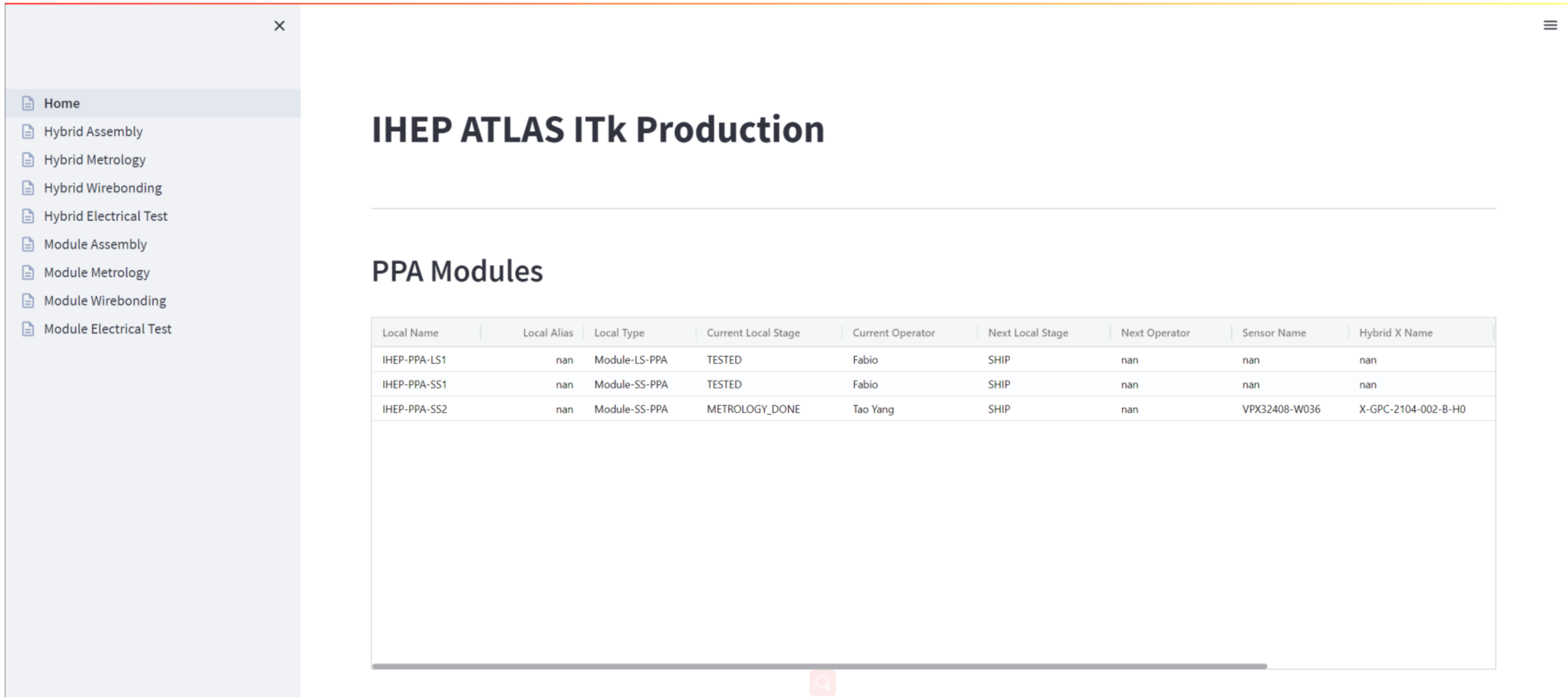
Weekly Report

Shudong WANG

Hardware

- Assembled IHEP-PPA-SS2

- Developing IHEP ATLAS ITk Production Management Page



- Home
- Hybrid Assembly
- Hybrid Metrology
- Hybrid Wirebonding
- Hybrid Electrical Test
- Module Assembly
- Module Metrology
- Module Wirebonding
- Module Electrical Test

IHEP ATLAS ITk Production

PPA Modules

Local Name	Local Alias	Local Type	Current Local Stage	Current Operator	Next Local Stage	Next Operator	Sensor Name	Hybrid X Name
IHEP-PPA-LS1	nan	Module-LS-PPA	TESTED	Fabio	SHIP	nan	nan	nan
IHEP-PPA-SS1	nan	Module-SS-PPA	TESTED	Fabio	SHIP	nan	nan	nan
IHEP-PPA-SS2	nan	Module-SS-PPA	METROLOGY_DONE	Tao Yang	SHIP	nan	VPX32408-W036	X-GPC-2104-002-B-H0

Constituent based W/Z boson tagging

- Introduction:
- Motivation: JetEtMiss group observed that the constituent based taggers showed impressive performance and best performance tagger is found (by far) for ParticleNet. Related to this, Xbb tagger team also observe better performance for GNN taggers compared to the current Xbb version, so a natural path is to try to see if this is also true for W and Z taggers.

- The conventional method for W/Z tagging:
- 1. Derive a set of high-level variables which summarize the information contained in the jet constituents.
- 2. Utilize these variables to perform cut based tagging or ML based tagging.

- The construction of these variables is almost always accompanied by information loss.

- Constituent based W/Z tagging:
- Combine and utilize the jet constituents' information using larger, complex and state-of-the-art ML/DL algorithms

Constituent based W/Z boson tagging

- Simulation Samples:
- Follow previous work on top tagging using UFO jet inputs: http://cdsweb.cern.ch/record/2777009/
- Boosted W/Z bosons are obtained from simulated W ’→WZ (→ $q\bar{q}q\bar{q}$) events with $m_{w'}$ = 2 TeV and cross section is reweighted to populate kinematic region of interest (0.2 TeV – 4 TeV)
- QCD jets obtained from events containing pairs of light quarks / gluons.
- Events are generated at leading-order with Pythia8 using the NNPDF2.3LO set of parton distribution functions and the A14 set of tuned parameters.
- Jet reconstruction, grooming, and truth labeling is identical to previous work (We just download their samples and use)

Jet requirements	W jet requirements	Z jet requirements
Jet $\|\eta\| < 2.0$ Jet $p_{\mathrm{T,truth}} > 200$ GeV Number of constituents $\geq$ 2 Jet mass > 40 GeV	dR(truth jet, MC truth W) < 0.75 Ungroomed truth jet mass > 50 GeV Number ghost associated b-hadrons == 1 Truth jet $\sqrt{d_{12}}$ > 55.25×exp(-2.34×10^{-3} × Jet $p_{\mathrm{T,truth}}$)	dR(truth jet, MC truth Z) < 0.75 Ungroomed truth jet mass > 50 GeV Truth jet $\sqrt{d_{12}}$ > 55.25×exp(-2.34×10^{-3} × Jet $p_{\mathrm{T,truth}}$)

Constituent based W/Z boson tagging

- Simulation Samples:
- The samples contain the flat ntuple (i.e. 1 jet / entry)
- Each jet has some high-level variable info (e.g. pt, eta, phi, D2, numConstituents, etc.) and 4 vector for the jet constituents are also stored (η, ϕ, p_T, E), we foucus on constituent level information now.

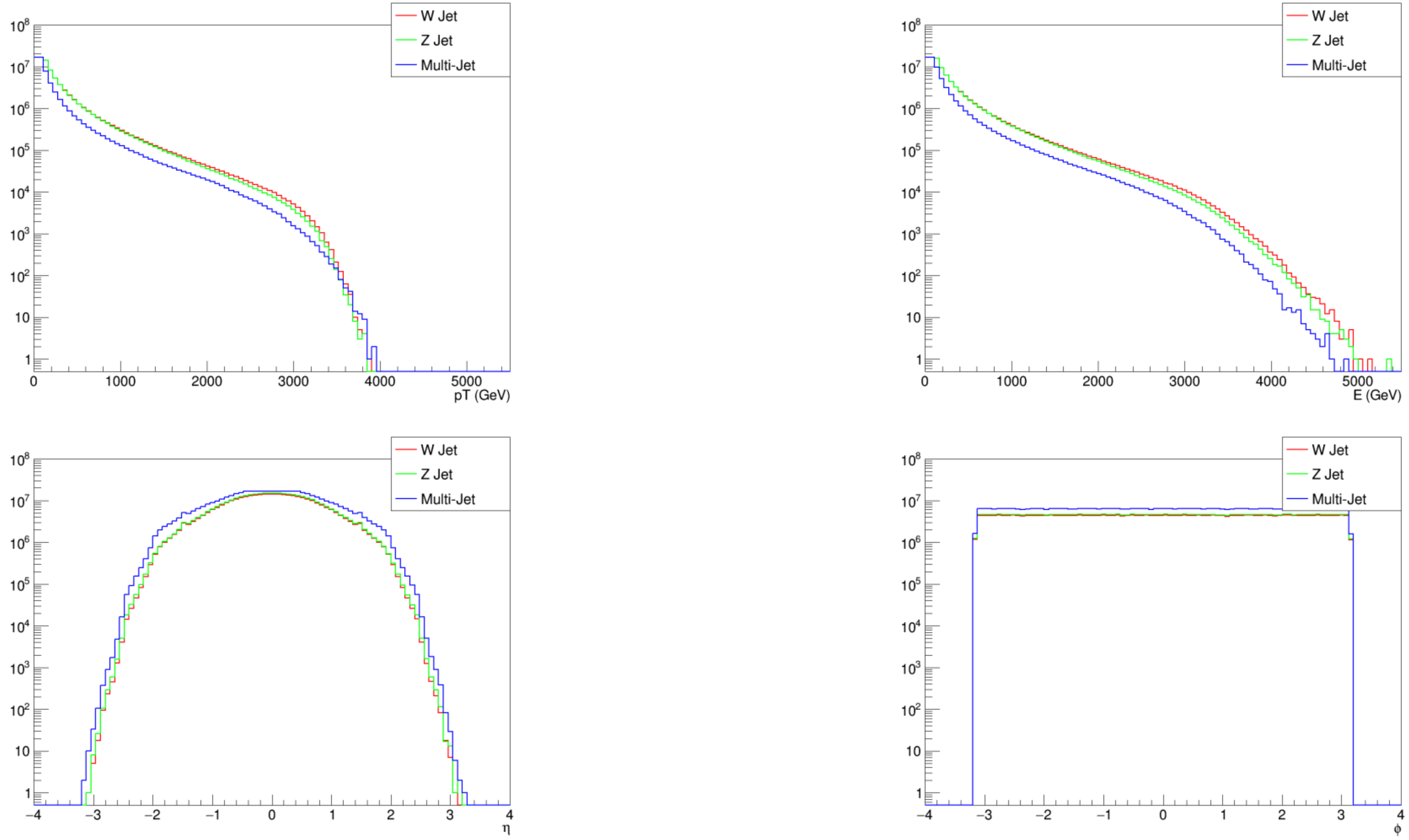
- Data process flow:
- Flat ntuple -> labeling (according to truth jet definitions mentioned in the previous slide)

  -> sampling (let #sig:#bkg == 1:1 to avoid *class-imbalance*) -> reweighting (according to pT)

  -> pre-processing(trim off undesired branches & add new vars) -> NN (ParticleNet) training

Constituent based W/Z boson tagging

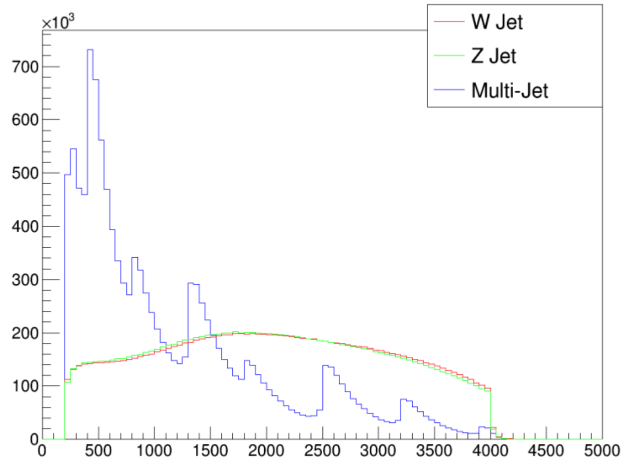
- Simulation Samples:
- Distributions of the of the jet constituent p_T, E, η, and ϕ.

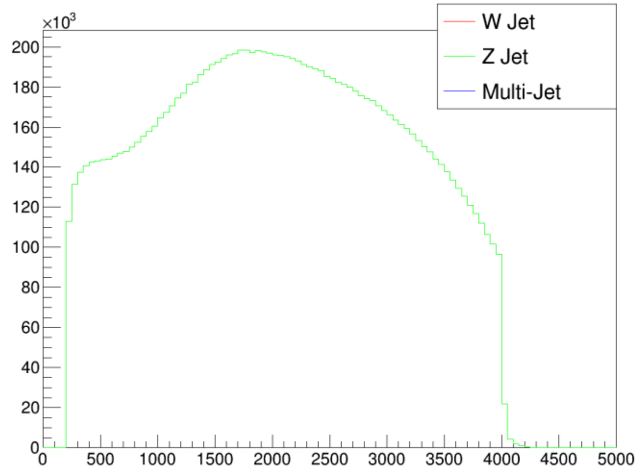


Constituent based W/Z boson tagging

- Simulation Samples:
- Jet pT and Training Weights
- Raw di-jet sample contains unphysical pT spectrum
- Derive weights which match background pT spectrum to signal

  - Prevents the tagger from associating signal jets with a particular pT, and assigning high scores to background jets which happen to have this pT.
- Weights are applied to loss function in network training



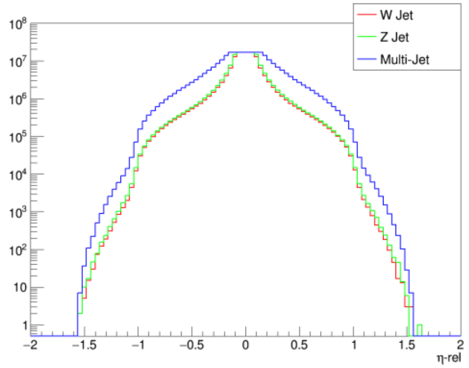


Constituent based W/Z boson tagging

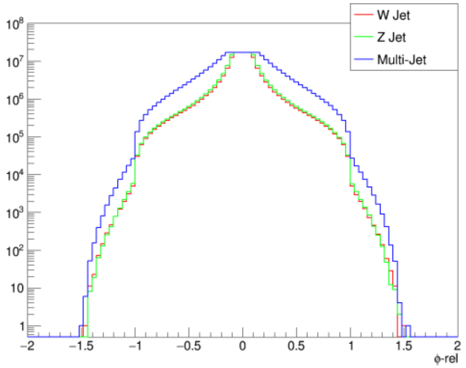
- Processing Samples:
- Constituent Level Pre-Processing:
- $\Delta\eta$ Difference in pseudo-rapidity between the particle and the jet axis
- $\Delta\phi$ Difference in azimuthal angle between the particle and the jet axis
- $\ln p_{\mathrm{T}}$ Logarithm of the particle’s pT
- $\ln E$ Logarithm of the particle's energy
- $\ln \frac{p_{\mathrm{T}}}{\sum_{\mathrm{jet}} p_{\mathrm{T}}}$ Logarithm of the particle’s pT relative to the total pT in jet
- $\ln \frac{E}{\sum_{\mathrm{jet}} E}$ Logarithm of the particle's energy relative to the total energy in jet
- ΔR Angular separation between the particle and the jet axis $\sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$

Constituent based W/Z boson tagging

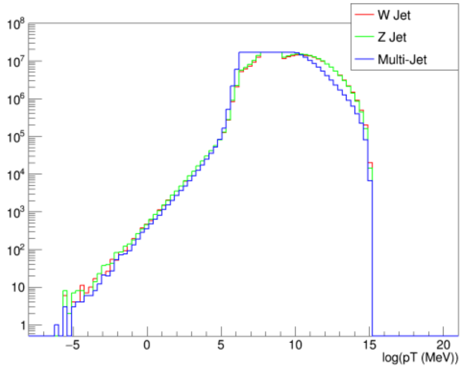
- Processing Samples:
- Distributions of the seven constituent-level quantities used as inputs to tagger



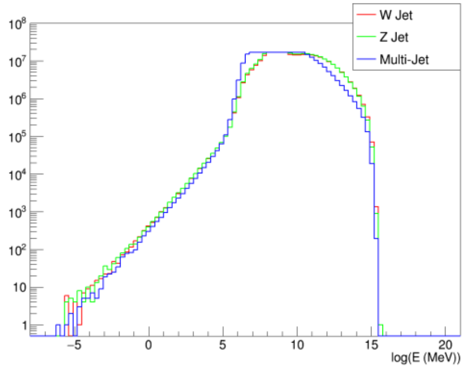
$\Delta\eta$



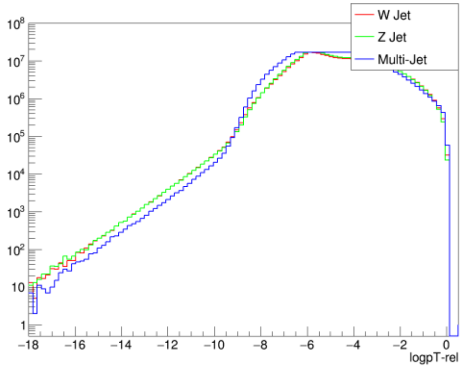
$\Delta\phi$



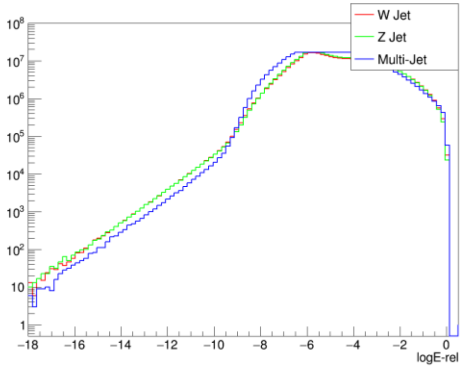
$\ln p_{\mathrm{T}}$



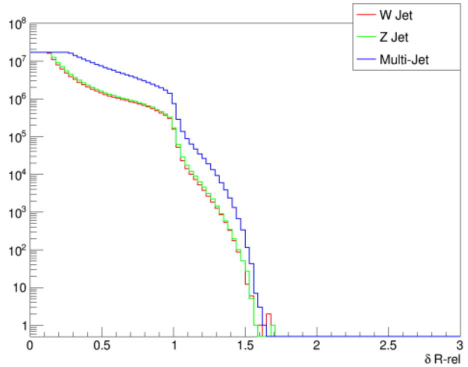
$\ln E$



$\ln \frac{p_{\mathrm{T}}}{\sum_{\mathrm{jet}} p_{\mathrm{T}}}$



$\ln \frac{E}{\sum_{\mathrm{jet}} E}$



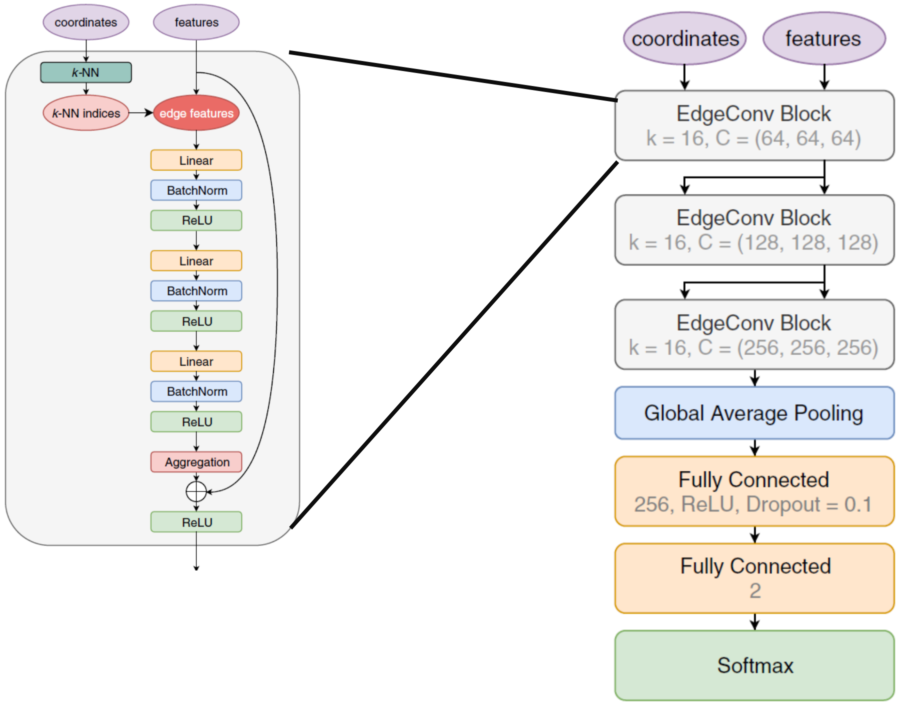
ΔR

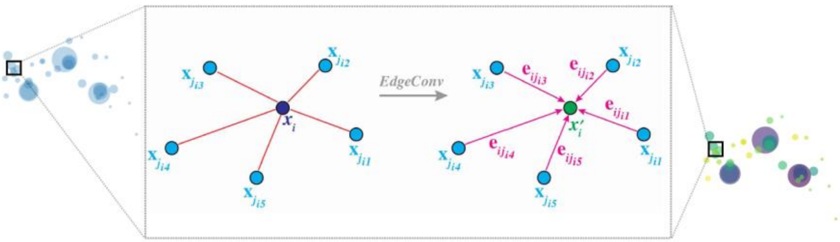
Constituent based W/Z boson tagging

- Training:
- Using ParticleNet, 1250W events (jets) for each class
- Train : Val : Test = 6: 2: 2
- use $(\Delta\eta, \Delta\phi)$ as coordinates to compute distance between particles

- **Using default setup of ParticleNet:**
- Number of EdgeConv blocks:3
- k nearest neighbors: 16
- Number of channels C = (C1, C2, C3) for each EdgeConv block

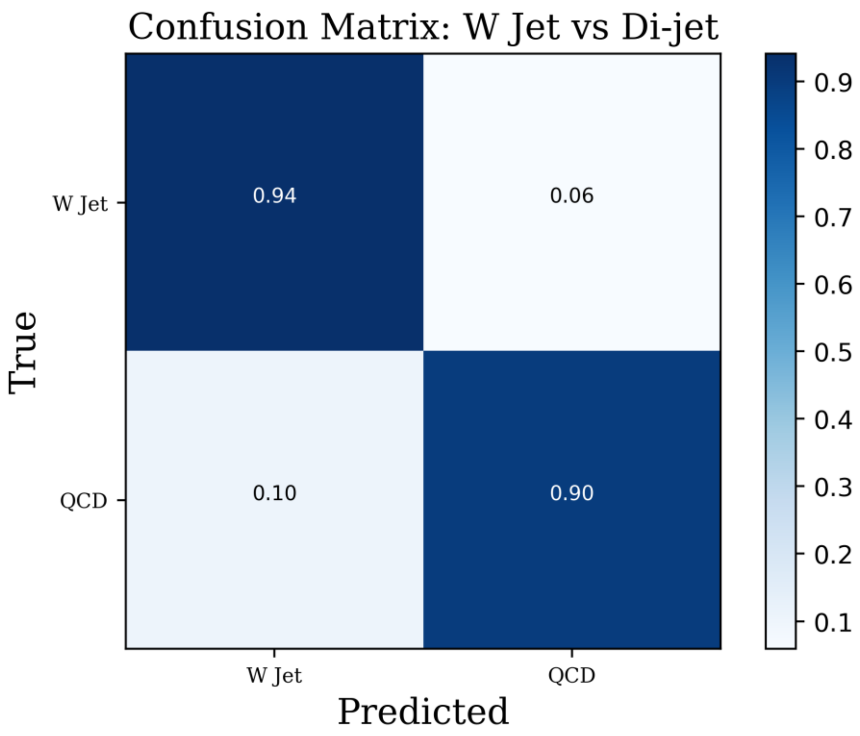
  is (64, 64, 64), (128,128, 128), and (256, 256, 256)
- Nodes of fully connected layer: 256
- Dropout probability: 0.1
- Starting learning rate: 5 x 10-3
- Learning rate scheduler: flat+decay
- Batch Size: 1000

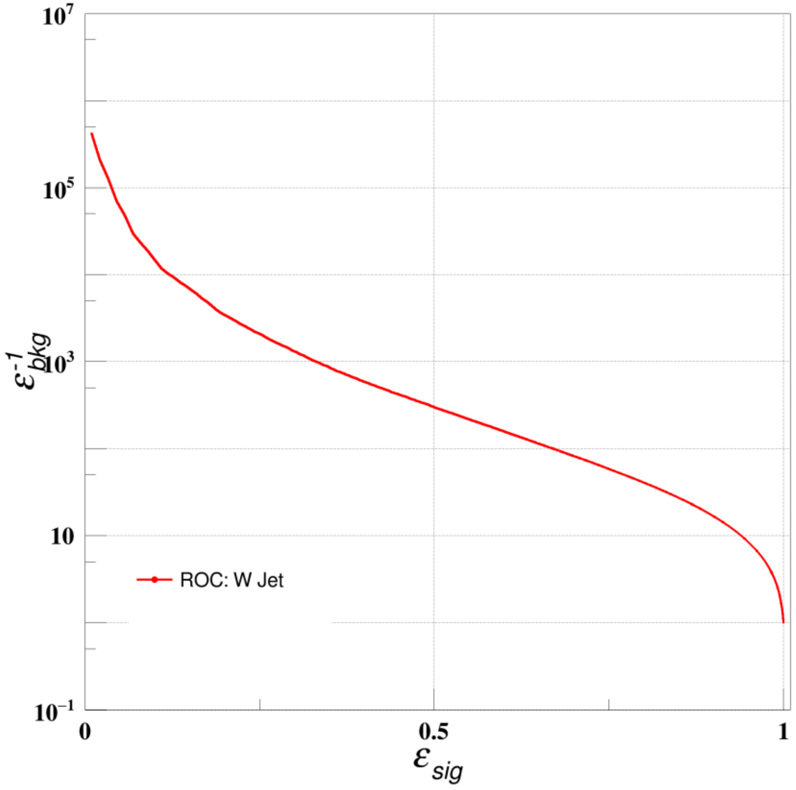




Constituent based W/Z boson tagging

- Preliminary result from ParticleNet:



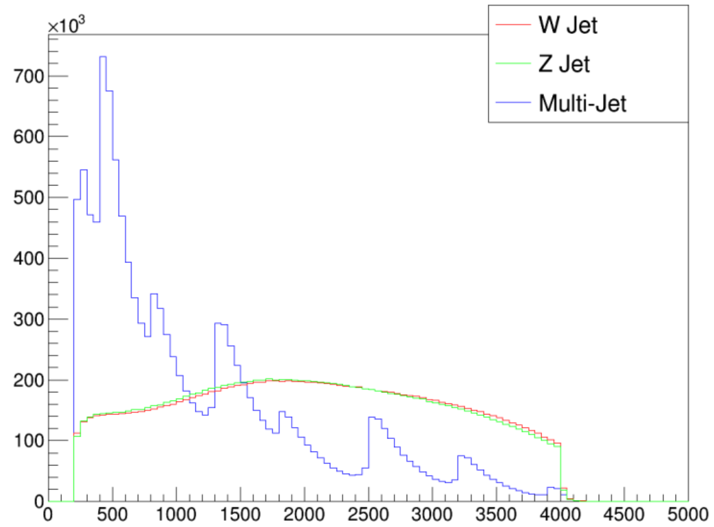


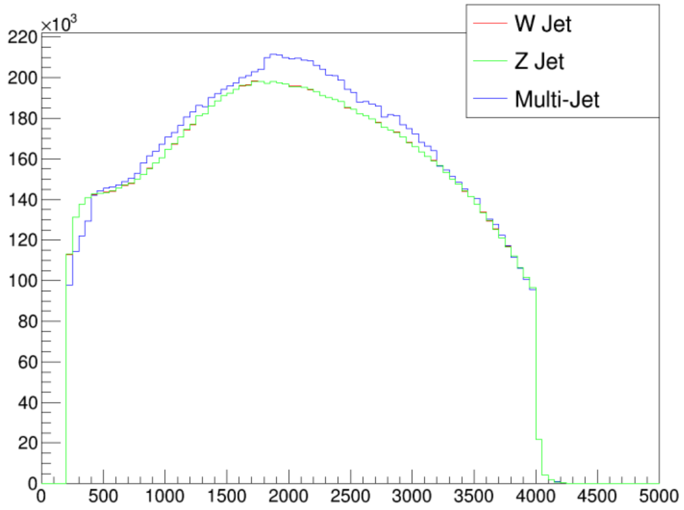
AUC: 0.975, ACC: 0.917

Constituent based W/Z boson tagging

- Problem found (biased sampling)

Result of biased sampling ?





- Weights gained by using all sig(13M+) & bkg (1.2B+) events, then do the sampling to get 1:1 sig/bkg ratio (12.5M: 12.5M) and apply weights*sf as training weights.
- Won't affect the result much, but need to be improved.