Tagger-Mass Decorrelation: Experience within CMS

Huilin Qu

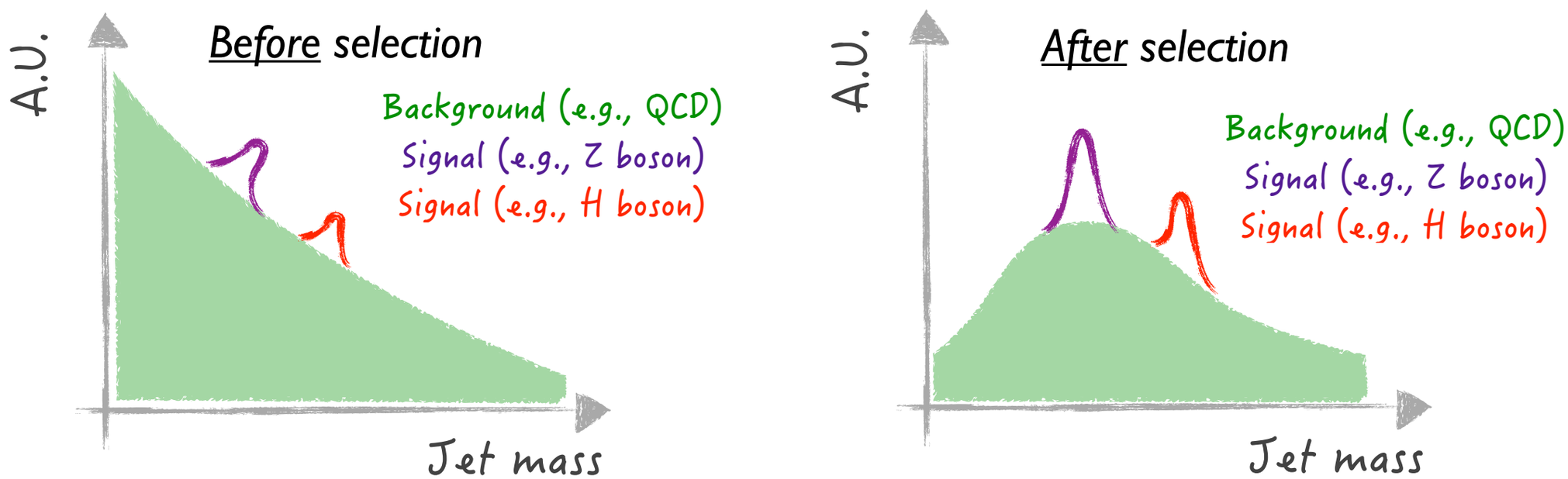
on behalf of the CMS collaboration

ML4Jets2020

January 16, 2020

Introduction

- Reconstruction and identification of Lorentz-boosted hadronically decaying particles (top/W/Z/H) is an important topic at the LHC
  - provides powerful handles for both searches for new physics and measurements of standard model processes
  - significant performance boost from machine-learning (ML) techniques
- But taggers are often correlated with the jet mass

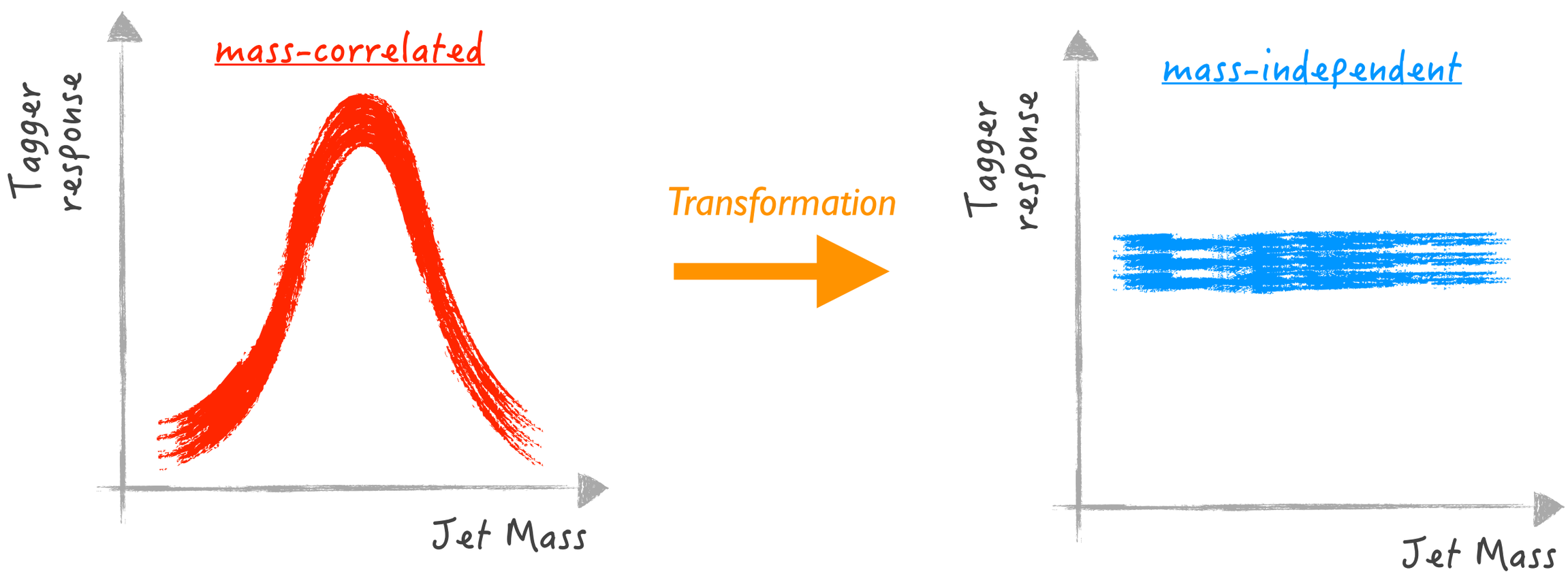


  - i.e., jet mass shape of the background becomes similar to that of the signal after selection with the tagger: **"Mass sculpting"**
  - depending on the analysis this may not be welcome...
- Various methods explored in CMS to reduce the tagger's correlation with the jet mass

METHOD 1: TRANSFORM TAGGER RESPONSE

- A tagger is mass-correlated because its response changes with the jet mass
  - a mass-independent tagger has a uniform response w.r.t the jet mass
- Mass decorrelation method 1: *the "brute-force" way*

Transforming the tagger response such that it no longer changes with the jet mass



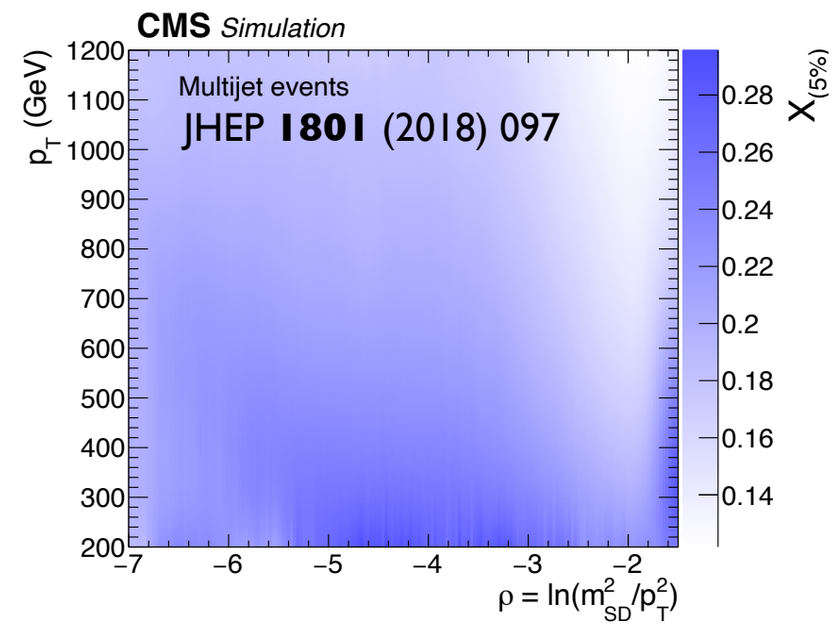
Designing Decorrelated Tagger (DDT)

- Designing Decorrelated Tagger (DDT) [JHEP **1605** (2016) 156]
  - transforms the tagger response as a function of the jet p_T and $\rho = \ln(m_{SD}^2/p_T^2)$:

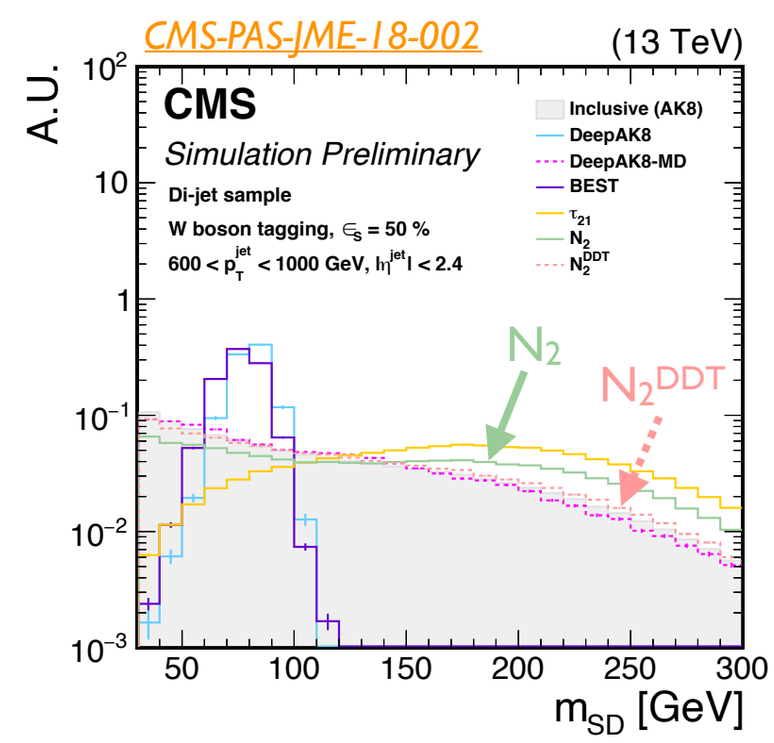
$$\mathrm{Tagger}^{DDT}(\rho, p_T) = \mathrm{Tagger}(\rho, p_T) - \mathrm{Tagger}^{(x\%)}(\rho, p_T)$$

    where $\mathrm{Tagger}^{(x\%)}(\rho, p_T)$ is the threshold for a background efficiency of x%, derived from simulated background (QCD) events
  - after the transformation, the selection $\mathrm{Tagger}^{DDT}>0$ (or <0) yields a constant background efficiency of x% across the m_{SD} and the p_T range

DDT map: $N_2^{(5\%)}$

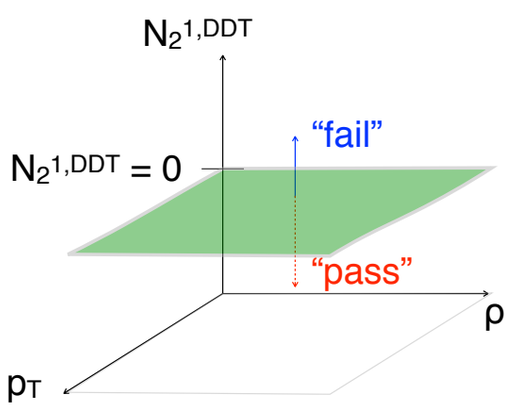


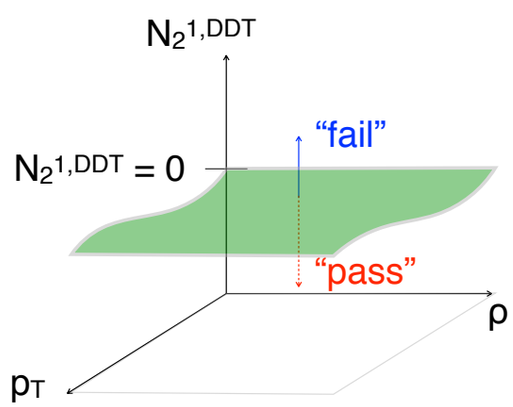
- N_2^{DDT}
  - N_2: generalized energy correlation functions [JHEP **1612** (2016 153] for 2-prong (W/Z/H) tagging
  - N_2^{DDT}: mass-decorrelated version of N_2 using the DDT method

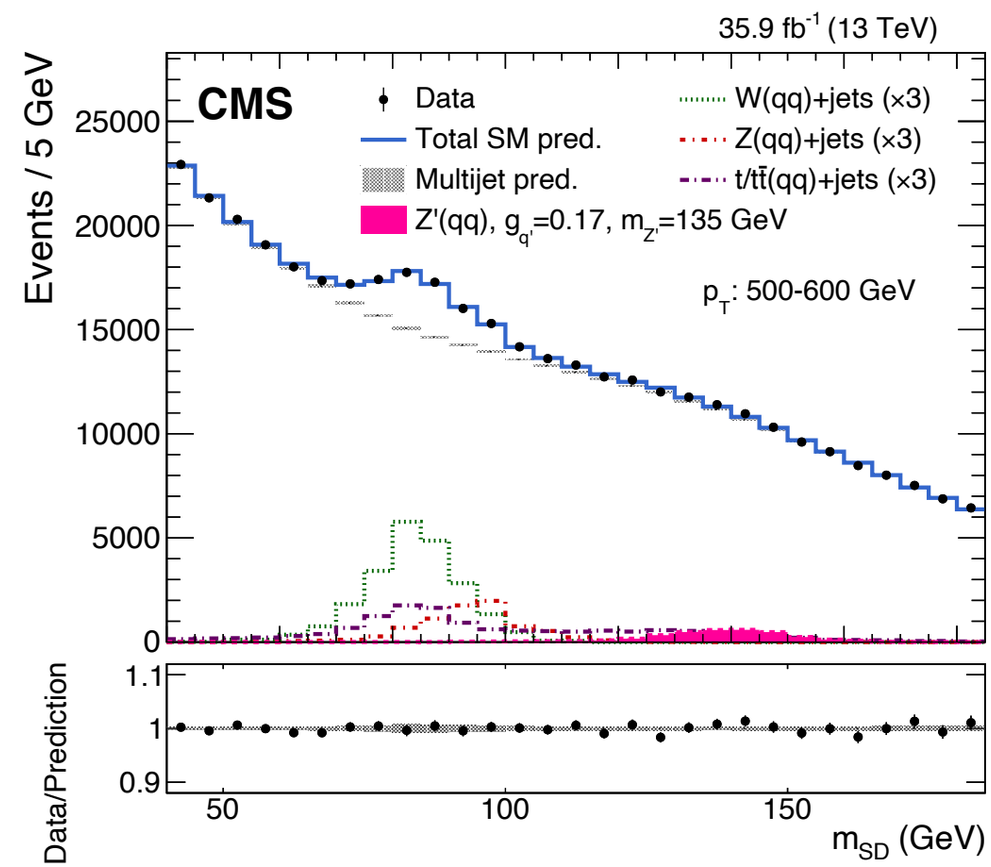


N_2^{DDT} in the Resonance Search

- N_2^{DDT} used in the CMS search for low mass resonances decaying into $q\bar{q}$ [JHEP **1801** (2018) 097]
- Signal enhanced with the selection $N_2^{DDT} < 0$
- Background shape estimated from events failing the N_2^{DDT} selection
  - pass-to-fail ratio $R_{p/f}(\rho, p_T)$ constant by design: same mass shapes in the pass and the fail regions
  - $R_{p/f}(\rho, p_T)$ allowed to float slightly in the fit to data to accommodate residual data/MC differences
- Fit to the m_{SD} distributions to extract the signal strength

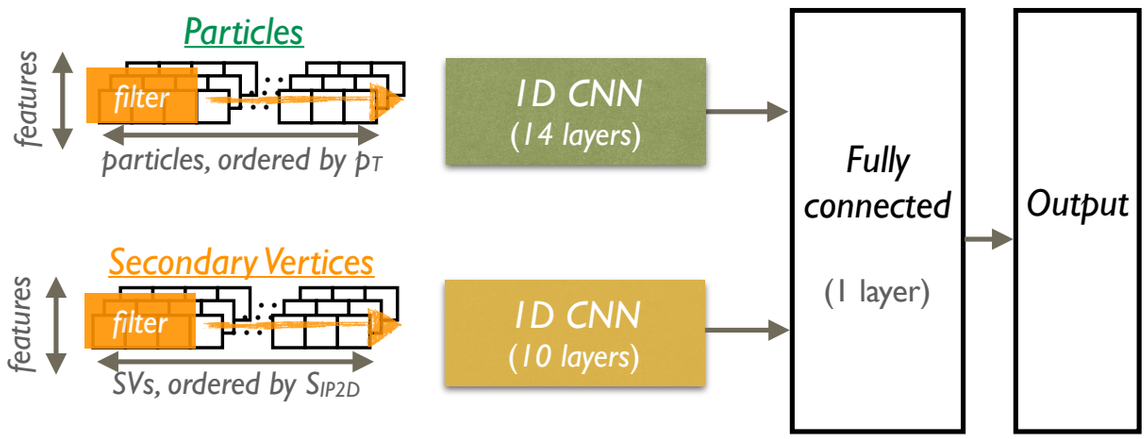






DDT for ML Taggers

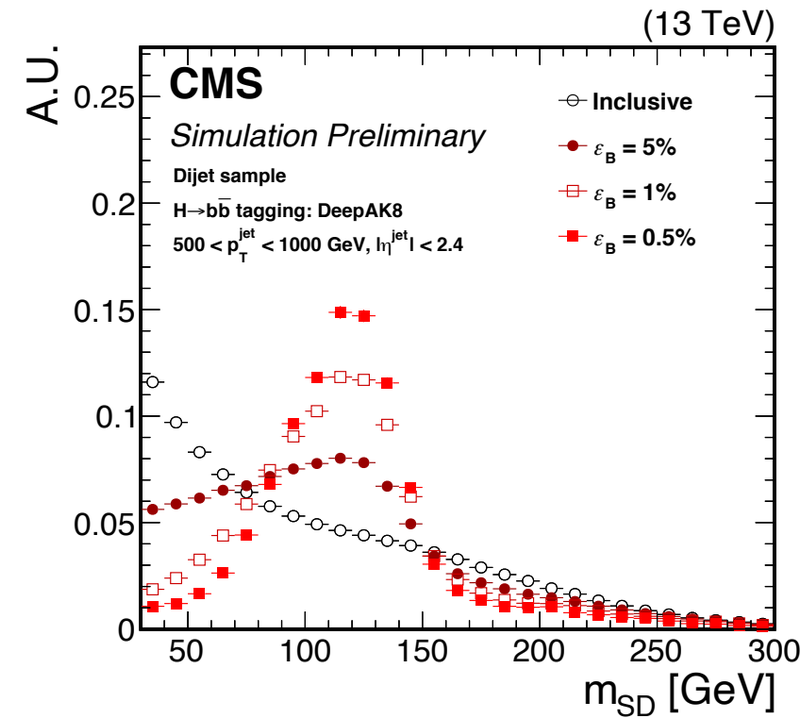
- DeepAK8
  - versatile multi-class tagger for t/W/Z/H tagging
  - directly uses jet constituents (PF candidates / secondary vertices)
  - 1D CNN based on the ResNet [arXiv: 1512.03385] architecture
  - significantly improved performance, but strong mass sculpting



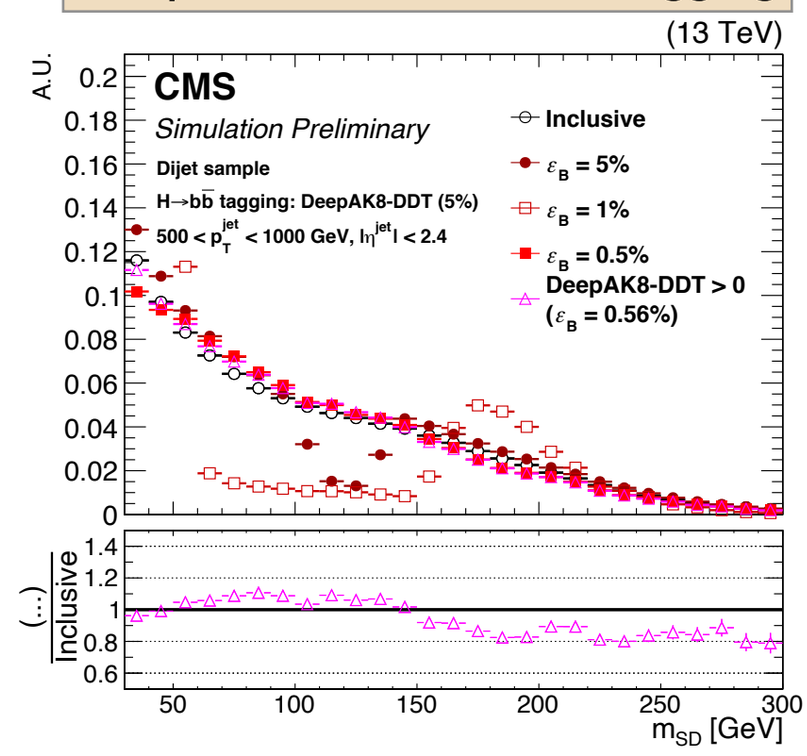
Category	Label
Higgs	H (bb)
	H (cc)
	H (VV*→qqqq)
Top	top (bcq)
	top (bqq)
	top (bc)
	top (bq)
W	W (cq)
	W (qq)
Z	Z (bb)
	Z (cc)
	Z (qq)
QCD	QCD (bb)
	QCD (cc)
	QCD (b)
	QCD (c)
	QCD (others)

- DeepAK8-DDT NEW
  - transforming the DeepAK8 scores following the DDT approach
  - good mass decorrelation achieved for the nominal working point (i.e., DeepAK8-DDT > 0)
  - unlike N_2^{DDT}, for ML taggers DDT achieves mass decorrelation only for the nominal WP
    - other selections show strong mass sculpting and cannot be used

DeepAK8: H→bb tagging



DeepAK8-DDT: H→bb tagging



CMS DP-2020/002

METHOD 2: MODIFY TRAINING PROCEDURE

- Mass decorrelation method 2: *the "active" way*

Modifying the training procedure/target
to prevent mass correlation

$L = L_{CE}$

Cross-Entropy loss

Modification →

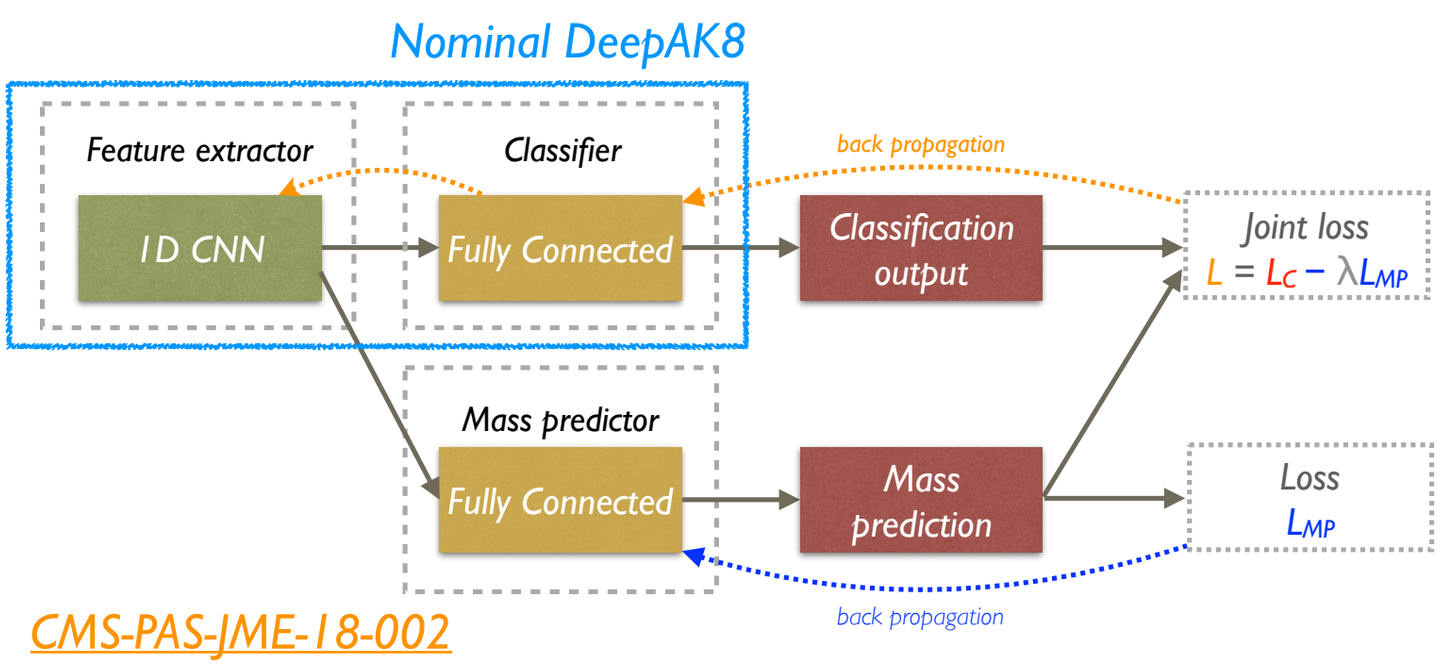
$L = L_{CE} + L_{MD}$

Cross-Entropy loss + *Mass-decorrelation loss*

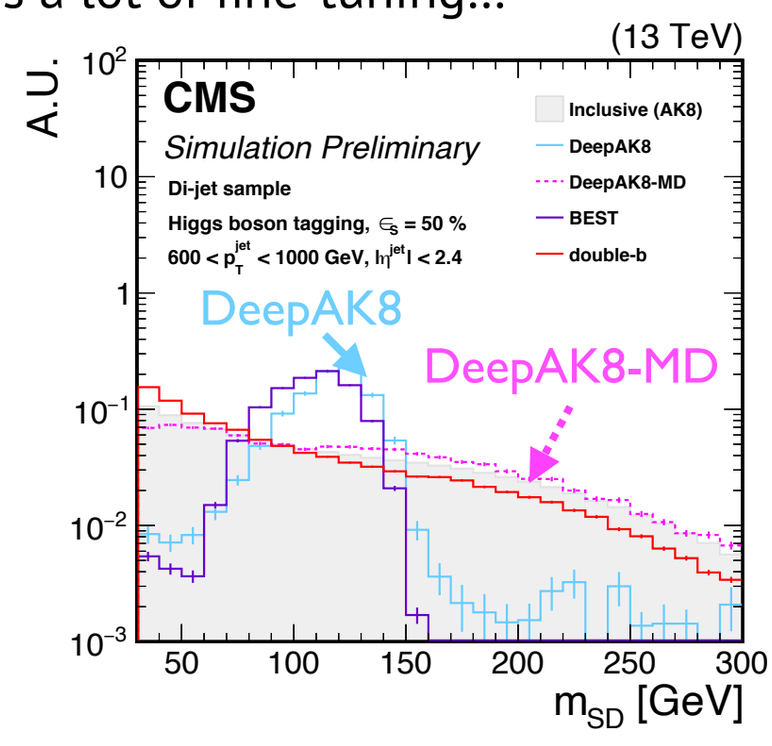
- Broadly speaking, this method involves choosing a metric to quantify the level of mass correlation and then minimize both the classification loss and this mass correlation metric
  - mass correlation can be measured with a number of metrics
    - KL divergence of the pass / fail mass shapes (e.g., CMS DeepDoubleB/C [CMS-DP-2018-046])
    - mutual information
    - a neural network — the GAN approach (e.g., CMS DeepAK8-MD)
    - ...

DeepAK8-MD

- DeepAK8-MD: mass-decorrelation using adversarial training [1611.01046]
  - added a mass prediction network to predict the jet mass from the learned features
    - higher mass prediction accuracy -> stronger correlation w/ the jet mass
  - accuracy of the mass prediction included in the loss function as a penalty
    - minimizing the joint loss -> improving classification accuracy while preventing mass correlation
  - in addition: signal/background samples reweighted to a ~flat (p_T, m_{SD}) distribution to aid the training
- The adversarial training approach works reasonably well
  - significantly reduced mass sculpting while still strong performance
  - however the training process is quite challenging and requires a lot of fine-tuning...

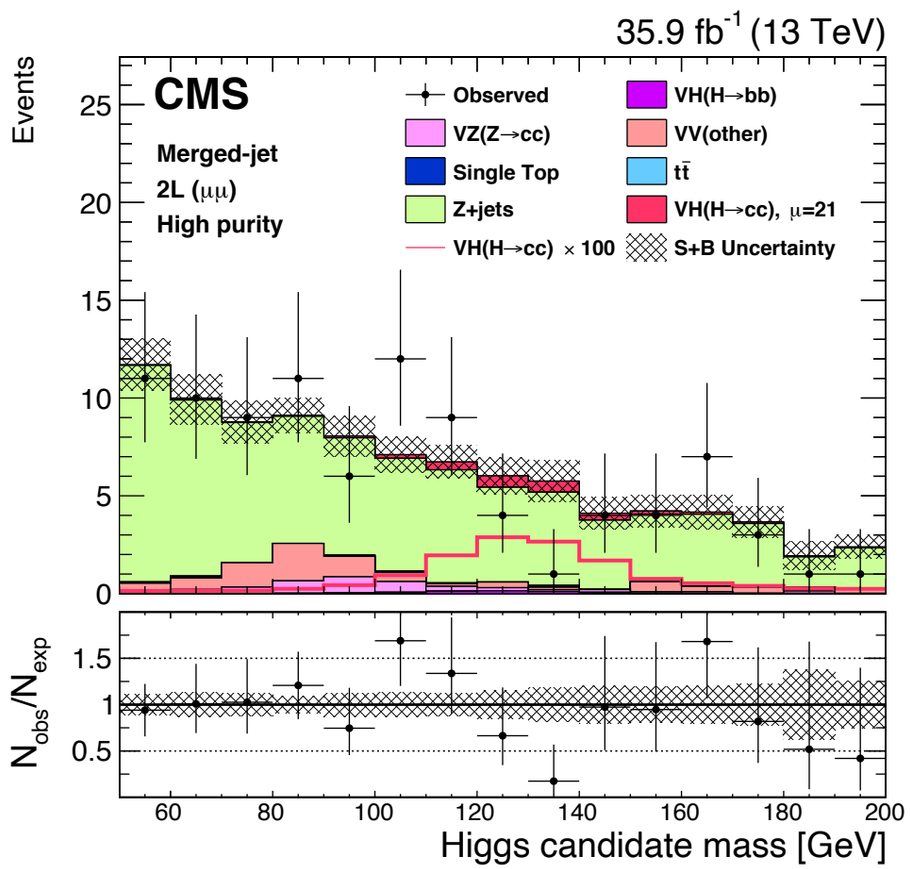


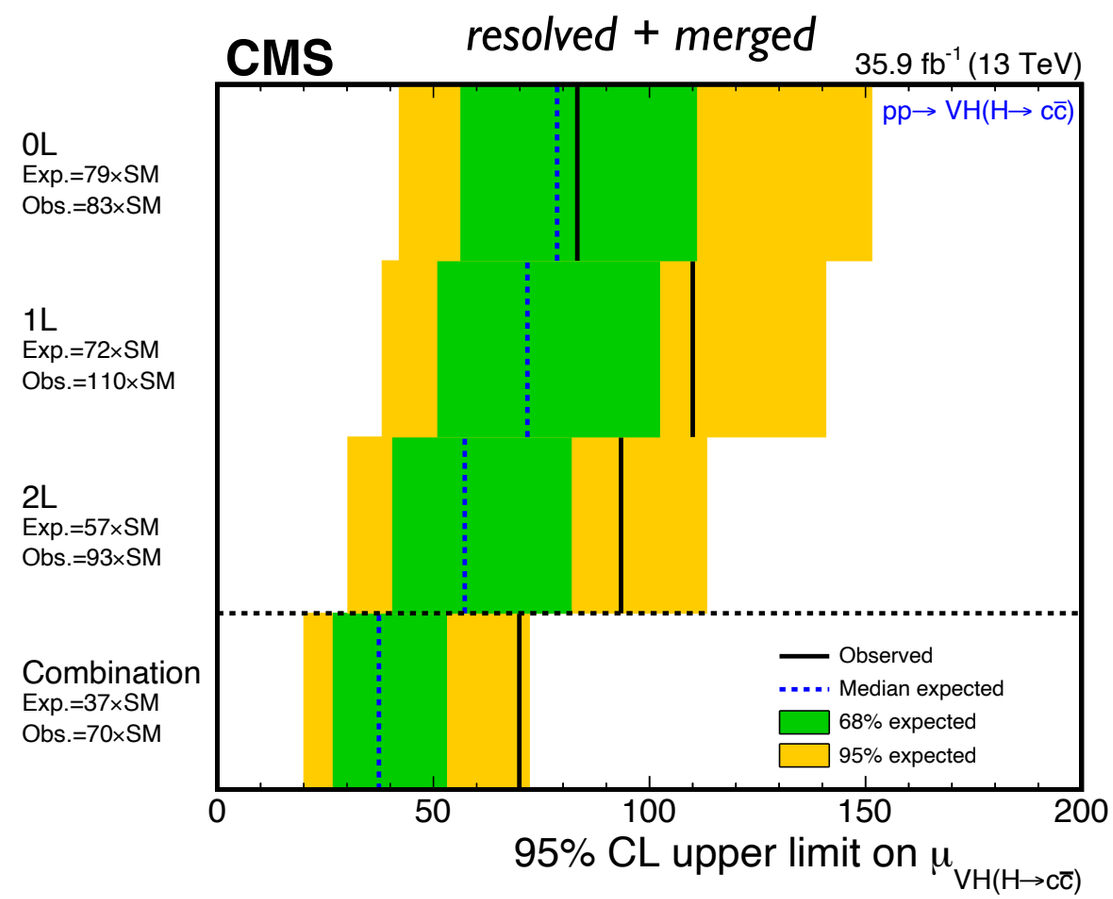
CMS-PAS-JME-18-002



DeepAK8-MD in the H→cc Search

- First direct search for H→cc in CMS [arXiv:1912.01662]
  - VH channel: V (W, Z) → ll, lν, νν
  - H→cc: resolved-jet topology + merged-jet topology
- DeepAK8-MD used in the merged-jet topology
  - adapted to R=1.5 jets (instead of R=0.8 jets) to increase acceptance at lower p_T (> ~200 GeV)
  - the DeepAK8-MD cc-tagging discriminant used to select cc-jet and suppress light-/bb-flavor jets
  - fit to the m_{SD} distribution to extract the H→cc signal

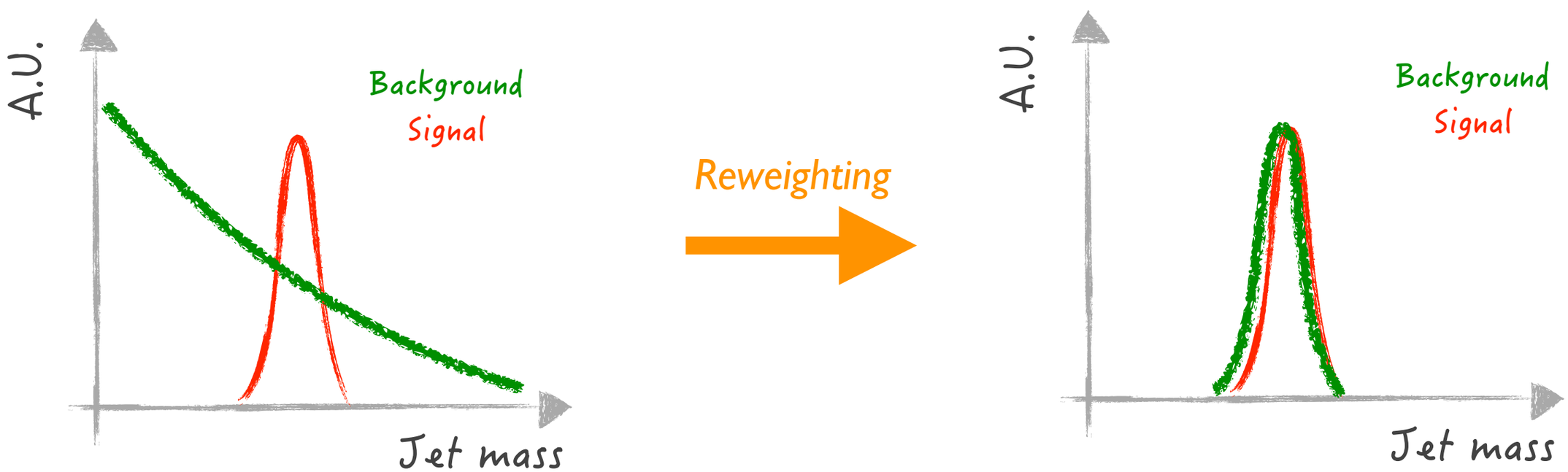




METHOD 3: REWEIGHT TRAINING SAMPLES

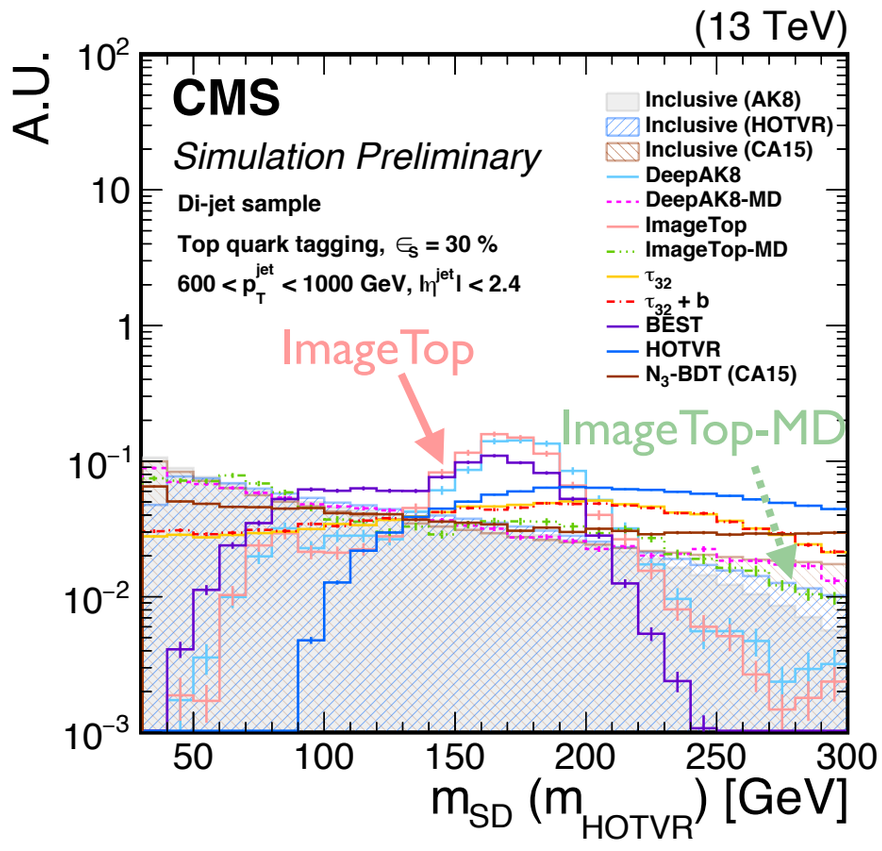
- For ML taggers, mass correlation arises because signal (t/W/Z/H) and background (QCD) jets have very different mass distributions
  - maximizing signal/background separation inevitably causes the tagger responses to depend on the jet mass
  - if signal and background jets have similar mass distributions, then mass sculpting simply cannot happen
- Mass decorrelation method 3: *the "passive" way*

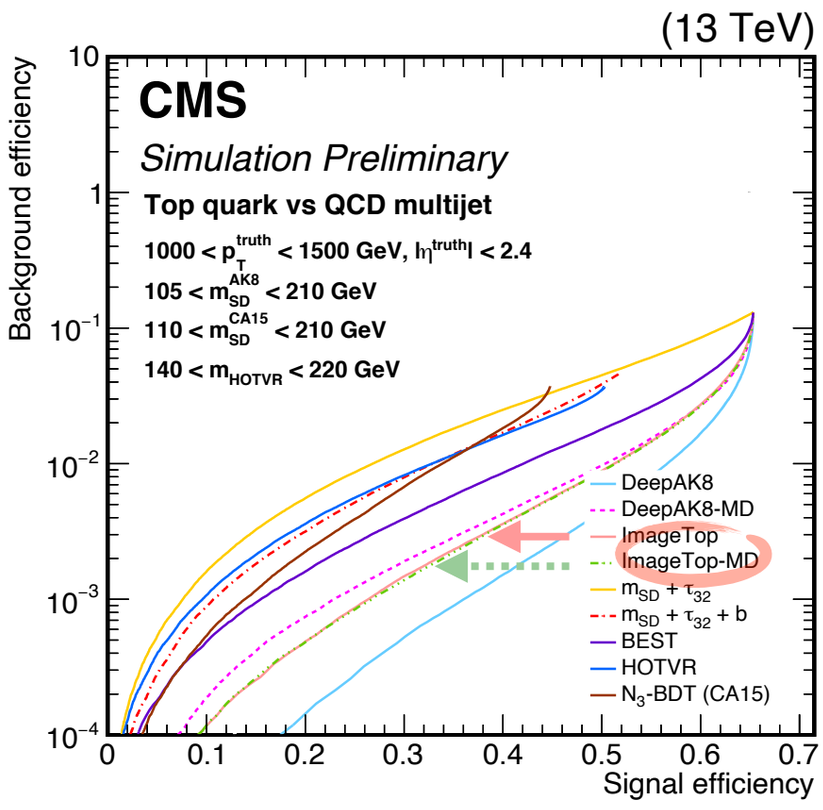
Reweighting the training samples such that signal and background jets have the same mass distributions



ImageTop-MD

- ImageTop-MD
  - ImageTop: top tagging algorithm based on jet images + subjet b-tagging
    - more details in Alejandro's talk
  - ImageTop-MD: mass decorrelated version by reweighting the QCD jets to match the mass shape of the top jets in the training
- The reweighting method achieves desired mass decorrelation
  - and almost no loss of performance



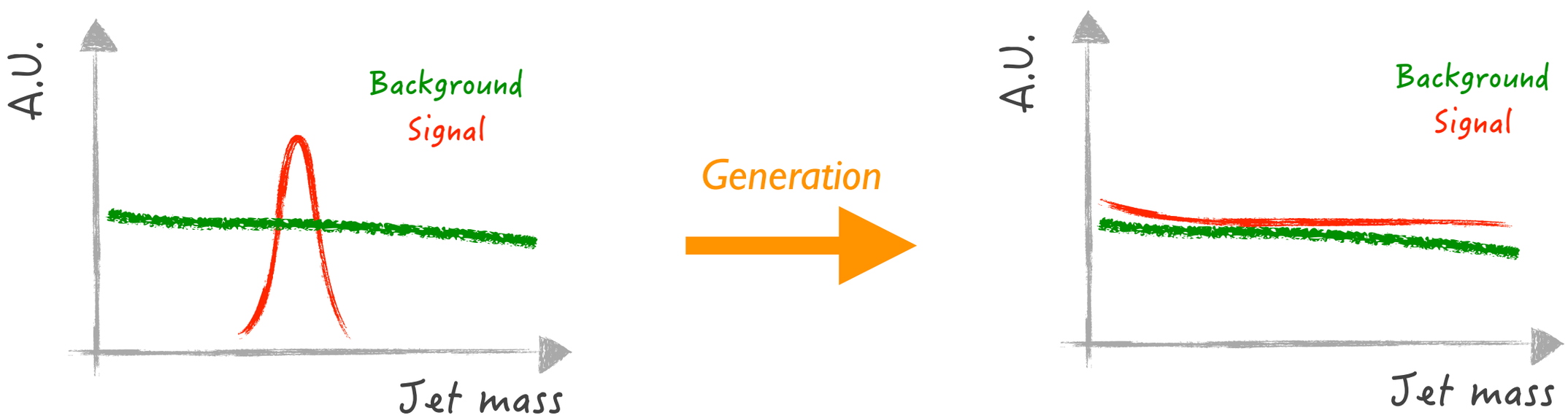


CMS-PAS-JME-18-002

METHOD 4: GENERATE SPECIAL TRAINING SAMPLES

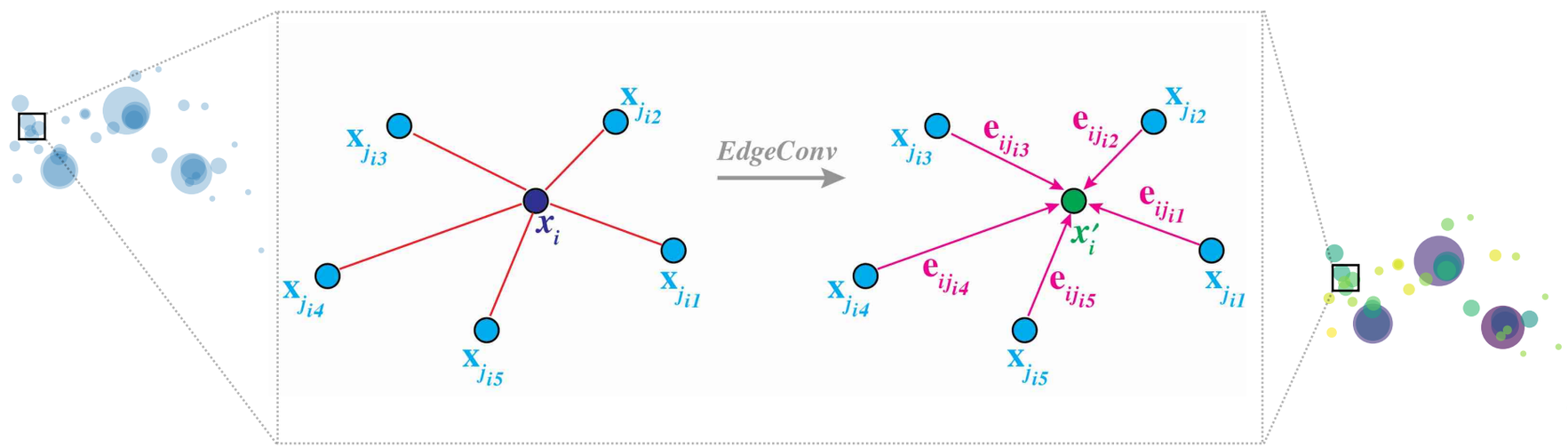
- The reweighting method works well for binary classification, but not sufficient for multi-class taggers
  - multiple signals, so cannot reweight the background mass shape to the signals
  - can possibly reweight everything to a flat / background-like mass distribution
    - but very low stats for signal away from the mass peak -> poor performance
- Instead of reweighting, can generate dedicated samples to populate the full mass range
- Mass decorrelation method 4: *the "actively-passive" way*

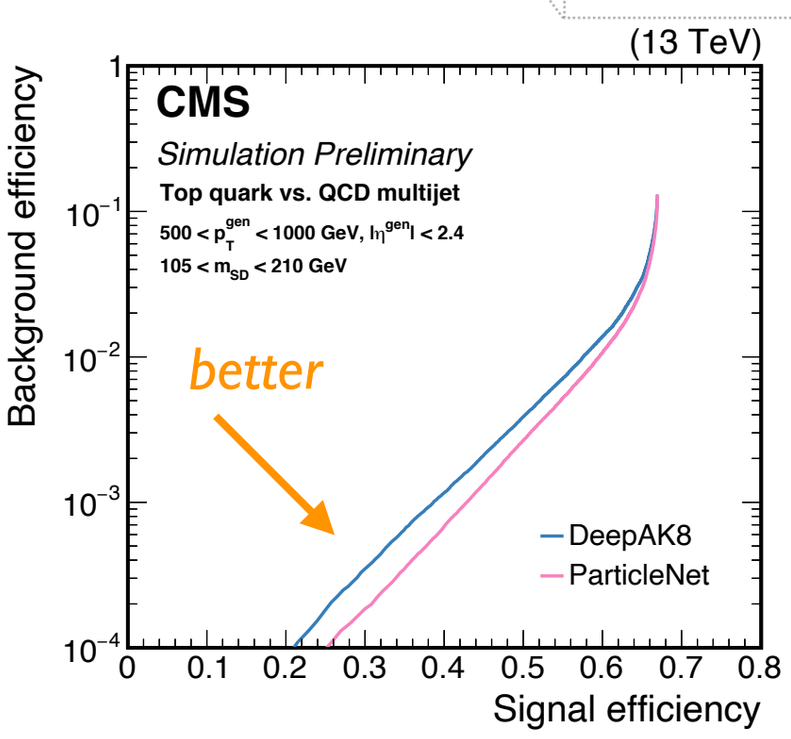
Generating a special training sample in which the signal particle has a flat mass distribution



PARTICLENET

- A graph neural network is explored with this mass decorrelation approach
- ParticleNet [arXiv: 1902.08570]
  - treating a jet as an unordered set of particles in space
  - using a permutation-invariant graph neural network architecture





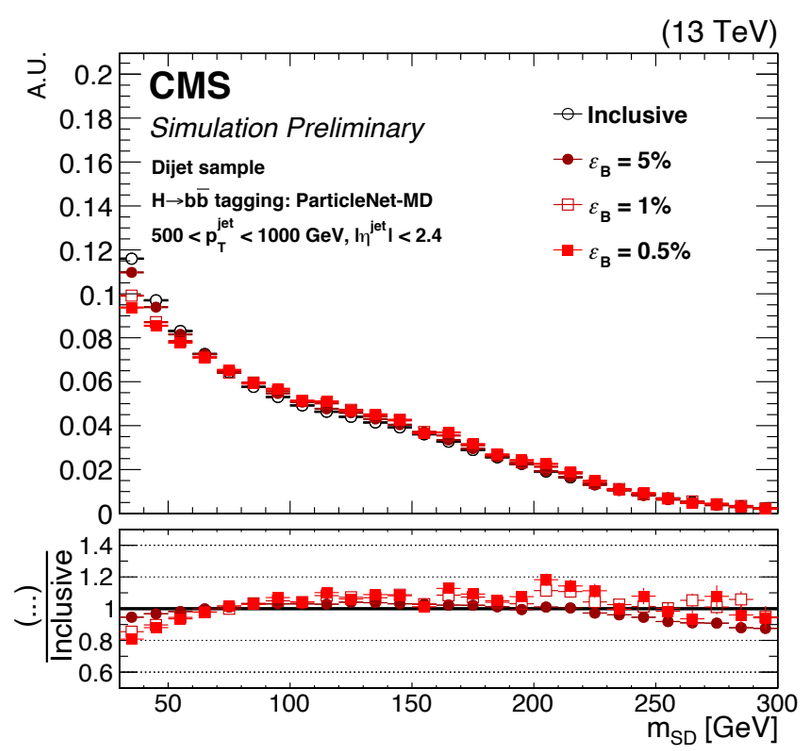
- The CMS implementation of ParticleNet NEW
  - multi-class tagger for t/W/Z/H tagging
  - same inputs as DeepAK8 (PF candidates/secondary vertices)
  - significant performance improvement

CMS DP-2020/002

PARTICLENET-MD

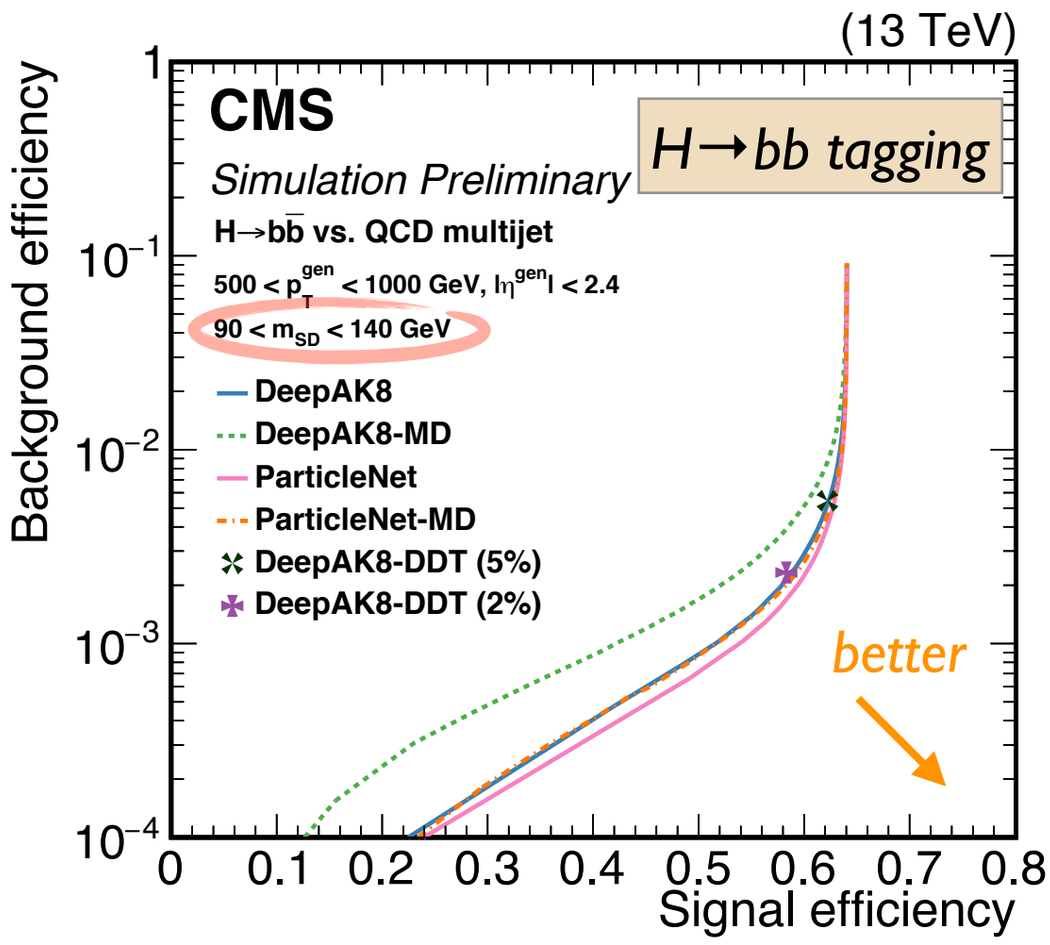
CMS DP-2020/002

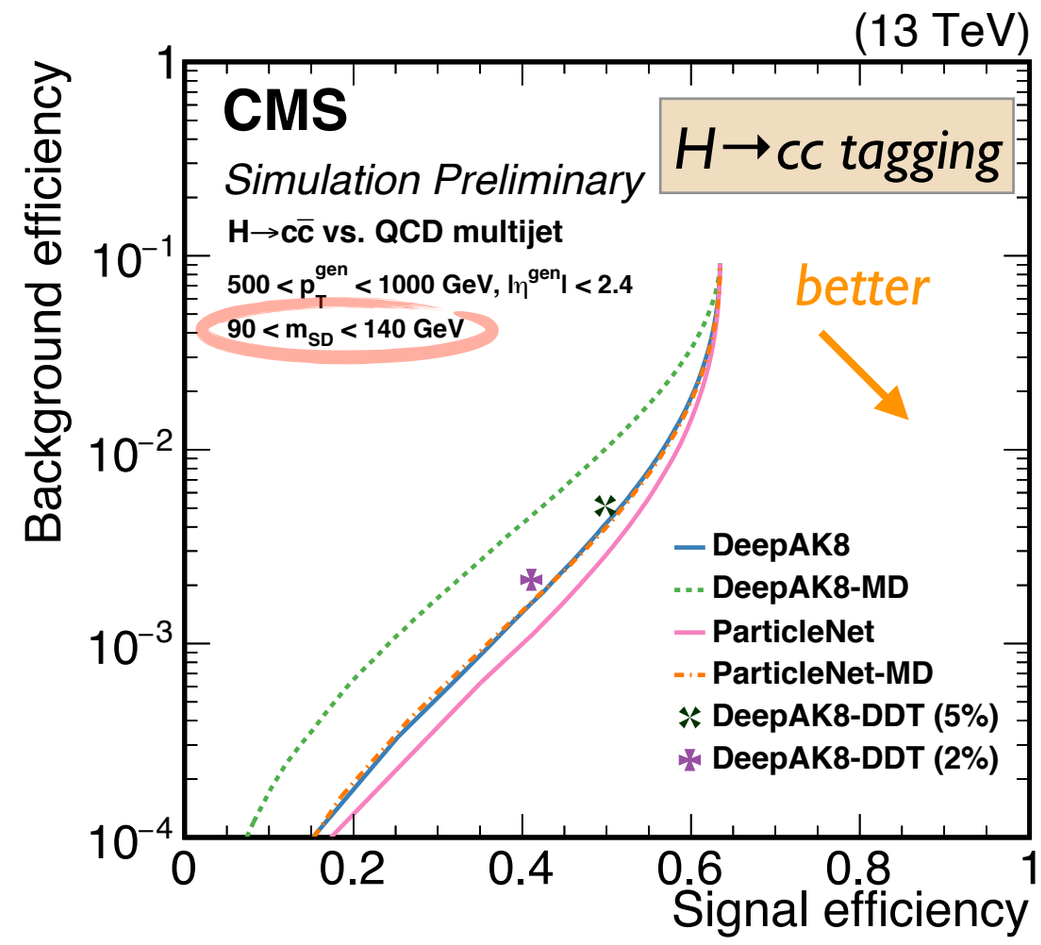
- ParticleNet-MD NEW
  - a generic mass-decorrelated 2-prong (W / Z / H / ...) tagger
    - w/ also flavour information: i.e., X->bb, X->cc and X->qq
  - trained using a dedicated signal sample
    - hadronic decays of a spin-0 particle X: $X \to b\bar{b}, X \to c\bar{c}, X \to q\bar{q}$
    - flat mass spectrum: $m_X \in [15, 250]$ GeV
  - signal and background further reweighted to a flat $[p_T, m_{SD}]$ distribution
  - using the ParticleNet graph neural network architecture
- Very good mass decorrelation with this approach
  - also very straightforward to train
    - no need to modify training procedure / loss



Comparison: Discrimination Power

- DeepAK8-DDT
  - two version investigated: constant background rate of 5% and 2%
  - performance close to (nominal) DeepAK8
- ParticleNet-MD
  - performance close to (nominal) DeepAK8: slightly better than DeepAK8-DDT
  - only small performance loss due to mass decorrelation compared to nominal ParticleNet



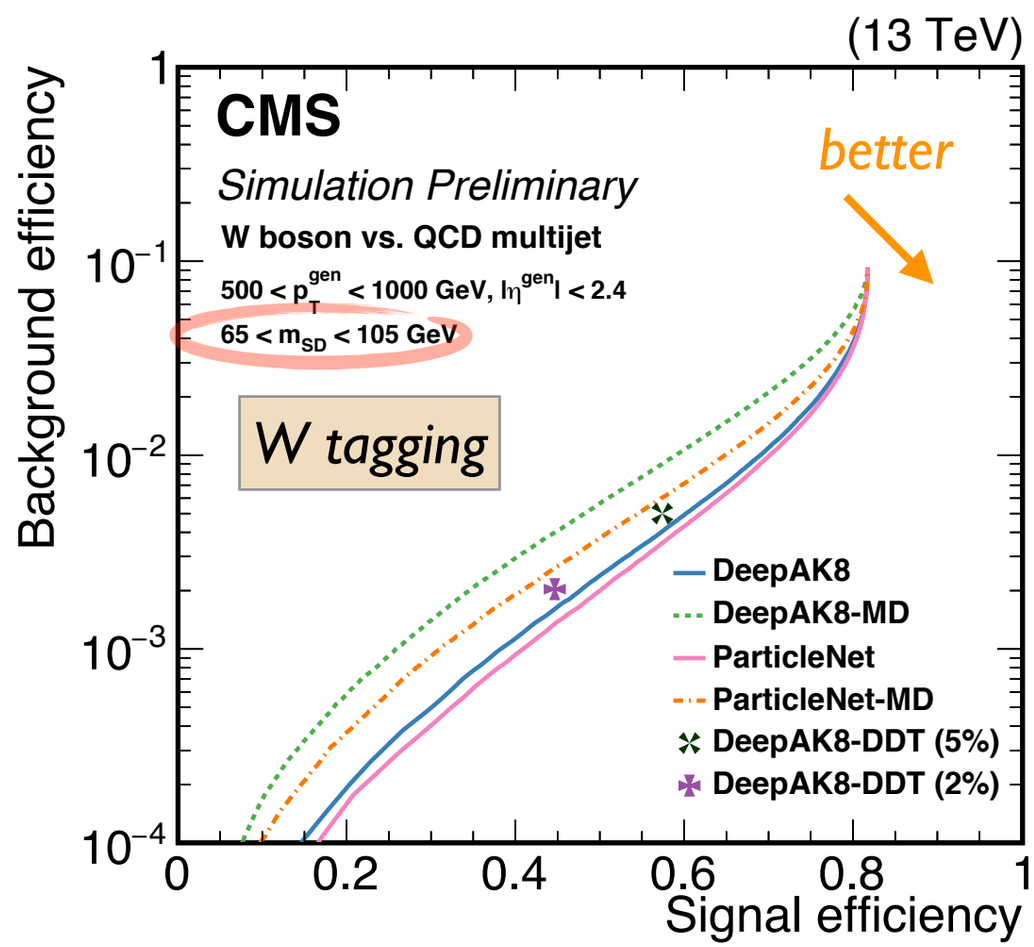


CMS DP-2020/002

Comparison: Discrimination Power (II)

- ParticleNet-MD

  - not trained on W decays; instead using an approximate discriminant:
    - [p(X→cc) + p(X→qq)] / [p(X→cc) + p(X→qq) + p(QCD)]
  - performance slightly weaker than DeepAK8-DDT but still much stronger than DeepAK8-MD



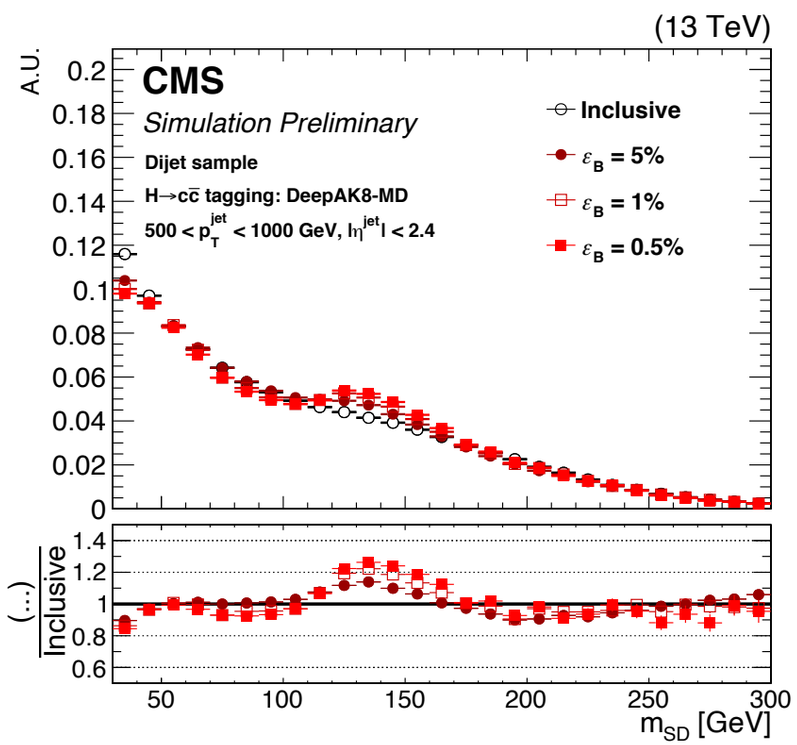
CMS DP-2020/002

Comparison: Mass Decorrelation

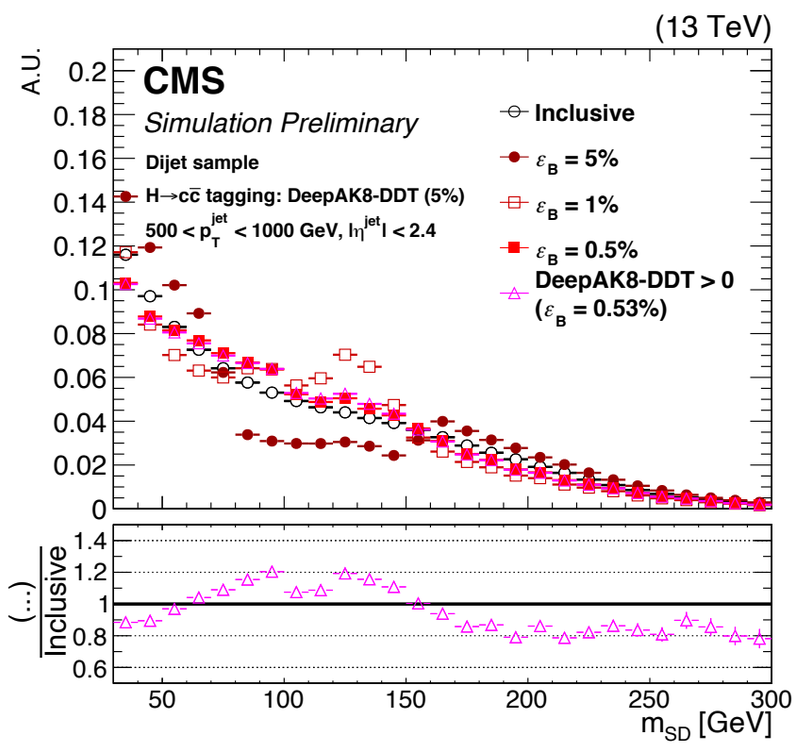
- Overall, good mass decorrelation for all methods
  - small residual mass sculpting for DeepAK8-MD
  - for DeepAK8-DDT, mass decorrelation only achieved for the nominal WP
  - ParticleNet-MD shows slightly better mass decorrelation

NEW

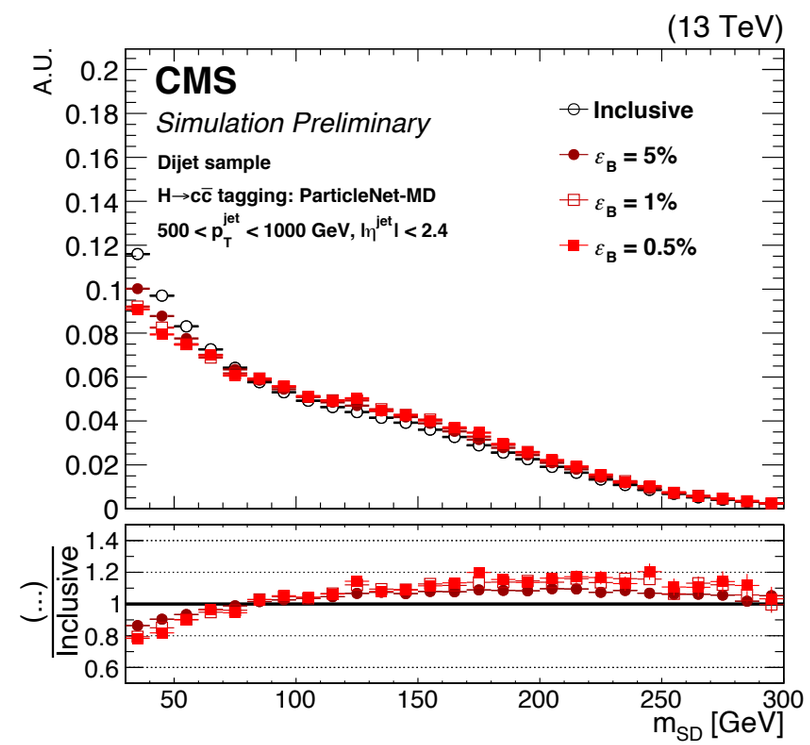
DeepAK8
H→cc tagging



DeepAK8-DDT (5%)
H→cc tagging



ParticleNet-MD
H→cc tagging



CMS DP-2020/002

SUMMARY

- Mass-decorrelated taggers useful for many physics analyses
- Several methods investigated in CMS to decelerate ML-based boosted jet taggers

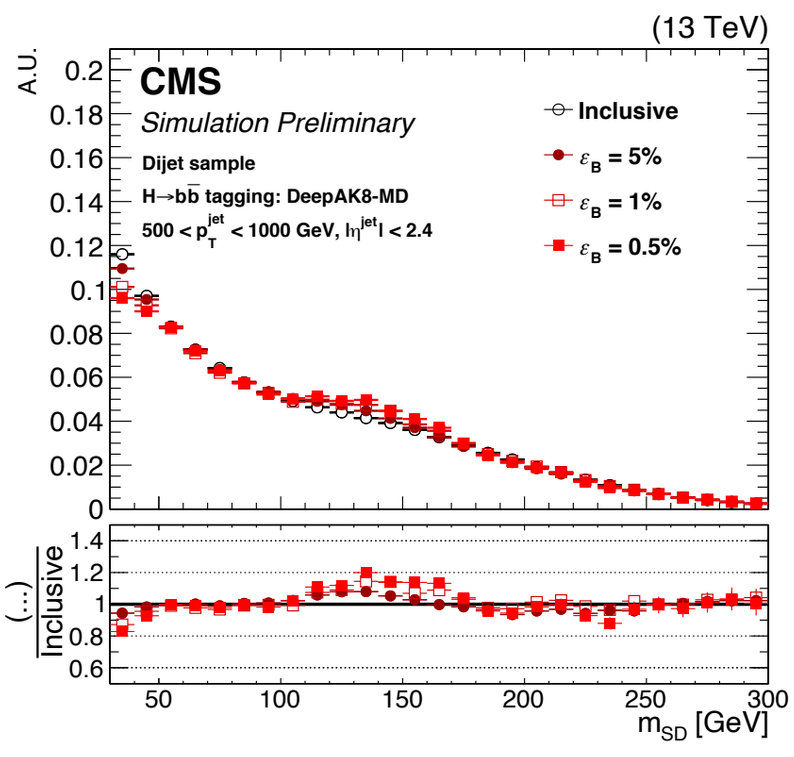
	Category	Method	Easy to implement	Mass decorrelation	Limitations
DeepAK8-DDT	t/W/Z/H-tagging	DDT	★★★	★★	*Mass decorrelation only for the nominal WP*
DeepAK8-MD	t/W/Z/H-tagging	adversarial training + reweighting	★	★★	*Very difficult to train*
ImageTop-MD	t-tagging	reweighting bkg. to signal	★★★	★★	*Only for binary classification*
ParticleNet-MD	2-prong tagging (w/ flavour)	special training samples	★★☆	★★★	*Need special samples; currently only for 2-prong tagging*

Backups

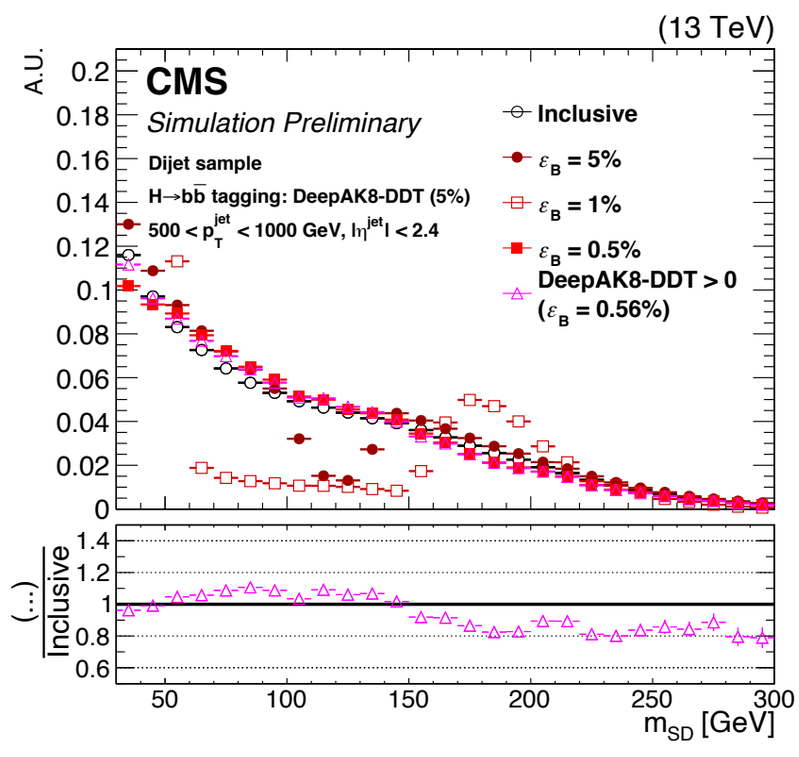
Comparison: Mass Decorrelation

NEW

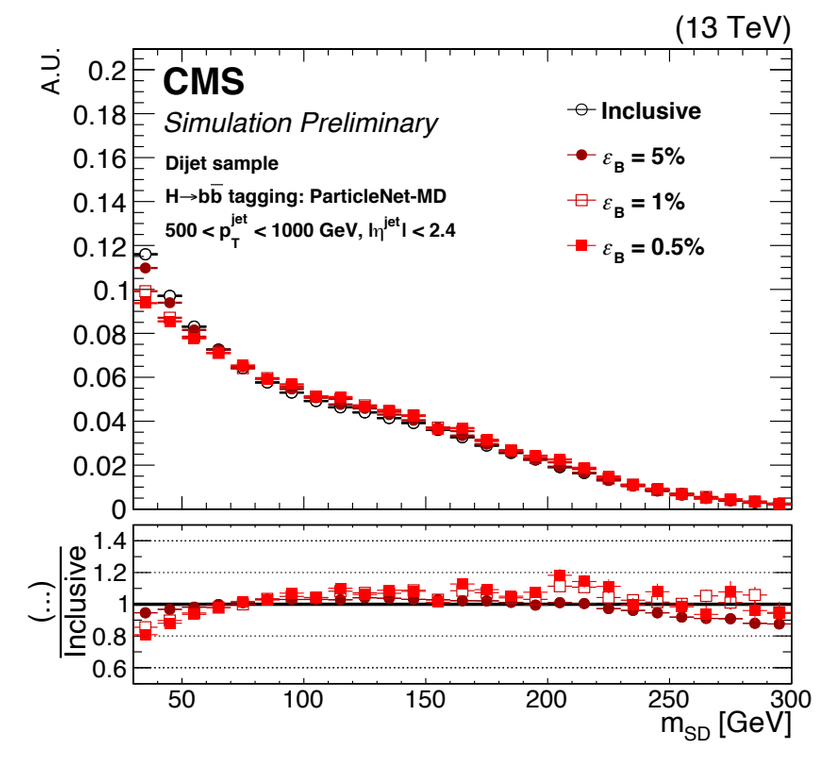
DeepAK8
H→bb tagging



DeepAK8-DDT (5%)
H→bb tagging

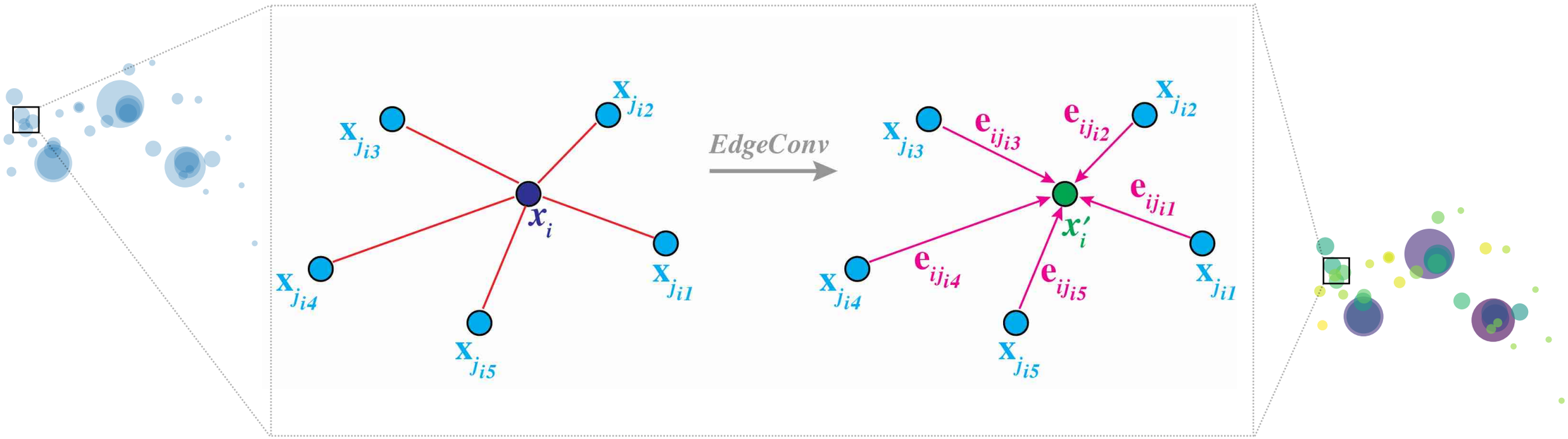


ParticleNet-MD
H→bb tagging



CMS DP-2020/002

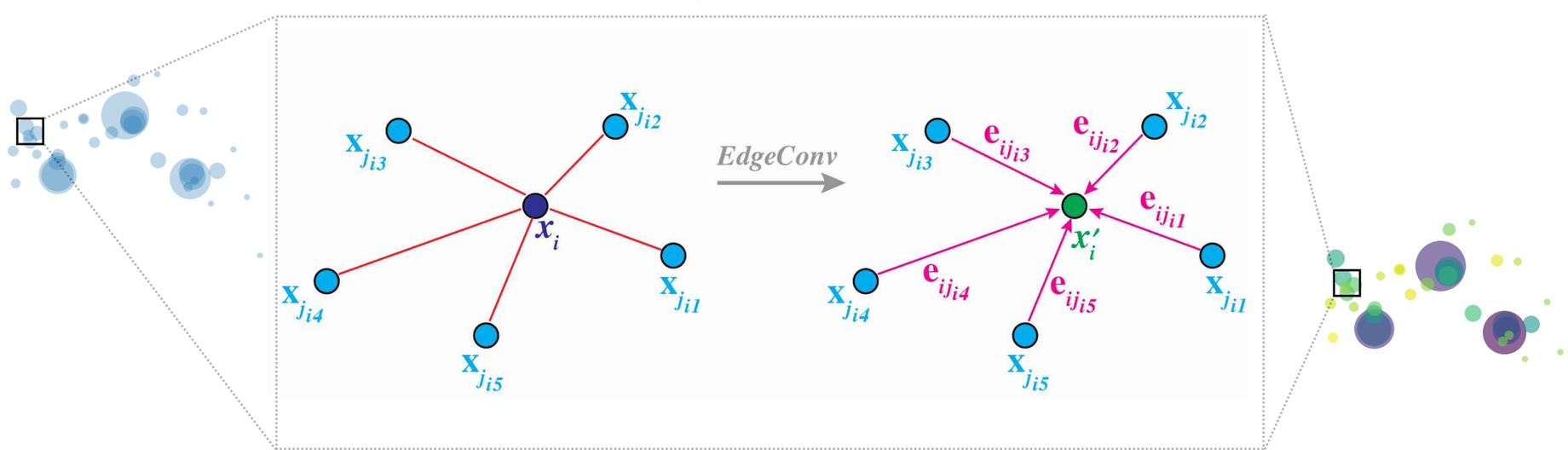
Convolution on Point Clouds



- Convolution on point clouds: *EdgeConv* [arXiv:1801.07829]
  - treating a point cloud as a graph: each point is a *vertex*
  - for each point, a local patch is defined by finding its *k-nearest neighbors*
  - designing a symmetric "convolution" function
    - define "edge feature" for each center-neighbor pair: $e_{ij} = h_\Theta(x_i, x_j)$
      - same h_Θ for all neighbor points, and all center points, for symmetry
    - aggregate the edge features in a symmetric way: $x_i' = \mathrm{mean}_j\, e_{ij}$

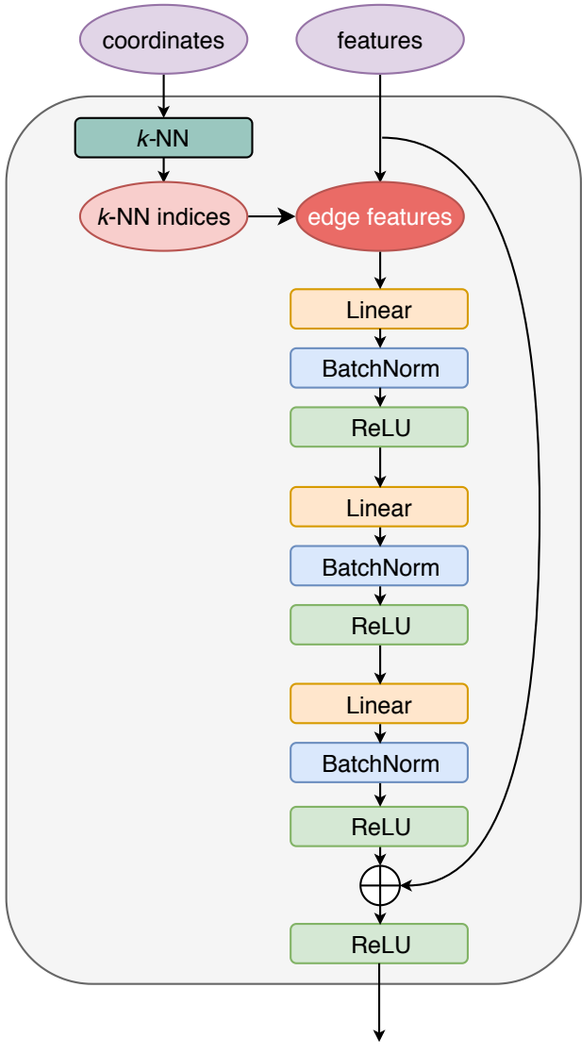
Dynamic Graph CNN

- *EdgeConv* shares many nice properties of regular CNNs
  - incorporates local neighborhood information
  - EdgeConv layers can be stacked to allow the network to learn both the local and global structures in a hierarchical manner
- Dynamic Graph CNN (DGCNN) [arXiv:1801.07829]
  - an EdgeConv operation outputs a set of learned features for each point
  - these features can also be interpreted as "coordinates" for each point (in a high-dim latent space)
  - such an interpretation enables us to recompute the "distances" between points and therefore dynamically update the k-nearest neighbor relations as going from one EdgeConv layer to another
  - found to be beneficial in the ML paper

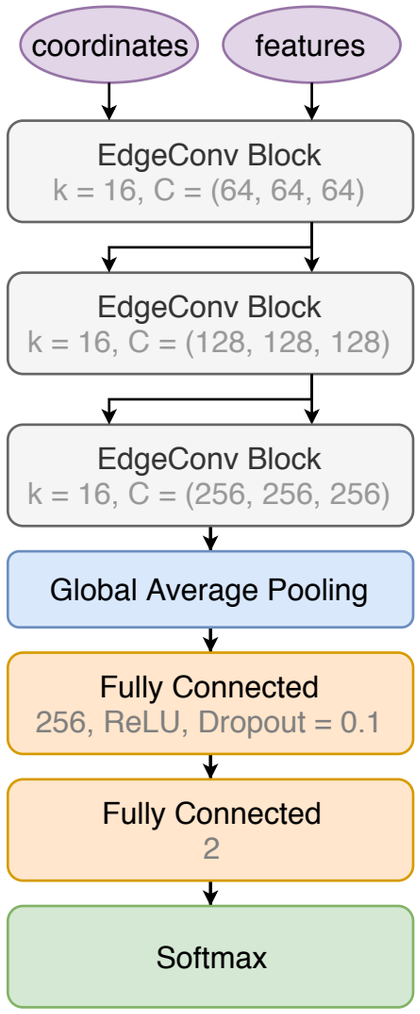


THE PARTICLENET ARCHITECTURE

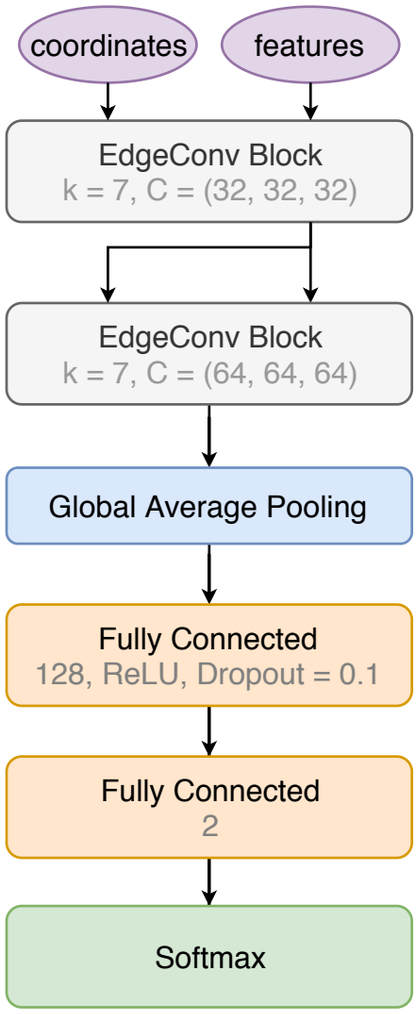
- Based on EdgeConv and DGCNN, we developed PARTICLENET, a customized architecture for jet tagging on particle clouds



EdgeConv block



ParticleNet architecture



ParticleNet-Lite

arXiv:1902.08570