Applications of Deep Machine Learning in Collider Physics

HPNP2021
"Higgs as a Probe of New Physics" Special Edition 2021
25–27 March 2021, Osaka University, Japan

Cheng-Wei Chiang
National Taiwan University
National Center for Theoretical Sciences

Outline

- Brief introduction to deep machine learning
- Example 1: tagging boosted W/Z jets[†]
  - Motivation
  - Our taggers and performance
- Example 2: Drell-Yan processes[‡]
  - Motivation
  - Our taggers and performance
- Summary

[†] Yu-Chen Janice Chen, CWC, Giovanna Cottin, David Shih
PRD 101 (2020) 5, 053001 (1908.08256 [hep-ph])
[‡] Spencer Chang, Ting-Kuo Chen, CWC
PRD 103 (2021) 3, 036016 (2007.14586 [hep-ph])

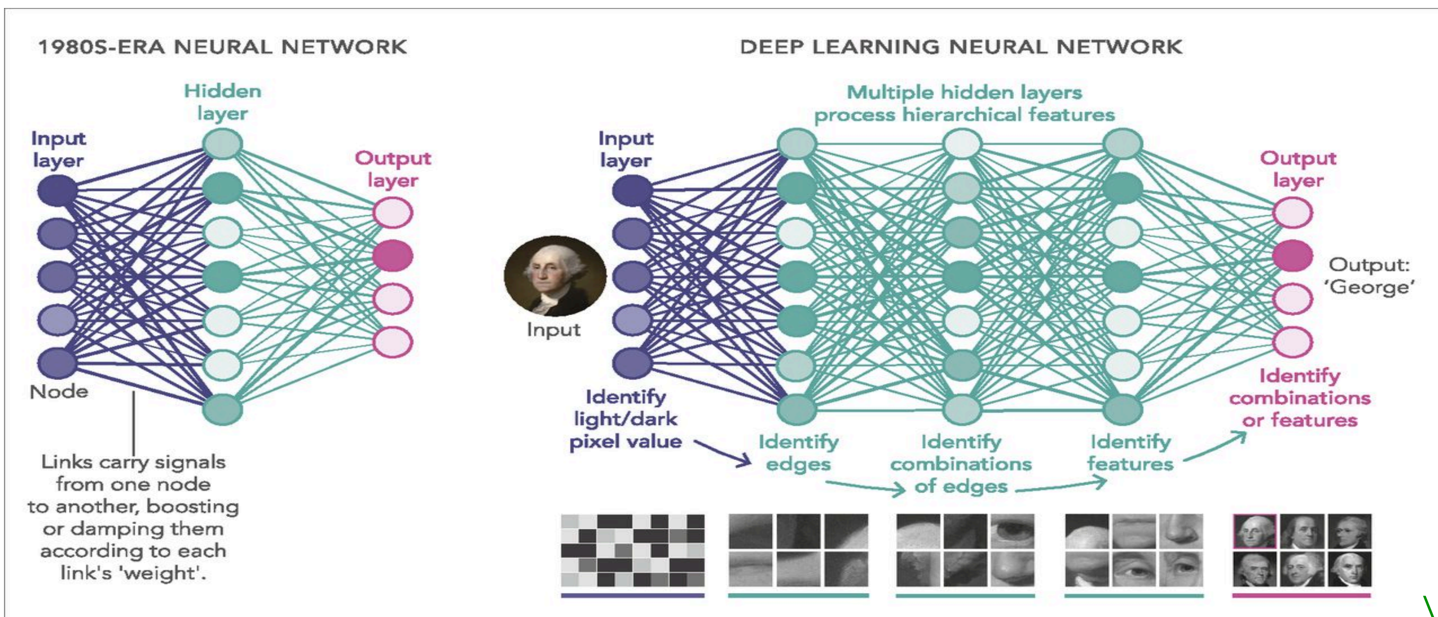
DEEP MACHINE LEARNING

Machine Learning

- ML is the tool used for large-scale data processing and is well suited for complex datasets with huge numbers of variables and features (patterns and regularities), especially for deep learning neural networks (NNs).
- The Universal Theorem: any function can be approximated by a neural network with at least one hidden layer.
- For a long time, given this theorem and the difficulty in complex networks, people have restricted themselves to shallow networks with only one hidden layer.
- Recently, people have realized that deeper, more complex networks with many hidden layers can "understand" higher levels of abstraction than shallow layers.

Resurgence of NN

- NNs became popular and then forgotten for a while.
- They have resurged in the last decade partly due to:
  - faster computers, with the use of GPUs versus the traditional use of CPUs,
  - better, deeper algorithms and NN designs, and
  - increasingly large datasets.



Waldrop 2019

shallow NN

deep NN

Common NN Types

- Dense neutral network (DNN): a network with standard fully-connected feed-forward layers that take flattened vectors as the input, prototypical for most tasks; sometimes also called multi-layer perceptron (MLP).
- Convolutional neural network (CNN)*: a network with special layers that filter data, suitable for computer vision.
  ➠ ideal for jet image recognition task in collider physics
- Recurrent neural network (RNN): a network that deals with sequences of variable length by defining a recurrence relation over these sequences, suitable for natural language processing and speech recognition tasks.

*Some evidence shows that neurons in CNNs are organized in a way similar to biological cells in the visual cortex of the human brain.

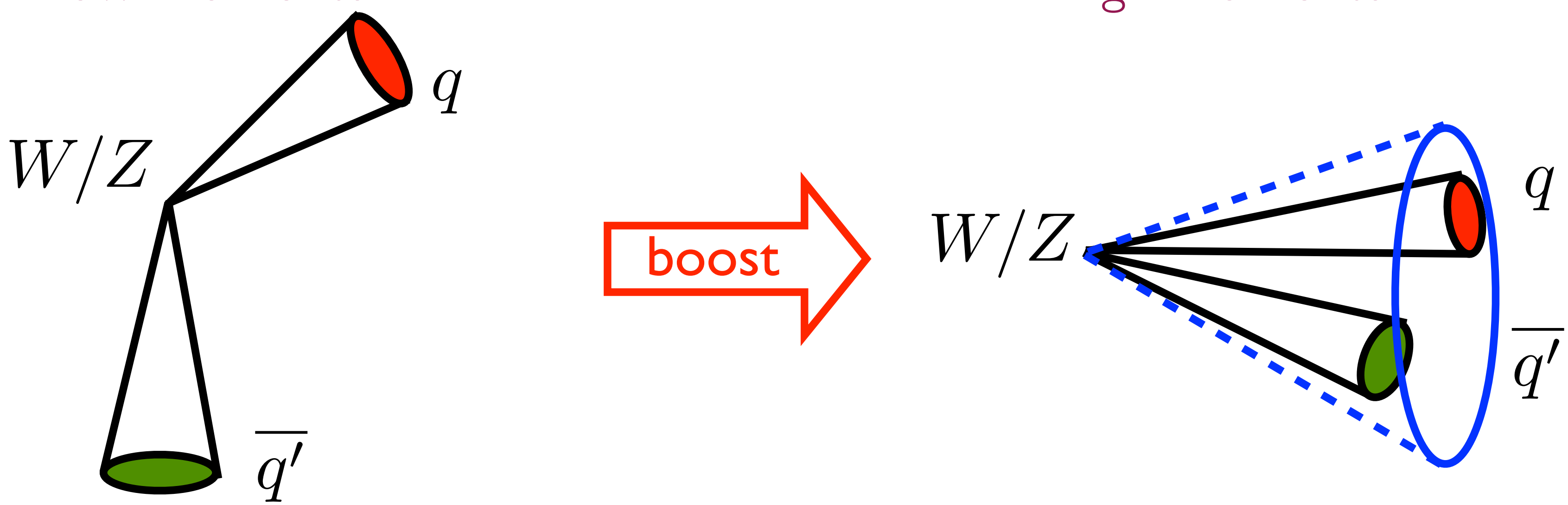
BOOSTED W/Z BOSON TAGGERS

Motivations

- Weak boson scatterings at high energy provide a direct probe of the EWSB mechanism.
- New physics particles, such as Z', W',or heavy Higgs, often decay to weak bosons.
- Such weak bosons are generally highly boosted and, when decaying hadronically, form one collimated jet.

low momentum

high momentum



two resolvable jets

one boosted fat jet

Motivations

- A lot of effort has been devoted to the important problem of tagging boosted resonances (i.e., identification or classification) through the understanding of jet substructure (how energy is distributed within the jet).

  Marzani, Soyez, and Spannowsky 2019
  Asquith et al. 2019
  Larkoski, Moult, and Nachman 2020

- Besides usual QCD jets (lighter quarks and gluons), the LHC produces new classes of jets with collimated prongs, derived from boosted W, Z, t-quark, or Higgs boson.
- Recently, there is enormous interest in the application of modern deep learning techniques to boosted resonance tagging because they can automate the process of feature engineering from high-dimensional, low-level inputs (e.g., jet constituents).

  de Oliveira et al 2016
  Larkoski, Moult, Nachman 2017

OUR TAGGERS AND THEIR PERFORMANCES

Sample Preparation

- Simulations:

parton-level processes
➠ `MG5 aMC@NLOv2.6.1`

showering and hadronization*
➠ `PYTHIA 8.2.19`

detector simulation
➠ `DELPHES 3.4.1 w/ CMS card`

jet reconstruction
➠ `FastJet 3.1.3`

*A dataset based on HERWIG showering and hadronization is also generated for the purpose of checking the reliability of our jet-tagging results.

- Jet selection:

m_H = 800 GeV

Jet sample	$p_T \in (350, 450)$ GeV, $\lvert\eta\rvert \leq 1$ jets with anti-k_T and $R = 0.7$ V-V merging : $\Delta R(V_1, V_2) < 0.6$ V-jet matching : $\Delta R(V, j) < 0.1$

- Sample sizes:

	Jet sample size		
	Training set	Validation set	Testing set
W^+	169.2k	18.8k	38k
W^-	178.2k	19.8k	40k
Z	157.5k	17.5k	35k

90% / 10%

This is how theorists generate large datasets for ML analyses.

Higher-Level Inputs

- Traditional analyses make use of higher-level observables:

Jet invariant mass

$$\mathcal{M}_J^2 = \left(\sum_{i\in J} E_i\right)^2 - \left(\sum_{i\in J} \mathbf{p}_i\right)^2$$

Jet charge

$$\mathcal{Q}_\kappa = \sum_{i\in J} q_i \times \left(\frac{p_T^i}{p_{T,J}}\right)^\kappa$$

where J denotes a jet, i runs over jet constituents (tracks) with $p_T > 500$ MeV, q_i is the integer charge of constituent i in units of proton charge, and κ is a free parameter.

- Q_κ is computed in this p_T-weighted scheme in the hope of minimizing mis-measurements from low-p_T particles.

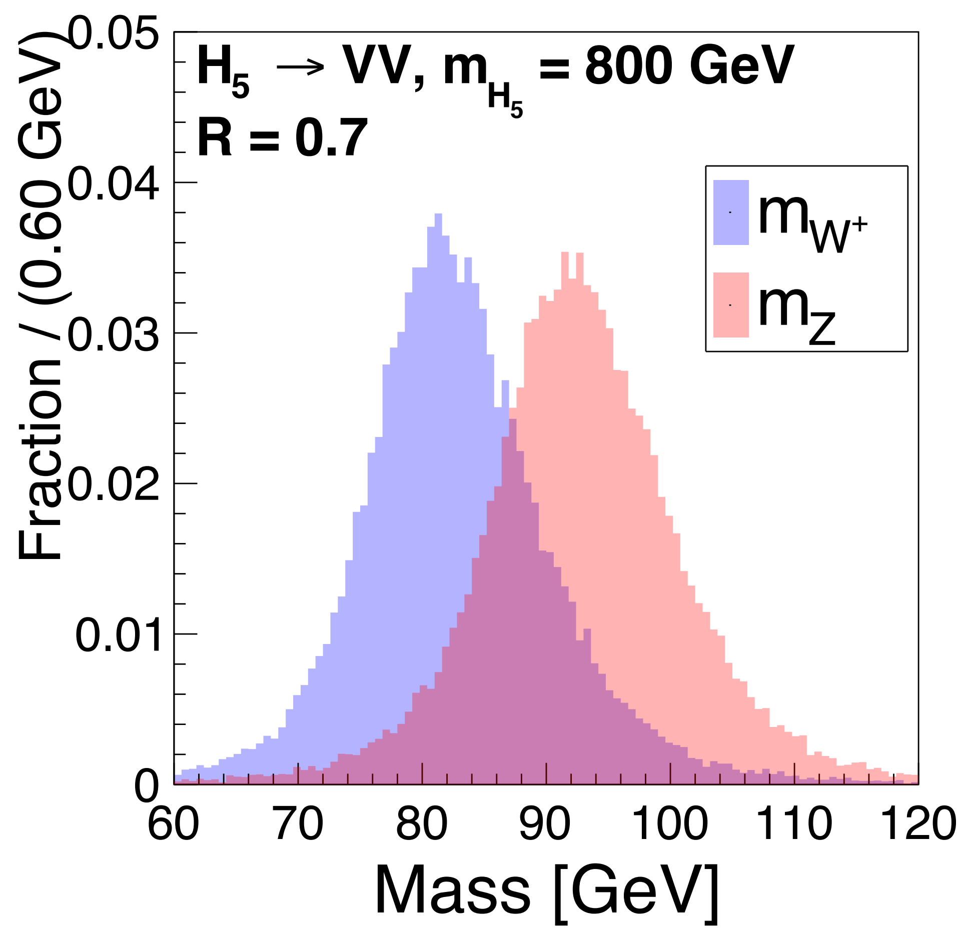
Higher-Level Inputs

- Traditional analyses make use of higher-level observables:

Jet invariant mass

$$\mathcal{M}_J^2 = \left(\sum_{i\in J} E_i\right)^2 - \left(\sum_{i\in J} \mathbf{p}_i\right)^2$$

- The broader widths in the mass distribution originate from a combination of showering, hadronization, jet clustering and detector effects.
  - ➠ no clear boundary
  - ➠ unable to distinguish W^+/W^-



Higher-Level Inputs

- Traditional analyses make use of higher-level observables:

Jet charge

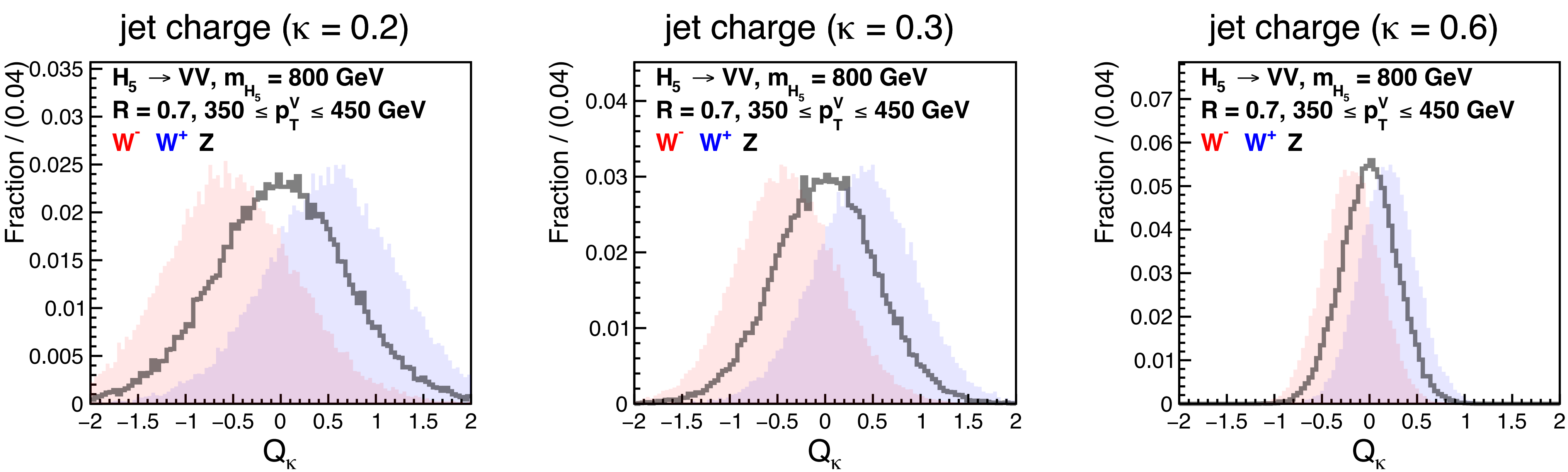
$$\mathcal{Q}_\kappa = \sum_{i \in J} q_i \times \left(\frac{p_T^i}{p_{T,J}} \right)^\kappa$$

p_T-weighted scheme:

$\kappa = 0$ ➠ equal-weight

$\kappa = 1$ ➠ proportional to p_T

Field, Feynman 1978

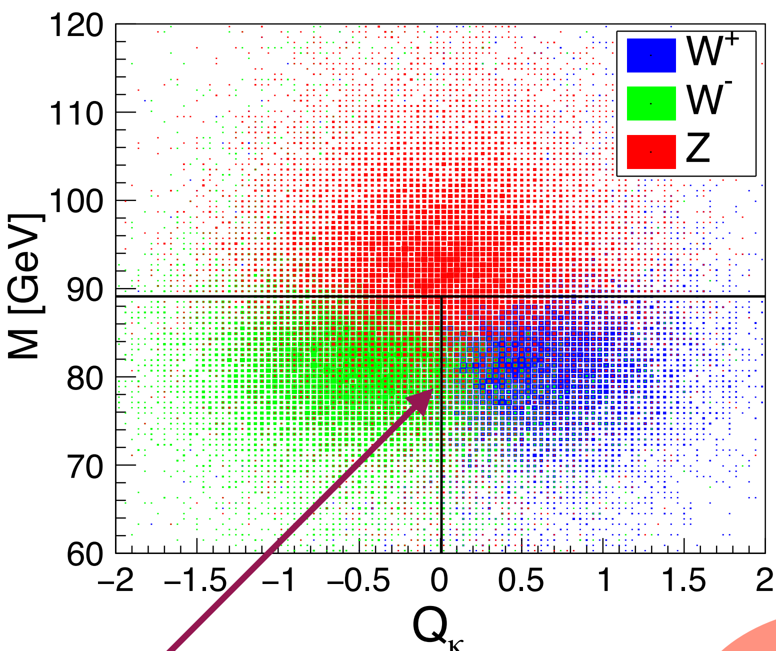


- The separation is not well and depends on the choices of weight factor κ, jet cone size R, etc.

Reference Taggers

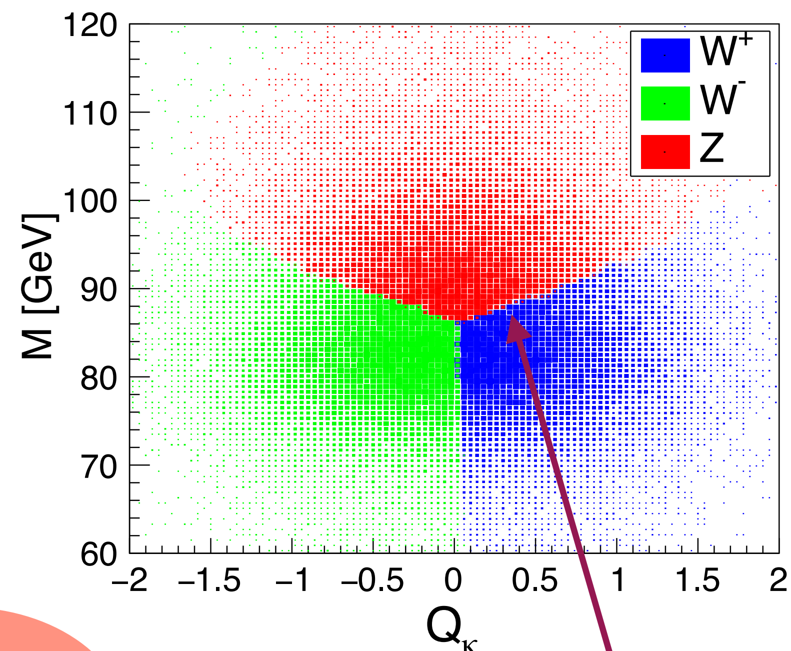
- For the ternary (W+/W−/Z) classification task, the reference taggers can be visualized as follows:

cut-based tagger

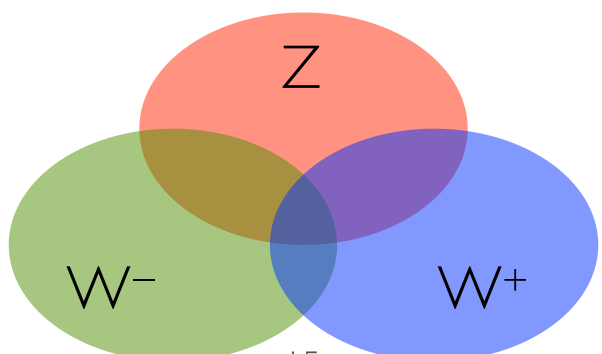


rectangular cuts

single-κ BDT (κ=0.3)



Y-shaped cuts



Jet Images and Channels

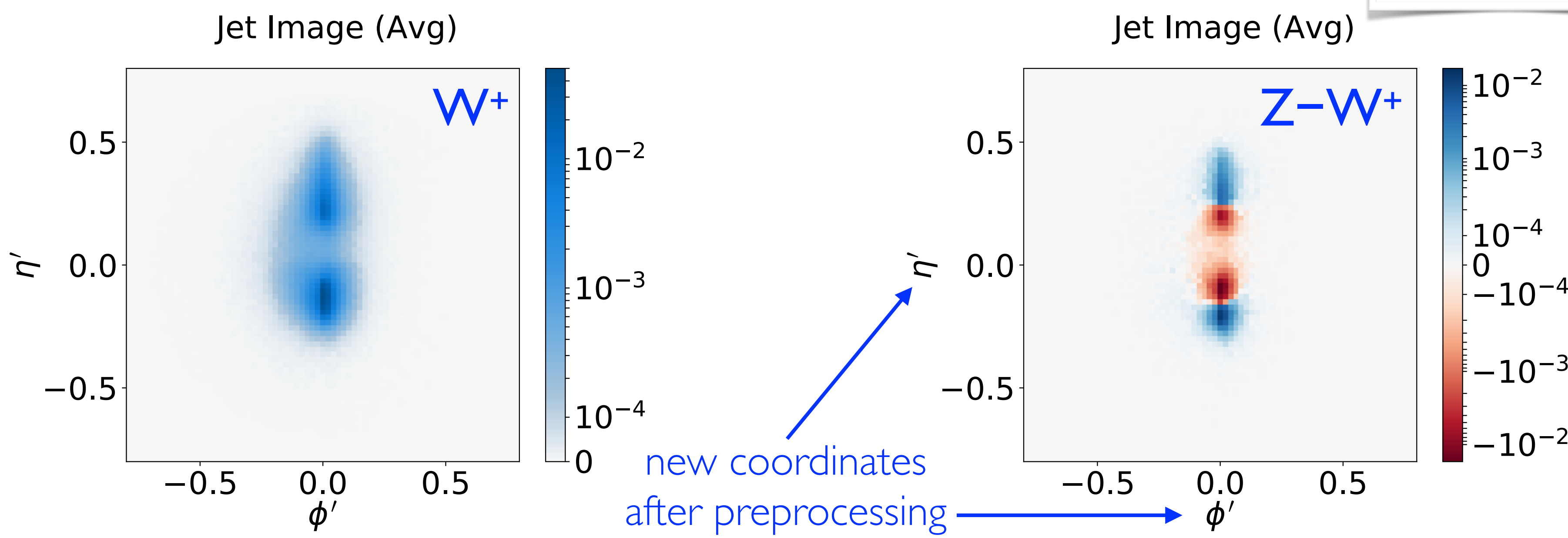
- Deep learning taggers studied in our work are based on jet images, utilizing lower-level inputs and processed by CNNs.
- Jet images are made from jets reconstructed in a box of $\Delta\eta = \Delta\phi = 1.6$ (central region) with 75×75 pixels.
  ➠ a resolution consistent with that of the CMS ECal
- The input variables or channels are Q_κ and p_T per pixel.

  ➠ now the sum $\sum_{i \in J}$ is done within each pixel

Lower-Level Inputs

- Preprocess each image, involving centralization, rotation and flipping (➠ jet with larger p_T is on $+\eta'$ axis).
- p_T channel:

colored bars in arbitrary units

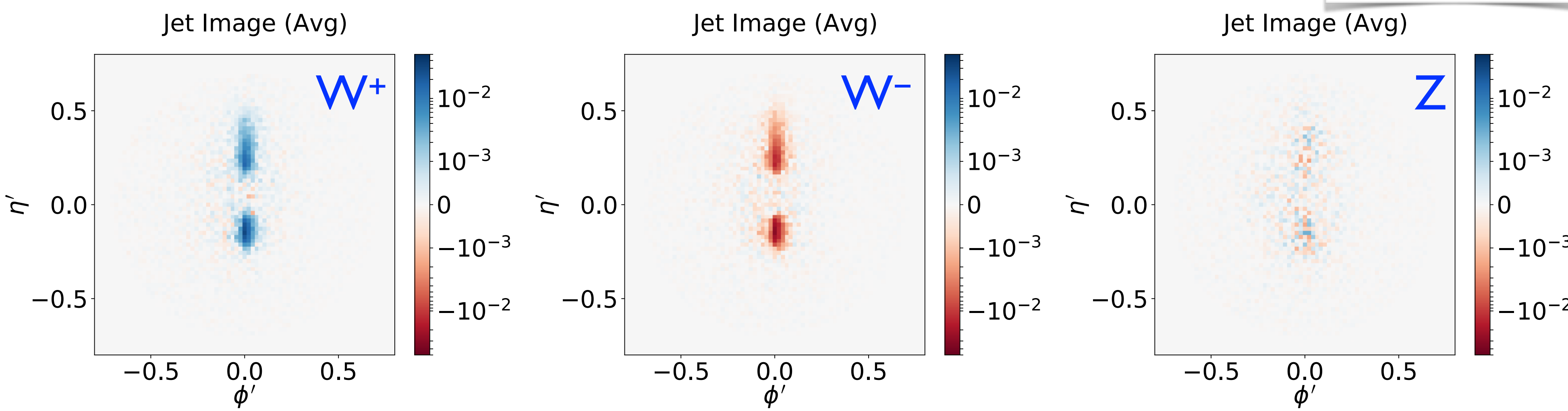


- W− average jet image is basically identical to that of W+.
- Z average jet image has a wider distribution in ΔR than W jets, as expected from its larger invariant mass.

Lower-Level Inputs

- Preprocess each image, involving centralization, rotation and flipping (➠ jet with larger p_T is on $+\eta'$ axis).
- Q_κ channel:

colored bars in arbitrary units



- The average Z jet charge image is close to zero as the constituent charges in different events tend to cancel out.

Our CNN Taggers

two architectures

- a deeper Q_κ network tends to overfit W^+/W^-
- a deeper p_T network helps identifying Z

Input Image	(75×75) pixels within $(\lvert\eta\rvert \leq 0.8,\ \lvert\phi\rvert \leq 0.8)$		
Neural Network	CNN	CNN^2	
Channels	p_T, $\mathcal{Q}_\kappa$	p_T	$\mathcal{Q}_\kappa$
Architecture	BN-32C6-MP2-128C4-MP2-256C6-MP2-512N-512N	BN-32C3-32C3-MP2-64C3-MP2-64C3-MP2-64C3-64C3-128C5-256C5-256N-256N	BN-32C3-32C3-MP2-64C4-64C4-MP2-256C6-MP2-256N
Settings	`Relu` Activation, `Padding=same`, *Dropout* $= 0.5$, l2 `Regularizer` $= 0.01$		
Preprocessing	*Centralization, Rotation, Flipping*		
Training	Adam Optimizer, `Minibatchsize`$=512$, *Cross entropy* loss		

activated to enable a deeper network

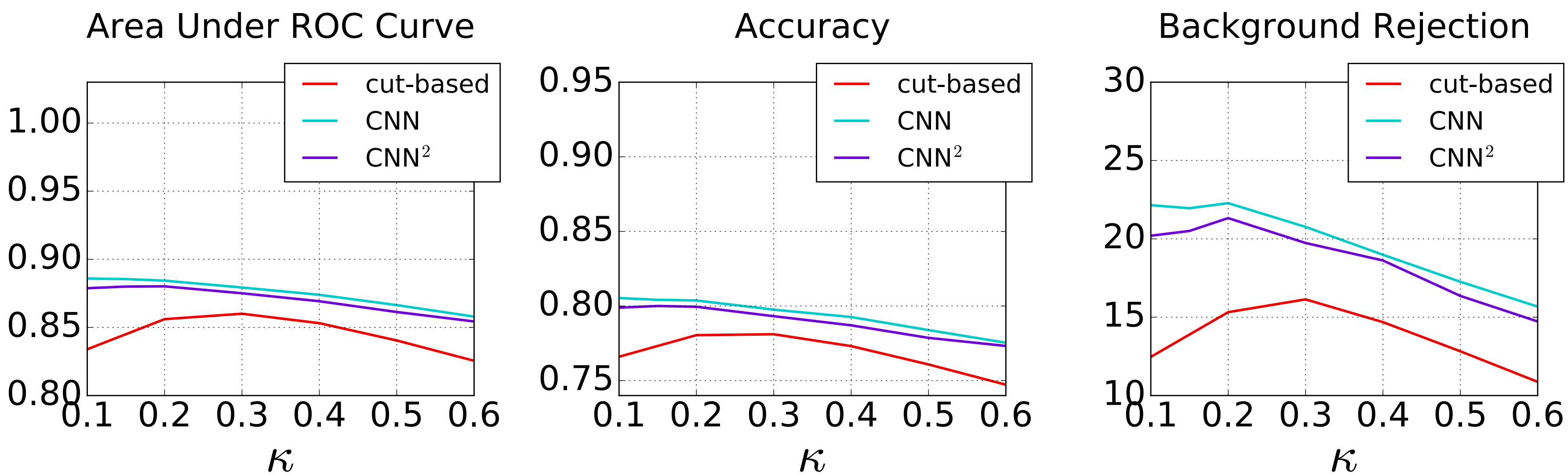
set to prevent overfitting

using Keras library with TensorFlow backend

W^-/W^+ Classification

- Only the Q_κ distribution is useful.

performance as a function of κ



- Qualitatively different κ dependence for cut-based taggers, while similar between CNNs.
- CNN is slightly better than CNN².
- CNNs have a smaller optimal κ.
  - ➠ κ = 0.3 for the single-κ BDT taggers, and
    κ = 0.15 for our CNN taggers

W^-/W^+ Classification

- **Performance metrics** for all taggers, except for the single-κ BDT, which is the same as the cut-based one.

Area under ROC curve

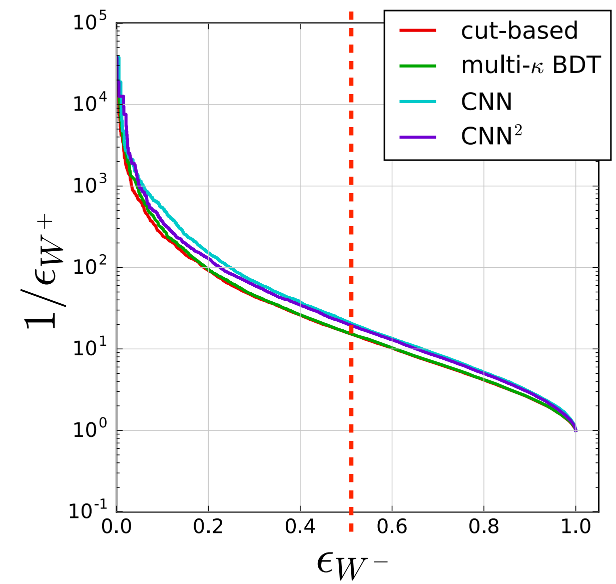
	R50	AUC	ACC
cut-based	16.1372	0.8600	0.7811
multi-κ BDT	16.0960	0.8615	0.7820
CNN	21.9559	0.8855	0.8042
CNN^2	20.5057	0.8800	0.8000

background rejection rate at a 50% signal efficiency working point, $(1/\varepsilon_b)|_{\varepsilon_s=50\%}$.

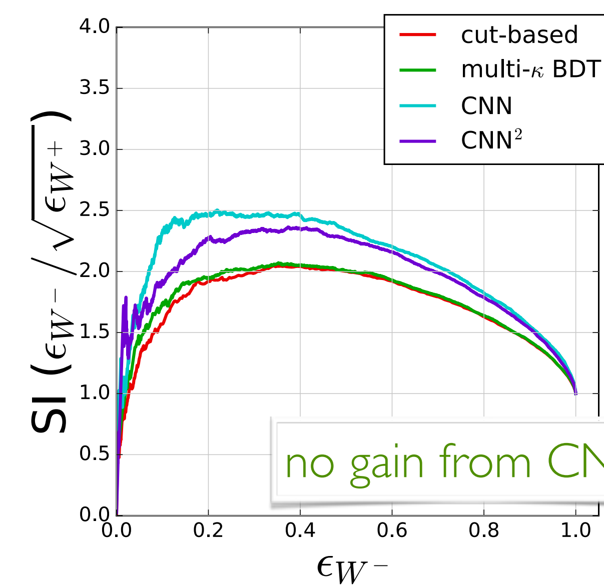
best accuracy

30-40% (36% for R50) improvement in bkgnd rejection rate

ROC curve



SIC curve



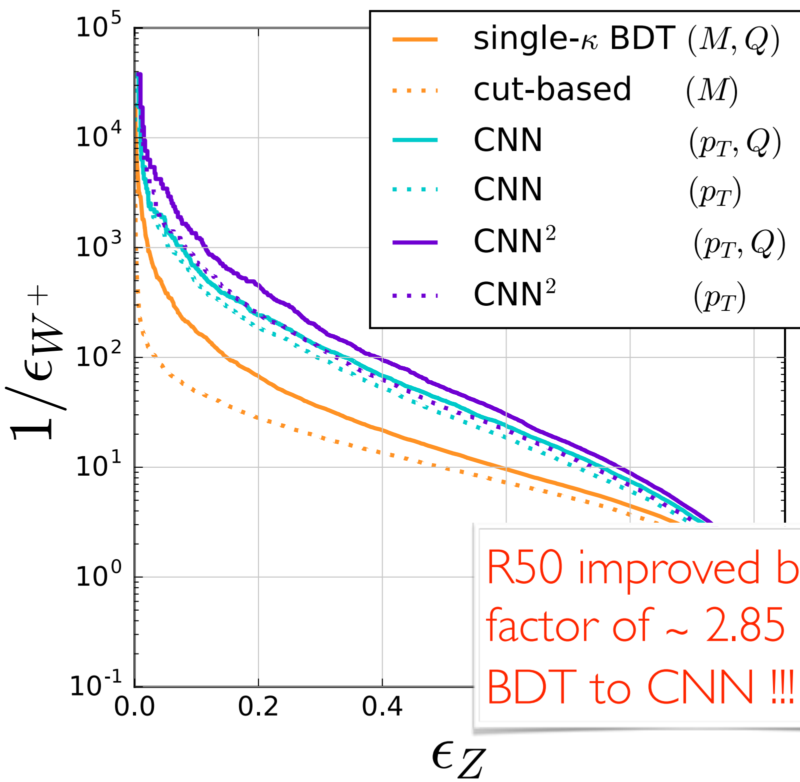
no gain from CNN^2

Z/W+ Classification

	R50	AUC	ACC
cut-based	9.9590	0.8118	0.7705
single-κ BDT	14.1638	0.8608	0.7875
multi-κ BDT	14.2383	0.8611	0.7880
CNN	40.4205	0.9091	0.8345
CNN^2	52.6028	0.9206	0.8452

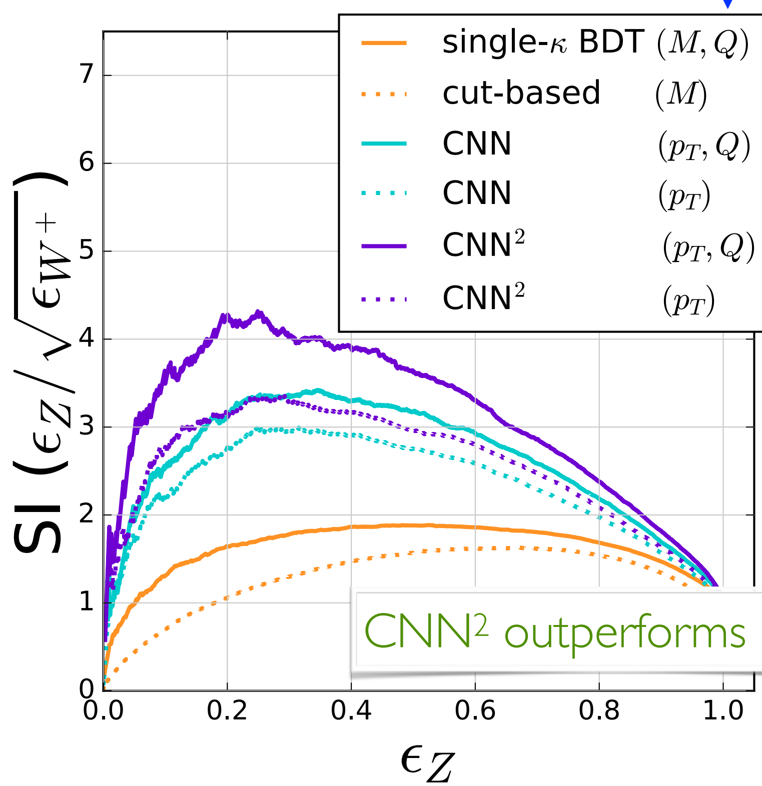
With or without Q:
In a wide range of working points, our CNN taggers enjoy a ~30% gain in the background rejection rate by incorporating Q_κ.

ROC curve



R50 improved by a factor of ~ 2.85 from BDT to CNN !!!

SIC curve



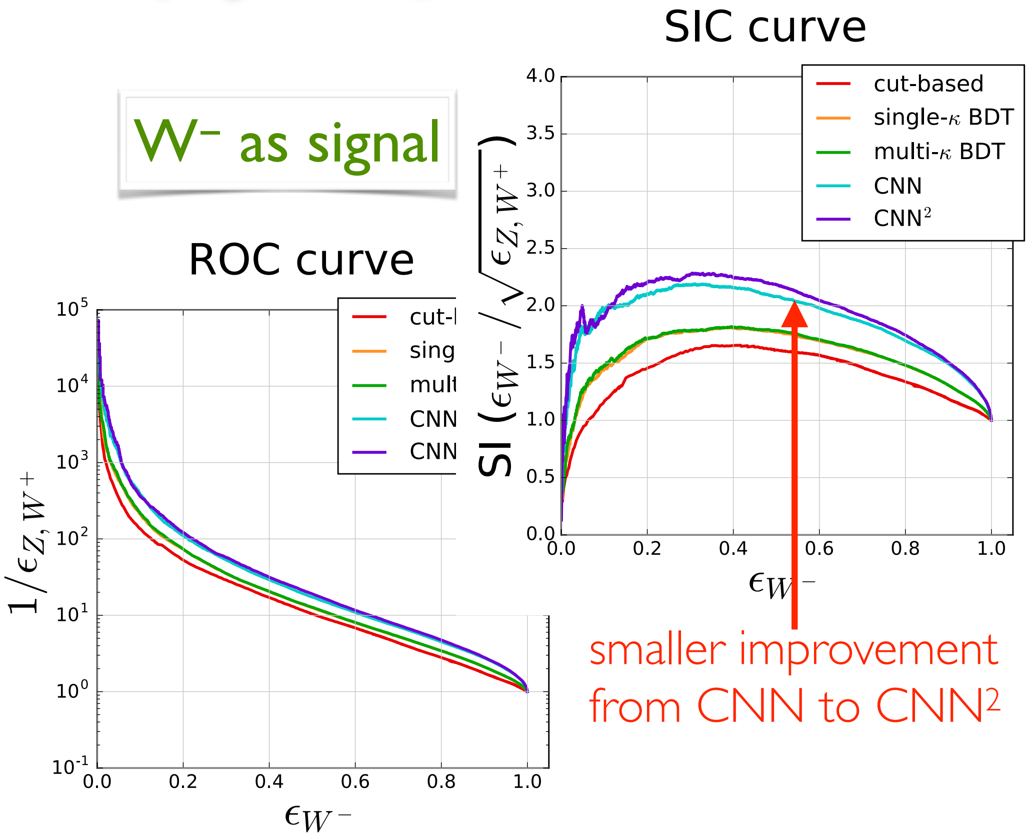
CNN^2 outperforms CNN

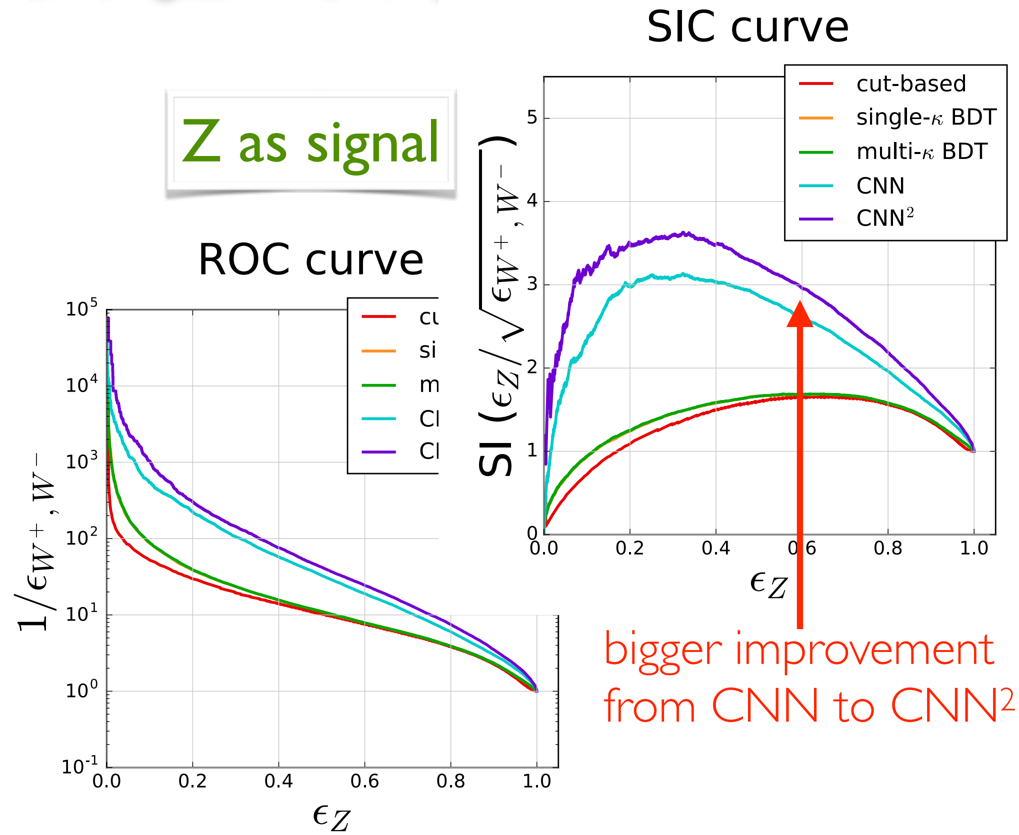
$W^+/W^-/Z$ Classification

- We compare the performance of the ternary taggers according to two metrics:
  (a) their overall accuracy
  $$\frac{\text{number of correct predictions}}{\text{total number of instances}}$$
  and
  (b) a "one-against-all" metric

  one class as "signal" $\leftrightarrow$ all the rest as "background"

W^- or Z Versus the Rest

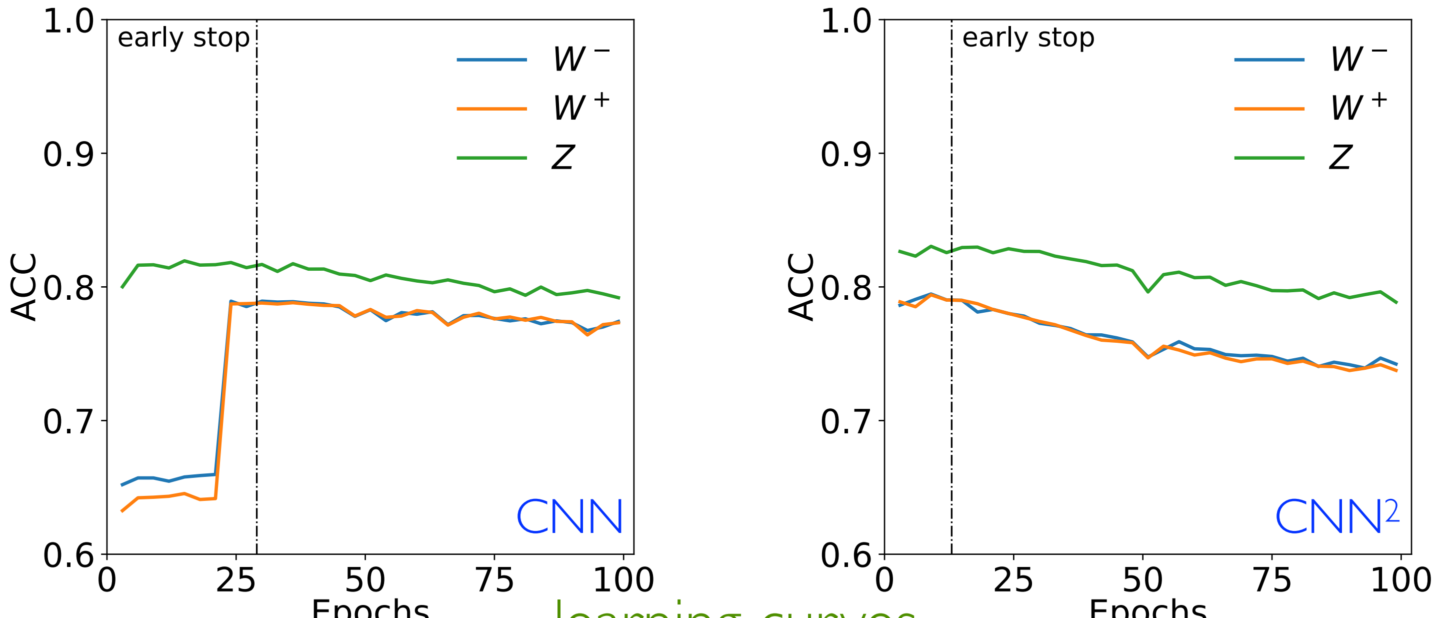




	overall	signal: W^-			signal: Z		
	ACC	R50	AUC	ACC	R50	AUC	ACC
cut-based	0.6581	8.0262	0.7893	0.7643	10.0882	0.8233	0.7839
single-κ BDT	0.6667	12.5230	0.8339	0.7576	11.0726	0.8363	0.7725
multi-κ BDT	0.6675	12.7115	0.8348	0.7579	11.0678	0.8366	0.7726
CNN	0.7197	17.3403	0.8715	0.7890	32.8981	0.8936	0.8170
CNN^2	0.7318	19.0907	0.8764	0.7950	42.1927	0.9088	0.8334

Phase Transition in DL

- A "phase transition" in the CNN architecture for W± samples around 25th epoch during training, but not CNN².
- The CNN tends to first learn characteristics of the Z sample, and then those of the W sample later.
- It is possible that the CNN² learns so fast that the performance in all classes saturates within one epoch.

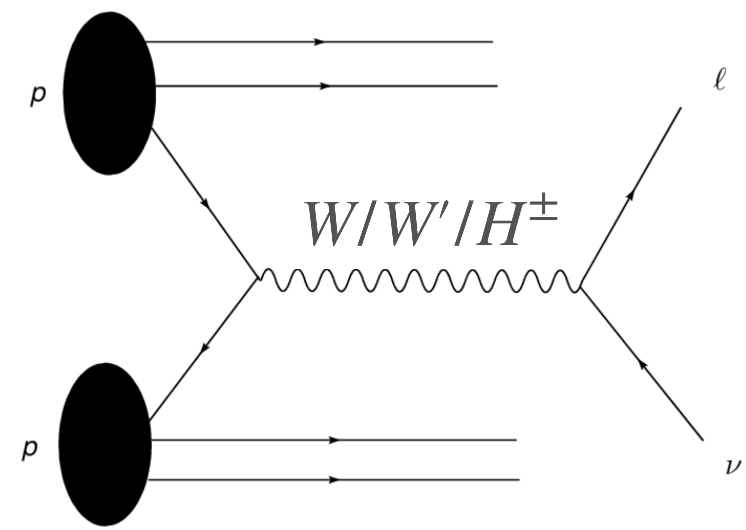


learning curves

SEARCHES FOR CHARGED BOSONS

Motivations

- Since the discovery of W boson through the eν decay channel in 1983, searches for W' and other charged bosonic resonances have continued.
- At LHC, the light leptonic channels are more favorable.
  ➠ hope to use these decay modes to determine mass, width, spin, and couplings to SM fermions



Ambiguities at LHC

- Unknown initial state: To study the Lorentz structure of a charged-current interaction by examining the angular distribution of $\ell^{\pm}$, we need to define a forward direction, e.g., in the q (not $\bar{q}$) direction. However, LHC is a symmetric machine.
- Missing longitudinal momentum: Since the colliding partons typically have a boosted c.m. frame, we need to identify the missing longitudinal momentum of the neutrino to correctly determine the distribution in $\cos\theta_{\rm CM}$. From kinematics, the longitudinal momentum can be solved from a quadratic equation assuming an on-shell mediating boson, but there is no event-by-event information to determine which of the two quadratic solutions is correct.

Classes of Interactions

- Vector/axial (VA): This class corresponds to a W' with $W'_\mu \bar{\psi}\gamma^\mu \chi$ or $W'_\mu \bar{\psi}\gamma^\mu \gamma_5 \chi$ fermionic couplings.
- Chiral (CH): This class corresponds to a W' with $W'_\mu \bar{\psi}\gamma^\mu \left(1-\gamma_5\right)\chi$ or $W'_\mu \bar{\psi}\gamma^\mu \left(1+\gamma_5\right)\chi$ fermionic couplings.
- Scalar (SC): This class corresponds to an $H^\pm$ with $H\bar{\psi}\chi$ or $H\bar{\psi}\gamma_5\chi$ Yukawa couplings.

- For a symmetric machine like LHC, we still cannot distinguish interactions with and without γ_5.*

*Interference between a W' and the SM W could in principle break this degeneracy, yet such effects are found to be negligible for the TeV-mass bosons considered in this study.

Our Goal

- We explore deep-learning-based approaches to tackle the problem of determining the spin and interaction type of a heavy charged boson through its leptonic decay channels.
- The above-mentioned ambiguities make event-by-event reconstruction by a NN also challenging, but classification based on a collection of events can still have significant distinguishing power.
- Two ways to input this collection of events:
  (a) simply feed them into the NN event by event as an array, or
  (b) combine a number of events and form a 2D histogram of a selected pair of variables as the input.

Our NN Models

- Consider three NN models in this analysis:
- FNNi: trained upon the kinematic information of individual events — a fully connected neural network.
- FNNh: trained upon flattened 2D histograms made from pairs of kinematic observables of a number of events.
- CNN: trained upon the 2D histograms mentioned above.

- Prepared $\approx$ 0.3M samples for each NP classes and SM.
- Will compare their performance in classifying different types of charged bosons and interactions.

OUR TAGGERS AND THEIR PERFORMANCES

Assumed Environment

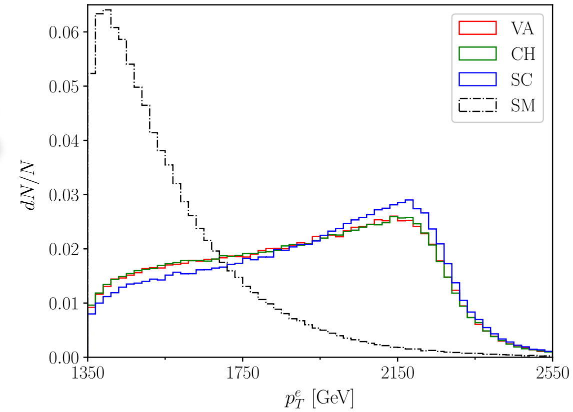
- Assume 14-TeV HL-LHC, with $L = 3\ \mathrm{ab}^{-1}$.
- Beyond the signal-only hypothesis testing, also include the SM W background. Khosa, Sanz, Soughton 2019
- Investigate scenarios of different S/B ratios.
- Study only the $e\nu$ channel, though the method can also be applied to $\mu\nu$ and improve the NN efficiency.
- Assume that the coupling strength and structure are universal to all generations in both quark and lepton sectors (even for $H^{\pm}$).
- To satisfy current bounds and to expect a 5σ discovery in the HL-LHC era, the boson mass has to be ≥ 4.5 TeV.
  ➠ will focus on 4.5 TeV (also explore 6 TeV)

0-Jet Samples

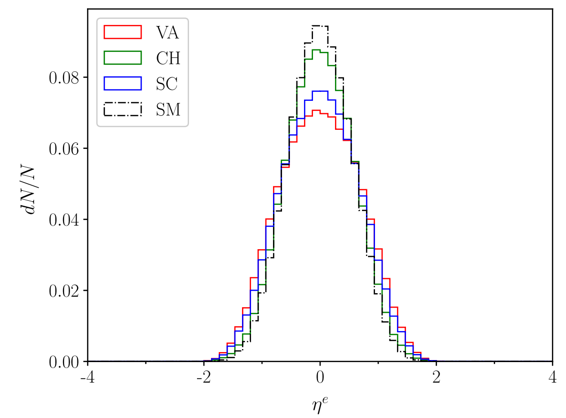
- Assume $M = 4.5$ TeV, $\Gamma_{\mathrm{NP}} \simeq 200$ GeV.
- Within selected phase space, expected number of SM 0-jet events is

$$B_0 = \sigma_{B_0} \times \mathcal{L} \approx 84$$

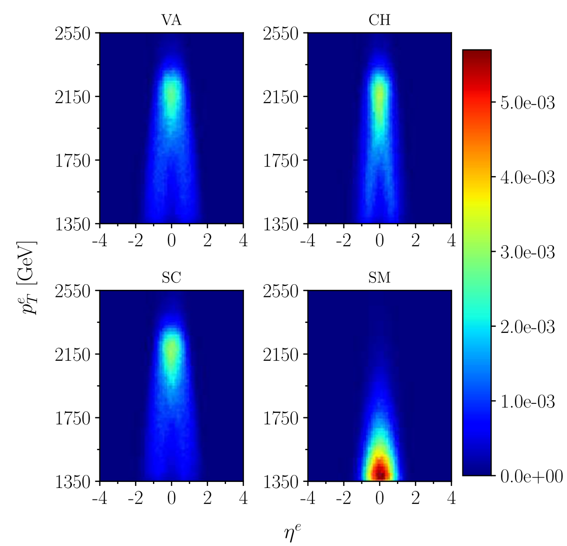
- (a) p_T^e distribution; (b) η^e distribution; (c) averaged image in η^e-p_T^e plane.
- VA and CH are basically identical in p_T^e, but very different in η^e.
  ➠ their difference in p_T^e in bottom plot is due to the η^e cut and normalization.



(a)



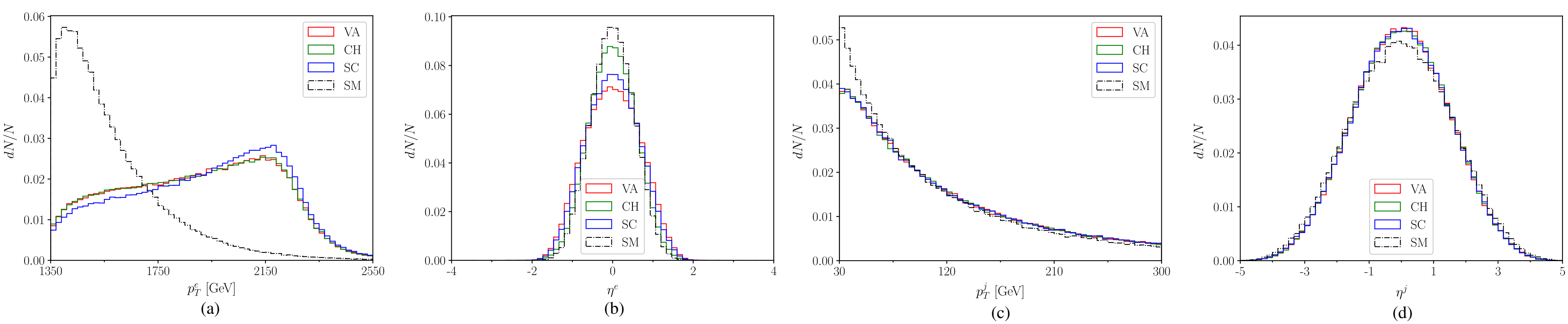
(c)



(e)

1-Jet Samples

- Within selected phase space, expected number of SM 1-jet events (including contributions from 0-jet samples) is
$$B_1 = \sigma_{B_1} \times \mathcal{L} \approx 58$$
- More kinematic variables: p_T^e, η^e, p_T^j, η^j, $\Delta\phi_{ej}$, $\not{E}_T$, $\Delta\phi_{e\not{E}_T}$, and $\Delta\phi_{j\not{E}_T}$, where last three being derived observables.
- Form "RGB" histograms by picking three pairs of them, according to physical relationship, principal component analysis, etc



Our NNs

TABLE III. Zero-jet and one-jet FNNi structure specifications.

	Zero jet	One jet
Input	p_T^e, η^e, ϕ^e	$p_T^e, \eta^e, p_T^j, \eta^j$ $\not{E}_T, \Delta\phi_{ej}, \Delta\phi_{e\not{E}_T}, \Delta\phi_{j\not{E}_T}$
Layers	batch normalization layer dense layer: 256[a] dense layer: 256	
Layer settings	hidden layer activation = `relu` output layer activation = `softmax`	
Compilation	loss = `categorical_crossentropy` optimizer = `adam` [47] metric = `accuracy`	

[a]This means that there are 256 nodes in the dense layer.

TABLE IV. Zero-jet and one-jet FNNh structure specifications.

	Zero jet	One jet
Input	Flattened 60×60 images p_T^e vs η^e	p_T^e vs η^e, p_T^e vs $\not{E}_T$, p_T^e vs $\Delta\phi_{ej}$
Layers	batch normalization layer dense layer: 1024 dense layer: 256	
Layer settings	hidden layer activation = `relu` output layer activation = `softmax`	
Compilation	loss = `categorical_crossentropy` optimizer = `adam` metric = `accuracy`	

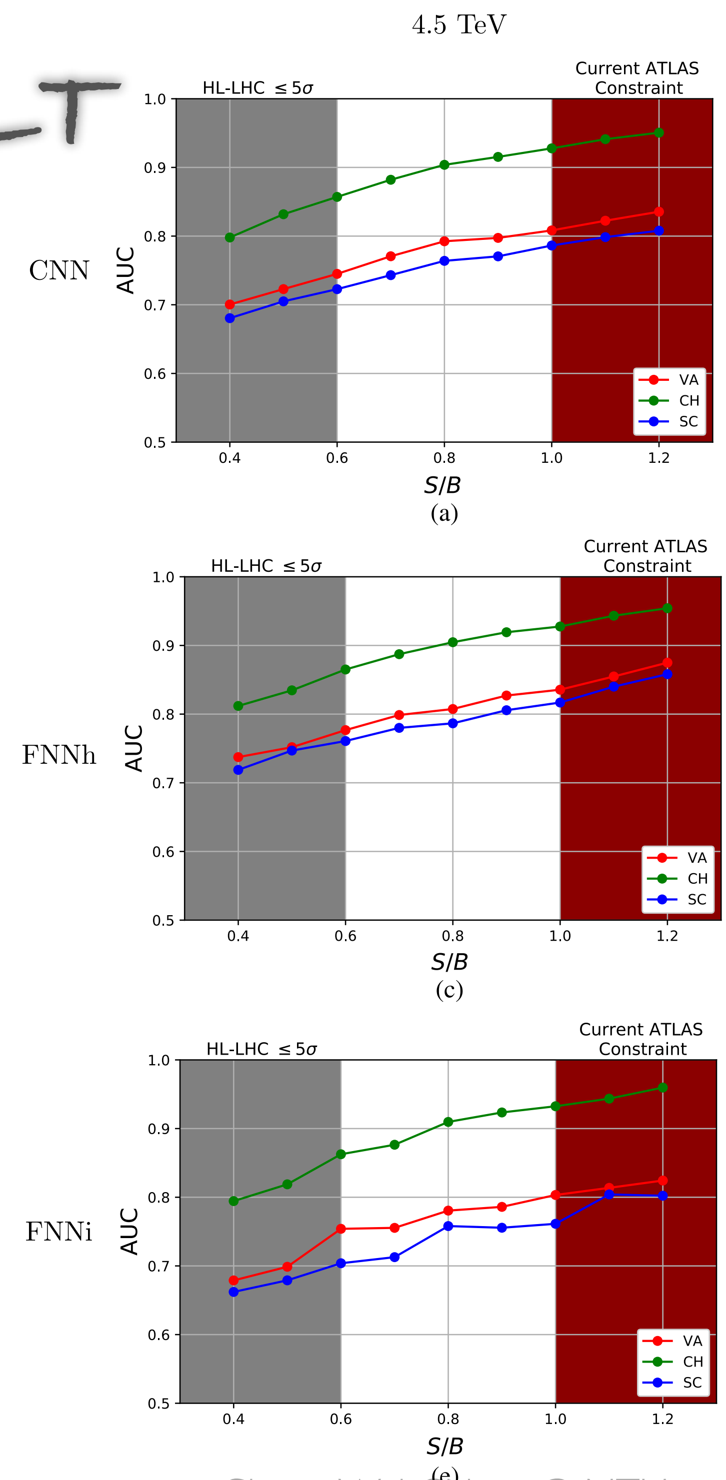
TABLE V. Zero-jet and one-jet CNN structure specifications.

	Zero jet	One jet
Input	60×60 images p_T^e vs η^e	RGB colors: p_T^e vs η^e, p_T^e vs $\not{E}_T$, p_T^e vs $\Delta\phi_{ej}$
Layers	batch normalization layer convolutional 2D layer: 3-32[a] max pooling 2D layer: 2-2[b] convolutional 2D layer: 3-32 max pooling 2D layer: 2-2 flatten layer dense layer: 128 dense layer: 64	
Layer settings	hidden layer activation = `relu` output layer activation = `softmax`	
Compilation	loss = `categorical_crossentropy` optimizer = `adam` metric = `accuracy`	

[a]This means that the filter kernel dimension is 3×3, and that there are 32 nodes in the convolutional layer.
[b]This means that the max pooling kernel dimension is 2×2, and that each stride is 2 pixels.

0-jet Result

- AUC as a function of S/B for 0-jet samples.
- Grey: not reach 5σ even for HL-LHC; Red: excluded by current ATLAS data.
- FNNh is slightly better than CNN, while FNNi is further worse.
- For all three NN models, CH has best performance, VA is slightly better than SC.
- Performance generally improves with S/B.

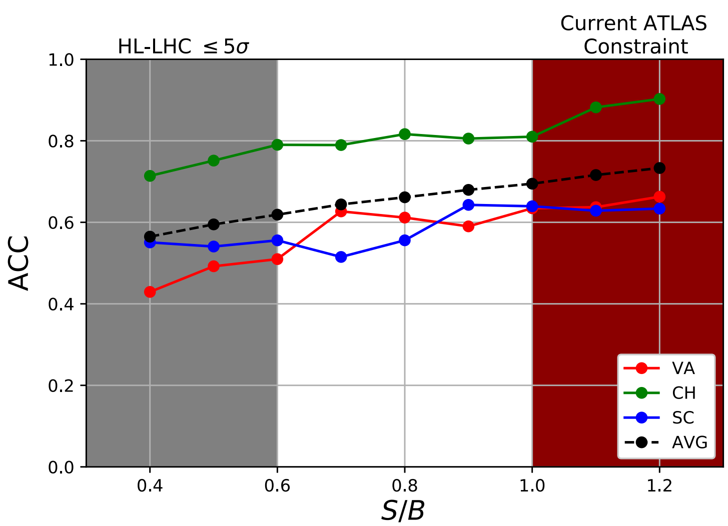


0-jet Result

- ACC (classwise true positive rate) as a function of S/B for 0-jet samples in FNNh.
- Compared to AUC which is evaluated using a sliding threshold, the ACC is more sensitive to model biases.
- Cross validation (CV) helps to stabilize the classwise accuracies and does not significantly alter the average ACC (global true positive rate).

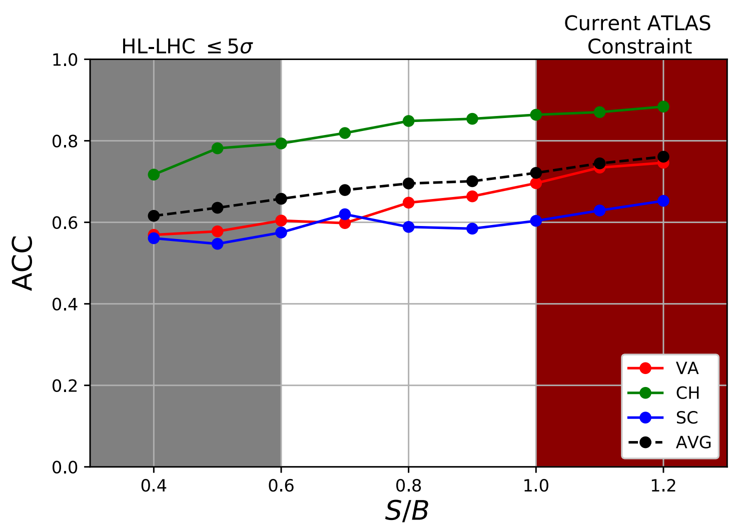
w/o CV

4.5 TeV



(a)

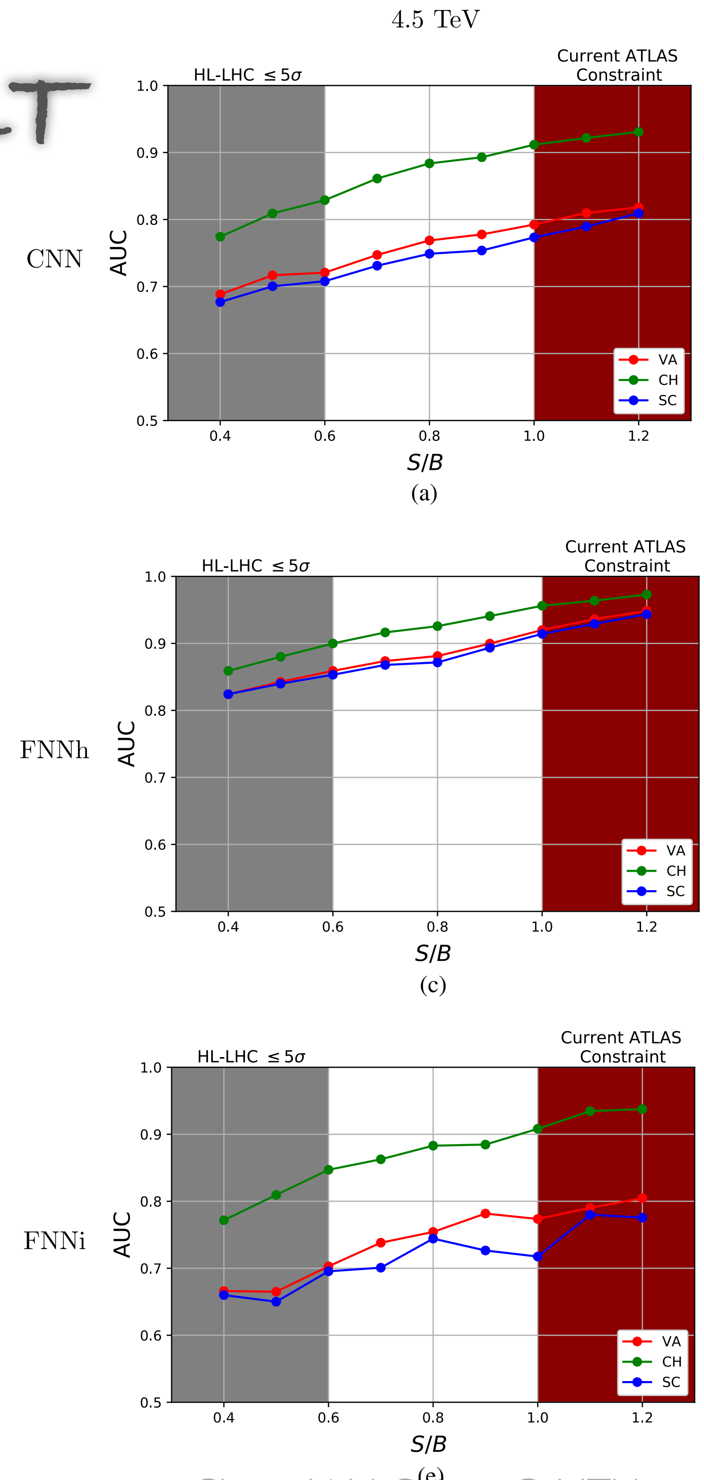
with CV



(b)

1-jet Result

- FNNh significantly outperforms CNN and FNNi.
- FNNh for 1-jet is better than for 0-jet, while CNN and FNNi have the opposite behavior.



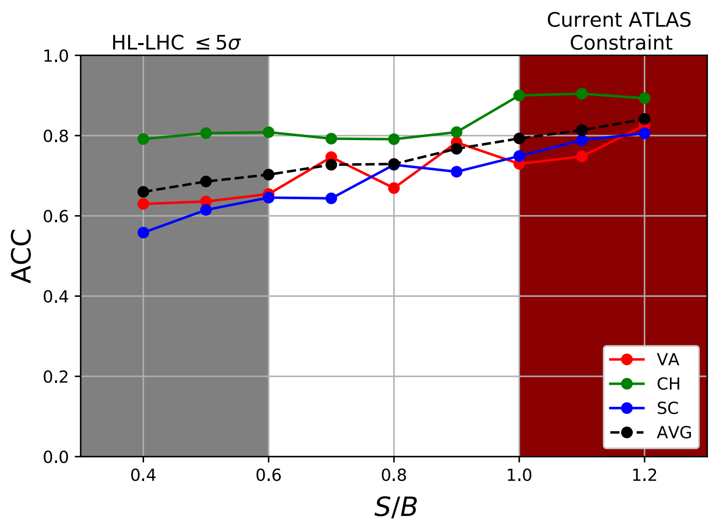
1-JET RESULT

- Again, cross validation helps stabilizing the FNNh.

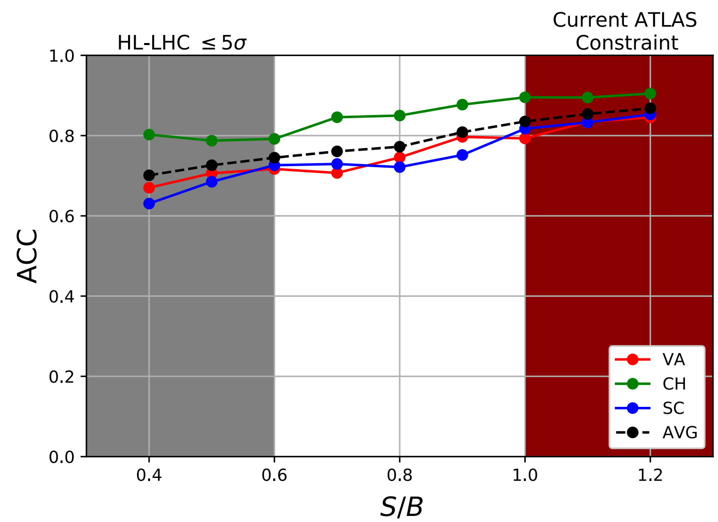
w/o CV

with CV

4.5 TeV

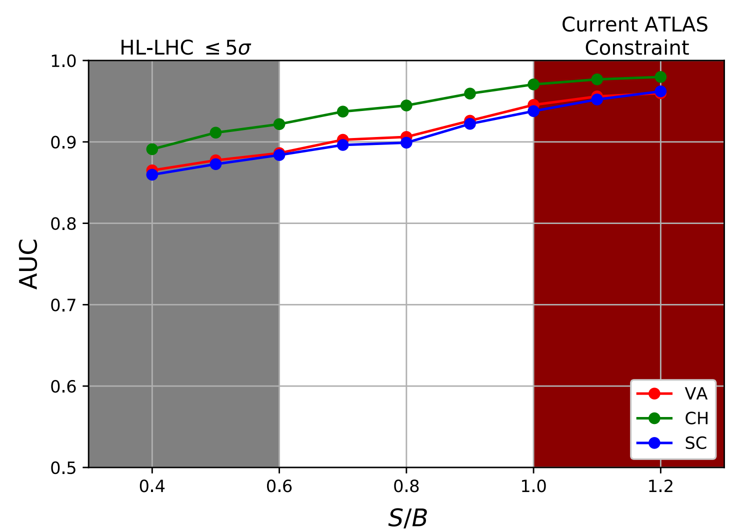


(a)



(b)

- AUC of FNNh after 10-fold CV:



(a)

Summary

- We show the power of modern machine learning techniques in collider physics by way of examples.
- We have built better taggers for
(a) boosted W/Z bosons through their hadronic decays, and
(b) the Drell-Yan processes through new charged bosons.
- For (a), CNN-based NN outperforms traditional cut-based or BDT analyses, and the charge channel is crucial in distinguishing W^+ and W^-.
- For (b), FNN-based NN on 2D histograms outperforms CNN. From 1-jet analysis, we see the power of NN for analyses with higher-dimensional kinematic variables.
- Modern machine learning is seen to have great potential in improving our abilities and efficiencies in analyzing data.

Backup Slides

Existing Jet Classifiers

- Jet flavor (light or heavy origin) tagging

Guest et al 2016

- Top tagging

Pearkes, Fedorko, Lister, Gay 2017
Egan, Fedorko, Lister, Pearkes, Gay 2017
Kasieczka, Plehn, Russell, Schell 2017
Butter, Kasieczka, Plehn, Russell 2018
Macaluso, Shih 2018
Butter et al 2019

- Quark/gluon tagging

Komiske, Metodiev, Schwartz 2017
Butter, Kasieczka, Plehn, Russell 2018
Macaluso, Shih 2018
Fraser, Schwartz 2018

- Boosted Z-jet tagging (from QCD-jets)

Larkoski, Salam, Thaler 2013
Larkoski, Moult, Neill 2016

- Boosted W-jet tagging (from QCD-jets)

Cui, Han, Schwartz 2011