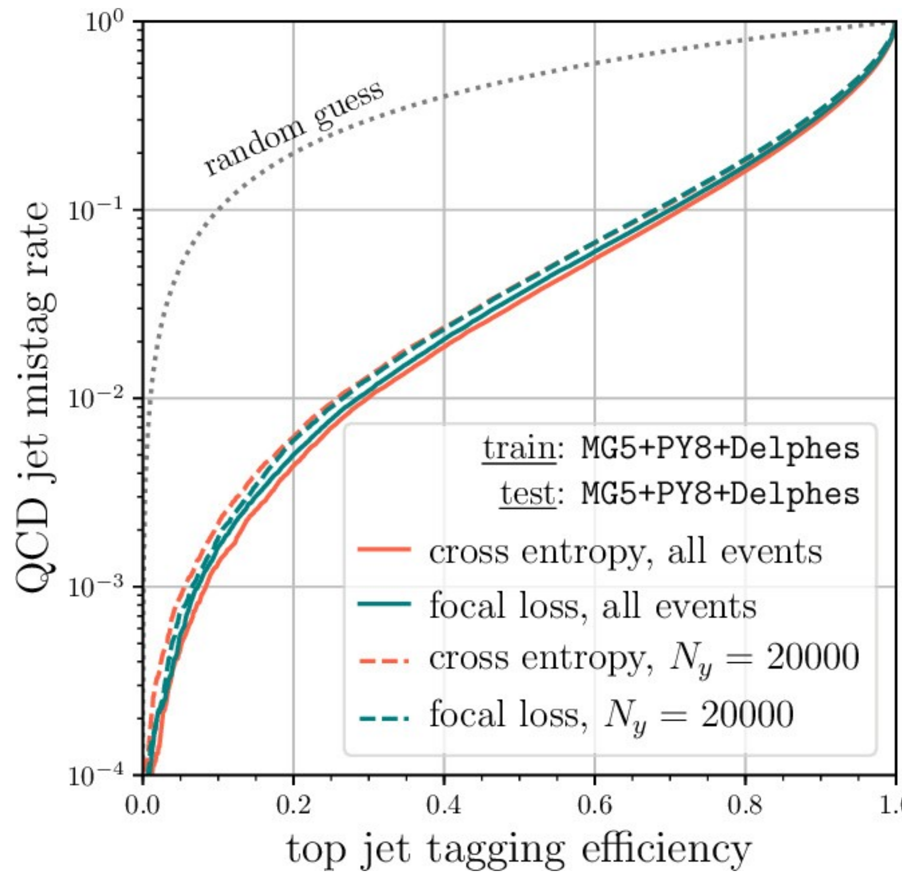
Focal Loss

To check whether our number of samples are enough, we consider a focal loss.

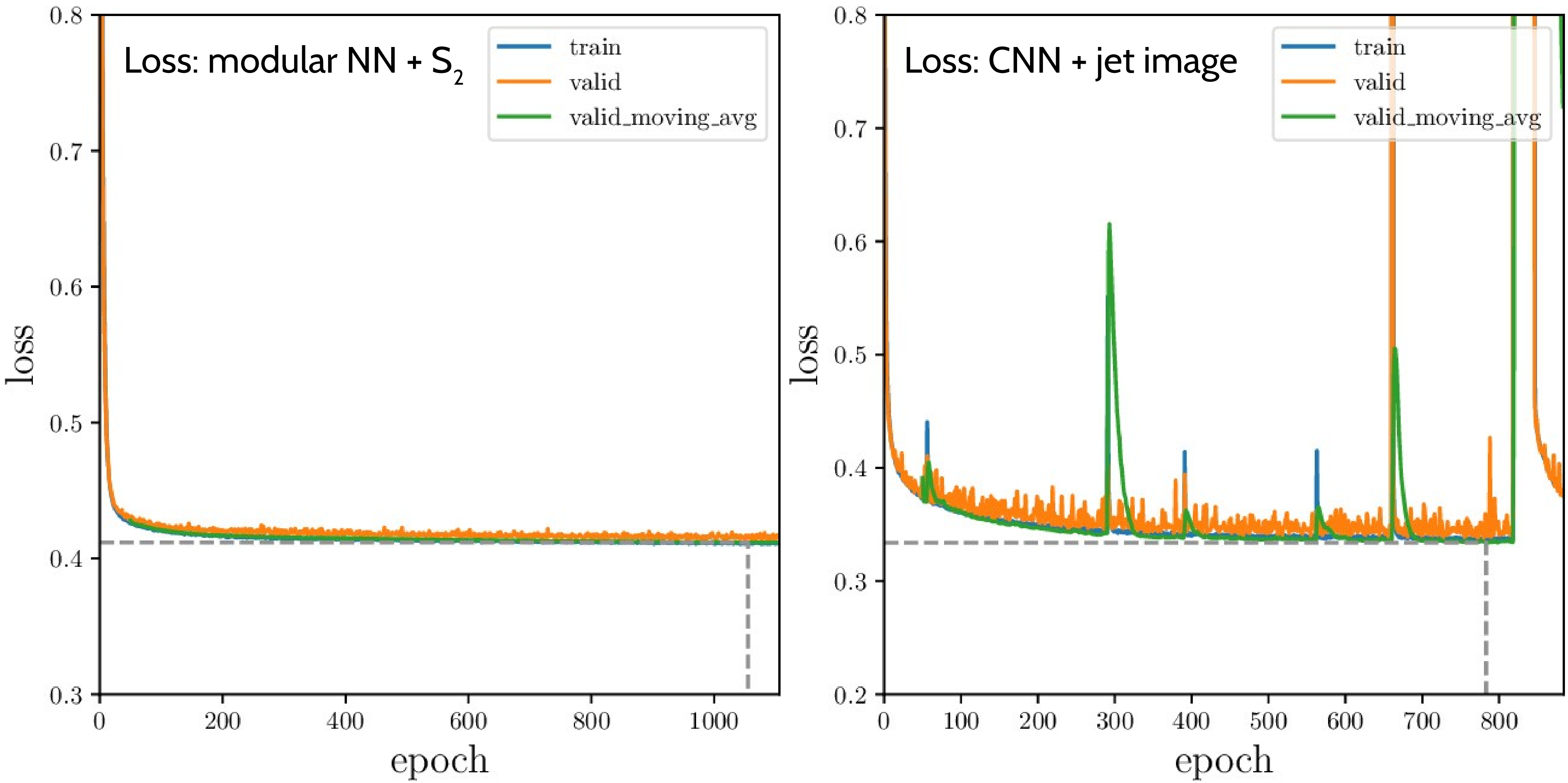
$$\mathcal{L} = \sum_{i=0}^{N_{\text{event}}} w^{(i)}(x)(1-\hat{y}^{(i)}(x))^{\gamma}\, y^{(i)}\, \log \hat{y}^{(i)}(x)$$

The focal loss can be considered as a perturbation from the cross-entropy, i.e., maximum likelihood estimation. The performance is slightly less in infinite stat. limit.

Focal loss penalize learning from easy samples, so that it may help in low statistics case.

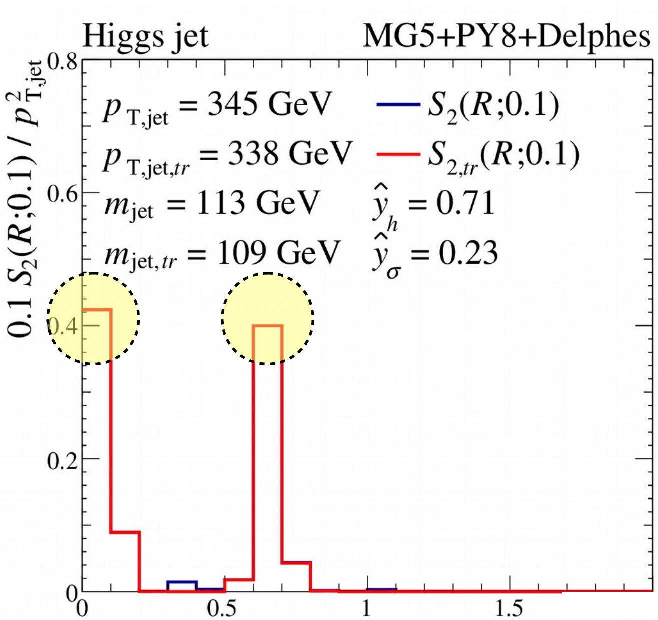


History of Loss Function during Training



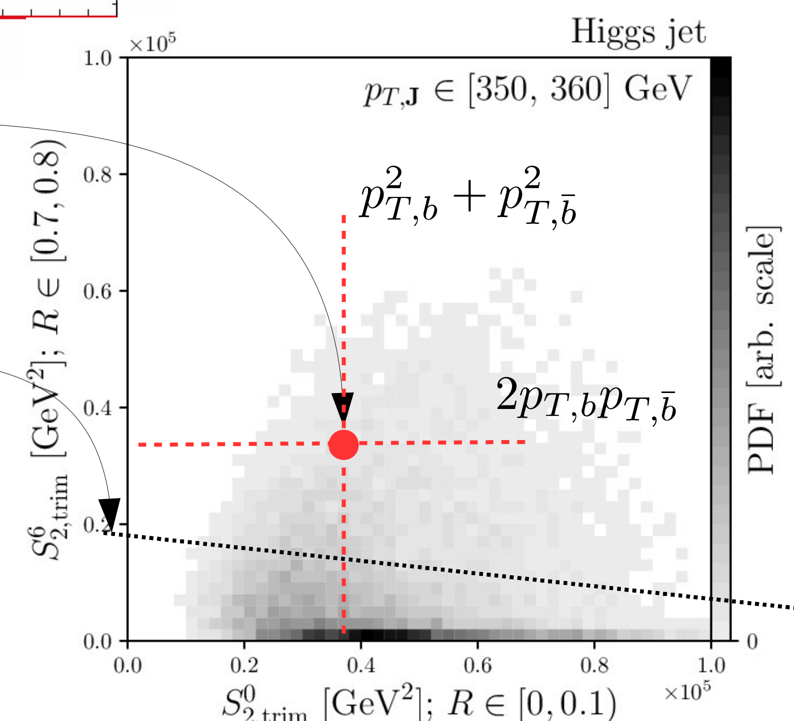
Modular NN + S_2 setup has less chance of learning batch-specific features.

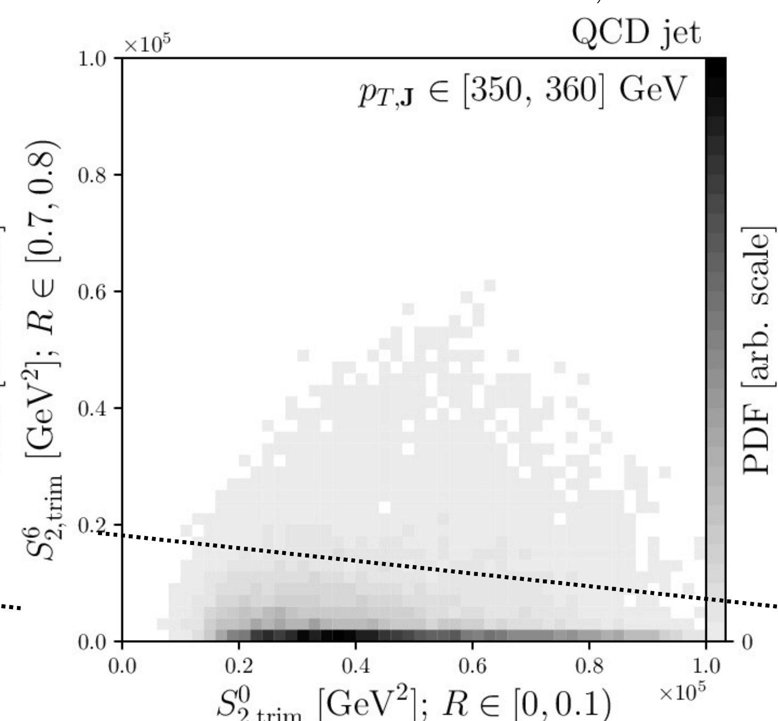
2D distribution of spectral intensity



Note that the Higgs jet samples are clustered in 20x2 dimensional phase space of the spectra.
It's hard to visualize the cluster,
but we can see a cluster of Higgs jet in
2D histogram of the projected input vector

$$(S_{2,\mathrm{trim}}(R=0), S_{2,\mathrm{trim}}(R=0.7)) \qquad \hat{R}_{b\bar{b}} = \frac{2m_h}{p_{T,\mathbf{J}}} \approx 0.7$$

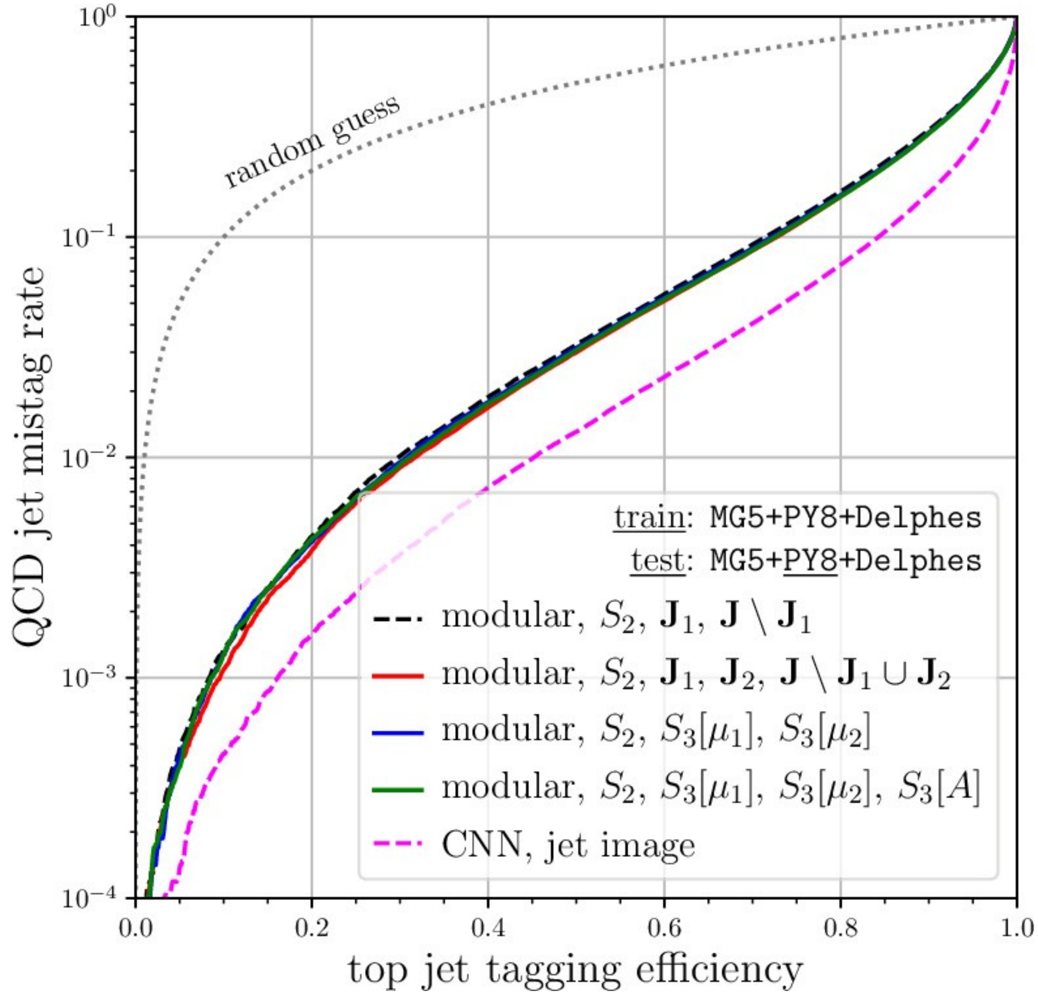




classification boundary

A linear classifier may work well.

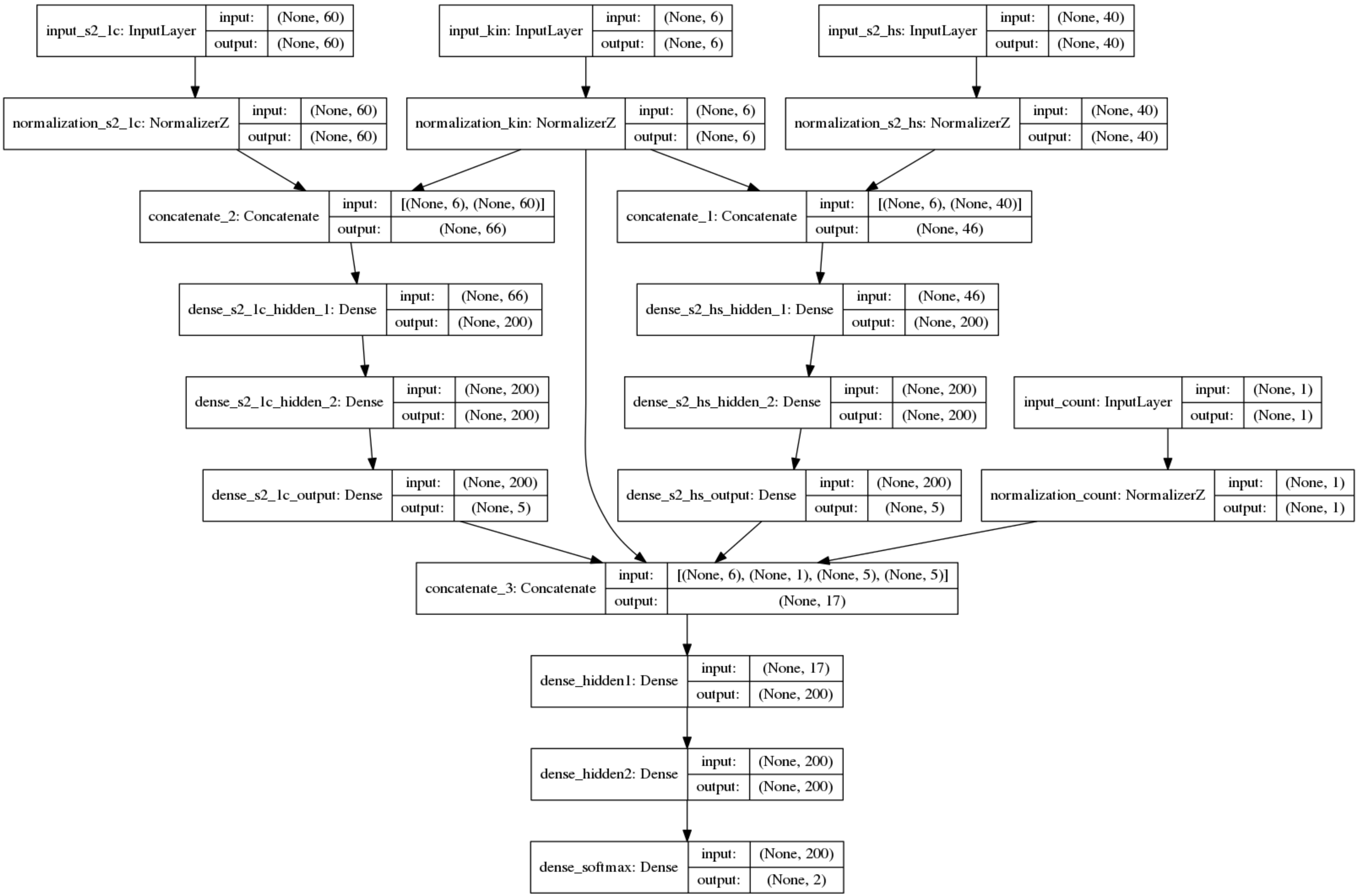
Including higher order terms...



They do not help much...

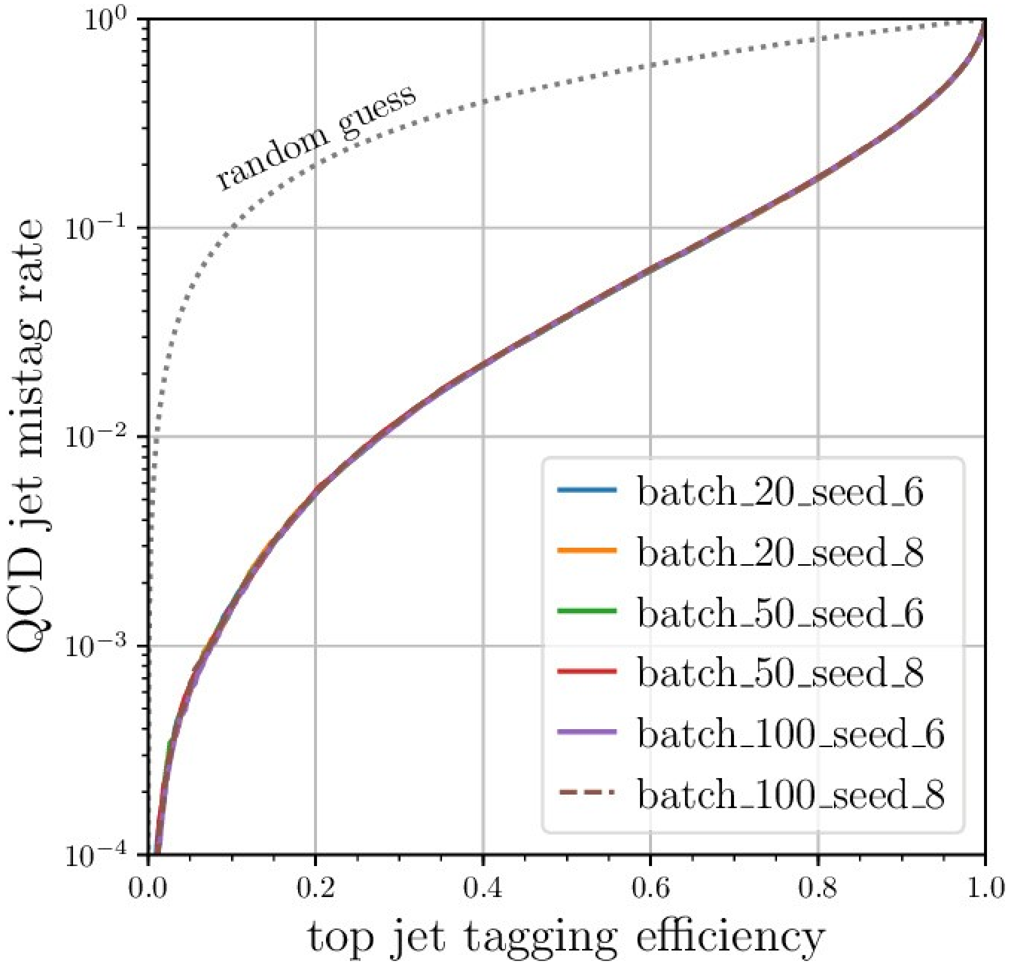
Network structure: modular S2

Network structure: modular S2 + N_{pixel}

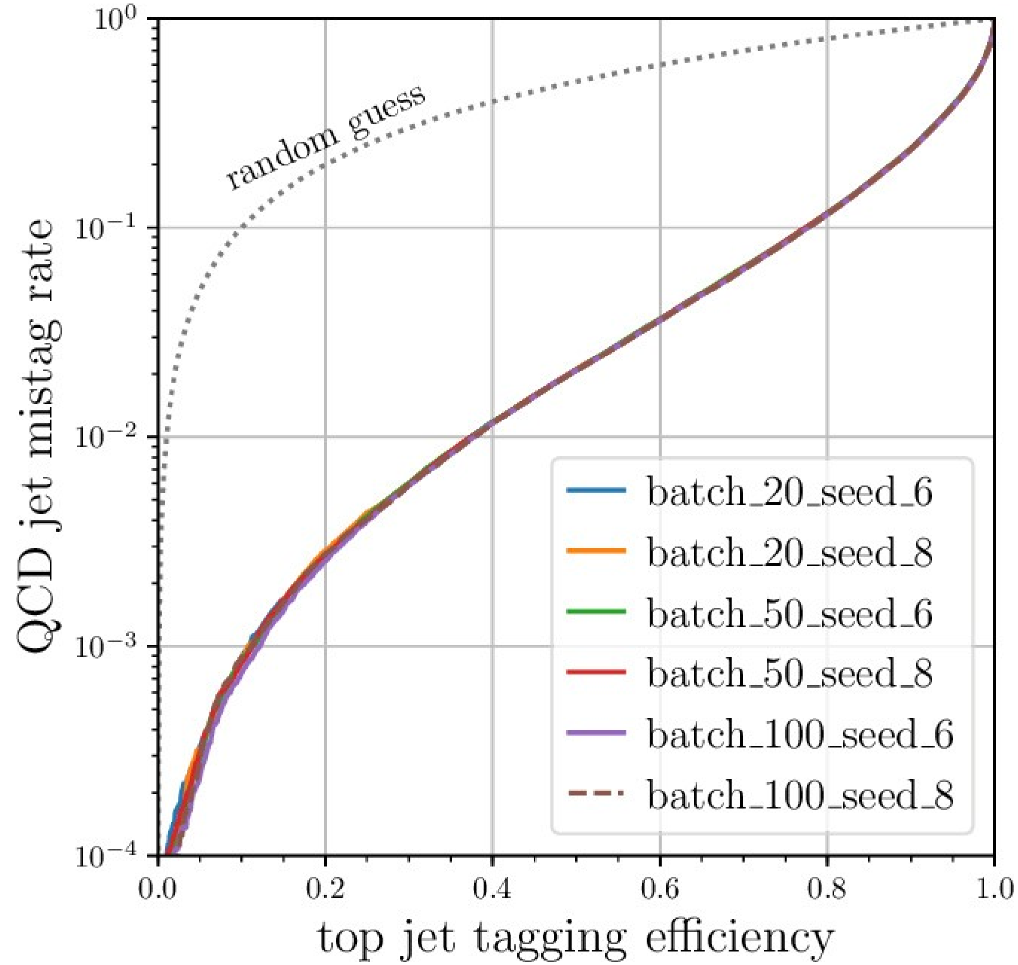


Network structure: CNN

Hyper-parameter scanning: modular S2

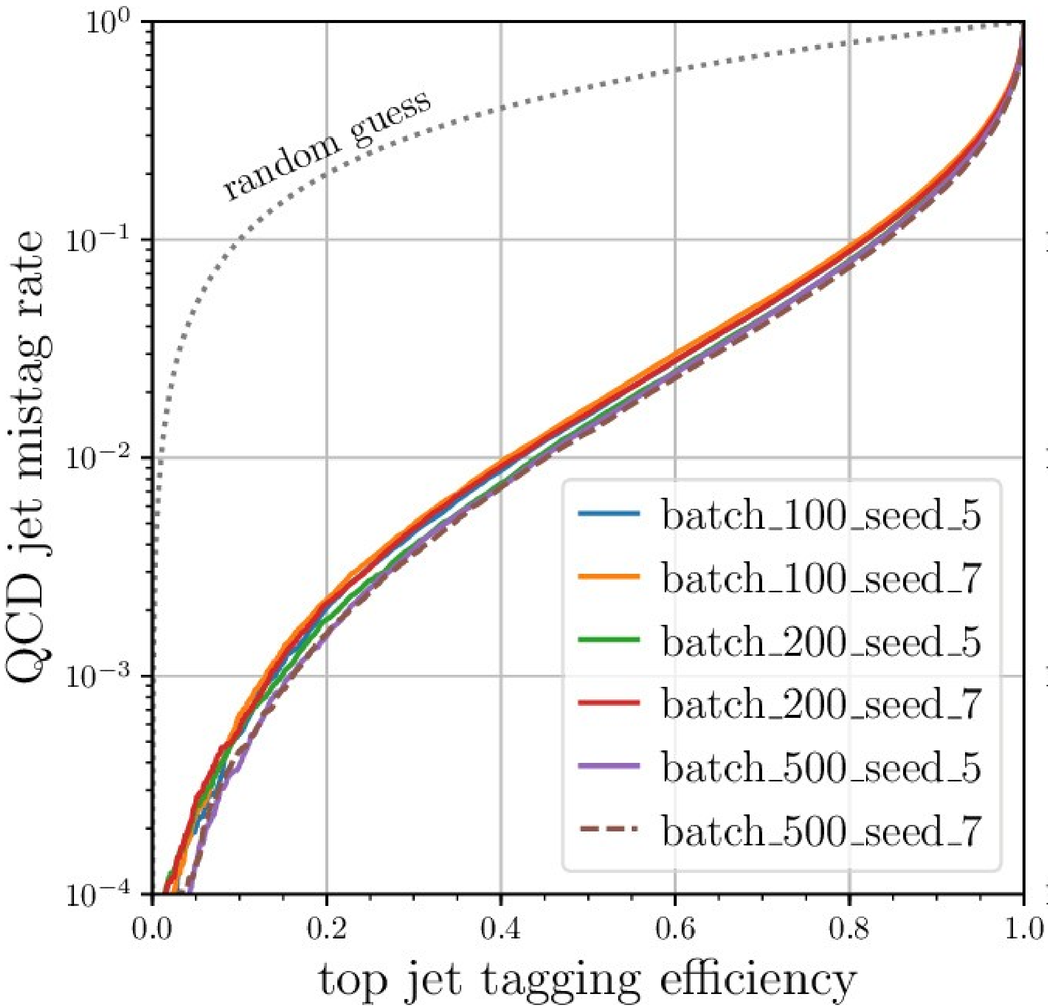


Train: MG5+PY8+Delphes
Test: MG5+PY8+Delphes

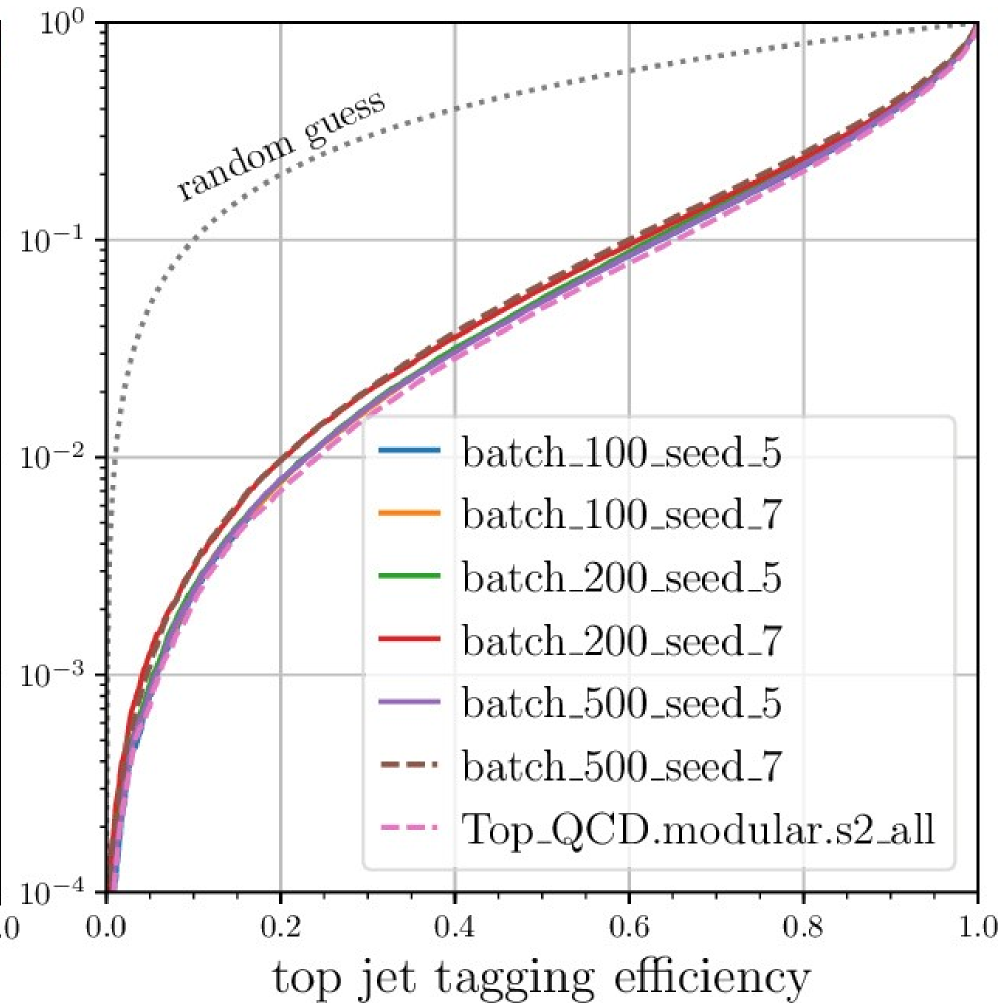


Train: MG5+PY8+Delphes
Test: MG5+HW7+Delphes

Hyper-parameter scanning: CNN

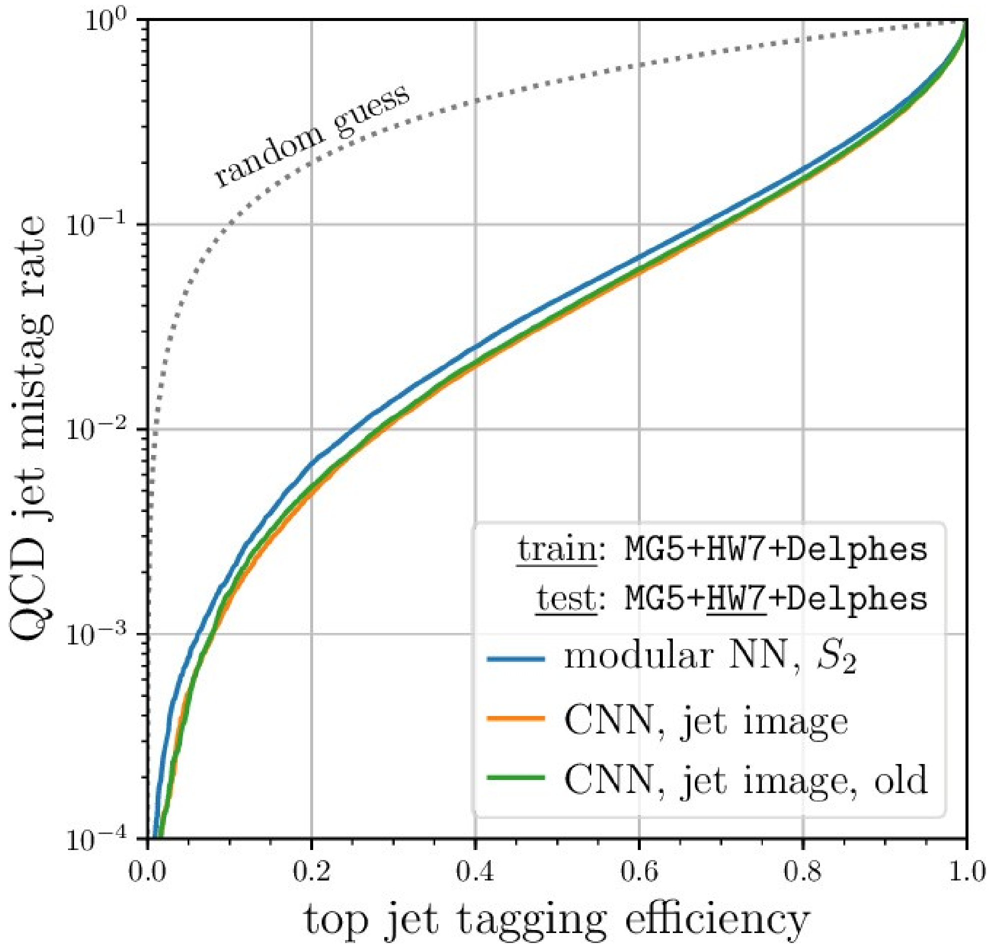


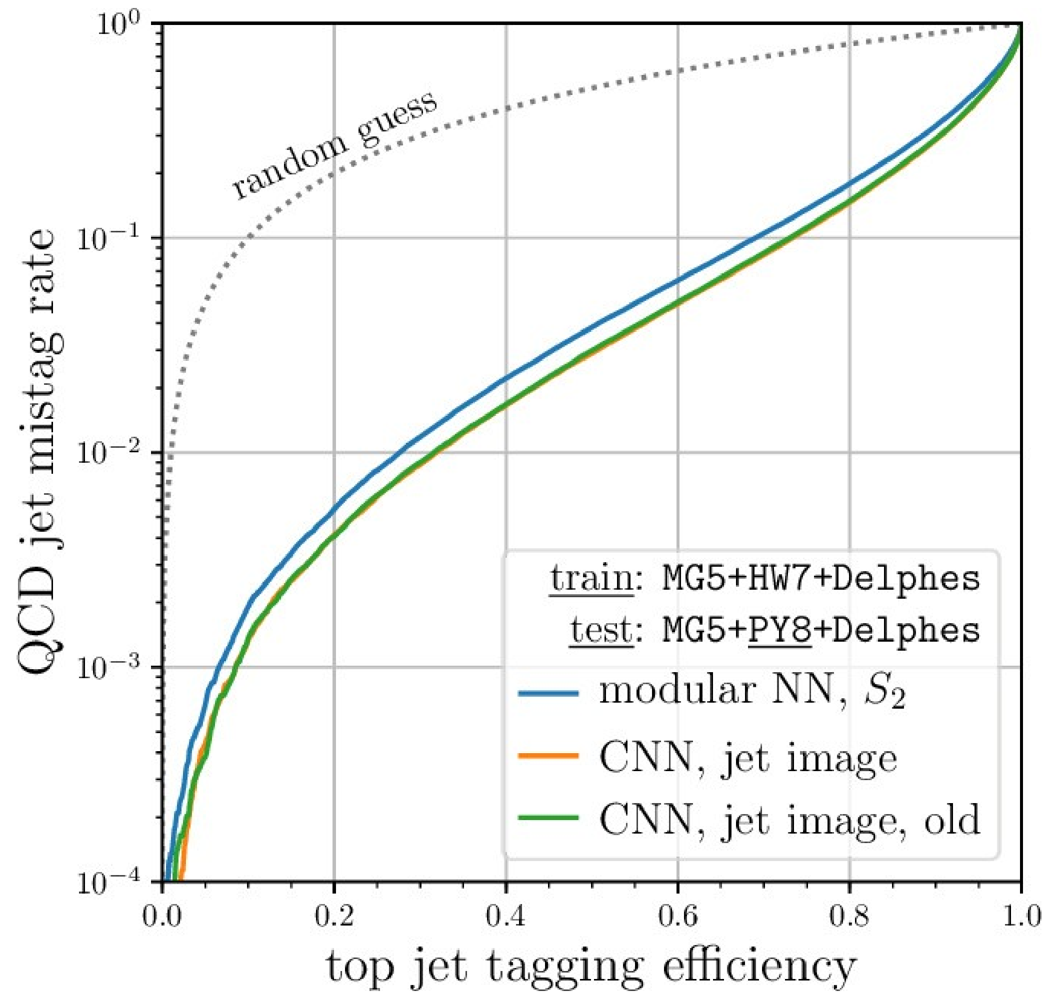
Train: MG5+PY8+Delphes
Test: MG5+PY8+Delphes



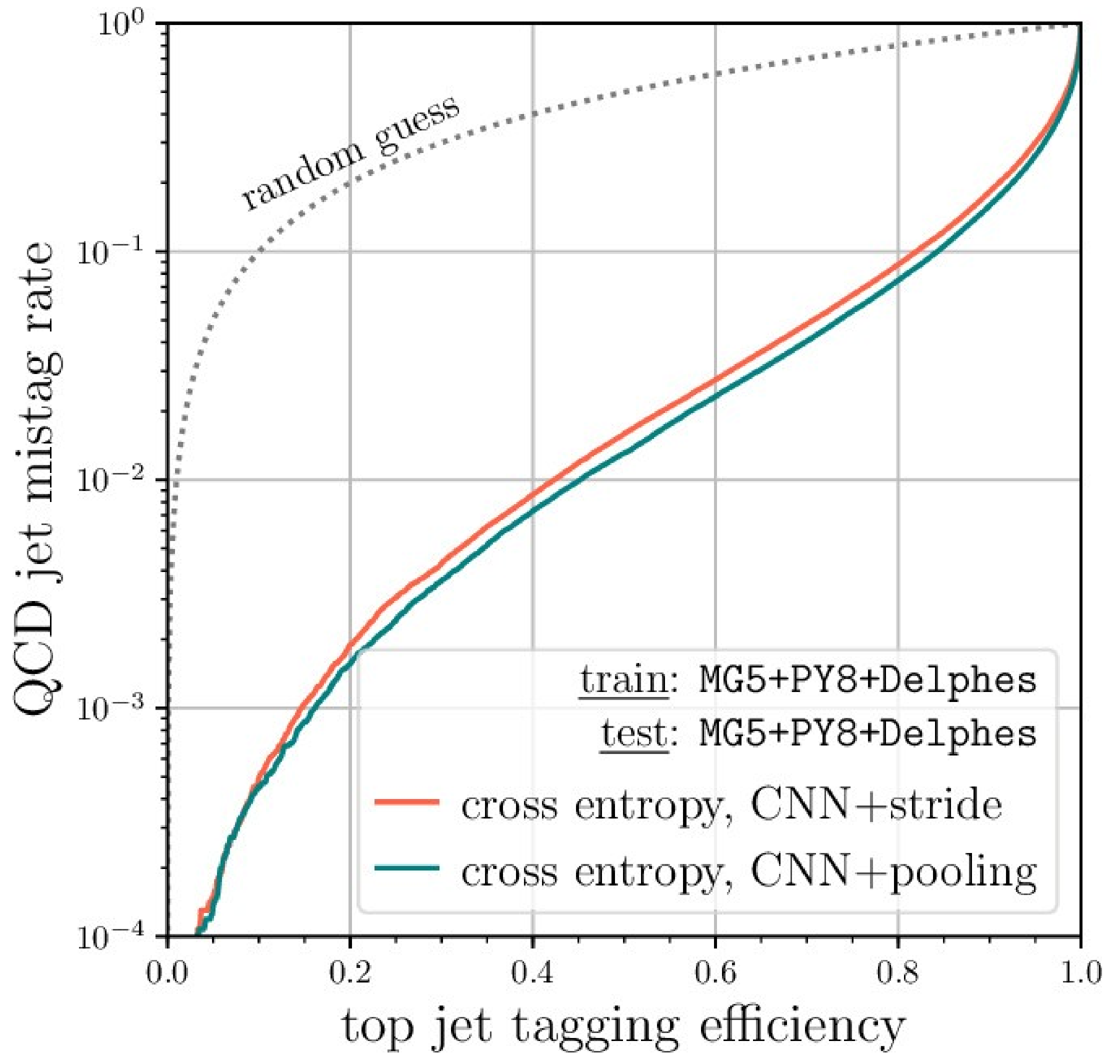
Train: MG5+PY8+Delphes
Test: MG5+HW7+Delphes

Train: HW7, varying test sample





Stride vs. Pooling



Interpretable Setup:

$$\sum_{n=0}^{\infty} \mathcal{O}\left[P_T^{2n}\right] \to \mathcal{O}\left[P_T^{2}\right] + \cancel{\cdots}$$

We may try the following two-level setup

Level1: substructure analyzer

$$\Phi[P_T; x_{\text{kin}}] = \int dR\, S_{2,ab}(R)\, w_{ab}^{(2)}(R; x_{\text{kin}})$$

: <u>IRC safe</u>

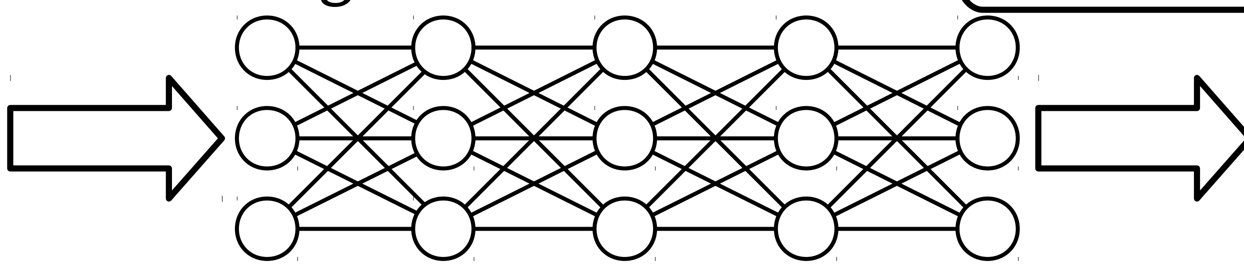
$$x_{\text{kin}} = \{p_{T,\text{jet}}, m_{\text{jet}}\}$$

$$\hat{R}_{b\bar{b}} = \frac{2m_h}{p_{T,h}}$$

Level2: kinematics analyzer

Use neural network to approximate the weight function.

x_{kin}



$w_{ab}^{(2)}(R; x_{\text{kin}})$

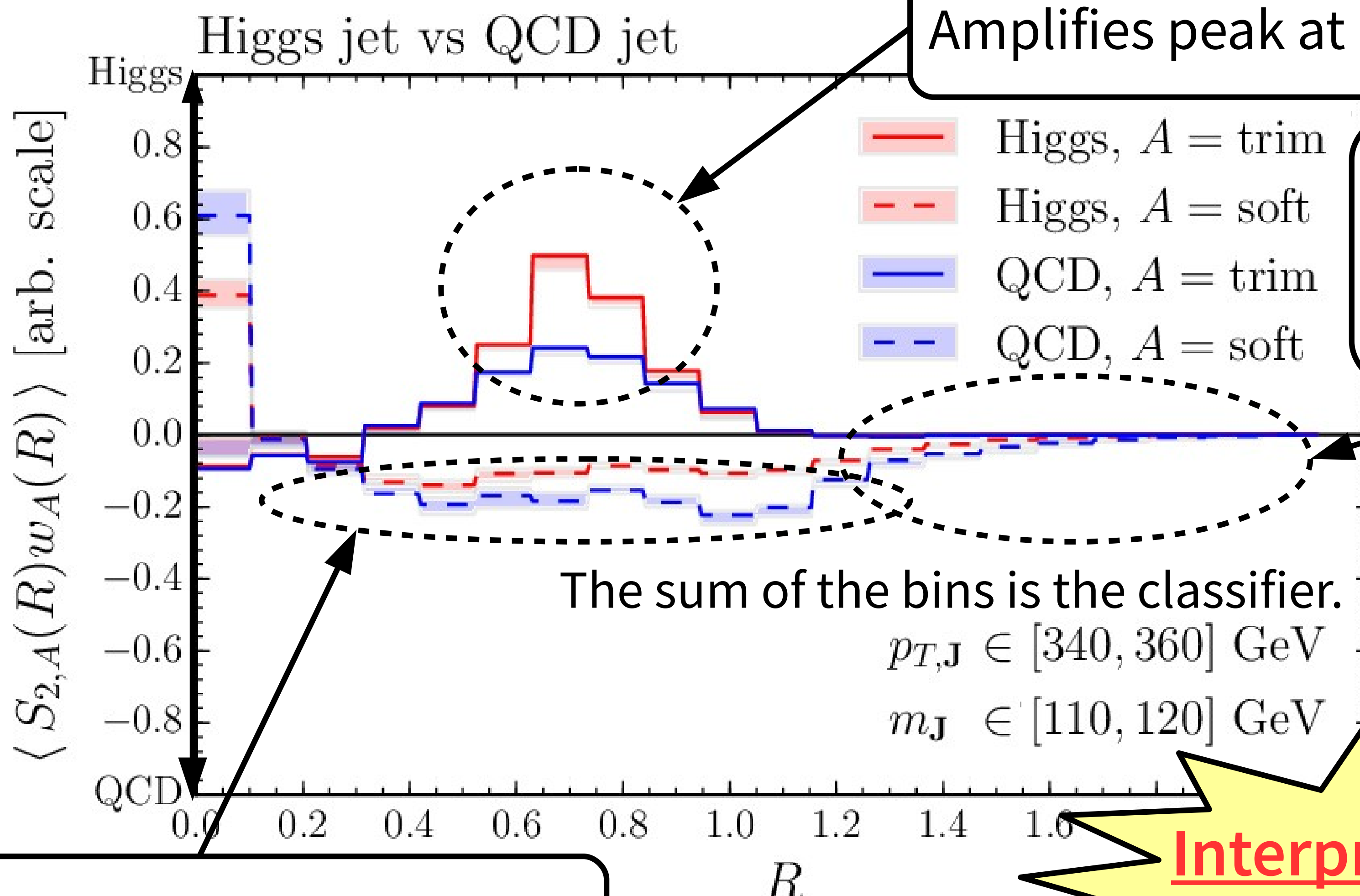
This architecture gives you two interpretable quantities:

$w_{ab}^{(2)}(R; x_{\text{kin}})$ shows the **<u>functional form</u>** of the energy correlator.

$S_{2,ab}(R)\, w_{ab}^{(2)}(R; x_{\text{kin}})$ shows the **<u>contribution</u>** to the classifier.

Average of the linear classifier outputs

$$\Phi[S_{2,ab}] = \int dR\, S_{2,\mathrm{trim}}(R) w^{(2)}_{\mathrm{trim}}(R) + \int dR\, S_{2,\mathrm{soft}}(R) w^{(2)}_{\mathrm{soft}}(R)$$



Amplifies peak at Rbb: **Higgs jet**

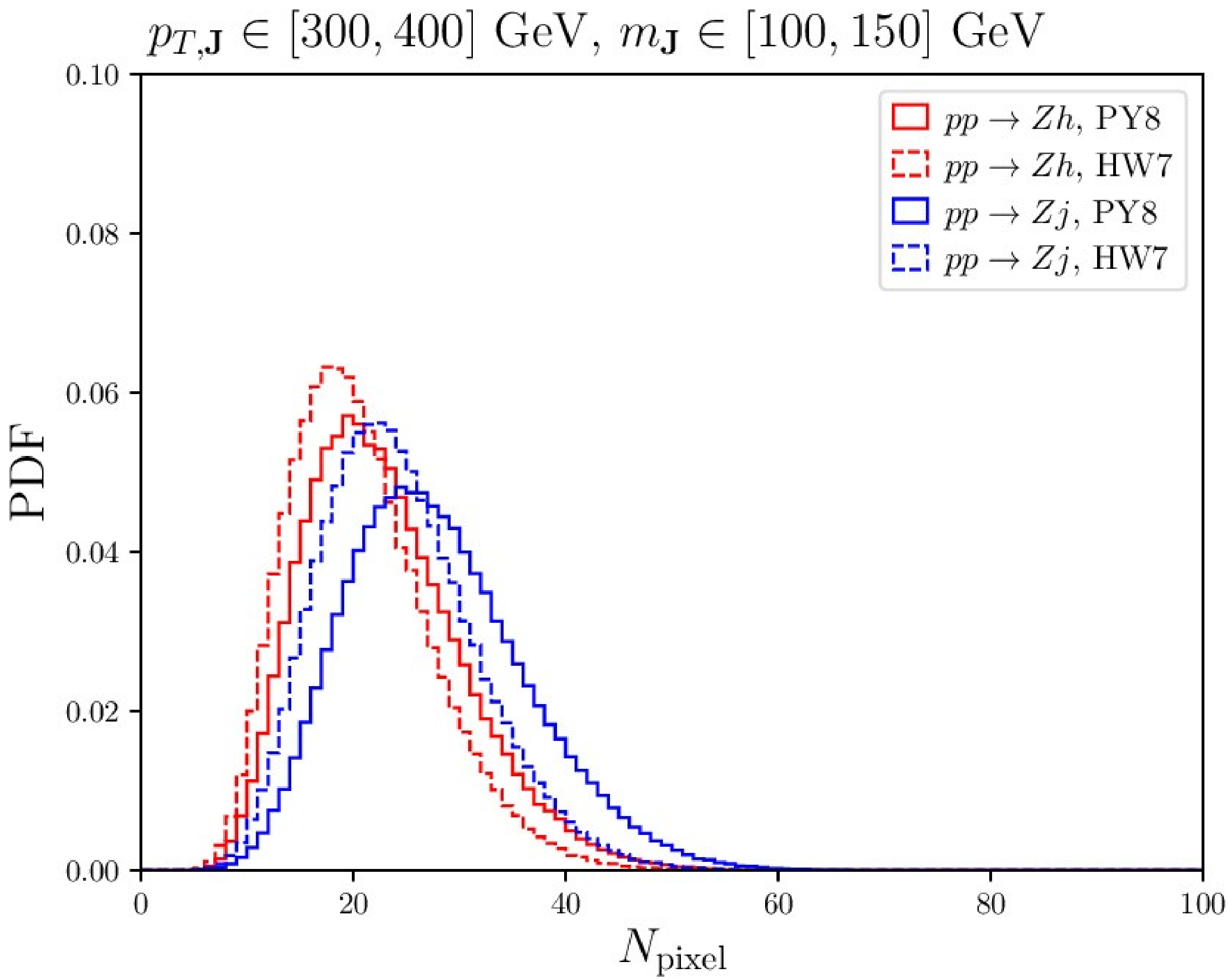
Large angle radiation: **QCD jet**

The sum of the bins is the classifier.

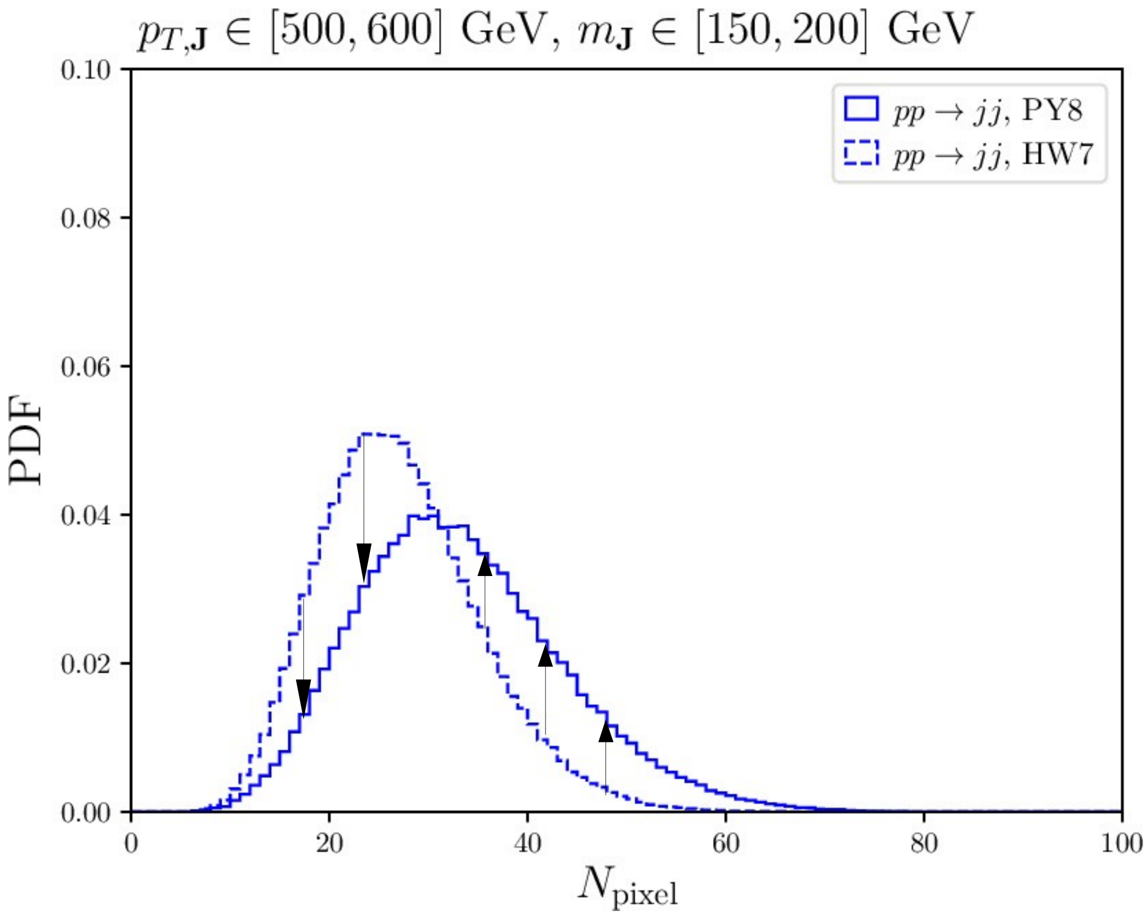
More soft activity: **QCD jet**

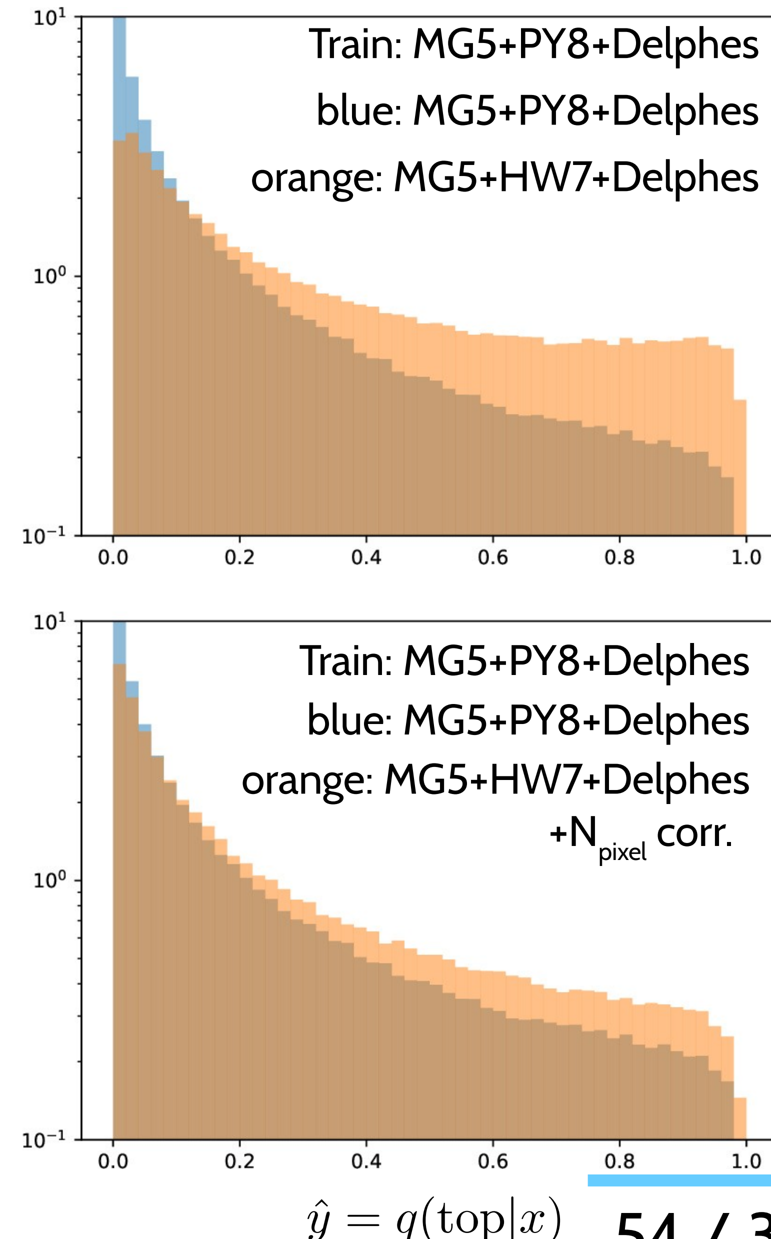
Interpretable

N_{pixel} distribution: Higgs jet and QCD jet

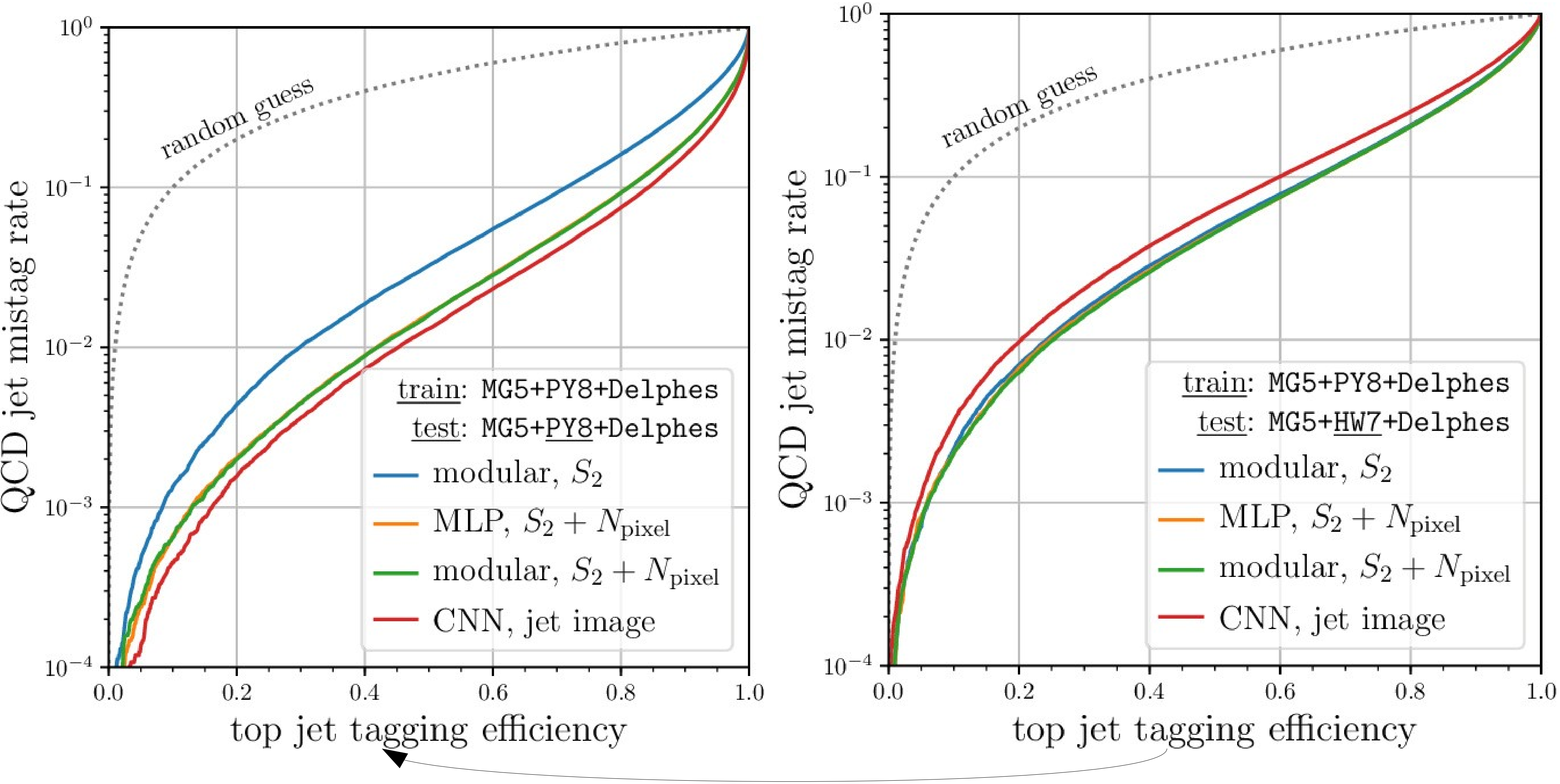


Correcting MC: reweighting HW7 to PY8





ROCs



ROC of the corrected MC will be close to that with the same train and test sample.