Recursive Neural Networks in quark/gluon Tagging

(arXiv:1711.02633)

Taoli Cheng

University of Chinese Academy of Sciences

11/04/2018 2nd IML Workshop@ CERN

Machine Learning in Jet Physics

DNN Architectures

- Fully connected NNs
- CNN
- RNN (LSTM)
- RecNN

Jet Tagging

- top tagging
- W tagging
- b tagging
- q/g tagging
- …

Main Idea: low level information -> -> output info.

(Classification)

Machine Learning in Jet Physics

DNN Architectures

- Fully connected NNs
- CNN
- RNN (LSTM)
- RecNN

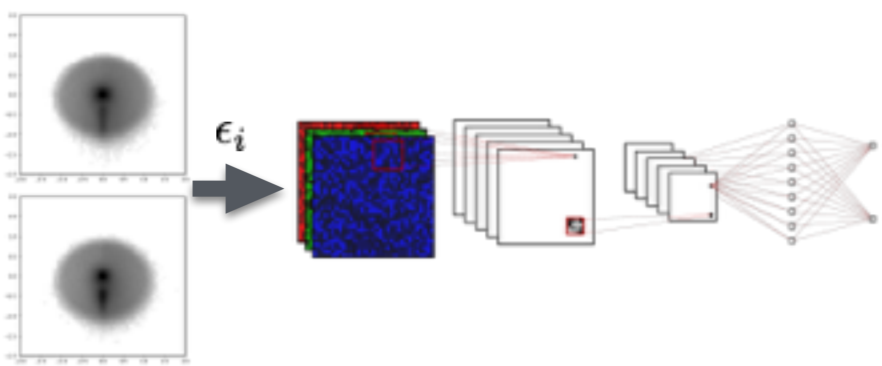
Jet Tagging

- top tagging
- W tagging
- b tagging
- q/g tagging
- …

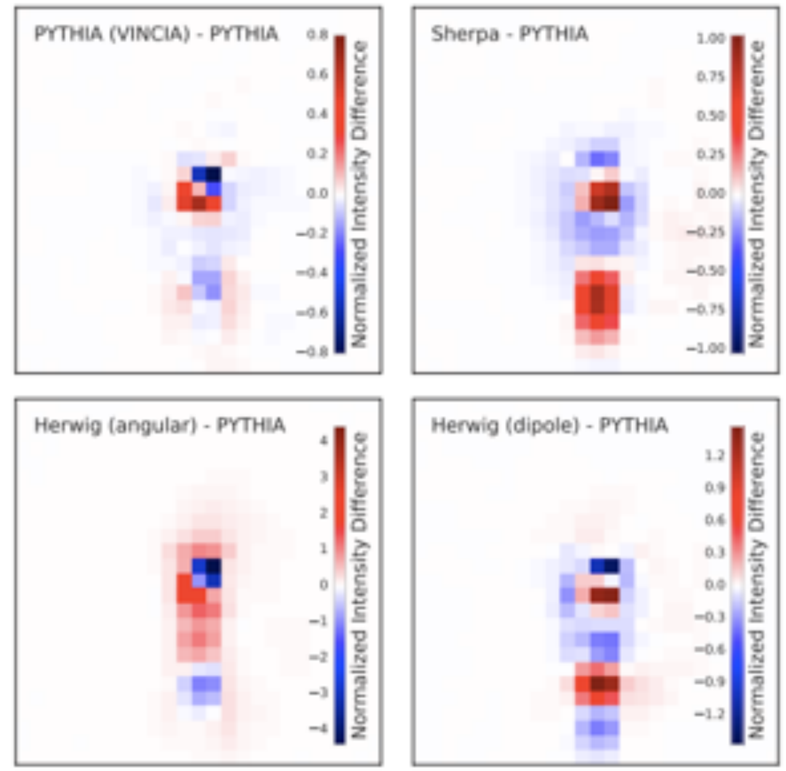
Main Idea: low level information <- <- output info.

(Interpretation)

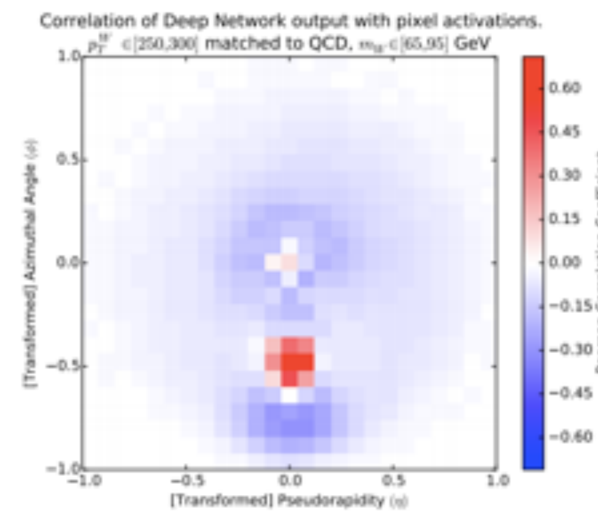
Machine Learning in Jets Physics



(P. Komiske, et al. arXiv: arXiv:1612.01551)



(J. Barnard, et al. arXiv:1609.00607)



(L. de Oliveira, et al. arXiv:1511.05190)

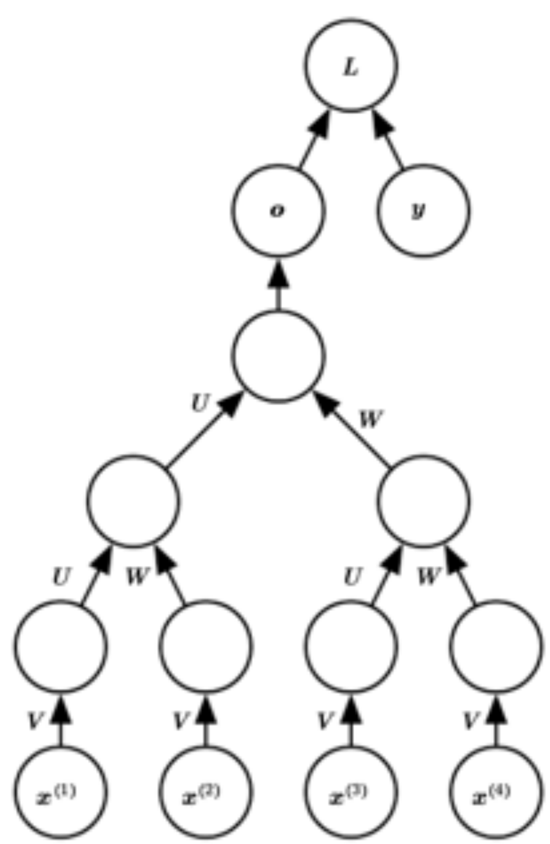
We are concerned with:

- input information
- the representation of the input information
- physics-motivated NNs architecture

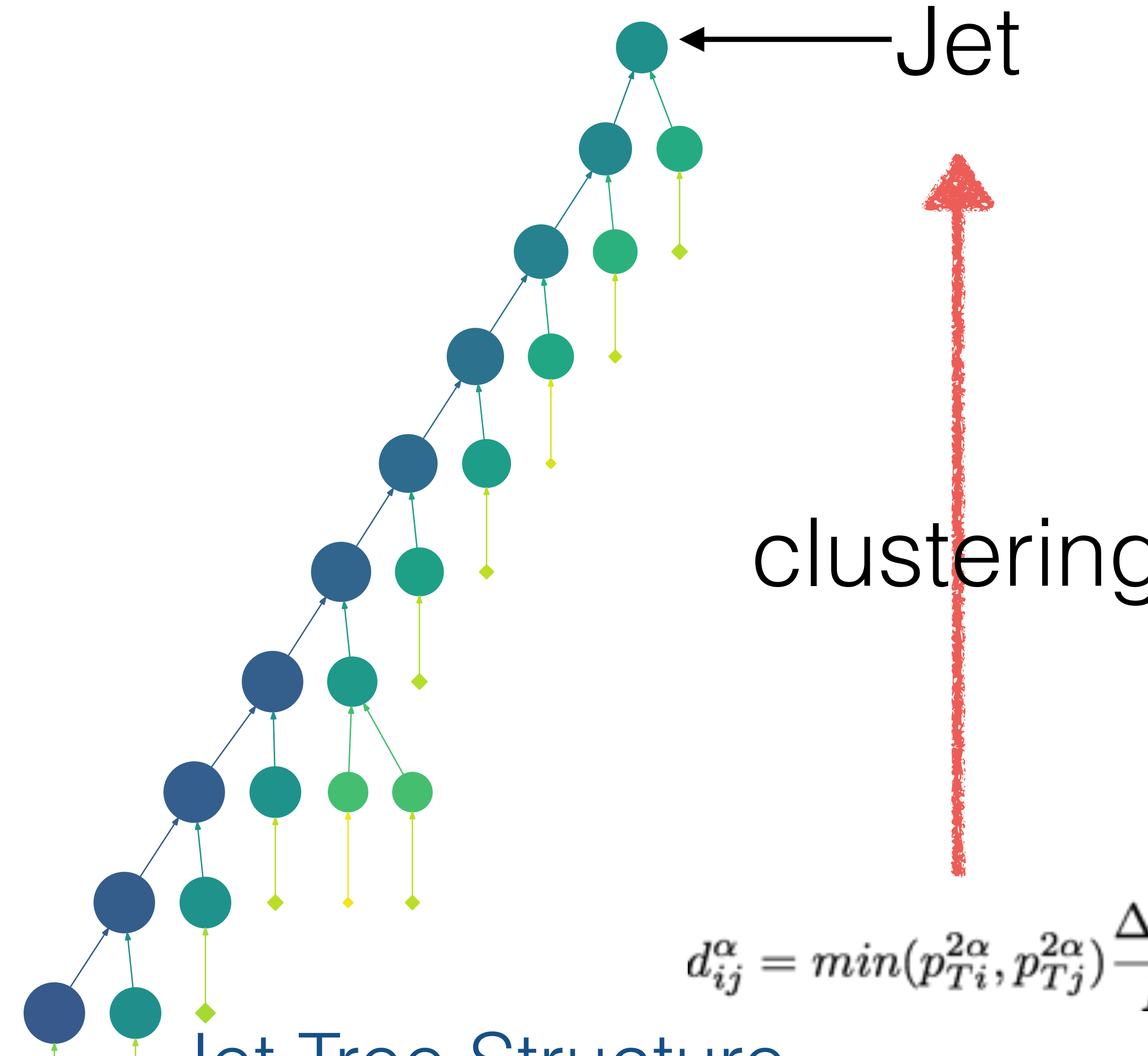
RecNN for Jets

Motivated by:

- problems in image approach: sparsity of jet images (5% - 10% active), fixed image size, (information loss from pixelization)
- natural tree-like structure of sequential jet clustering history
- implementation in event-level



Recursive Neural Nets (RecNN)



$$d_{ij}^{\alpha} = min(p_{Ti}^{2\alpha}, p_{Tj}^{2\alpha})\frac{\Delta R_{ij}^2}{R^2}$$

Jet Tree Structure

RecNN for Jets

Motivated by:

- problems in image approach: sparsity of jet images (5% - 10% active), fixed image size, (information loss from pixelization)
- natural tree-like structure of sequential jet clustering history
- implementation in event-level

[G. Louppe, K. Cho, C. Becot, K. Cranmer, arXiv: 1702.00748]

QCD-Aware Recursive Neural Networks for Jet Physics

Gilles Louppe,[1] **Kyunghyun Cho**,[1] **Cyril Becot**,[1] and **Kyle Cranmer**[1]

[1]*New York University*

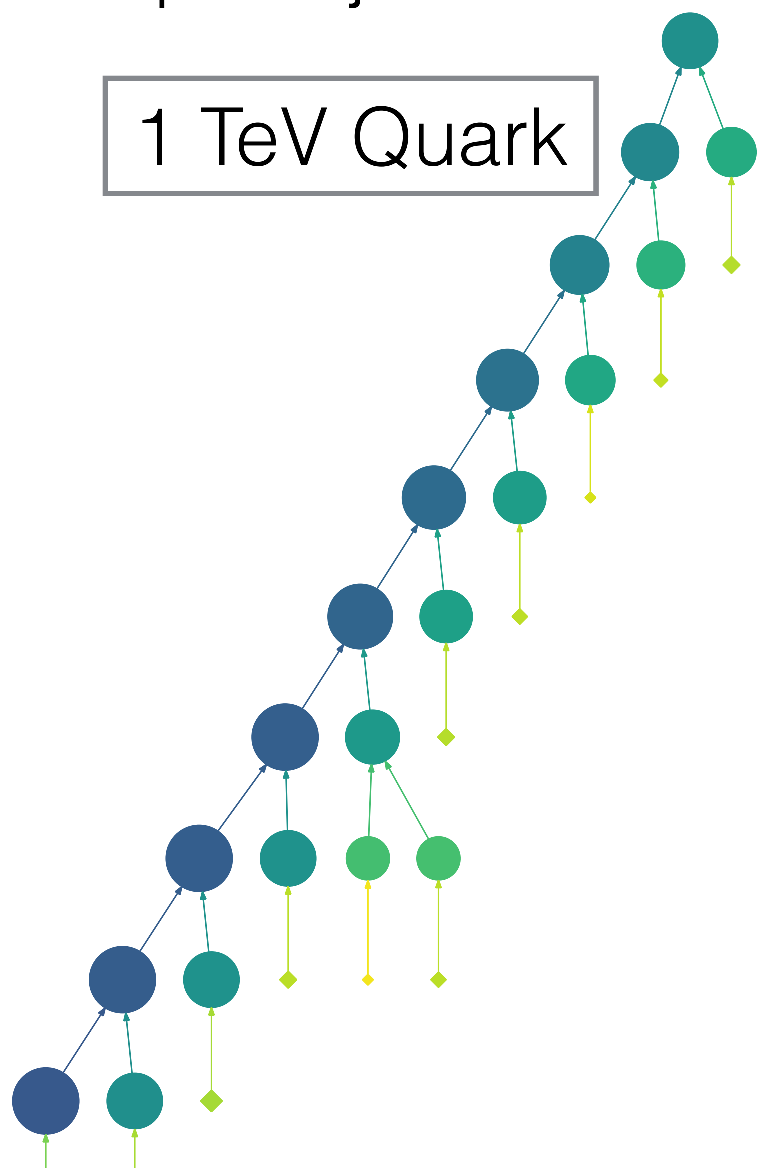
Recent progress in applying machine learning for jet physics has been built upon an analogy between calorimeters and images. In this work, we present a novel class of recursive neural networks built instead upon an analogy between QCD and natural languages. In the analogy, four-momenta are like words and the clustering history of sequential recombination jet algorithms is like the parsing of a sentence. Our approach works directly with the four-momenta of a variable-length set of particles, and the jet-based tree structure varies on an event-by-event basis. Our experiments highlight the flexibility of our method for building task-specific jet embeddings and show that recursive architectures are significantly more accurate and data efficient than previous image-based networks. We extend the analogy from individual jets (sentences) to full events (paragraphs), and show for the first time an event-level classifier operating on all the stable particles produced in an LHC event.

RecNN for Quark/Gluon Tagging

Quark jet v.s. Gluon jet $\longrightarrow$ Different Radiation Patterns

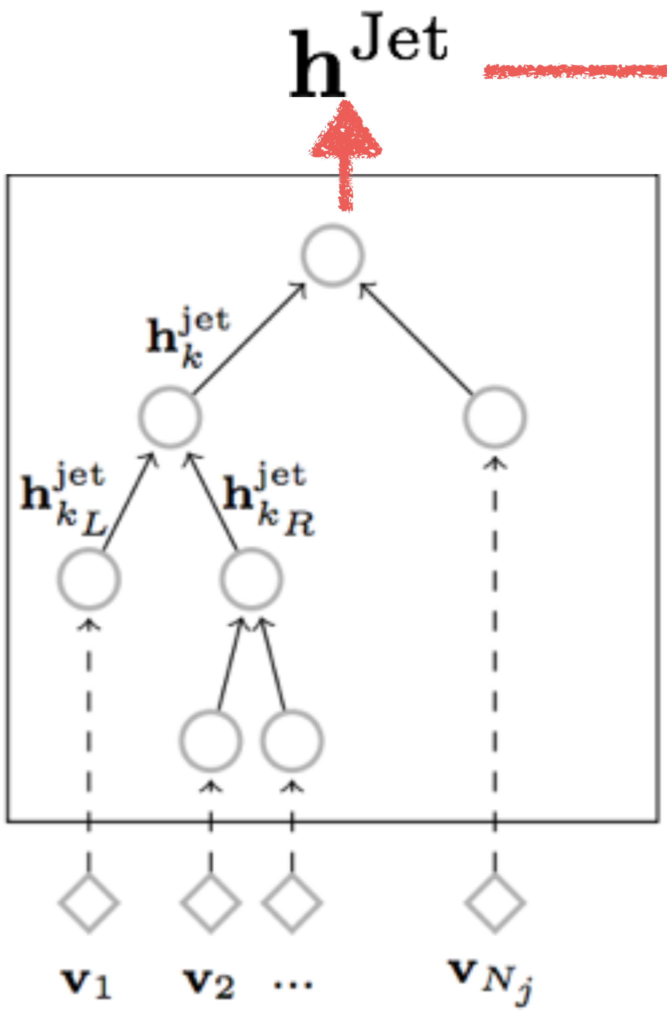
$$\langle N \rangle_g / \langle N \rangle_q \sim C_A / C_F \sim 2$$

~50 for quark jets and ~90 for gluon jets @ pt = 1 TeV



Conventionally, track count has been the most powerful discriminant for q/g tagging

RecNN & Jet Embedding



(taken from G. Louppe, K. Cho, C. Becot, K. Cranmer, arXiv: 1702.00748)

Classifier

$R^f \to R^q$

$$\boxed{\mathbf{h}_k = \sigma\left(W_h \begin{bmatrix} \mathbf{h}_{k_L}^{\mathrm{jet}} \\ \mathbf{h}_{k_R}^{\mathrm{jet}} \\ \mathbf{u}_k \end{bmatrix} + b_h\right)} \quad \mathbf{u}_k = \sigma(W_u \mathbf{x}_k + b_u)$$

$(p_T, \eta, \phi, \ldots)_i$ and particle flow identification

- One hot vector $((i_{\text{neutral hadron}}, i_{\text{photon}}, i_+, i_-), i = 0 \text{ or } 1)$
- pt-weighted charge $Q_k^{\mathrm{rec}} = \dfrac{Q_{k_L}^{\mathrm{rec}}(p_T^{k_L})^\kappa + Q_{k_R}^{\mathrm{rec}}(p_T^{k_R})^\kappa}{(p_T^k)^\kappa}$

* with recursively defined pt-weighed charge, we can include the particle flow information in one variable which is well defined for all the nodes

Workflow

Pythia8

↓

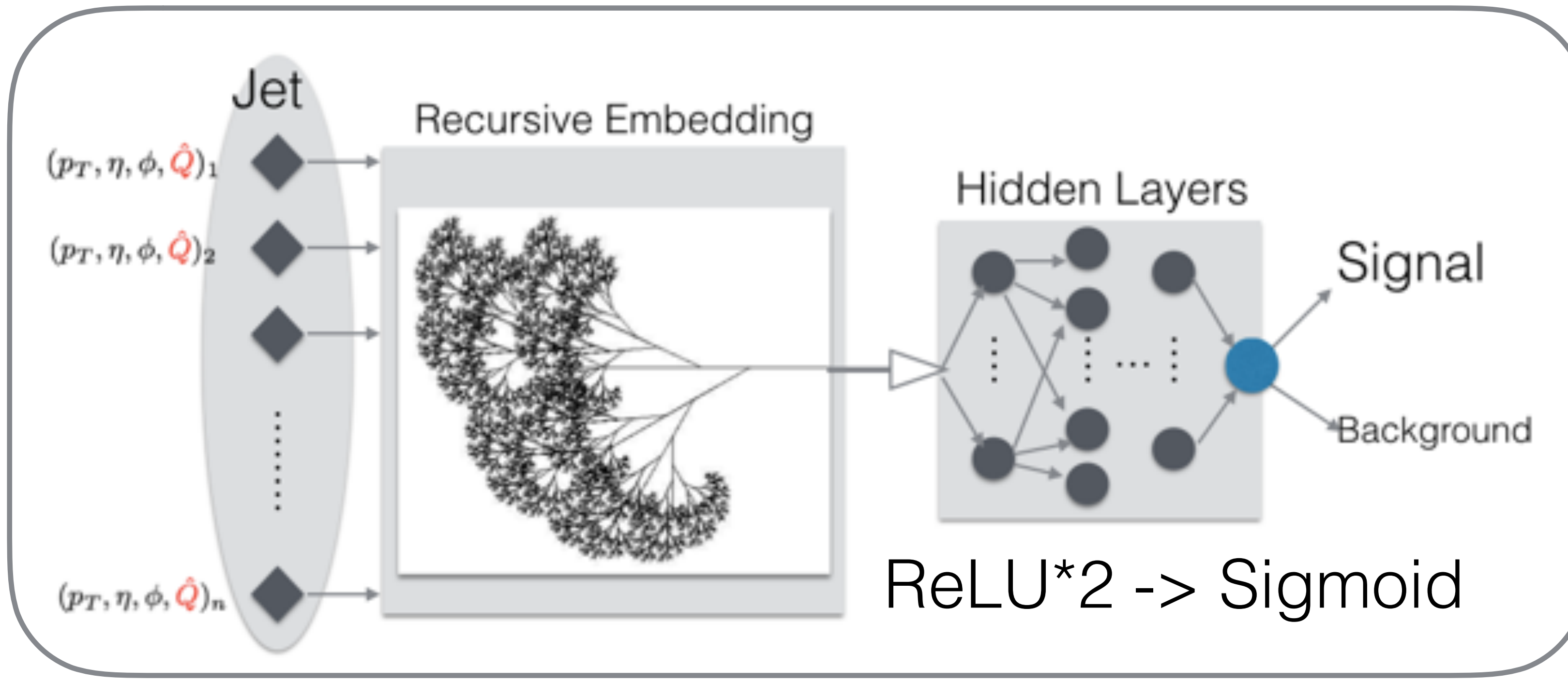
Delphes

[DeepJets, J. Barnard, E. N. Dawe, M. J. Dolan, N. Rajcic, arXiv: 1609.00607]

↓

Data Preprocessing →

RecNN



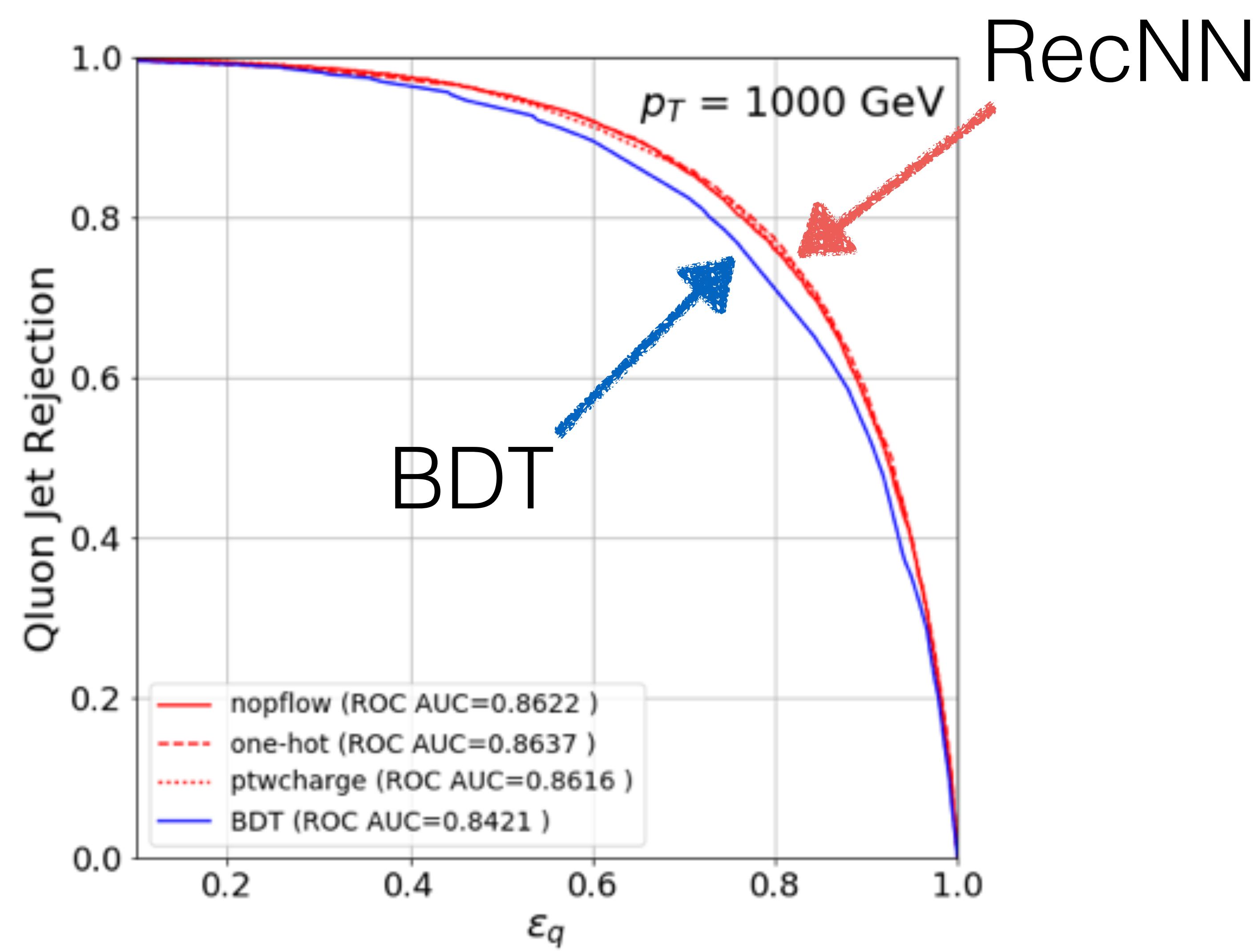
- Measure: ROC (AUC), background rejection rate @ $\epsilon_s = 50\%$
- Particle Flow Identification: one-hot vectors, or pt weighted charge

Main Results

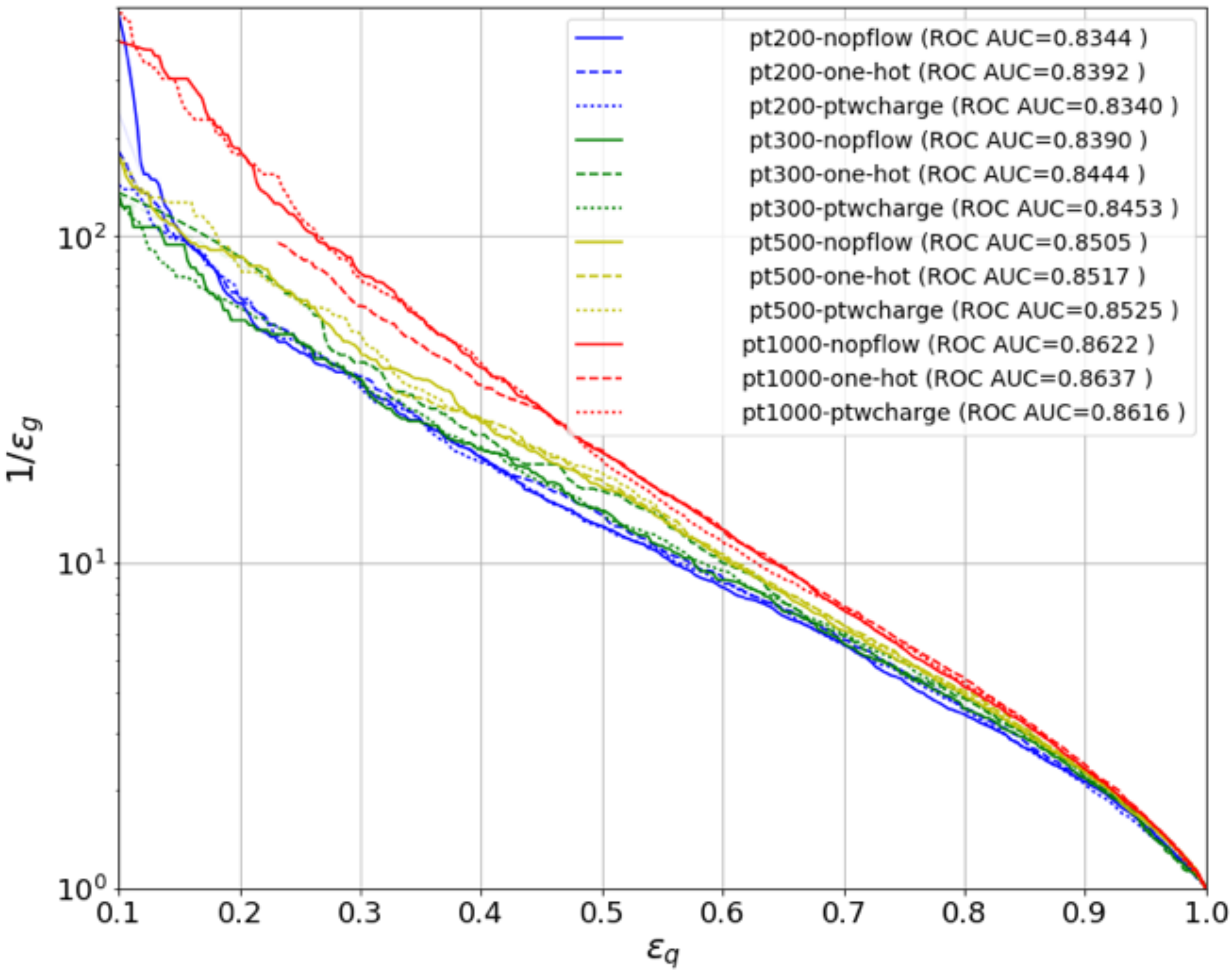
Baseline: BDT (jet mass m/p_T, jet girth $\sum_{i\in\mathrm{Jet}} \frac{p_T^i}{p_T^J} r_i$, charged particle count $\#_{\mathbf{charged}}$)

For RecNN,

- no particle flow identification
- one-hot vectors
- pt-weighted charge instead

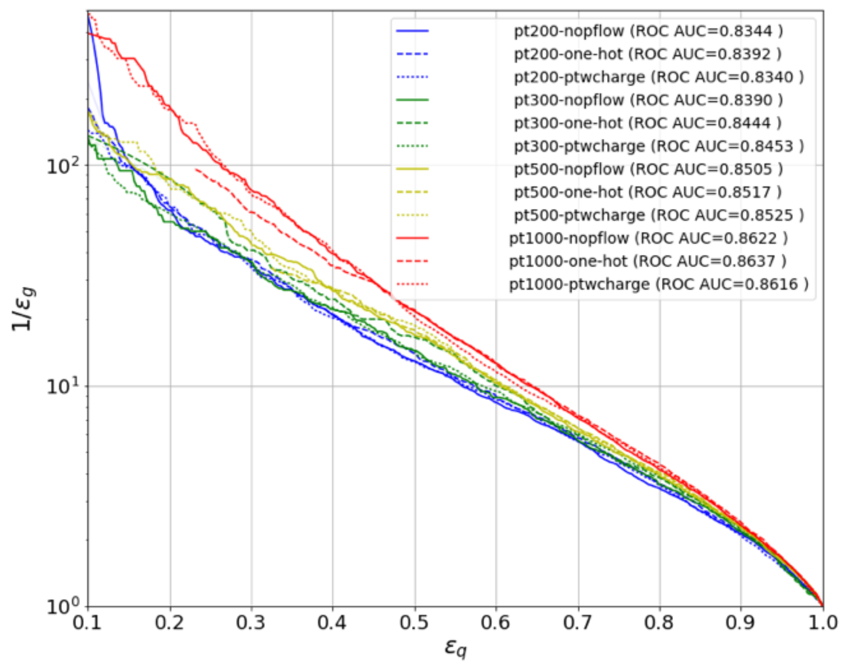


Main Results



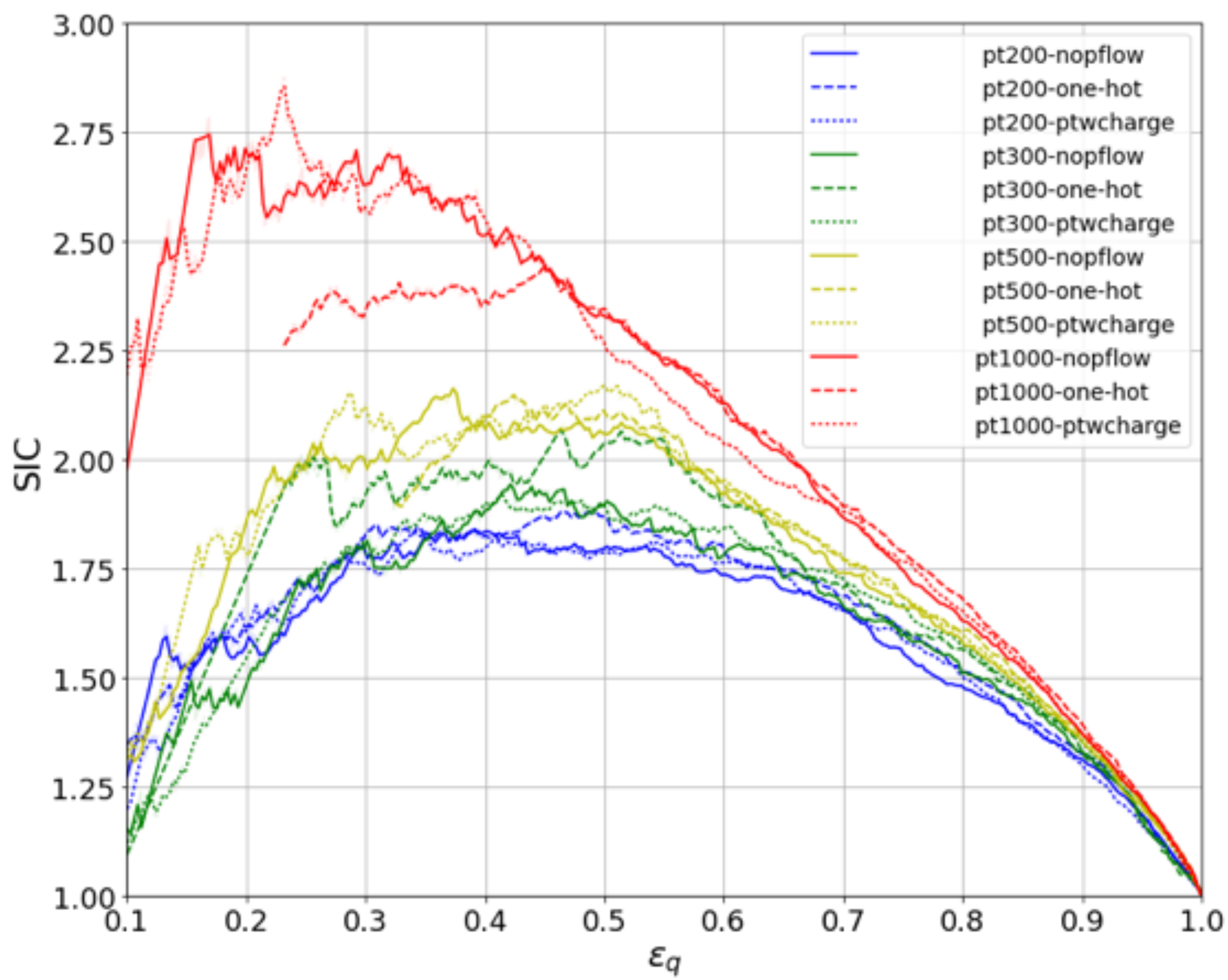
Jet pts: 200, 300, 500, 1000 GeV

Main Results



$$\sigma \equiv \frac{S}{\sqrt{B}} \rightarrow \frac{\epsilon_S S}{\sqrt{\epsilon_B B}} = \left(\frac{\epsilon_S}{\sqrt{\epsilon_B}}\right)\sigma \rightarrow \mathrm{SI} = \frac{\epsilon_S}{\sqrt{\epsilon_B}}$$

Significance Improvement



Jet pts: 200, 300, 500, 1000 GeV

Variants

[pt=200 GeV]

Variants	AUC	$R_{\epsilon=50\%}$
Baseline	0.8344	12.9
R=0.7	0.8210	12.4
$W_h \to R^{q\times 2q}$	0.8268	12.3
$W_h \to R^{q\times 2q}$ with one-hot	0.8313	13.7
$\mathbf{x}$=(p_T, η, ϕ)	0.8291	11.8
$\mathbf{x}$=(η, ϕ)	0.8249	11.9
$\mathbf{x}$=(p_T)	0.8264	11.6
only one-hot	0.8255	11.9
$\mathbf{x}$=($Q^{\text{rec}}_{\kappa=50\%}$)	0.8234	11.3

$$\mathbf{h}_k = \sigma\left(W_h \begin{bmatrix} \mathbf{h}^{\text{jet}}_{k_L} \\ \mathbf{h}^{\text{jet}}_{k_R} \\ \mathbf{u}_k \end{bmatrix} + b_h\right)$$

Variants in input information

- particle flow identification doesn't help significantly
- the discriminating information for q/g tagging is RecNN mainly reside in the tree structure itself

Variants

$$\mathbf{h}_k = \sigma\left(W_h \begin{bmatrix} \mathbf{h}^{\text{jet}}_{k_L} \\ \mathbf{h}^{\text{jet}}_{k_R} \end{bmatrix} + b_h\right)$$

[pt=200 GeV]

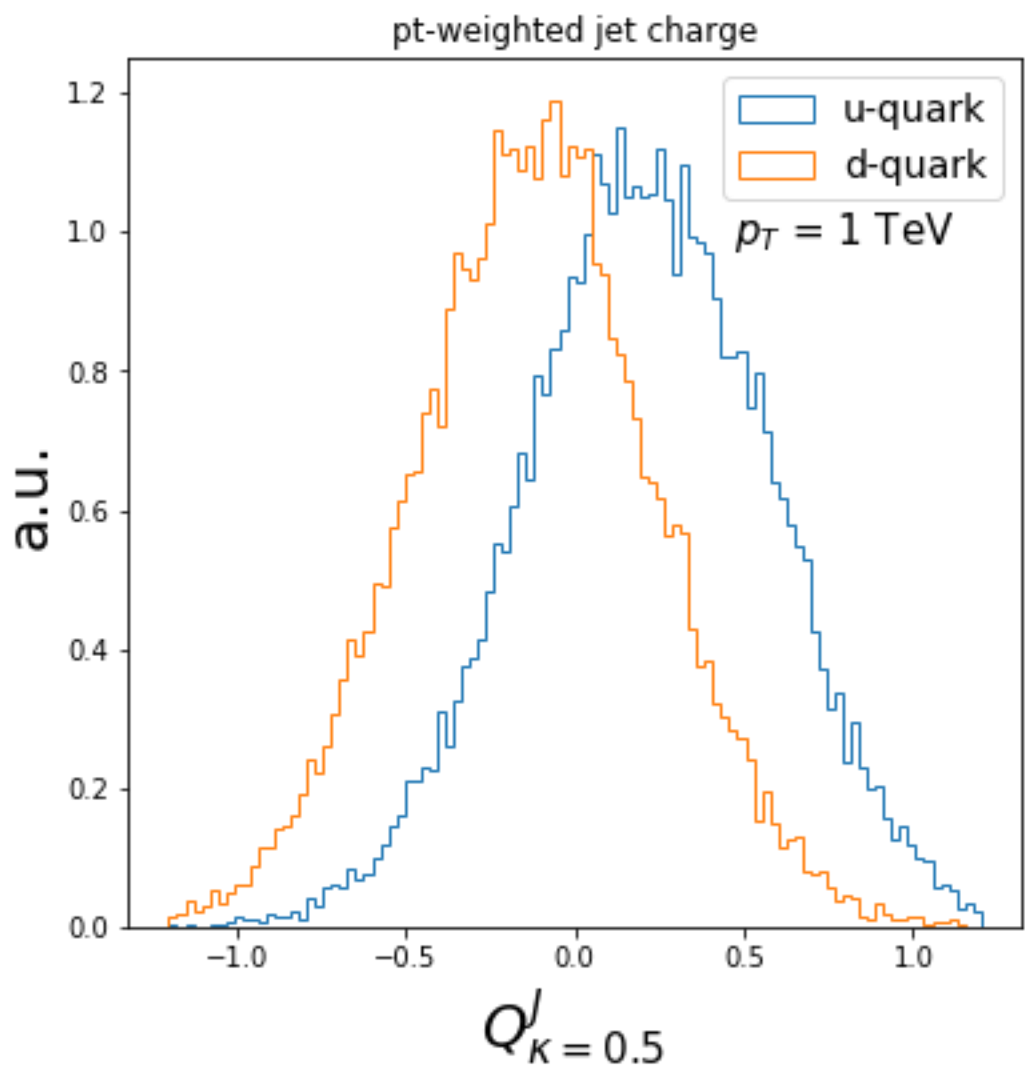
Variants	AUC	$R_{\epsilon=50\%}$
Baseline	0.8344	12.9
R=0.7	0.8210	12.4
$W_h \to R^{q\times 2q}$	0.8268	12.3
$W_h \to R^{q\times 2q}$ with one-hot	0.8313	13.7
$\mathbf{x}=(p_T,\, \eta,\, \phi)$	0.8291	11.8
$\mathbf{x}=(\eta,\, \phi)$	0.8249	11.9
$\mathbf{x}=(p_T)$	0.8264	11.6
only one-hot	0.8255	11.9
$\mathbf{x}=(Q^{\text{rec}}_{\kappa=50\%})$	0.8234	11.3

Variants in input information

- particle flow identification doesn’t help significantly
- the discriminating information for q/g tagging is RecNN mainly reside in the tree structure itself

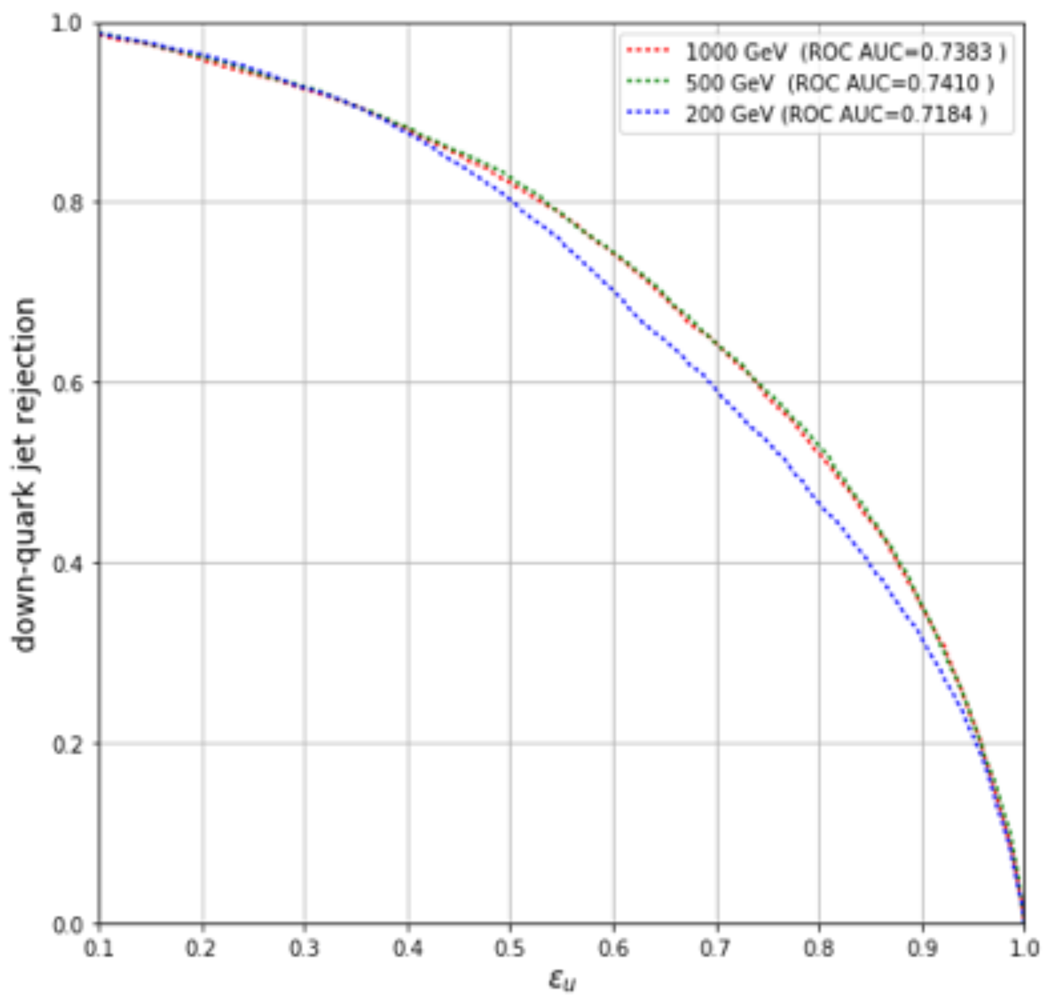
Jet Charge

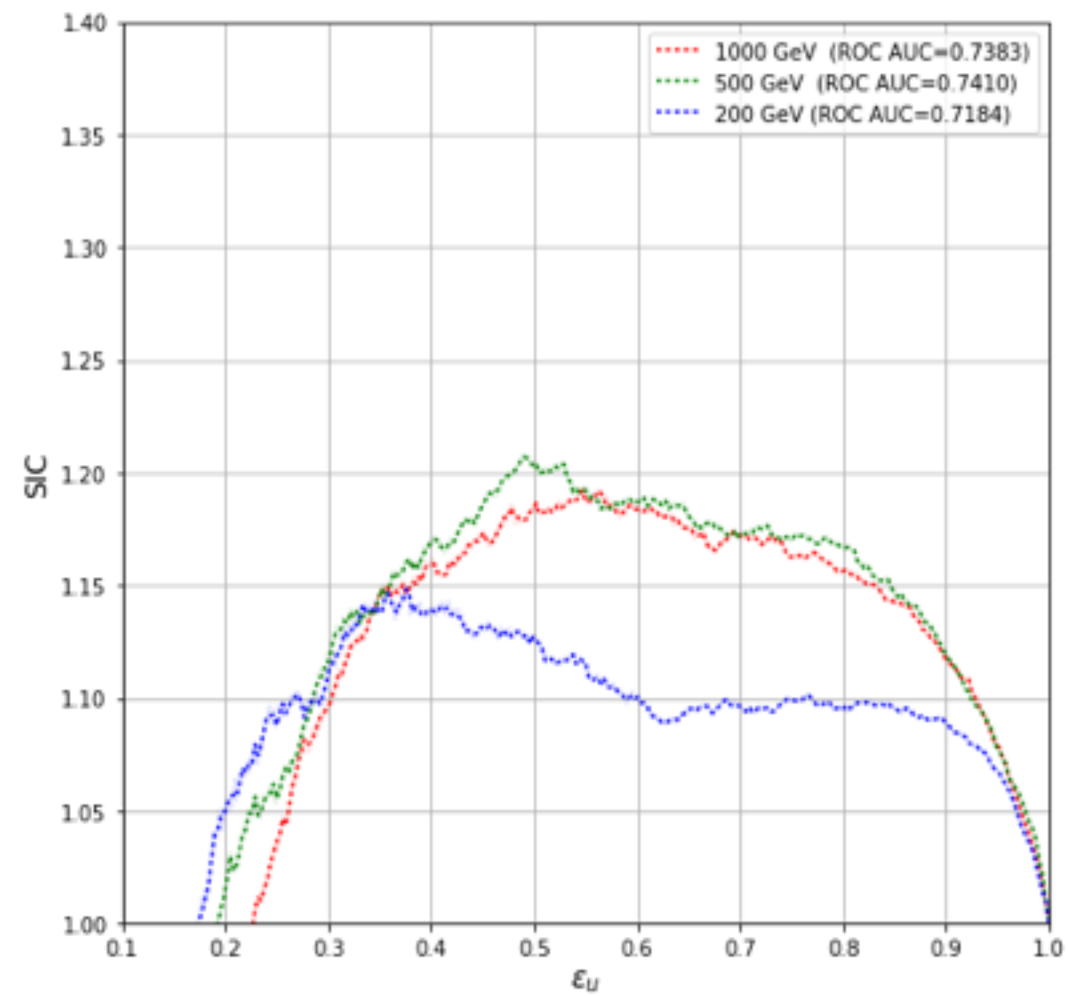
pt weighted jet charge $Q^J_\kappa = \sum_{i \in J} (\frac{p^i_T}{p^J_T})^\kappa q_i$



Jet Charge

u/d discrimination





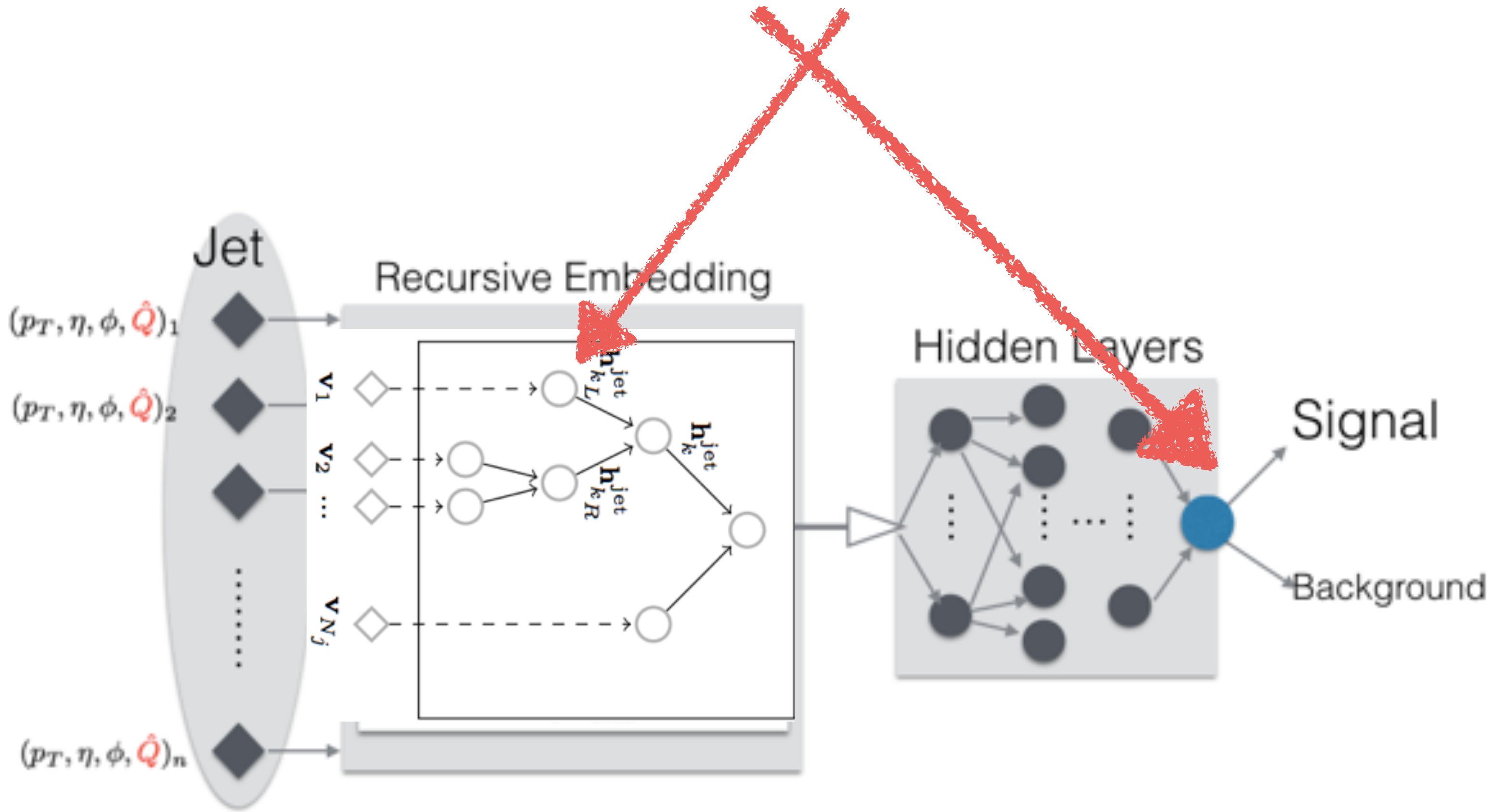
RecNNs with pt-weighted charge

* one-hot implementation doesn't work here

Visualisation

Sensitivity indicated by gradients

$||df/dh||$

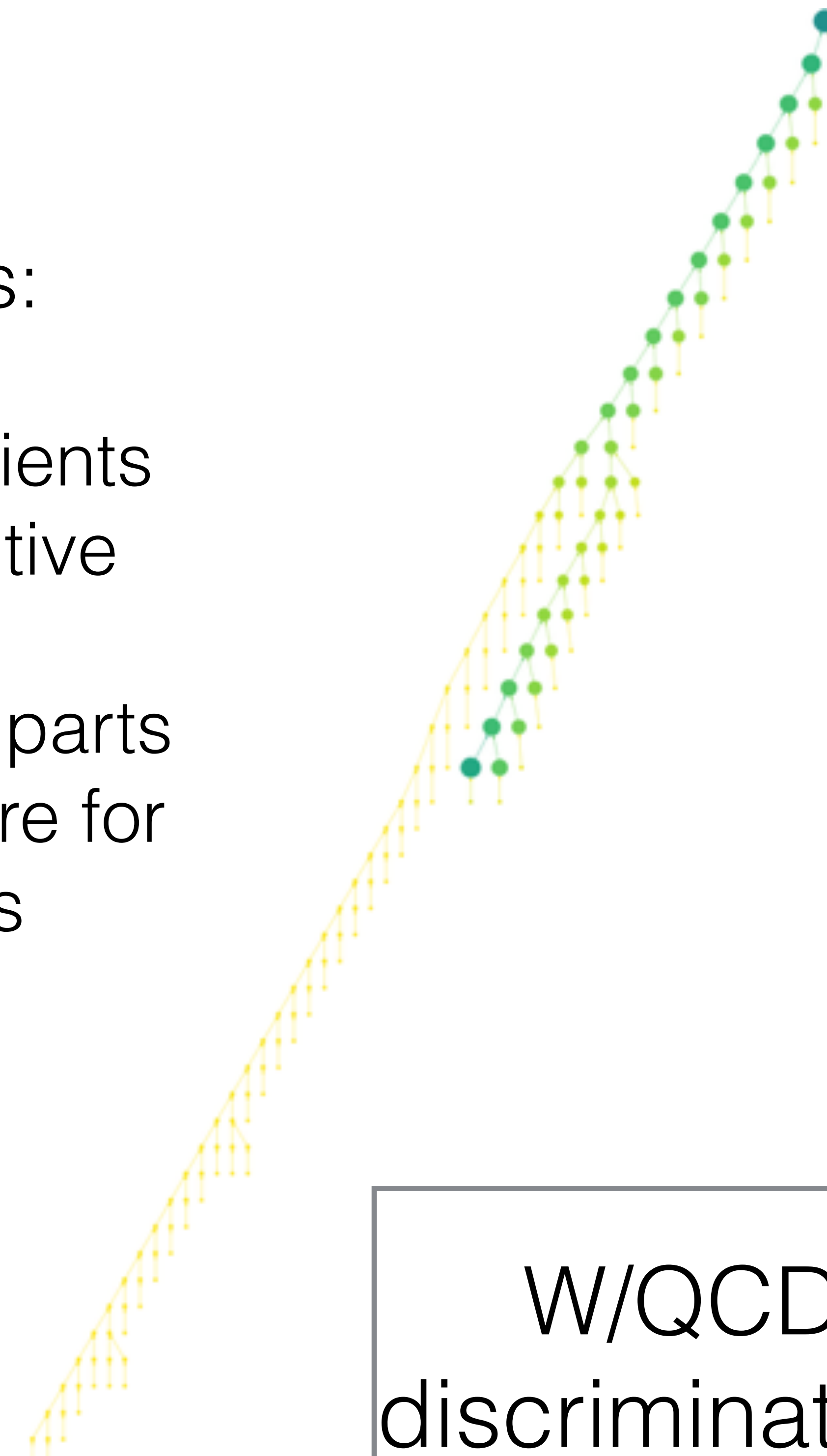


(work in progress)

Visualisation

Observations:

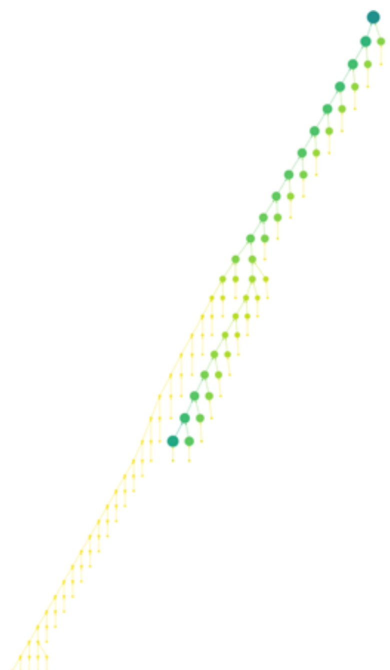
The largest gradients (thus most sensitive nodes) reside in different parts of the tree structure for different tasks



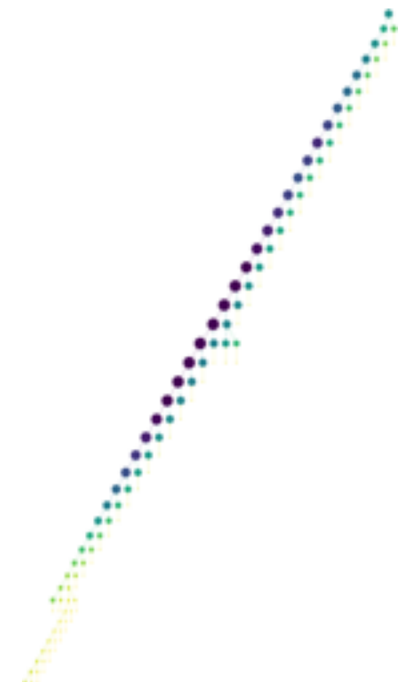
q/g discrimination

W/QCD discrimination

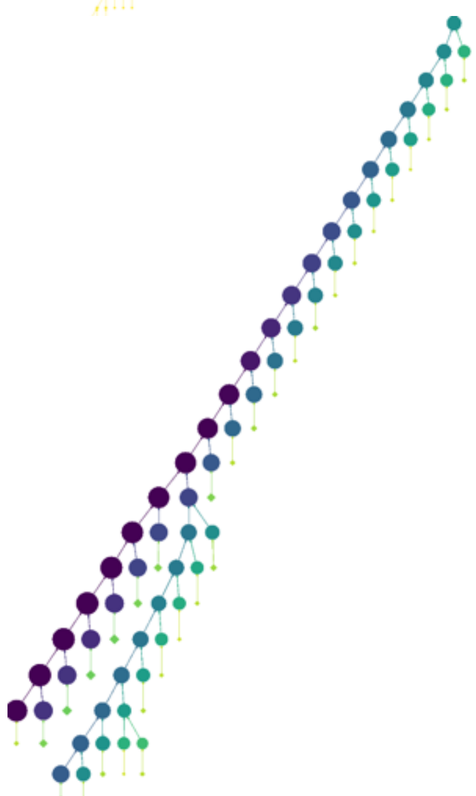
Visualisation



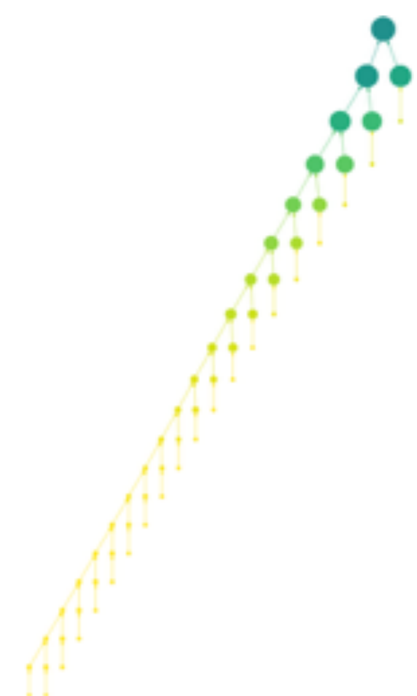
W Tagging



Top Tagging



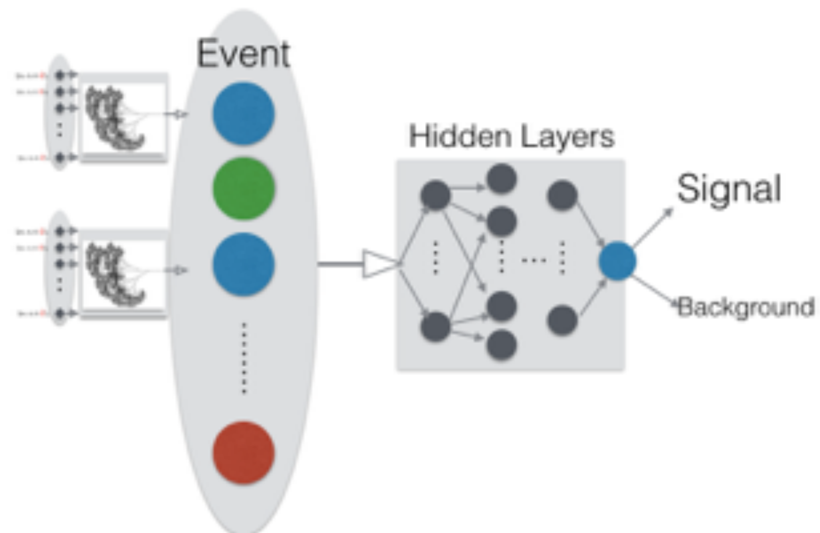
q/g Tagging

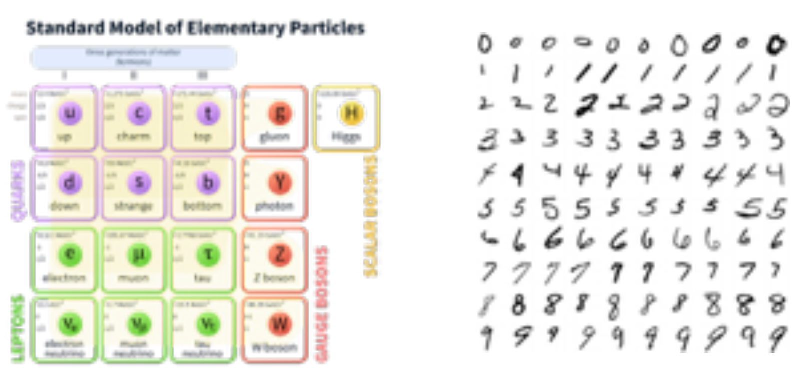


u/d Tagging

(work in progress)

Outlook

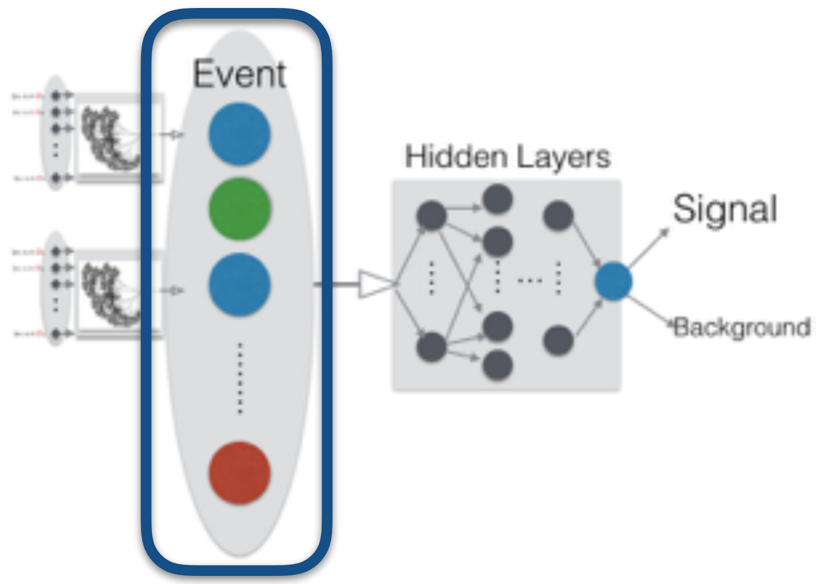


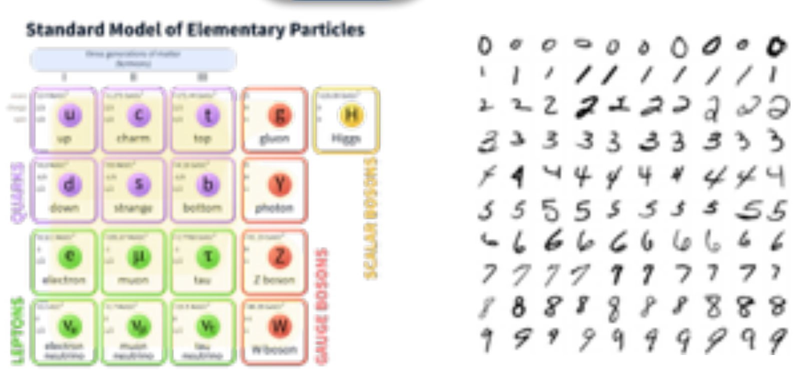


SM as MNIST

- Event-level analysis
- Multiclass classification
- Jet Algorithms
- Applications

Outlook

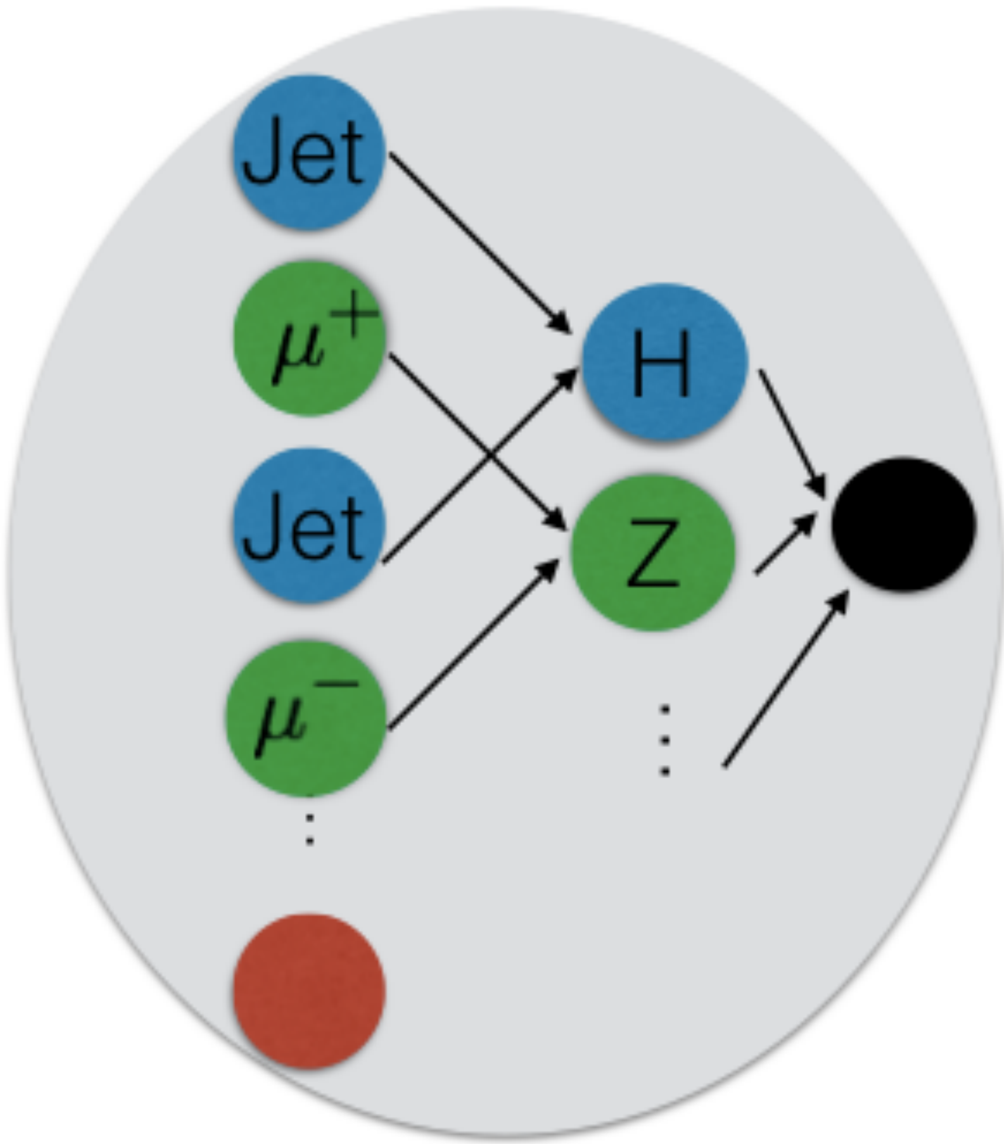




SM as MNIST

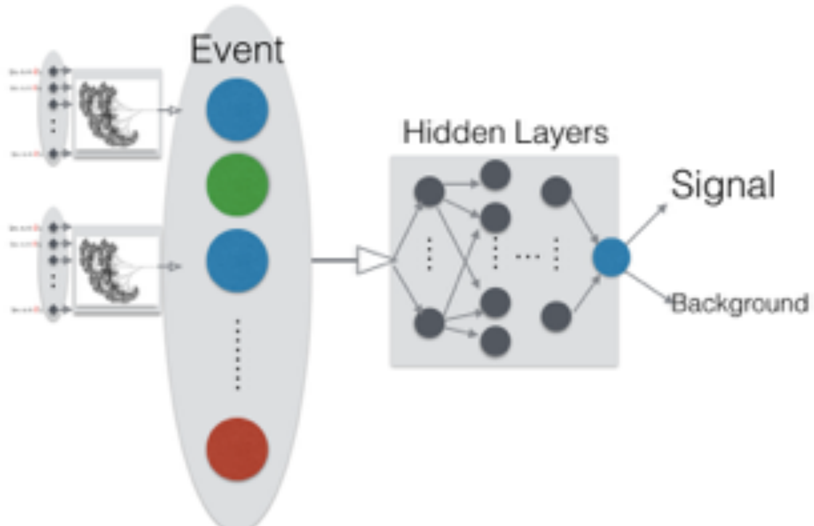
- Event-level analysis
- Multiclass classification
- Jet Algorithms
- Applications

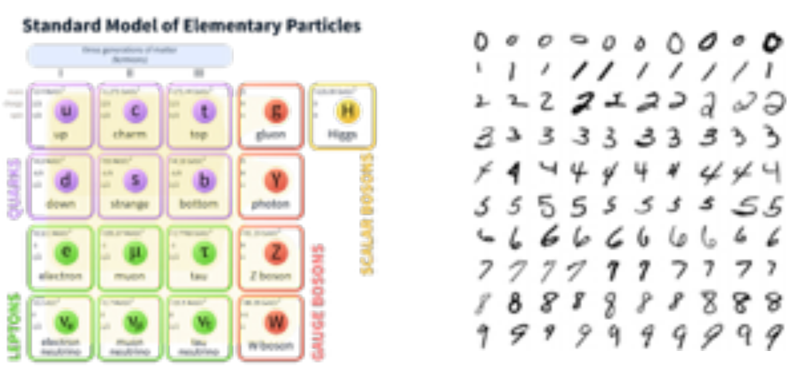
Outlook



- Event-level analysis
- Multiclass classification
- Jet Algorithms
- Applications

Outlook





SM as MNIST

- Event-level analysis
- Multiclass classification
- Jet Algorithms
- Applications

Summary

What has been done:

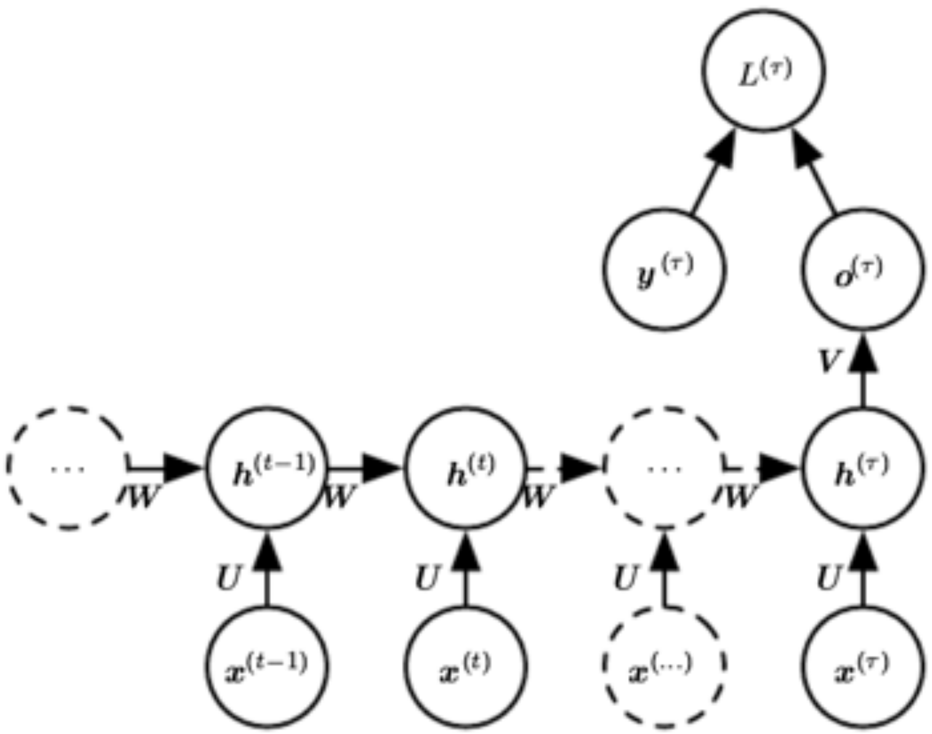
- examined performance of RecNNs in q/g tagging in detail
- explored different variants of the networks (which shows that the main information is included in the tree-structure itself)

What to expect:

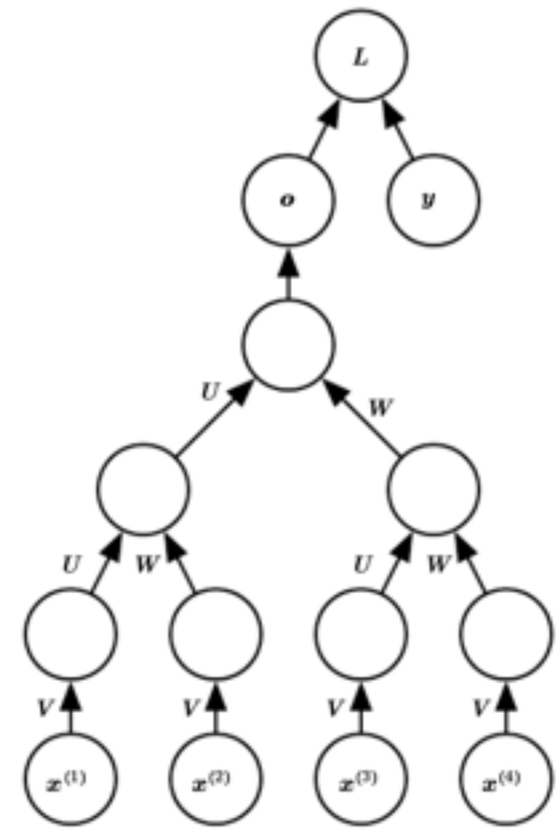
- Visualisation study
- Full event analysis
- Jet clustering implemented with DNNs

Thank you!

Backup



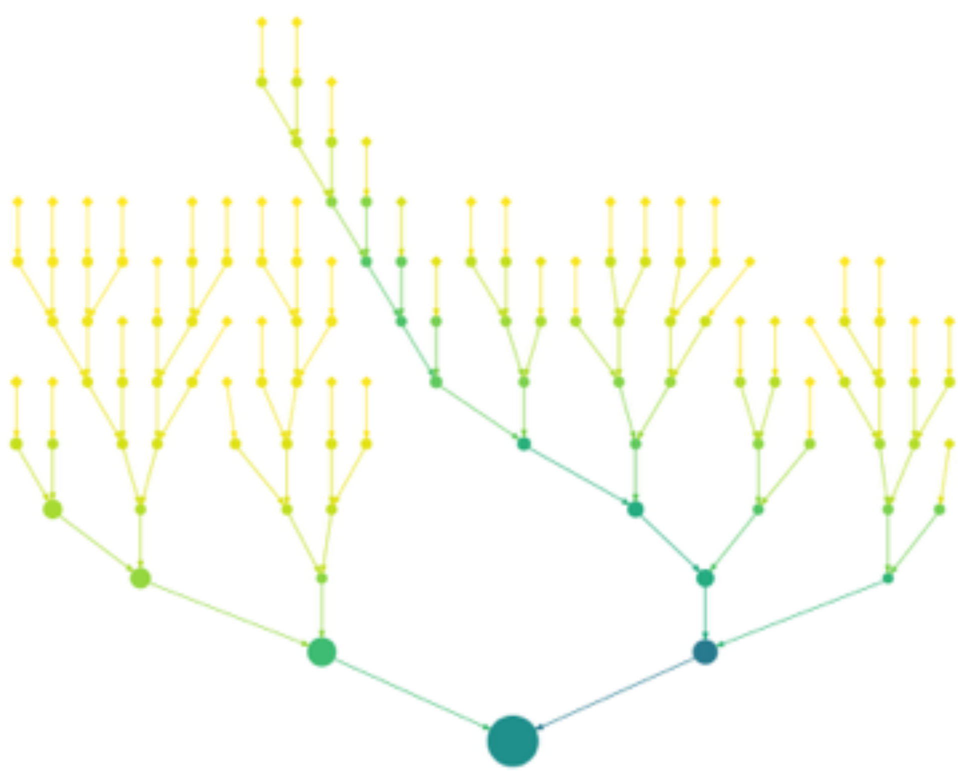
RNN



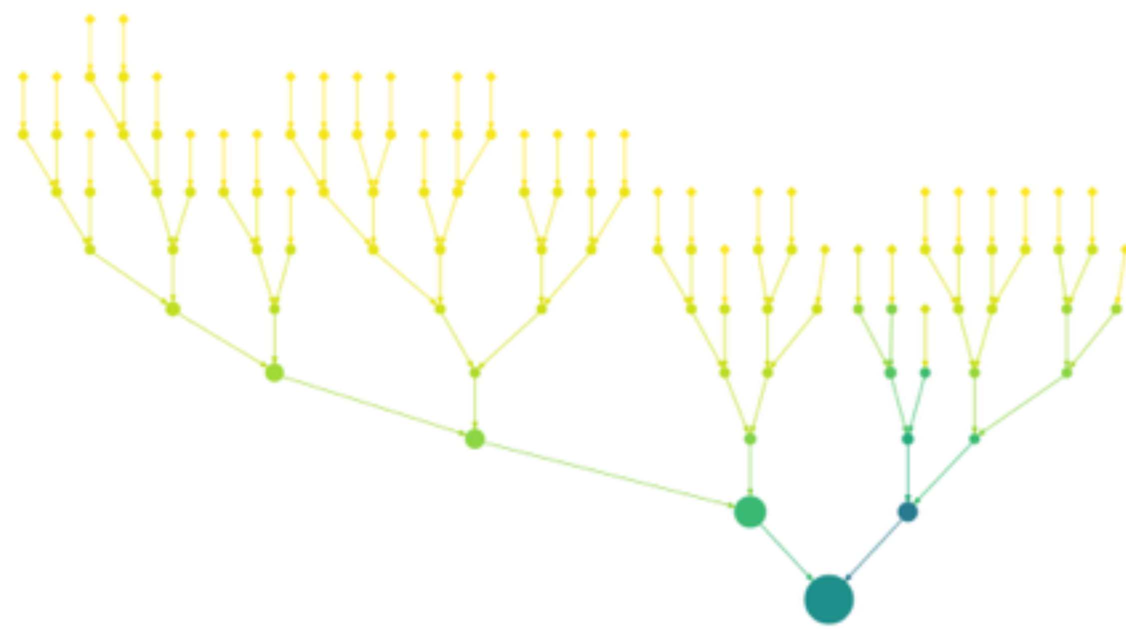
RecNN

Backup

W/QCD Tagging



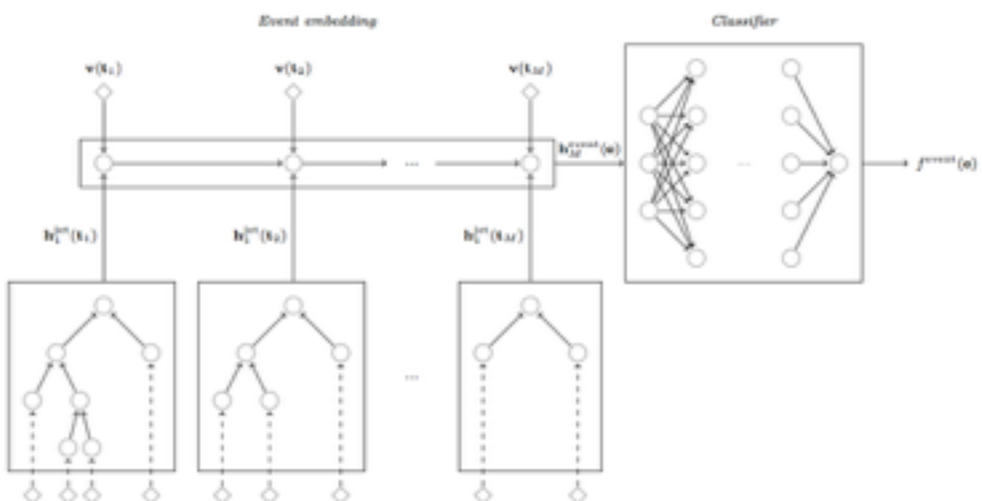
W Jet



QCD Jet

(work in progress)

Backup



In Gilles' work, it has been shown that only the 2 hardest jets give the best results. Other soft jets even deflect performance

Input	ROC AUC	$R_{\epsilon=80\%}$
Hardest jet		
$\mathbf{v}(\mathbf{t}_j)$	0.8909 ± 0.0007	5.6 ± 0.0
$\mathbf{v}(\mathbf{t}_j)$, $\mathbf{h}_j^{\text{jet}(k_t)}$	**0.9602 ± 0.0004**	**26.7 ± 0.7**
$\mathbf{v}(\mathbf{t}_j)$, $\mathbf{h}_j^{\text{jet}(\text{desc}-p_T)}$	0.9594 ± 0.0010	25.6 ± 1.4
2 hardest jets		
$\mathbf{v}(\mathbf{t}_j)$	0.9606 ± 0.0011	21.1 ± 1.1
$\mathbf{v}(\mathbf{t}_j)$, $\mathbf{h}_j^{\text{jet}(k_t)}$	0.9866 ± 0.0007	156.9 ± 14.8
$\mathbf{v}(\mathbf{t}_j)$, $\mathbf{h}_j^{\text{jet}(\text{desc}-p_T)}$	**0.9875 ± 0.0006**	**174.5 ± 14.0**
5 hardest jets		
$\mathbf{v}(\mathbf{t}_j)$	0.9576 ± 0.0019	20.3 ± 0.9
$\mathbf{v}(\mathbf{t}_j)$, $\mathbf{h}_j^{\text{jet}(k_t)}$	0.9867 ± 0.0004	152.8 ± 10.4
$\mathbf{v}(\mathbf{t}_j)$, $\mathbf{h}_j^{\text{jet}(\text{desc}-p_T)}$	**0.9872 ± 0.0003**	**167.8 ± 9.5**
No jet clustering, desc-p_T on $\mathbf{v}_i$		
$i = 1$	0.6501 ± 0.0023	1.7 ± 0.0
$i = 1, \ldots, 50$	**0.8925 ± 0.0079**	**5.6 ± 0.5**
$i = 1, \ldots, 100$	0.8781 ± 0.0180	4.9 ± 0.6
$i = 1, \ldots, 200$	0.8846 ± 0.0091	5.2 ± 0.5
$i = 1, \ldots, 400$	0.8780 ± 0.0132	4.9 ± 0.5

Right now, the pre-clustering is still necessary. However, it will be interesting if we combine the jet clustering and event-level analysis.