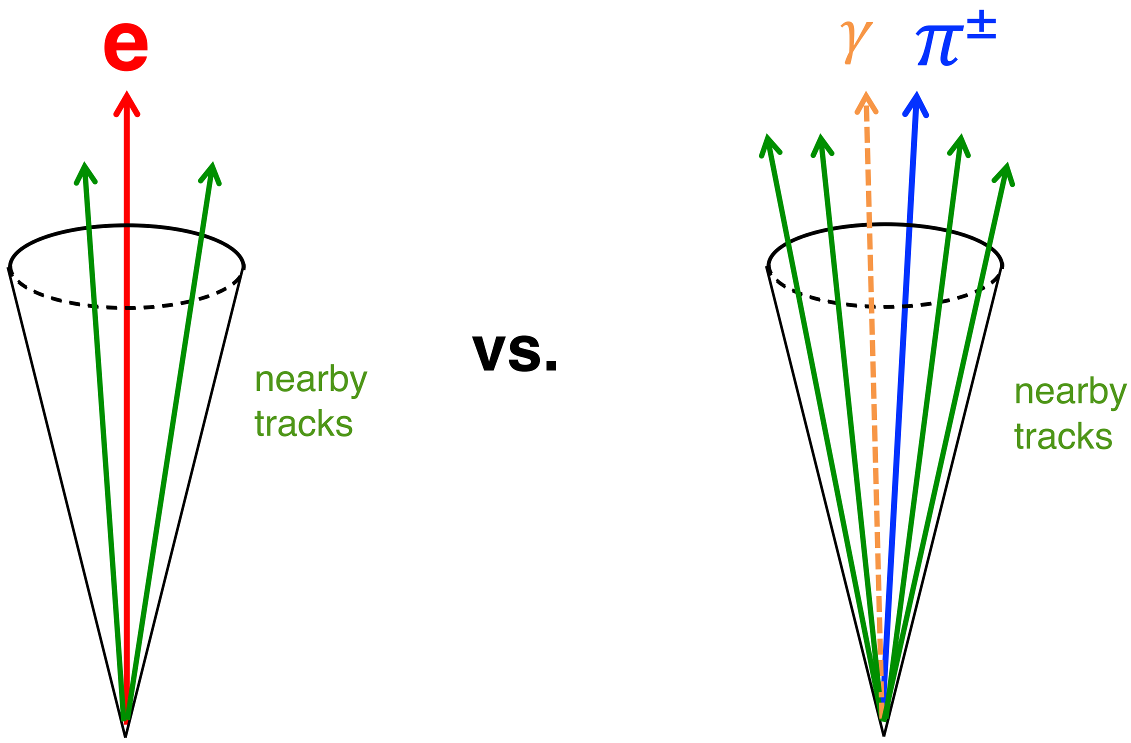
# Machine Learning for (Future) Colliders

**Ben Hooberman,**
**University of Illinois at Urbana-Champaign**

# Outline

- Classification techniques at colliders
- Brief overview of neural networks
- Machine learning use cases at (future) colliders
  - Fast simulation
  - Tracking with unsupervised learning
  - Jet classification
  - Particle ID
  - Event-based classification
- Summary

# Intro: Classification at Colliders




prompt electron
**(signal)**

jet → electron fake
**(background)**

How do we identify electrons at LHC?

## 1. Cut-based selection

– Apply requirements on human-designed **features**

**ATLAS Run 1 electron features [1]:**

| Category | Description | Variable |
|---|---|---|
| *loose* | | |
| Acceptance | $\|\eta\| < 2.47$ | |
| Hadronic leakage | In $\|\eta\| < 0.8$ and $\|\eta\| > 1.37$: ratio of $E_T$ in the first layer of the hadronic calorimeter to $E_T$ of the EM cluster | $R_{\mathrm{had},1}$ |
| | In $0.8 < \|\eta\| < 1.37$: ratio of $E_T$ in whole hadronic calorimeter to $E_T$ of the EM cluster | $R_{\mathrm{had}}$ |
| Middle layer of the EM | Ratio of energies in $3 \times 7$ cells over $7 \times 7$ cells | $R_\eta$ |
| | Lateral width of the shower | $w_{\eta 2}$ |
| Front layer of the EM | Total shower width | $w_{\mathrm{stot}}$ |
| | Energy difference of the largest and second largest energy deposits in the cluster divided by their sum | $E_{\mathrm{ratio}}$ |
| Track quality and track–cluster matching | Number of hits in the pixel detector (>0) | |
| | Number of hits in the silicon detectors ($\geq 7$) | |
| | $\|\Delta\eta\|$ between the cluster position in the first layer and the extrapolated track (<0.015) | $\Delta\eta_1$ |


[1] EPJC 74 (2014) 2941

# Classification Techniques at Colliders

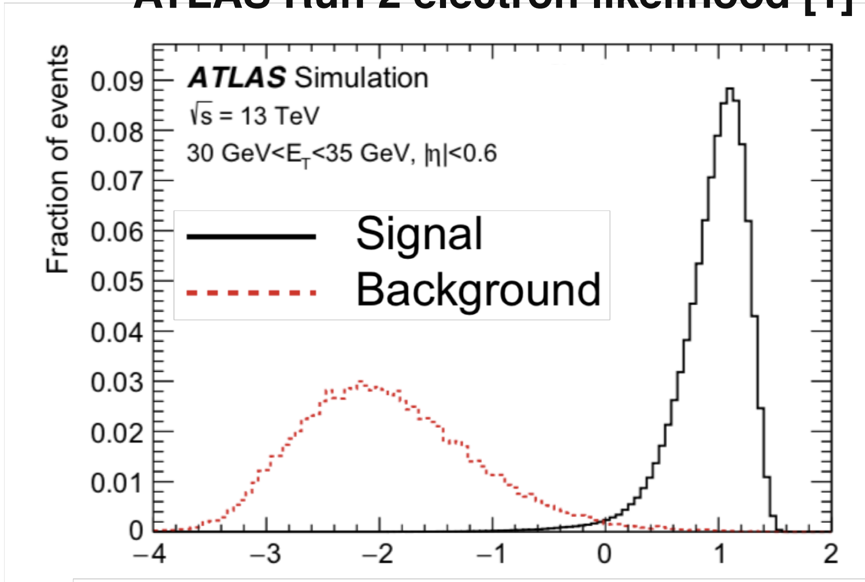
## 1. Cut-based selection

– Apply requirements on human-designed **features**

## 2. Multi-Variate Algorithms (MVA)

– Combine features using *neural networks, boosted decision trees, likelihoods, etc.*
– **Exploit *correlations* between features**

**ATLAS Run 2 electron likelihood [1]**




Log-transformed likelihood discriminant

[1] EPJC 79 (2019) 639

# Classification Techniques at Colliders

## 1. Cut-based selection

– Apply requirements on human-designed **features**

## 2. Multi-Variate Algorithms (MVA)

– Combine features using *neural networks*, *boosted decision trees*, *likelihoods, etc.*

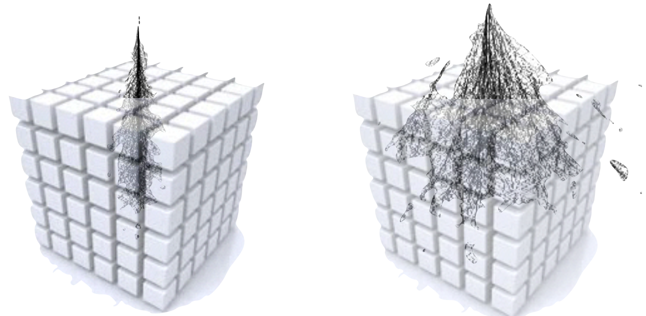
– **Exploit *correlations* between features**

## 3. Deep Learning

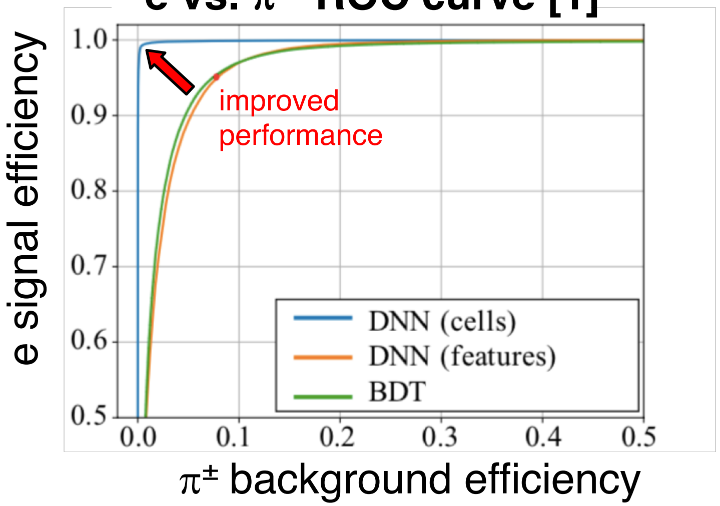
– Feed *low-level data* (e.g. calorimeter cells) directly to **deep neural networks**

– ***Potential to exploit information not contained in features***

single particle showers in a high-granularity 3D calorimeter




**e vs. $\pi^\pm$ ROC curve [1]**




[1] BH, Farbin, Khattak, Pacela, Pierini, Vlimant, Spiropulu, Wei, Proceedings of the Deep Learning for Physical Sciences Workshop at Neural Information and Processing Systems (NIPS17)

# Classification Techniques at Colliders

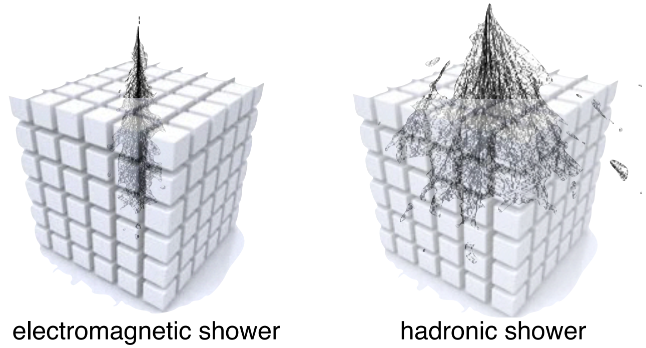
## 1. Cut-based selection
- Apply requirements on human-designed **features**

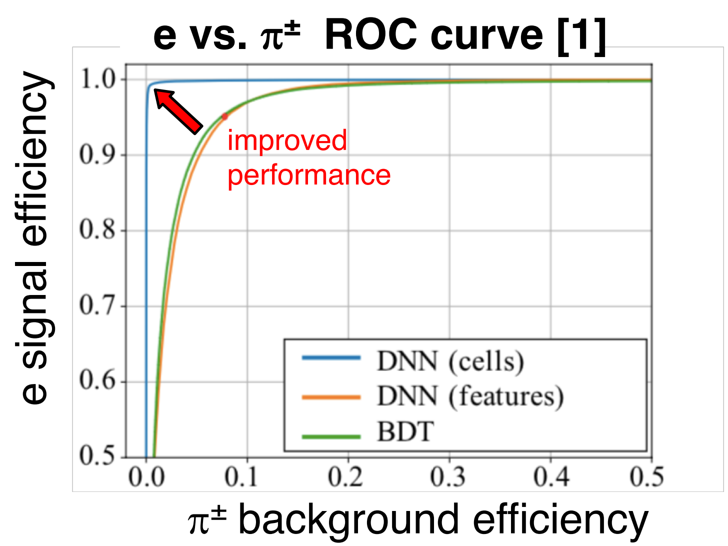
**machine learning**

## 2. Multi-Variate Algorithms (MVA)
- Combine features using *neural networks, boosted decision trees, likelihoods, etc.*
- **Exploit *correlations* between features**

## 3. Deep Learning
- Feed *low-level data* (e.g. calorimeter cells) directly to **deep neural networks**
- ***Potential to exploit information not contained in features***

single particle showers in a high-granularity 3D calorimeter




[1] BH, Farbin, Khattak, Pacela, Pierini, Vlimant, Spiropulu, Wei, Proceedings of the Deep Learning for Physical Sciences Workshop at Neural Information and Processing Systems (NIPS17)

# Example: “In-painting” with Deep Learning [1]

corrupted image

deep learning

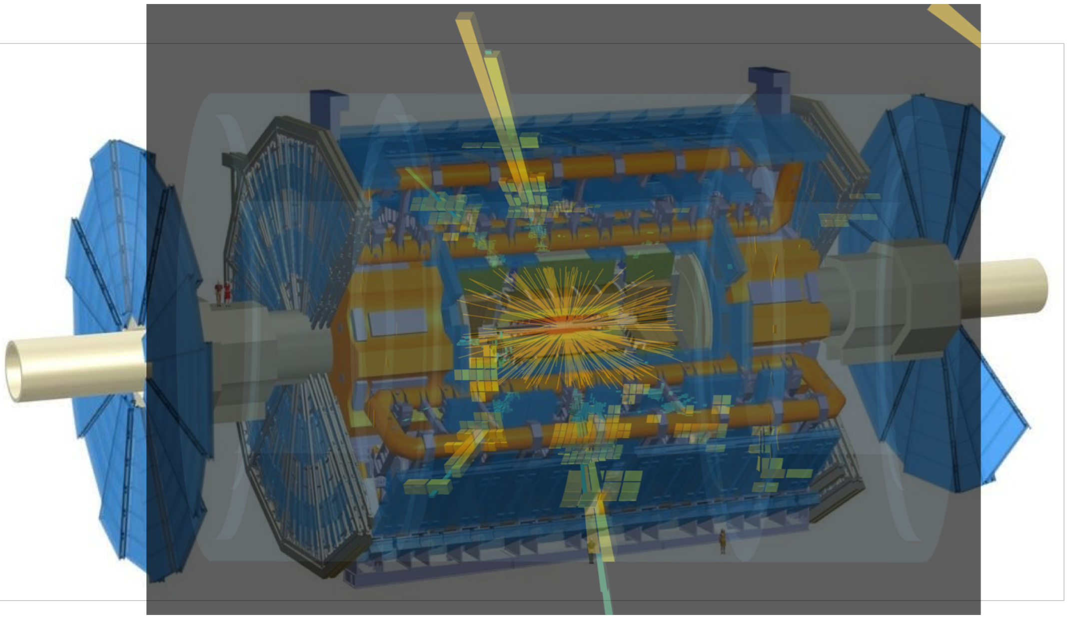
“in-painted” image
using deep neural networks [2]

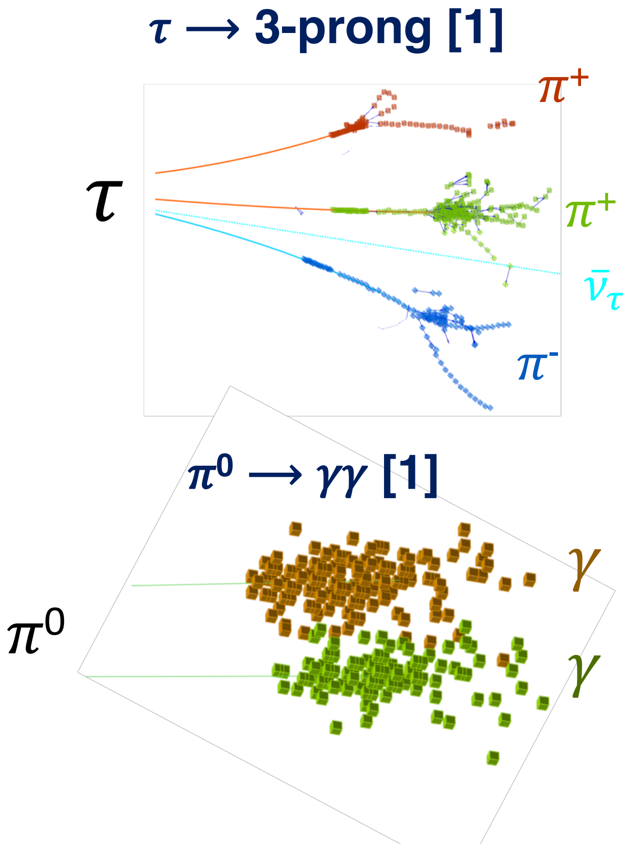
- Make progress by understanding the **structure** of the data
  - Not just more computational power and larger datasets

[1] from slides by Jessie Thaler

[2] Ulyanov, Vedaldi, Lempitsky, “Deep Image Prior,” 1711.10925

# Machine Learning at Colliders






- Particle detectors record **enormous volumes of complex 3D "images"**
  - ➢ Multiple sub-detectors, cell sizes, complex η-dependence, 3D structure, etc…
  - ➢ Use machine learning techniques to **exploit all available information**

# Neural Network Architectures

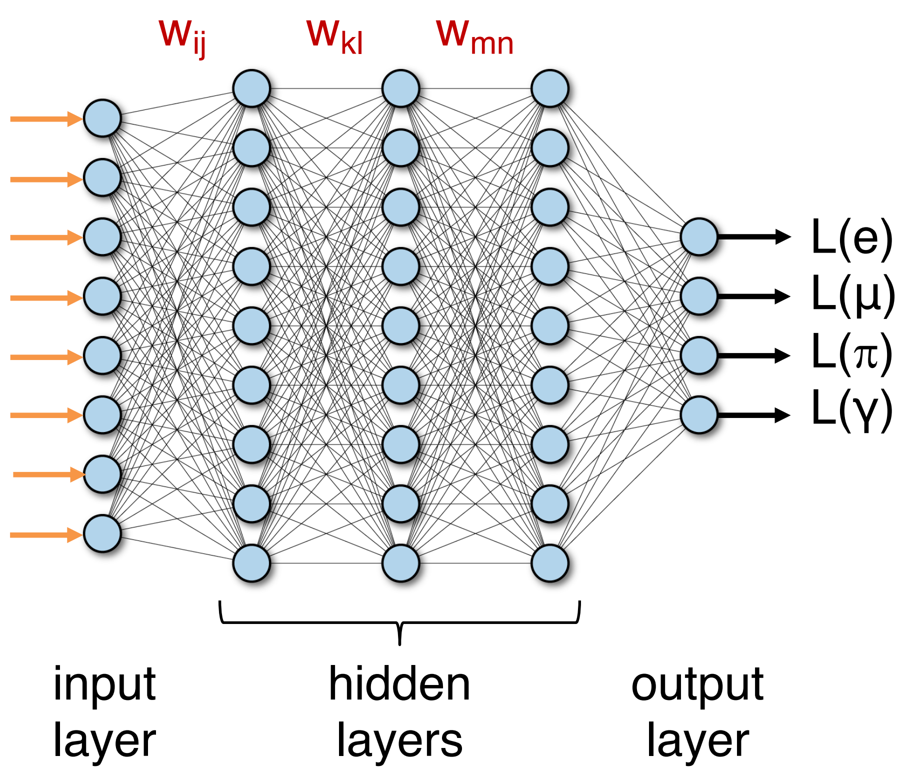
- **Fully-Connected Networks (FCN)**
  - Multiple layers of **fully inter-connected neurons** with variable weights
  - Structure-agnostic → widely applicable

**inputs** can be…

**features**

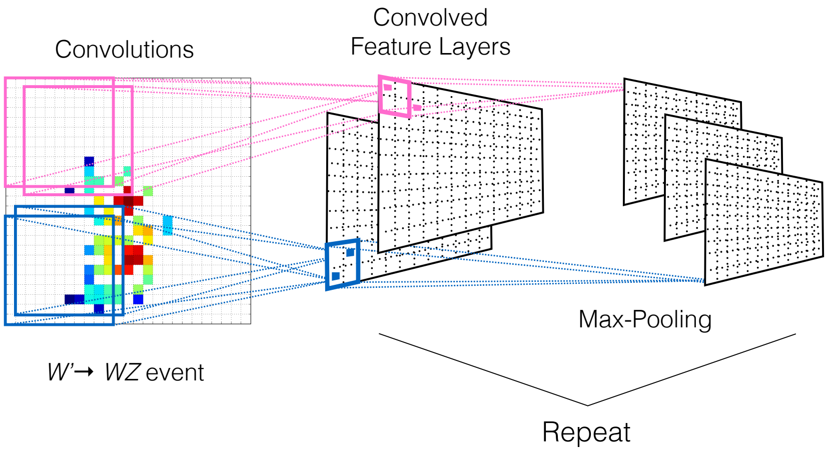
or

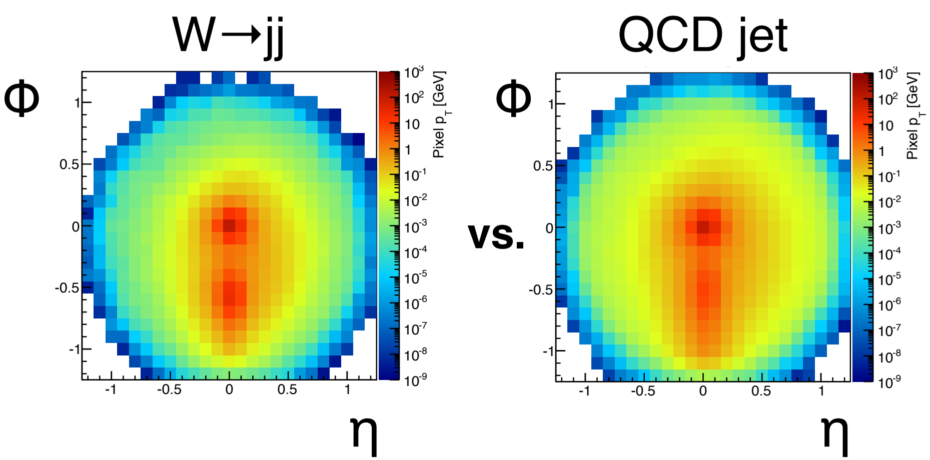
**low-level data**
(calo cells, track / cluster / particle flow p4's, etc.)



# Neural Network Architectures

- **Fully-Connected Networks (FCN)**
  - Multiple layers of **fully inter-connected neurons** with variable weights
  - Structure-agnostic → widely applicable
- **Convolutional Neural Networks (CNN)**
  - Specialized layers ("convolutional filters") identify structures at different scales
  - **Computer vision / imaging** applications
  - Assumes **fixed-length** input data

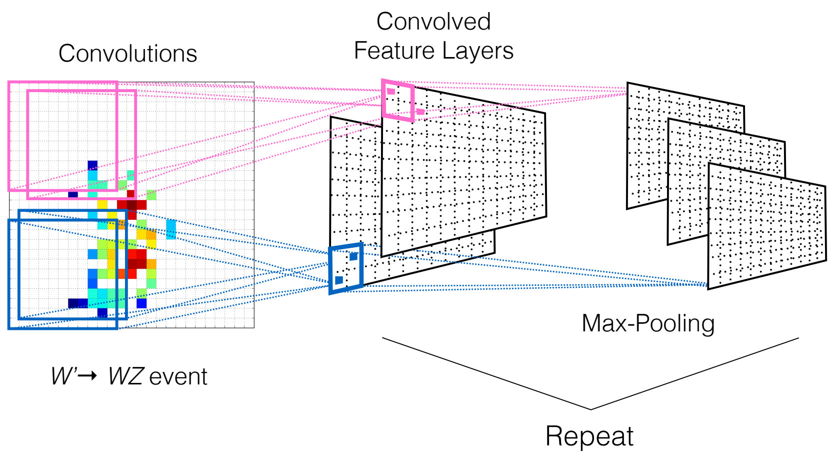


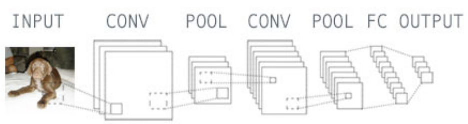




[1] de Oliveira, Kagan, Mackey, Nachmann, Schwartzman, "Jet Images – Deep Learning Edition", JHEP07 (2016) 069

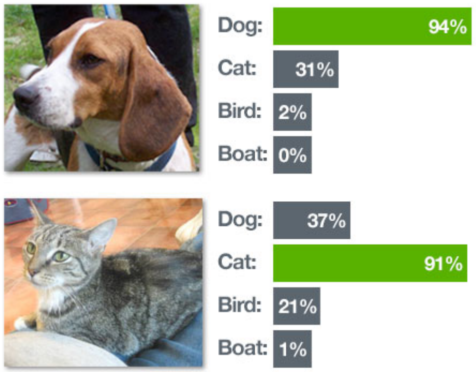
# Neural Network Architectures

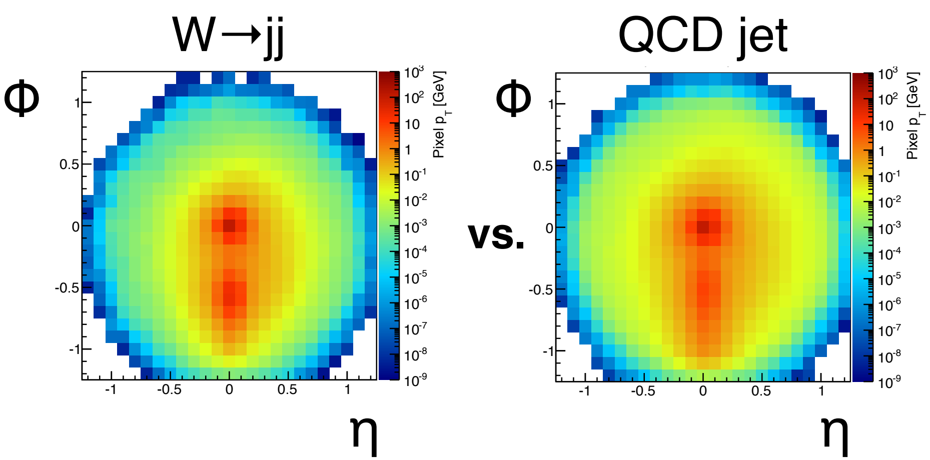
- **Fully-Connected Networks (FCN)**
  - Multiple layers of **fully inter-connected neurons** with variable weights
  - Structure-agnostic → widely applicable
- **Convolutional Neural Networks (CNN)**
  - Specialized layers (“convolutional filters”) identify structures at different scales
  - **Computer vision / imaging** applications
  - Assumes **fixed-length** input data




exploits extensive computer vision R&D



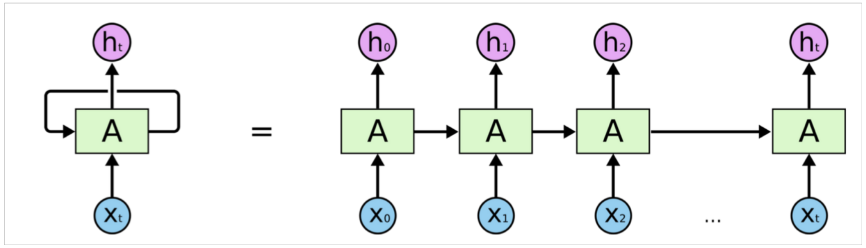

[1] de Oliveira, Kagan, Mackey, Nachmann, Schwartzman, “Jet Images – Deep Learning Edition”, JHEP07 (2016) 069

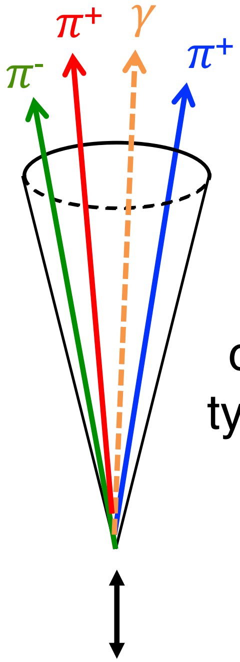
# Neural Network Architectures

- **Fully-Connected Networks (FCN)**
  - Multiple layers of **fully inter-connected neurons** with variable weights
  - Structure-agnostic → widely applicable

- **Convolutional Neural Networks (CNN)**
  - Specialized layers ("convolutional filters") identify structures at different scales
  - **Computer vision / imaging** applications
  - Assumes **fixed-length** input data

- **Recurrent Neural Networks (RNN)**
  - Cyclical structures allow for **variable-length** input data
    - ➢ e.g. Particle Flow Candidate p4's
  - **Language processing** applications




jet ⟷ sentence
constituents ⟷ words
type, $p_T$, η, Φ ⟷ letters

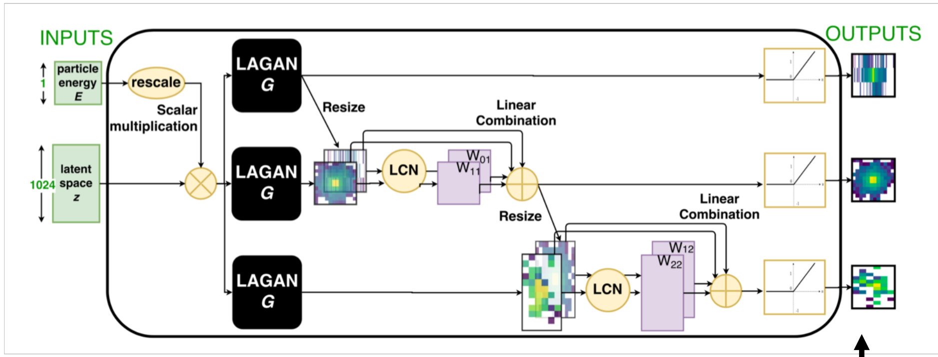
"pm_pt3.5_eta1.1_phi0.2 pp_pt5.6_eta0.3_phi1.8 g_pt10.5_eta1.4_phi0.3 pp_pt3.5_eta1.1_phi1.2."

exploits extensive language processing and translation R&D (e.g. google translate)

Louppe, Cho, Becot, Cranmer, QCD-Aware RNNs for Jet Physics, 1702.00748
Cheng, RNNs for Quark/Gluon Tagging, CSBS (2018) 2:3
ATLAS, b-tagging with RNNs, ATL-PHYS-PUB-2017-003

# Neural Network Architectures

- **Fully-Connected Networks (FCN)**
  - Multiple layers of **fully inter-connected neurons** with variable weights
  - Structure-agnostic → widely applicable
- **Convolutional Neural Networks (CNN)**
  - Specialized layers ("convolutional filters") identify structures at different scales
  - **Computer vision / imaging** applications
  - Assumes **fixed-length** input data
- **Recurrent Neural Networks (RNN)**
  - Cyclical structures allow for **variable-length** input data
    - e.g. Particle Flow Candidate p4's
  - **Language processing** applications
- **Generative Adversarial Networks (GAN)**
  - Generate ensembles of pseudo-data
  - **Fast simulation** applications




generated output images (for 3 ATLAS ECAL layers)

Paganini, de Oliveira, Nachman, CaloGAN for 3D particle showers, [PRD 97, 014021 (2018)]

# ML Use Cases at Colliders

- **Fully-Connected Networks (FCN)**
  - Multiple layers of **fully inter-connected neurons** with variable weights
  - Structure-agnostic → widely applicable
- **Convolutional Neural Networks (CNN)**
  - Specialized layers ("convolutional filters") identify structures at different scales
  - **Computer vision / imaging** applications
  - Assumes **fixed-length** input data
- **Recurrent Neural Networks (RNN)**
  - Cyclical structures allow for **variable-length** input data
    - e.g. Particle Flow Candidate p4's
  - **Language processing** applications
- **Generative Adversarial Networks (GAN)**
  - Generate ensembles of pseudo-data
  - **Fast simulation** applications

**classification**

- objects: jet classification, particle ID, etc.
- events: $t\bar{t}H(b\bar{b})$ vs. $t\bar{t}+b\bar{b}$, SUSY vs. $t\bar{t}$ , etc.
- "supervised" (labeled data) or "unsupervised"

measurements with **regression**

- objects: jet and lepton energies and angles
- events: total / hadronic / missing energy, $m_H$

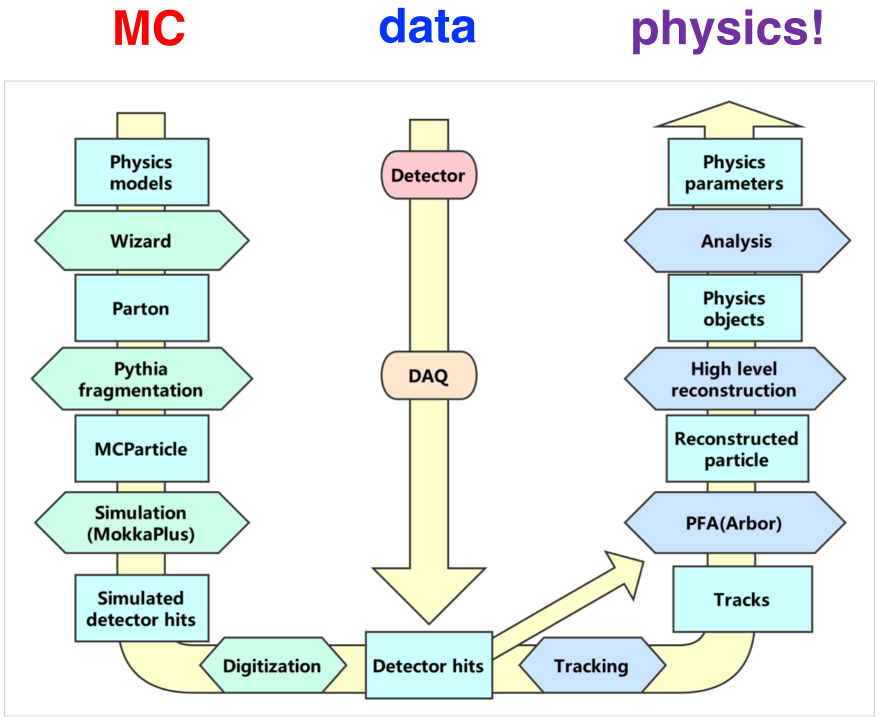
**fast simulation**

e.g. particle showers in calorimeters

# Collider Analysis Flow Chart

**1. Generation**
of truth information

**2. Simulation**
of detector response




**4. Analysis**
of physics objects

**3. Reconstruction**
of physics objects

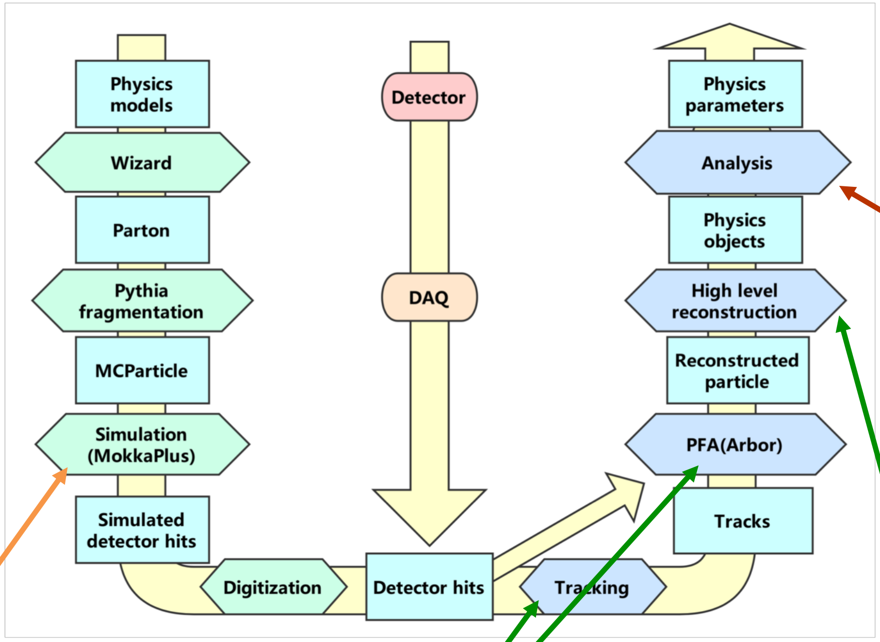
# Machine Learning Use Cases

MC data physics!

**1. Generation**
of truth information

**2. Simulation**
of detector response

**generative models**
(e.g. calorimeter showers)




**4. Analysis**
of physics objects

**event classification**
(e.g. ttH vs. tt+bb)

**event regression**
(e.g. $M_{Higgs}$)

**3. Reconstruction**
of physics objects

**object classification**
(e.g. particle ID, b-tagging)

**object regression**
(e.g. E, $\theta$, $\phi$)

**unsupervised classification**
(e.g. tracking, clustering,
track-cluster matching)

# Fast Simulation with GANs

- Future colliders will require enormous MC event samples:

CEPC operation plan:

| Operation mode | $\sqrt{s}$ (GeV) | $L$ per IP ($10^{34}$ cm$^{-2}$s$^{-1}$) | Years | Total $\int L$ (ab$^{-1}$, 2 IPs) | Event yields |
|---|---|---|---|---|---|
| $H$ | 240 | 3 | 7 | 5.6 | $1 \times 10^6$ |
| $Z$ | 91.2 | 32 (*) | 2 | 16 | $\boxed{7 \times 10^{11}}$ |
| $W^+W^-$ | 158–172 | 10 | 1 | 2.6 | $2 \times 10^7$ (†) |

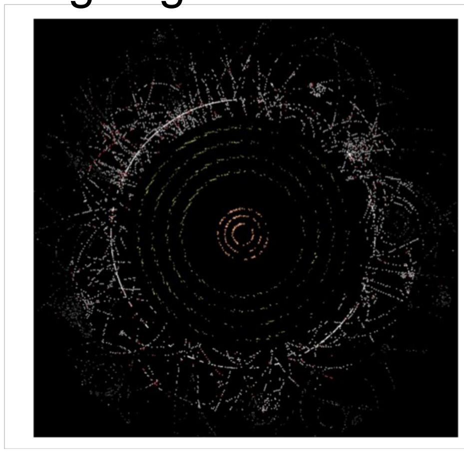
**~1 trillion Z bosons!**
(20× more than in 3000 fb$^{-1}$ LHC14 data)

- 50-70% of ATLAS computing resources (billions of CPU hours/year) spent on simulation [1] → dominated by particle showers in calorimeters
- Can use GANs to quickly generate large ensembles of calorimeter showers
- Alternative: “frozen shower” approach [2]

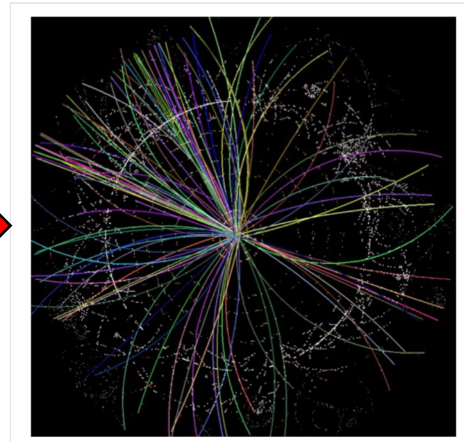
[1] CERN-RRB-2015-014
[2] ATLAS Fast Shower Simulation

# Tracking with ML

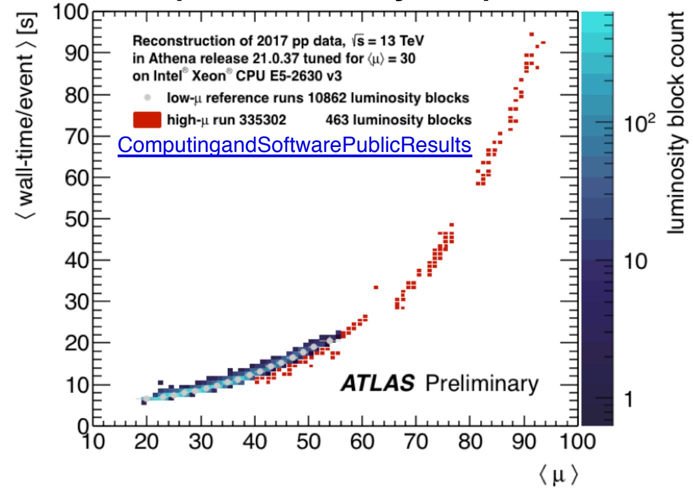
going from hits…



to tracks…



is computationally expensive:

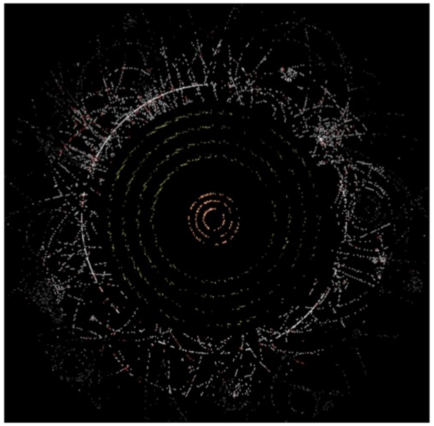


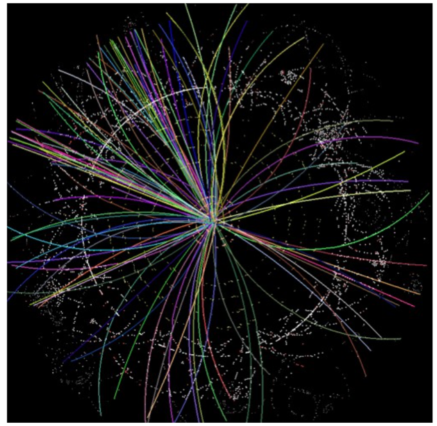

- Major challenge for HL-LHC and future hadron colliders!

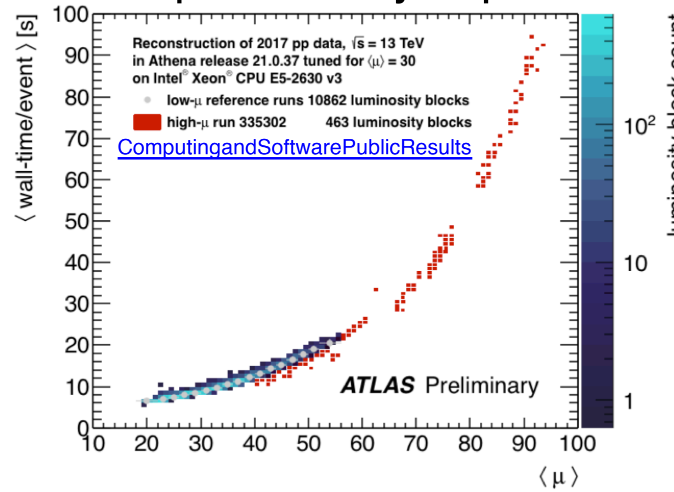
Images from J.R. Vlimant, CEPC Oxford [Workshop]

# Tracking with ML

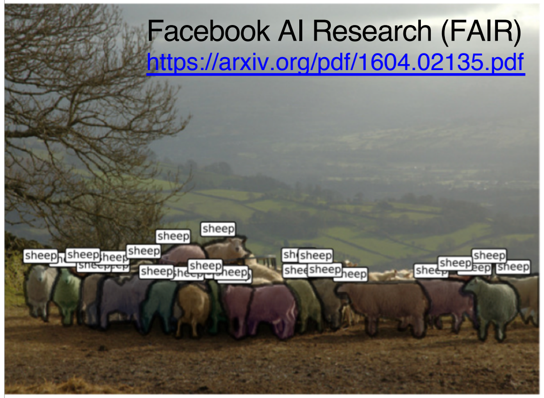
going from hits… to tracks… is computationally expensive:

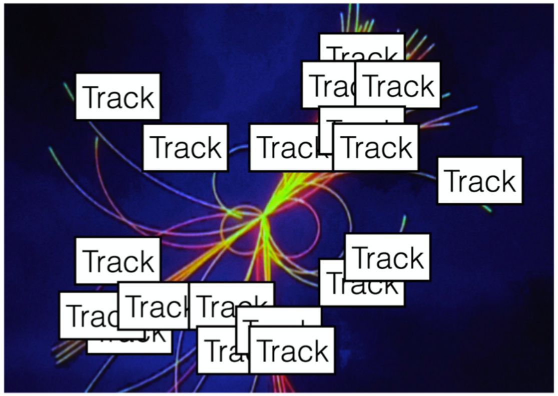







- Major challenge for HL-LHC and future hadron colliders!
- Can leverage **unsupervised learning** techniques to group hits into tracks
- Subject of TrackML challenge

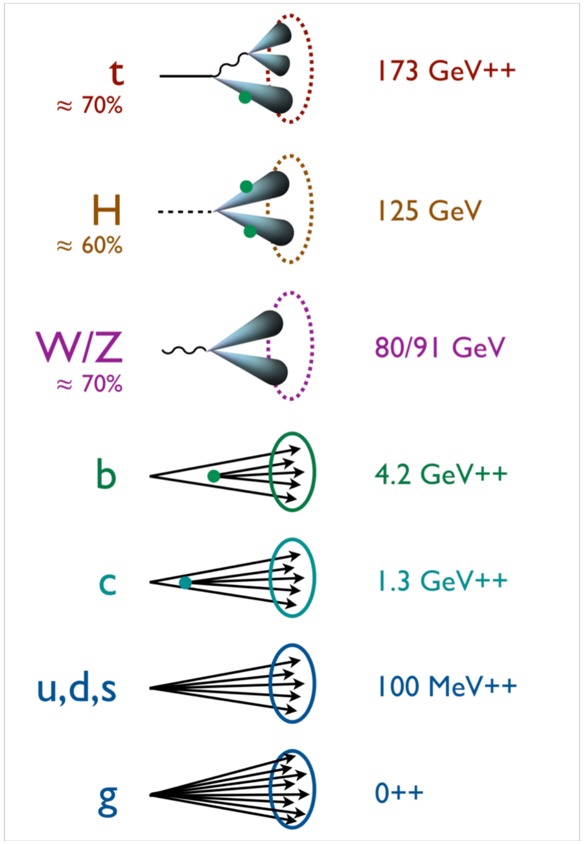


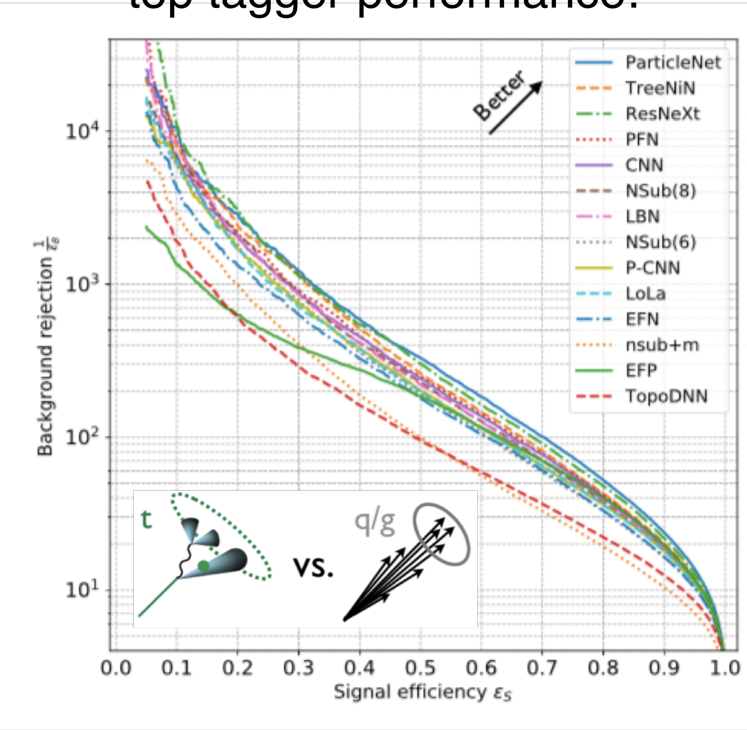

Images from J.R. Vlimant, CEPC Oxford Workshop

# Jet Classification at LHC [1]

++ = mass from QCD radiation




top tagger performance:

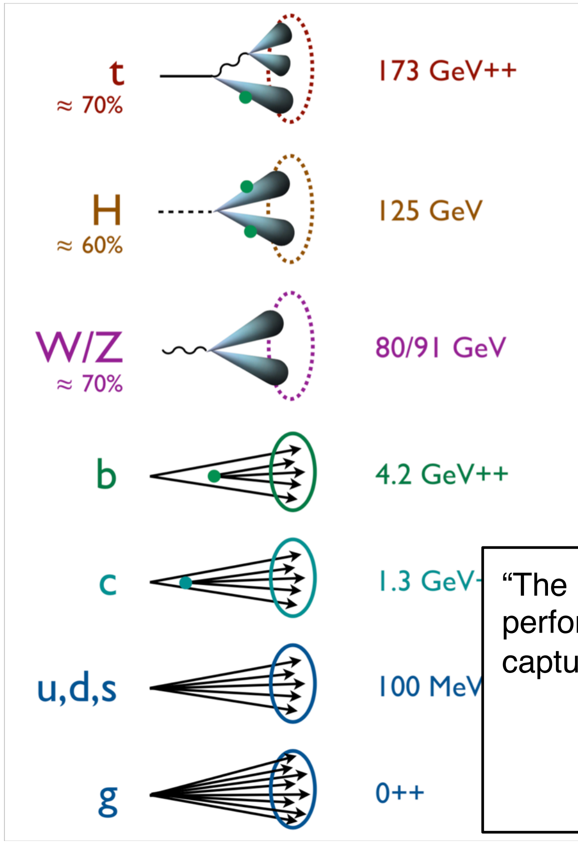



[1] from slides by Jessie Thaler
see also recent reviews:
Larkoski, Moult, Nachman, 1709.04464,
Marzani, Soyez, Spannowsky, 1901.10342

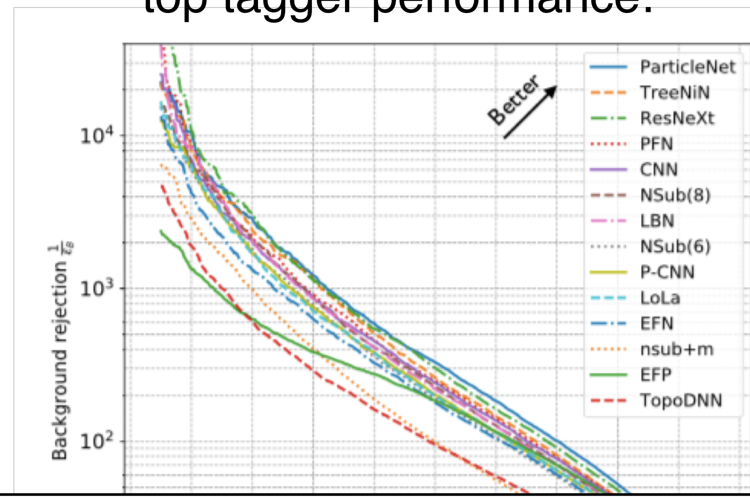
- Deep learning approach often provides best performance for jet classification tasks

# Jet Classification at LHC [1]

++ = mass from QCD radiation




top tagger performance:




deep learning

"The FCN with one subset consisting of fourteen observables shows nearly no degradation of performance. This indicates that these fourteen expert-designed observables could have captured the most necessary information for separating quark and gluon jets."

- Luo, Wang, Xu, Zhu, quark/gluon discrimination with FCNs, [Science China Phys., Mech. Astron. (2019) 62:991011]

[1] from [slides] by Jessie Thaler

see also recent reviews:

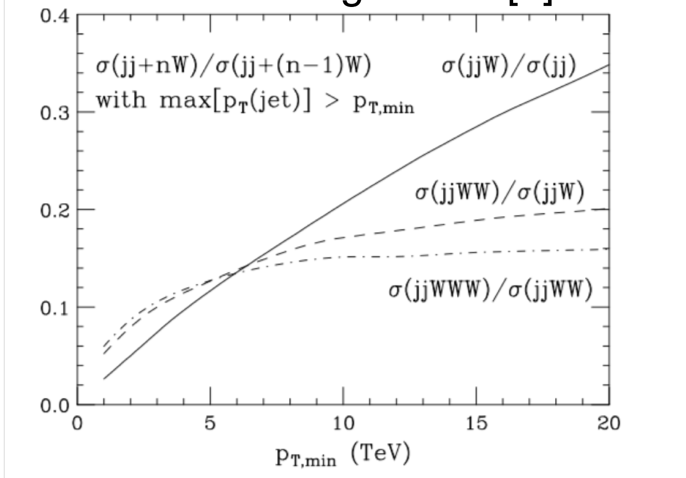
Larkoski, Moult, Nachman, [1709.04464],

Marzani, Soyez, Spannowsky, [1901.10342]

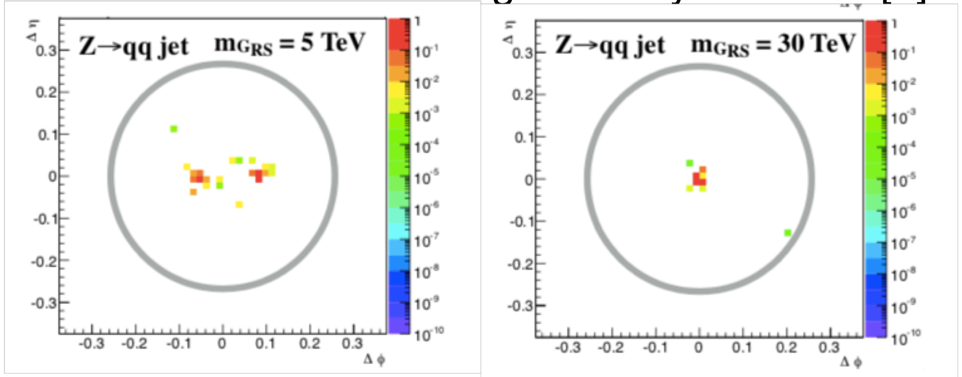
- Deep learning approach often provides best performance for jet classification tasks
  - But not always… possible to design clever features!

# Jets at Future Hadron Colliders

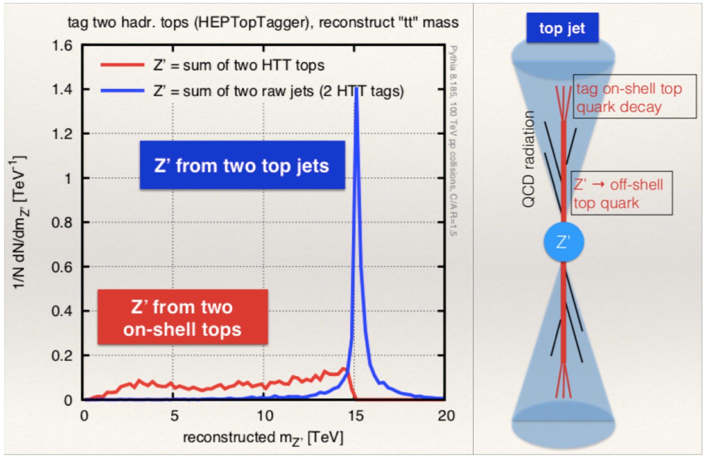
- High $p_T$ jets radiate W bosons at large rates [1]




- Large Lorentz boost → highly collimated W / Z bosons → calo granularity is crucial [2]




- top decays become as collimated as b decays at LHC → top quarks vs. top jets [3]

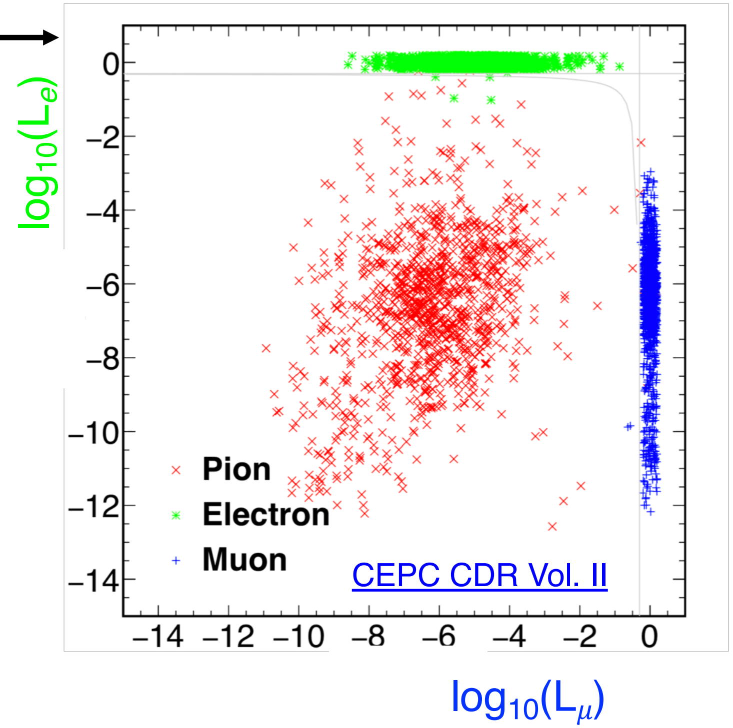



- "Boosted techniques will be essential at 100 TeV [hadron collider]" [3]
- Good use case for deep learning


[1] Arkani-Hamed, Mangano, Han, Wang, "Physics Opportunities at a 100 TeV pp collider", Physics Reports 652 (2016) 1-49
[2] Mangano et al., "Physics at a 100 TeV pp collider: SM processes", 1607.01831
[3] Salam, "Principles of Multi-TeV boosted objects", Higgs & BSM at 100 TeV workshop

# Particle ID at Future $e^+e^-$ Collider

- Electrons and muons are crucial for precision $Z \longrightarrow \ell\ell$ electroweak measurements and leptonic Higgs decays
- LICH algo [1] calculates ***e*** and $\mu$ likelihoods $L_e$, $L_\mu$ using 24 features combined into BDT
  - dE/dx, # ECAL / HCAL hits, spatial shower shape info, energy distribution, fractal dimension [2]
  - Similar approach for b-tagging with LCFIPlus [3]

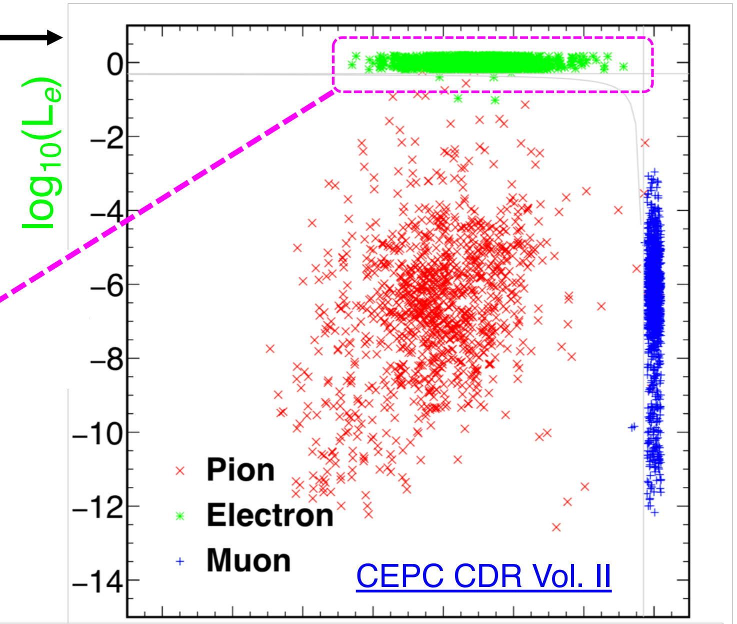




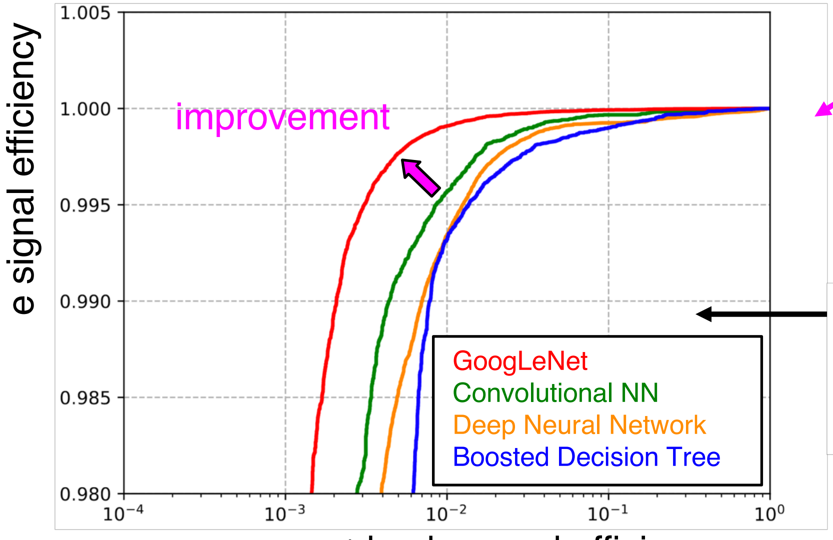
[1] Yu, Ruan, Boudry, Videau, EPJC 77 (2017) no. 9
[2] Ruan, Jeans, Boudry, Brient, Videau, PRL 112(1), 012001, 2014
[3] Suehara, Tanabe, NIM A 808 (2016) 109, and see talk from Wei-Ming Yao

# Particle ID at Future $e^+e^-$ Collider

- Electrons and muons are crucial for precision $Z \longrightarrow \ell\ell$ electroweak measurements and leptonic Higgs decays
- LICH algo [1] calculates ***e*** and $\mu$ likelihoods $L_e$, $L_\mu$ using 24 features combined into BDT
  - dE/dx, # ECAL / HCAL hits, spatial shower shape info, energy distribution, fractal dimension [2]
  - Similar approach for b-tagging with LCFIPlus [3]




**ROC curve for e vs. π± classifier**
with high granularity 3D calorimeter:




Calorimetry with Deep Learning: Particle Identification and Simulation for Collider Physics

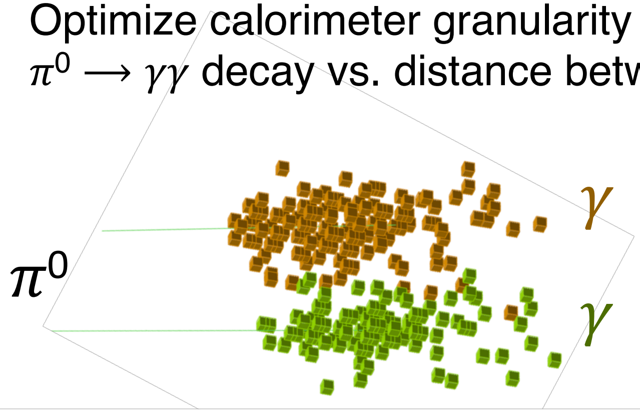
Dawit Belayneh[1], Federico Carminati[2], Amir Farbin[3], Benjamin Hooberman[4], Gulrukh Khattak[25], Miaoyuan Liu[6], Junze Liu[4], Dominick Olivito[7], Vitória Barin Pacela[8], Maurizio Pierini[2], Alexander Schwing[4], Maria Spiropulu[9], Sofia Vallecorsa[2], Jean-Roch Vlimant[9], Wei Wei[4], and Matt Zhang[a4]

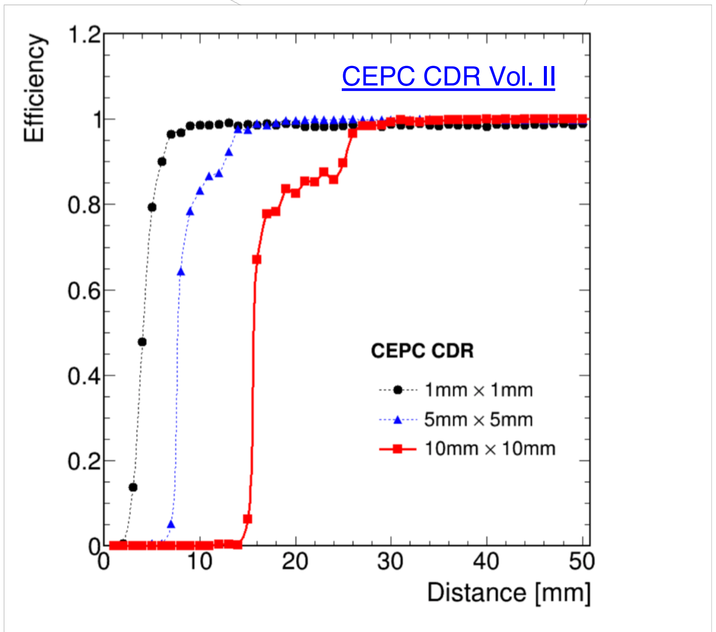
to appear soon

see also: de Oliveira, Nachman, Paganini, EM showers beyond shower shapes, 1806.05667

# $\pi^0(\gamma\gamma)$ vs. $\gamma$ discrimination at Future $e^+e^-$ Collider

- $\pi^0(\gamma\gamma)$ reconstruction crucial for τ and heavy flavor physics
  - Optimize calorimeter granularity by determining efficiency to reconstruct both photons from $\pi^0 \longrightarrow \gamma\gamma$ decay vs. distance between $\gamma$ calorimeter impact points, for different cell sizes

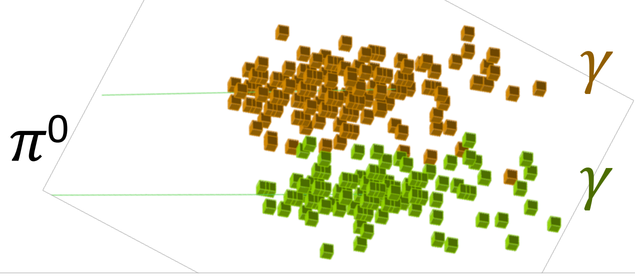


# $\pi^0(\gamma\gamma)$ vs. $\gamma$ discrimination at Future e+e- Collider

- $\pi^0(\gamma\gamma)$ reconstruction crucial for τ and heavy flavor physics
  - Optimize calorimeter granularity by determining efficiency to reconstruct both photons from $\pi^0 \longrightarrow \gamma\gamma$ decay vs. distance between $\gamma$ calorimeter impact points, for different cell sizes

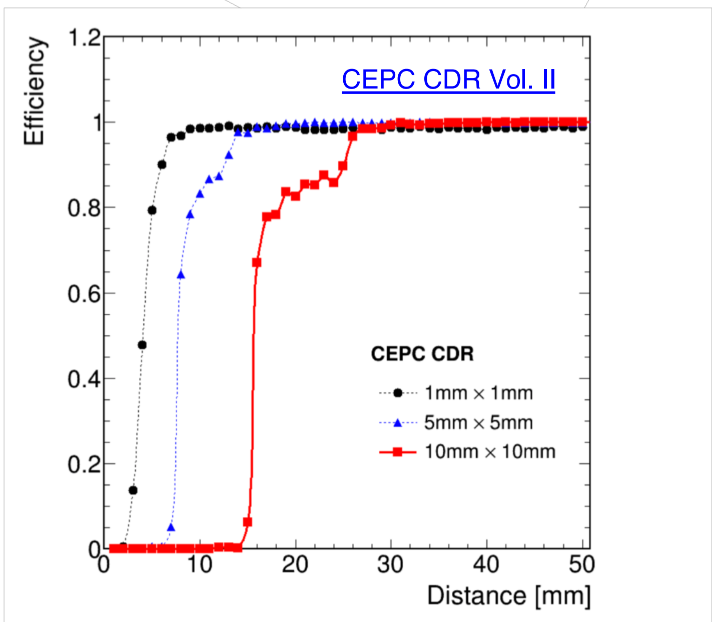
**good use case for CNN imaging!**



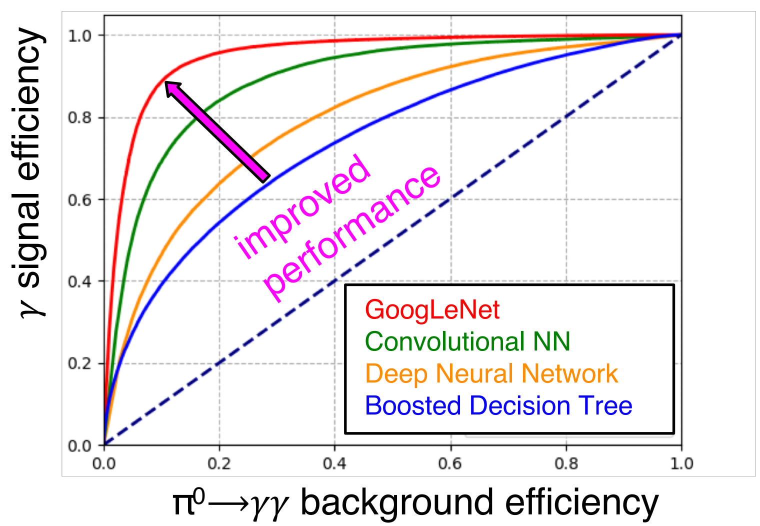

Calorimetry with Deep Learning: Particle Identification and Simulation for Collider Physics

Dawit Belayneh[1], Federico Carminati[2], Amir Farbin[3], Benjamin Hooberman[4], Gulrukh Khattak[25], Miaoyuan Liu[6], Junze Liu[4], Dominick Olivito[7], Vitória Barin Pacela[8], Maurizio Pierini[2], Alexander Schwing[4], Maria Spiropulu[9], Sofia Vallecorsa[2], Jean-Roch Vlimant[9], Wei Wei[4], and Matt Zhang[a4]

to appear soon




**ROC curve for $\gamma$ vs. $\pi^0 \longrightarrow \gamma\gamma$ classifier**
with high granularity CLIC LDC calorimeter:



# Performance Requirements

Future $e^+e^-$ collider performance requirements [1]

| Physics process | Measurands | Detector subsystem | Performance requirement |
|---|---|---|---|
| $ZH, Z \to e^+e^-, \mu^+\mu^-$ <br> $H \to \mu^+\mu^-$ | $m_H, \sigma(ZH)$ <br> $\mathrm{BR}(H \to \mu^+\mu^-)$ | Tracker | $\Delta(1/p_T) =$ <br> $2 \times 10^{-5} \oplus \frac{0.001}{p(\mathrm{GeV}) \sin^{3/2}\theta}$ |
| $H \to b\bar{b}/c\bar{c}/gg$ | $\mathrm{BR}(H \to b\bar{b}/c\bar{c}/gg)$ | Vertex | $\sigma_{r\phi} =$ <br> $5 \oplus \frac{10}{p(\mathrm{GeV}) \times \sin^{3/2}\theta} (\mu\mathrm{m})$ |
| $H \to q\bar{q}, WW^*, ZZ^*$ | $\mathrm{BR}(H \to q\bar{q}, WW^*, ZZ^*)$ | ECAL <br> HCAL | $\sigma_E^{\mathrm{jet}}/E =$ <br> $3 \sim 4\%$ at 100 GeV |
| $H \to \gamma\gamma$ | $\mathrm{BR}(H \to \gamma\gamma)$ | ECAL | $\Delta E/E =$ <br> $\frac{0.20}{\sqrt{E(\mathrm{GeV})}} \oplus 0.01$ |

[1] CEPC CDR Vol. II

# Possible Performance Enhancements with Deep Learning

Future $e^+e^-$ collider performance requirements [1]

| Physics process | Measurands | Detector subsystem | Performance requirement |
|---|---|---|---|
| $ZH, Z \to e^+e^-, \mu^+\mu^-$<br>$H \to \mu^+\mu^-$ | $m_H, \sigma(ZH)$<br>$\mathrm{BR}(H \to \mu^+\mu^-)$ | Tracker | $\Delta(1/p_T) =$<br>$2 \times 10^{-5} \oplus \frac{0.001}{p(\mathrm{GeV})\sin^{3/2}\theta}$ |
| $H \to b\bar{b}/c\bar{c}/gg$ | $\mathrm{BR}(H \to b\bar{b}/c\bar{c}/gg)$ | Vertex | $\sigma_{r\phi} =$<br>$5 \oplus \frac{10}{p(\mathrm{GeV})\times\sin^{3/2}\theta}(\mu\mathrm{m})$ |
| $H \to q\bar{q}, WW^*, ZZ^*$ | $\mathrm{BR}(H \to q\bar{q}, WW^*, ZZ^*)$ | ECAL<br>HCAL | $\sigma_E^{\mathrm{jet}}/E =$<br>$3 \sim 4\%$ at 100 GeV |
| $H \to \gamma\gamma$ | $\mathrm{BR}(H \to \gamma\gamma)$ | ECAL | $\Delta E/E =$<br>$\frac{0.20}{\sqrt{E(\mathrm{GeV})}} \oplus 0.01$ |

- b/c-tagging RNN [2,3]
- quark / gluon discrimination RNN / CNN [4-6]

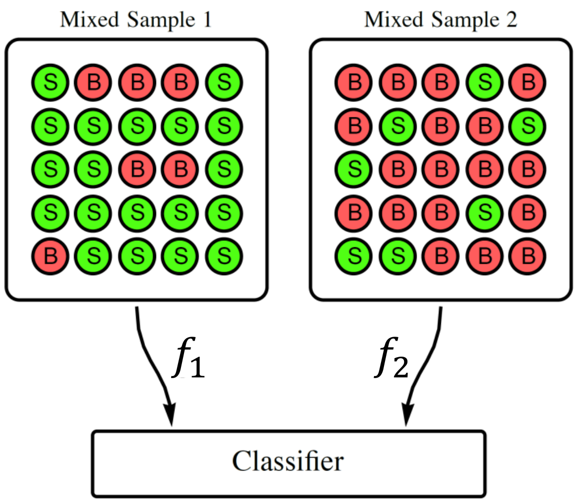
- jet energy regression with NNs [4,7]

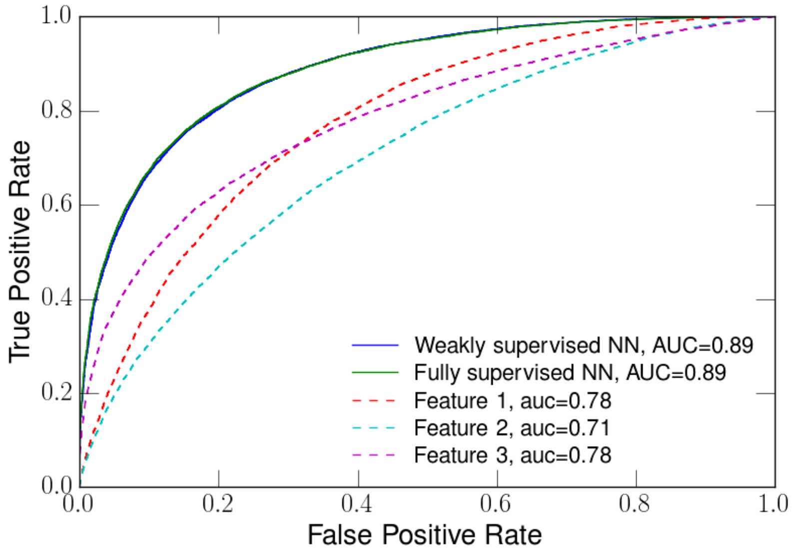
- $\gamma$ vs. $\pi^0(\gamma\gamma)$ discrimination with CNNs [8]
- $E_\gamma$, $\Delta\theta_{\gamma\gamma}$ regression for improved $M_{\gamma\gamma}$ resolution [8]


[1] CEPC CDR Vol. II
[2] ATLAS, b-tagging with RNNs, ATL-PHYS-PUB-2017-003
[3] CMS, Heavy flavor identification with DNNs, CMS-DP-2017-005
[4] Larkoski, Moult, Nachman, 1709.04464 and references therein
[5] ATLAS, q/g tagging with jet images, ATL-PHYS-PUB-2017-017
[6] Cheng, RNNs for Quark/Gluon Tagging, CSBS (2018) 2:3
[7] ATLAS, NN approach to jet calibration, ATL-PHYS-PUB-2018-013
[8] BH et al., Calorimetry with Deep Learning, NeurIPS2017

# Training with Data: Weak Supervised Learning

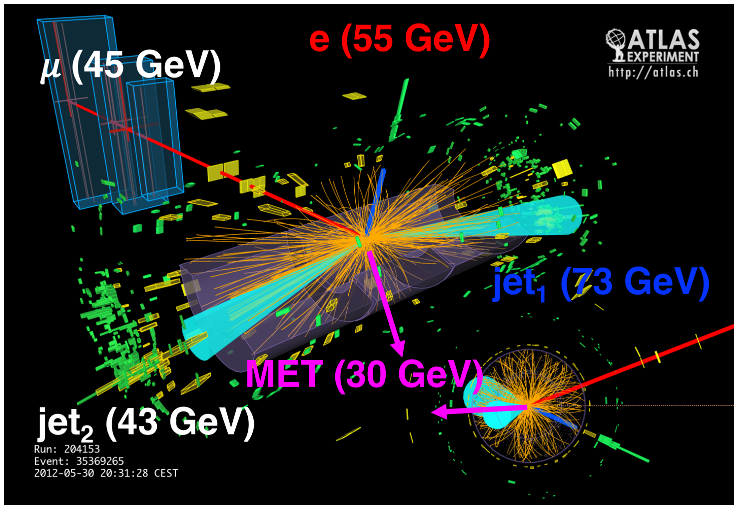
- Challenge: we can't rely on MC to model low-level inputs perfectly
- **Train on data** using **weakly-supervised learning** with signal-enriched and background-enriched data samples
- Rely on MC only for signal and background fractions of the two samples (and vary this fraction to estimate systematic uncertainty)

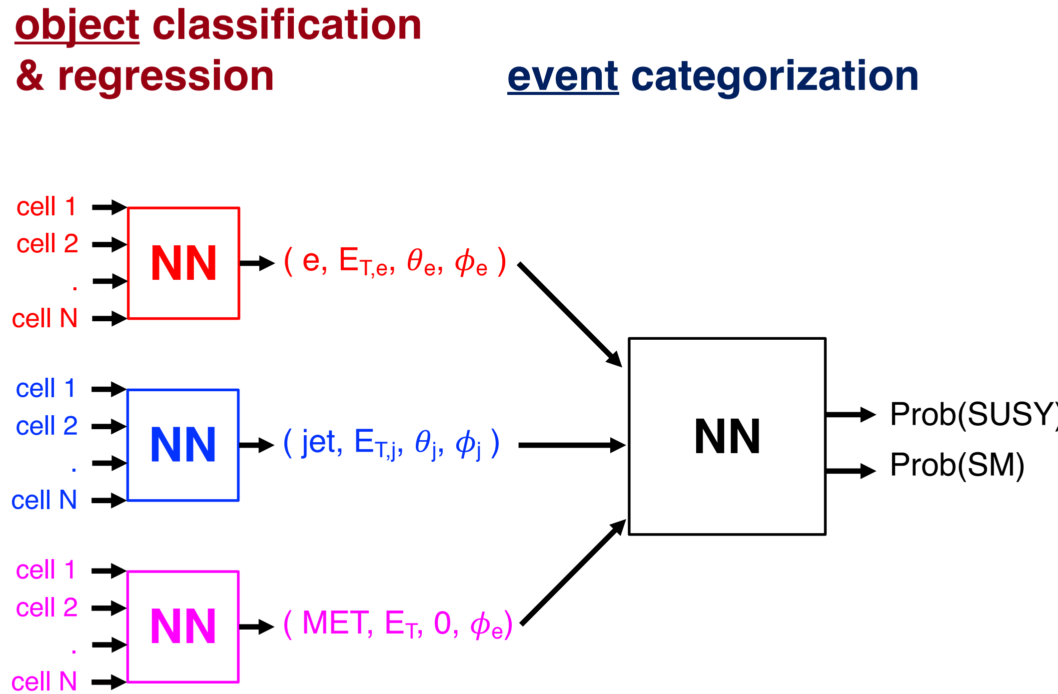



Dery, Nachman, Rubbo, Schwartzman, "Weakly Supervised Classification in High Energy Physics", JHEP 05 (2017) 145

# (One) Strategy for ML Event Classification



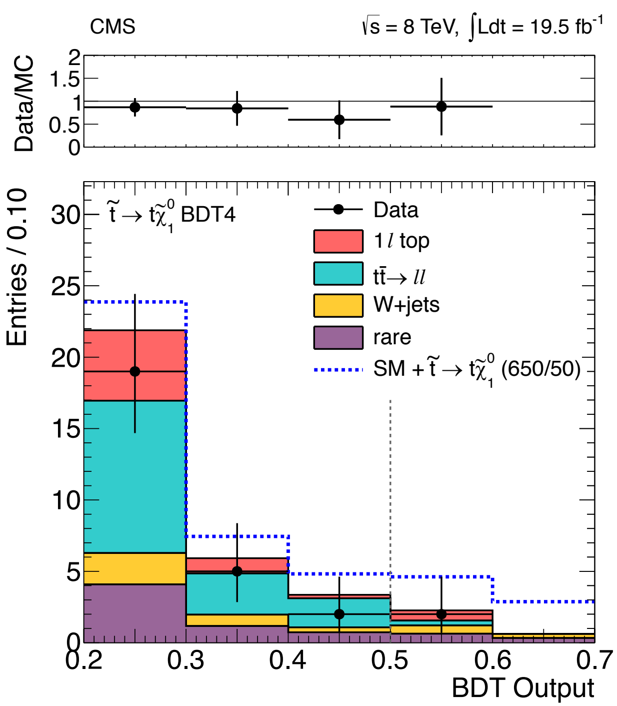




- Factorize the problem: **object tagging** + **event classification**
  - Use **cells** to *classify type* and *measure p4's* of physics **objects** (e, $\mu$, $\tau$, $\gamma$, j, MET)
  - Use **object types and p4's** to *categorize* **events** (e.g. SM vs. SUSY) with e.g. RNNs

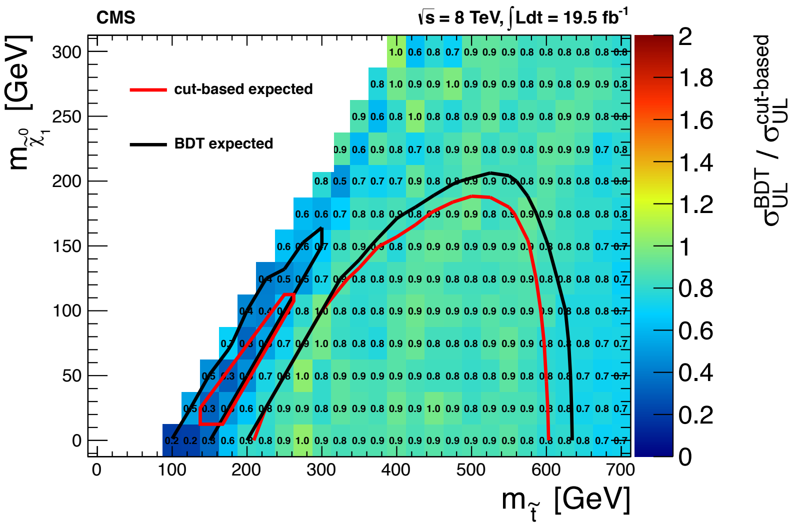
# Classifying Events with BDTs

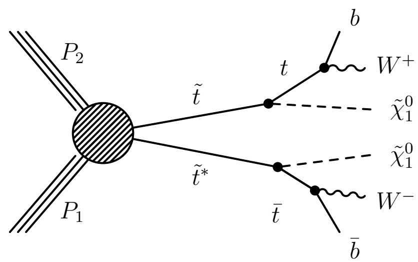
- Feature-based BDT event classifiers in wide use by ATLAS & CMS
- Be wary of “tails” → also perform cut-based analysis, and compare results

CMS Run 1 stop 1L search, [EPJC 73(12):2677, **2013**]




BDT vs. cut-based approach:



# Alternative Methods for Event Classification

1. Deep learning approach: feed low-level data (e.g. particle flow object p4's) directly to DNN

2. Transform low-level data to "rapidity-mass matrices" and train shallow NNs [1,2]

3. Matrix Element Likelihood Analysis (MELA)
   - Interesting option for future $e^+e^-$ colliders (with excellent detector resolution)

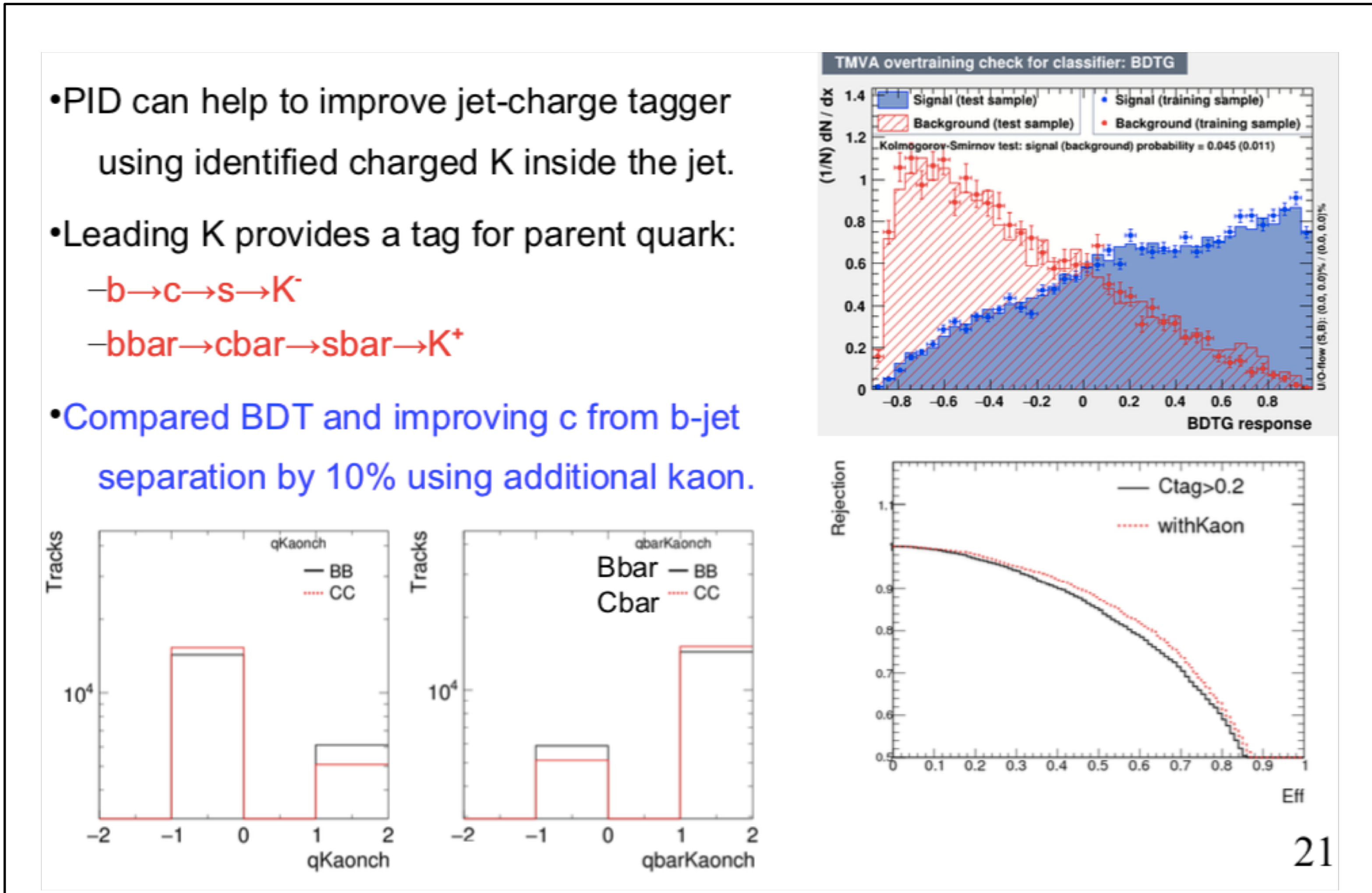
[1] Chekanov, "Imaging particle collision data for event classification using machine learning", NIM A 931 (2019) 92
[2] Chekanov, "Machine learning using rapidity-mass matrices for event classification problems in HEP", 1810.06669

# Summary

- Wide variety of machine learning techniques available for collider **classification**, **regression**, and **fast simulation** tasks
- Feature-based classifiers widely used in ATLAS and CMS and under study for future colliders
- **Deep learning** approach with low-level inputs has been shown to provide better performance for some problems

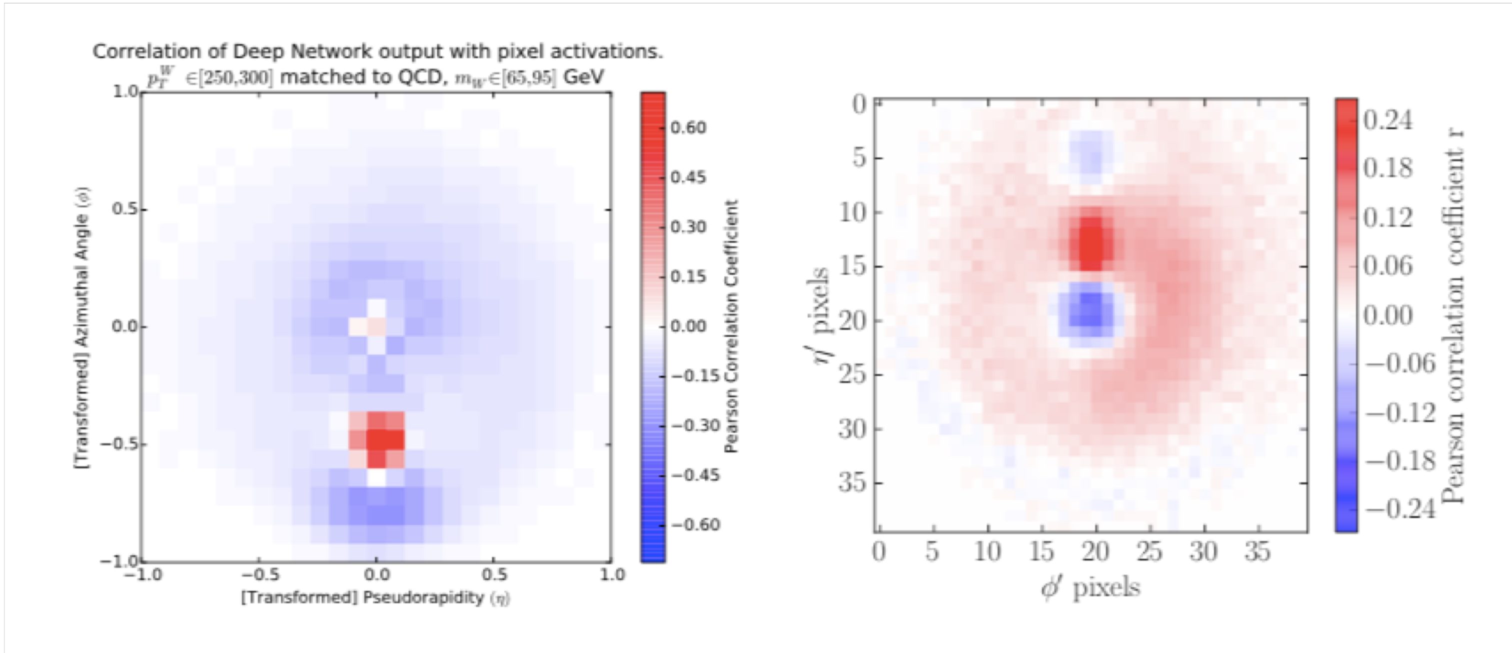
# Additional Material

# B-tagging with ML




[Slide](#) from Wei-Ming Yao

# Looking Inside the Black Box




- **Low-level correlations**: Correlations between the network inputs and outputs can show which areas of the input space are most useful for discrimination. For a jet image $J$, this results in another image $C$ where the pixel intensity is the correlation between the network output $N$ and the pixel intensity, $C_{ij} = \rho(J_{ij}, N(J))$. This only identifies linear information about the network output but can illustrate how this is distributed non-linearly in space. Examples are shown in Fig. 33 for $W$ and top tagging. Extensions to non-linear generalizations of the correlation coefficient are also possible.

Larkoski, Moult, Nachman, 1709.04464,

# Looking Inside the Black Box

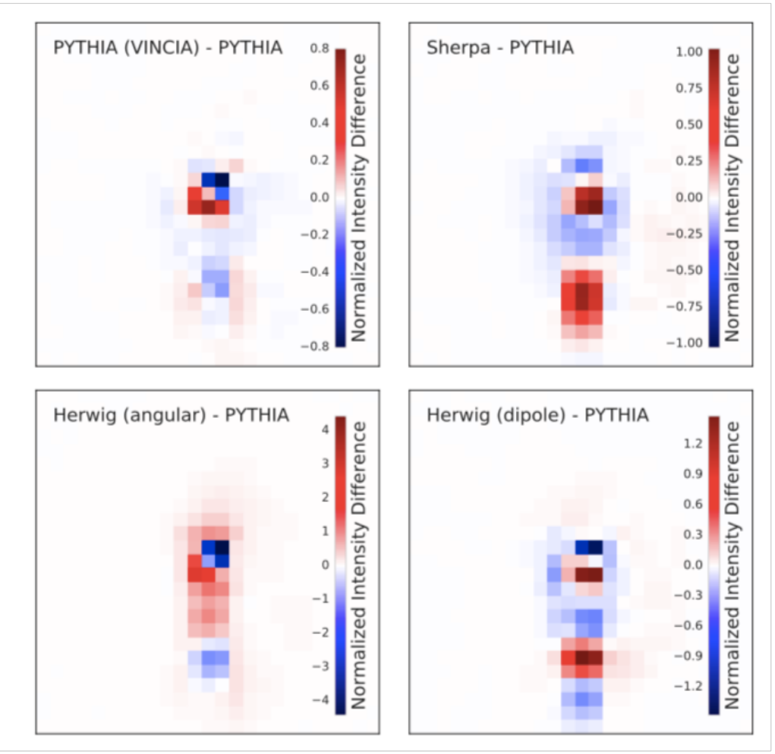
- **High-level correlations**: The joint distribution of standard physically-inspired features (e.g. jet mass) and the network output (or intermediate node activations) illustrate if and how the network is learning about known physical effects.
- **High-level input**: Building a new classifier that combines the network output and a standard physically-inspired feature can demonstrate to what extent the information about that feature is learned by the network.
- **Redacted phase space**: Studying the distribution of inputs and the network performance after conditioning on standard physically-inspired features can help to visualize what new information the network is using from the jet. Training the network on inputs that have been conditioned on specific values of known features can also be useful for this purpose.

- **Re-weighted phase space**: A complementary approach to redacting is to re-weight phase space so that the marginal likelihood ratio for standard physically-inspired features is unity, $p_s(m)/p_b(m) = 1$, where $m$ is a feature of the full image $J$ and $p_{s,b}(m) = \int p_{s,b}(J)\delta(m(J) = m)$ is the marginal probability distribution. With this weighting, the known feature $m$ is not useful for classification. Reference [572] named this 'planning'.
- **Weights**: The activations for the various layers can sometimes be useful in identifying what the network is learning. This is particularly true for convolutional layers where the filters encode activated features. An interesting further step is to convolve the filters with the average image from the two classes and then visualize their difference.
- **Most activating images**: A complementary approach to visualizing the network weights is to find which sets of inputs most activate a particular node or the entire network. In the case of jet images, one can plot the average of the $n$ most activating images.

Larkoski, Moult, Nachman, 1709.04464,

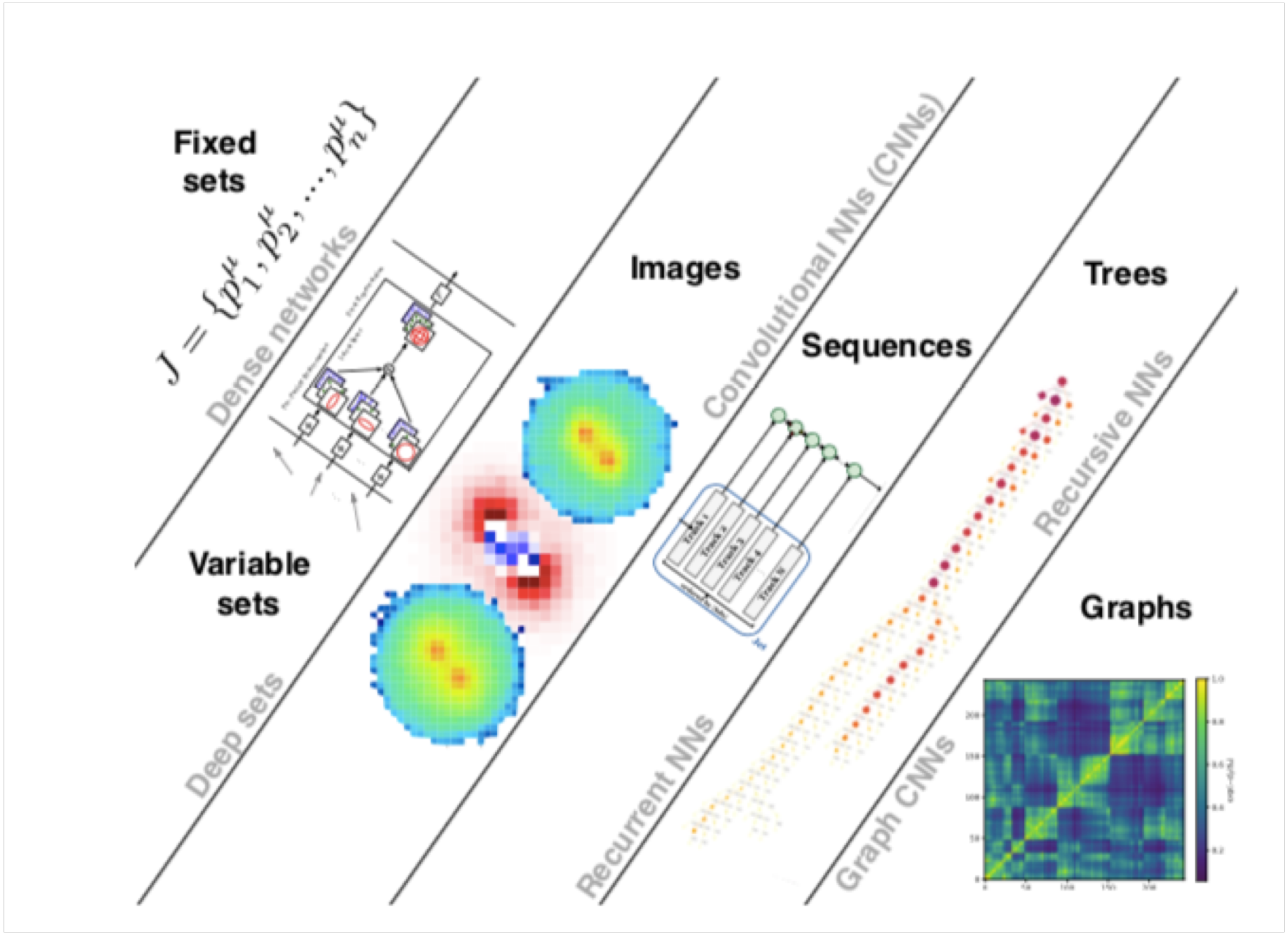
# Systematic Uncertainties

[Barnard, Dawe, Dolan, Rajcic, 1609.00607]




- jet images (or sequences) depend on the generator → 3 approaches
  - Build in dependence on nuisance parameters to network
    [Baldi, Cranmer, Faucett, Sadowski, Whiteson, 1601.07913]
  - Minimize network dependence on nuisance parameters
    [Louppe, Kagan, Cranmer, 1611.01046]
  - Train with data using weak supervised learning

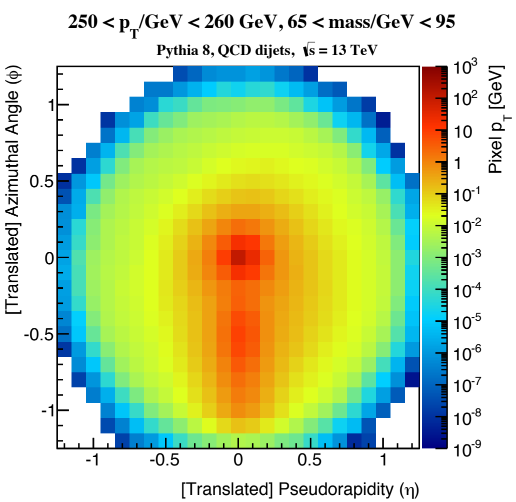
# Data Representations

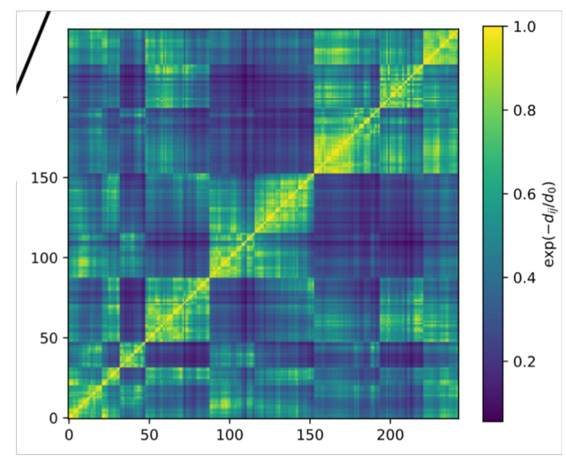



Larkoski, Moult, Nachman, [1709.04464](https://arxiv.org/abs/1709.04464),

# Data Representations: Fixed- vs. Variable-Length

- **Images**
  - **Fixed length** representation
  - Inputs are individual pixel/cell energies
  - Use **computer vision** techniques (**CNN/DNN**)
- **Sequences / trees**
  - **Variable length** representation
  - Inputs are tracks, clusters, or PF candidates
  - Use **language processing** (RNN/LSTM/GRU)
- **Graphs**
  - **Variable length** representation
  - Inputs are tracks, clusters, or PF candidates
  - Use **Message-Passing Neural Networks** (MPNN)



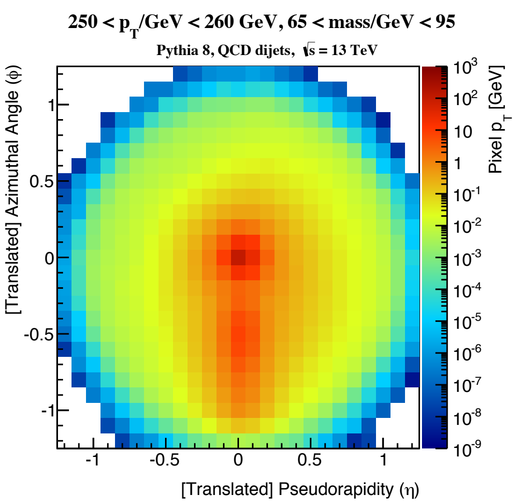

Jet-Images – Deep Learning Edition, 1511.05190

QCD-Aware RNNs for Jet Physics, 1702.00748

Neural Message Passing for Jet Physics, NIPS DLPS 107

# Data Representations: Fixed- vs. Variable-Length

- **Images**
  - **Fixed length** representation
  - Inputs are individual pixel/cell energies
  - Use **computer vision** techniques (**CNN/DNN**)
- **Sequences / trees**
  - **Variable length** representation
  - Inputs are tracks, clusters, or PF candidates
  - Use **language processing** (RNN/LSTM/GRU)
- **Graphs**
  - **Variable length** representation
  - Inputs are tracks, clusters, or PF candidates
  - Use **Message-Passing Neural Networks** (MPNN)

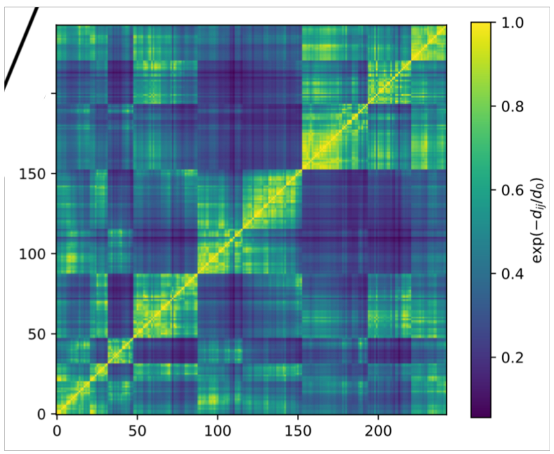



Easy to visualize
Benefit from image processing literature
Exploits full info (including distance btw pixels)
Images are sparse and w/o clear edges
Preprocessing is non-trivial
Non-trivial convolutional filters for non-uniform cell sizes

Jet-Images – Deep Learning Edition, 1511.05190

Captures all constituents with full granularity
Can handle sparse images and non-uniform cell sizes
Exploits calo. reconstruction (cells→clusters)
No unique order / structure
Each node is connected only to neighbor

QCD-Aware RNNs for Jet Physics, 1702.00748




Captures all constituents with full granularity
Can learn jet clustering algo (adjacency matrix)
Can be non-local
Non-trivial to visualize

Neural Message Passing for Jet Physics, NIPS DLPS 107

# Data Representations: Variable-Length Representations

*order-independent* variable-length representations!

**Energy Flow Networks: Deep Sets for Particle Jets**

Patrick T. Komiske, Eric M. Metodiev, Jesse Thaler

arXiv:1810.05165 [hep-ph]

(Submitted on 11 Oct 2018 (v1), last revised 11 Jan 2019 (this version, v2))

A key question for machine learning approaches in particle physics is how to best represent and learn from collider events. As an event is intrinsically a variable-length unordered set of particles, we build upon recent machine learning efforts to learn directly from sets of features or "point clouds". Adapting and specializing the "Deep Sets" framework to particle physics, we introduce Energy Flow Networks, which respect infrared and collinear safety by construction. We also develop Particle Flow Networks, which allow for general energy dependence and the inclusion of additional particle-level information such as charge and flavor. These networks feature a per-particle internal (latent) representation, and summing over all particles yields an overall event-level latent representation. We show how this latent space decomposition unifies existing event representations based on detector images and radiation moments. To demonstrate the power and simplicity of this set-based approach, we apply these networks to the collider task of discriminating quark jets from gluon jets, finding similar or improved performance compared to existing methods. We also show how the learned event representation can be directly visualized, providing insight into the inner workings of the model. These architectures lend themselves to efficiently processing and analyzing events for a wide variety of tasks at the Large Hadron Collider. Implementations and examples of our architectures are available online in our EnergyFlow package.

**Dynamic Graph CNN for Learning on Point Clouds**

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, Justin M. Solomon

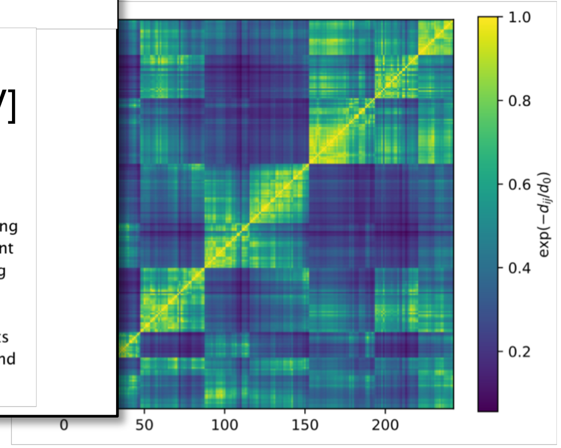
arXiv:1801.07829 [cs.CV]

(Submitted on 24 Jan 2018)

Point clouds provide a flexible and scalable geometric representation suitable for countless applications in computer graphics; they also comprise the raw output of most 3D data acquisition devices. Hence, the design of intelligent computational models that act directly on point clouds is critical, especially when efficiency considerations or noise preclude the possibility of expensive denoising and meshing procedures. While hand-designed features on point clouds have long been proposed in graphics and vision, however, the recent overwhelming success of convolutional neural networks (CNNs) for image analysis suggests the value of adapting insight from CNN to the point cloud world. To this end, we propose a new neural network module dubbed EdgeConv suitable for CNN-based high-level tasks on point clouds including classification and segmentation. EdgeConv is differentiable and can be plugged into existing architectures. Compared to existing modules operating largely in extrinsic space or treating each point independently, EdgeConv has several appealing properties: It incorporates local neighborhood information; it can be stacked or recurrently applied to learn global shape properties; and in multi-layer systems affinity in feature space captures semantic characteristics over potentially long distances in the original embedding. Beyond proposing this module, we provide extensive evaluation and analysis revealing that EdgeConv captures and exploits fine-grained geometric properties of point clouds. The proposed approach achieves state-of-the-art performance on standard benchmarks including ModelNet40 and S3DIS.

**phs**

- **/ariable length** epresentation
- nputs are tracks, clusters, or PF candidates
- Jse **Message-Passing Neural Networks** (MPNN)




Easy to visualize
Benefit from image processing literature
Exploits full info (including distance btw pixels)
Images are sparse and w/o clear edges
Preprocessing is non-trivial
Non-trivial convolutional filters for non-uniform cell sizes

Captures all constituents with full granularity
Can handle sparse images and non-uniform cell sizes
Exploits calo. reconstruction (cells⟶clusters)
**No unique order / structure**
Each node is connected only to neighbor

Captures all constituents with full granularity
Can learn jet clustering algo (adjacency matrix)
Can be non-local
Non-trivial to visualize

Jet-Images – Deep Learning Edition, 1511.05190

QCD-Aware RNNs for Jet Physics, 1702.00748

Neural Message Passing for Jet Physics, NIPS DLPS 107

# Data Representations: Variable-Length Representations

*order-independent* variable-length representations!

**Energy Flow Networks: Deep Sets for Particle Jets**

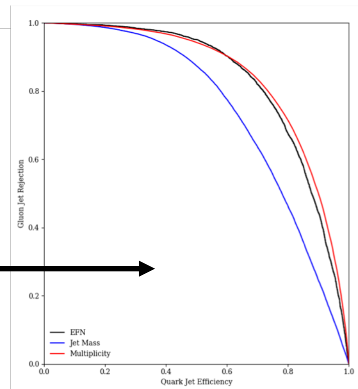
Patrick T. Komiske, Eric M. Metodiev, Jesse Thaler

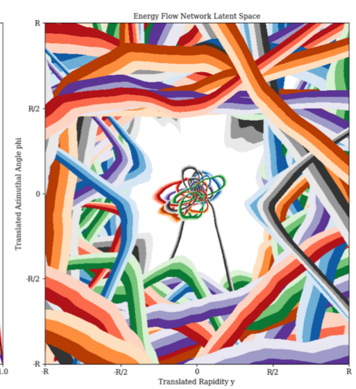
arXiv:1810.05165 [hep-ph]

(Submitted on 11 Oct 2018 (v1), last revised 11 Jan 2019 (this version, v2))

A key question for machine learning approaches in particle physics is how to best represent and learn from collider events. As an event is intrinsically a variable-length unordered set of particles, we build upon recent machine learning efforts to learn directly from sets of features or "point clouds". Adapting and specializing the "Deep Sets" framework to particle physics, we introduce Energy Flow Networks, which respect infrared and collinear safety by construction. We also develop Particle Flow Networks, which allow for general energy dependence and the inclusion of additional particle-level information such as charge and flavor. These networks feature a per-particle internal (latent) representation, and summing over all particles yields an overall event-level latent representation. We show how this latent space decomposition unifies existing event representations based on detector images and radiation moments. To demonstrate the power and simplicity of this set-based approach, we apply these networks to the collider task of discriminating quark jets from gluon jets, finding similar or improved performance compared to existing methods. We also show how the learned event representation can be directly visualized, providing insight into the inner workings of the model. These architectures lend themselves to efficiently processing and analyzing events for a wide variety of tasks at the Large Hadron Collider. Implementations and examples of our architectures are available online in our EnergyFlow package.

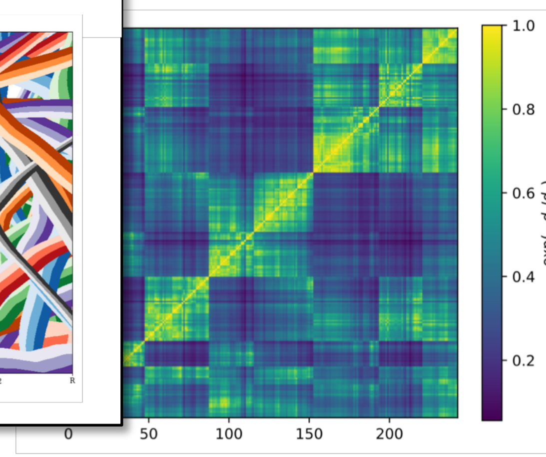
- Code with simple examples at https://energyflow.network/
- *q / g* tagging "out-of-the-box" from Anil at UIUC




phs

- /ariable length epresentation
- nputs are tracks, clusters, or PF candidates
- Jse **Message-Passing Neural Networks** (MPNN)




Easy to visualize
Benefit from image processing literature
Exploits full info (including distance btw pixels)
Images are sparse and w/o clear edges
Preprocessing is non-trivial
Non-trivial convolutional filters for non-uniform cell sizes

Captures all constituents with full granularity
Can handle sparse images and non-uniform cell sizes
Exploits calo. reconstruction (cells⟶clusters)
**No unique order / structure**
Each node is connected only to neighbor

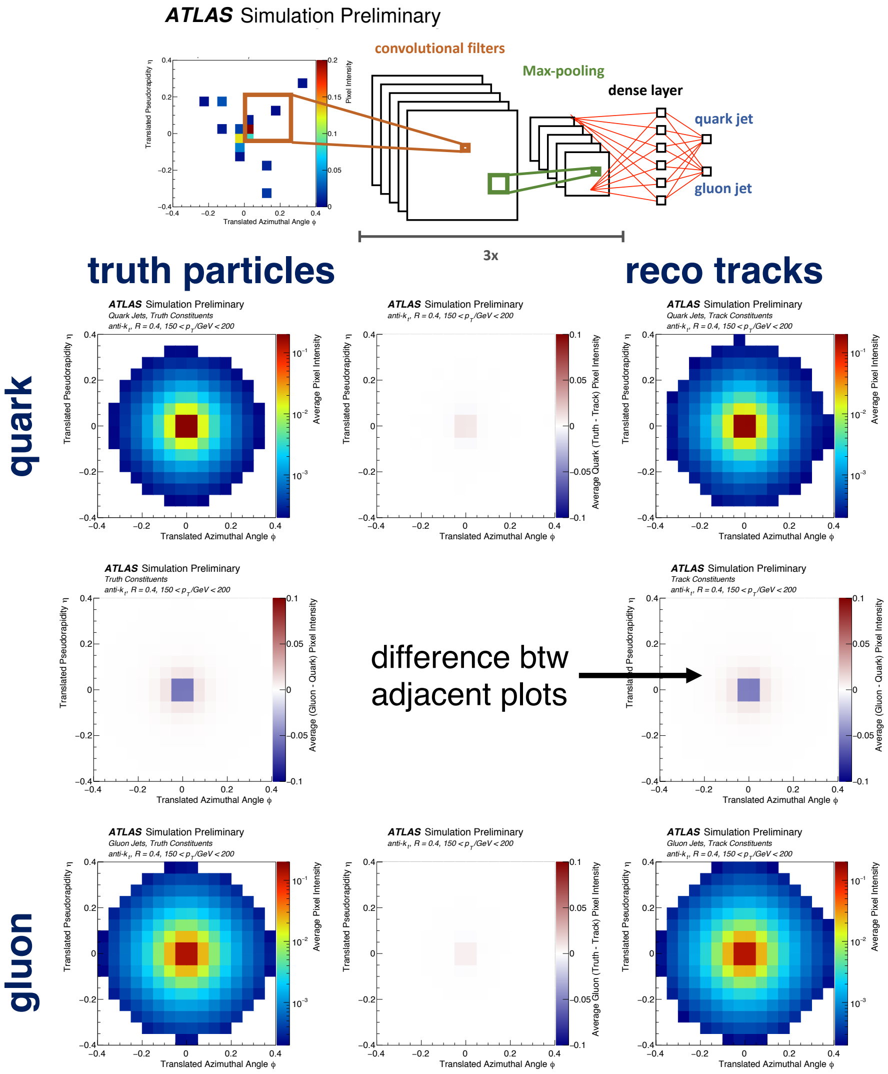
Captures all constituents with full granularity
Can learn jet clustering algo (adjacency matrix)
Can be non-local
Non-trivial to visualize

Jet-Images – Deep Learning Edition, 1511.05190

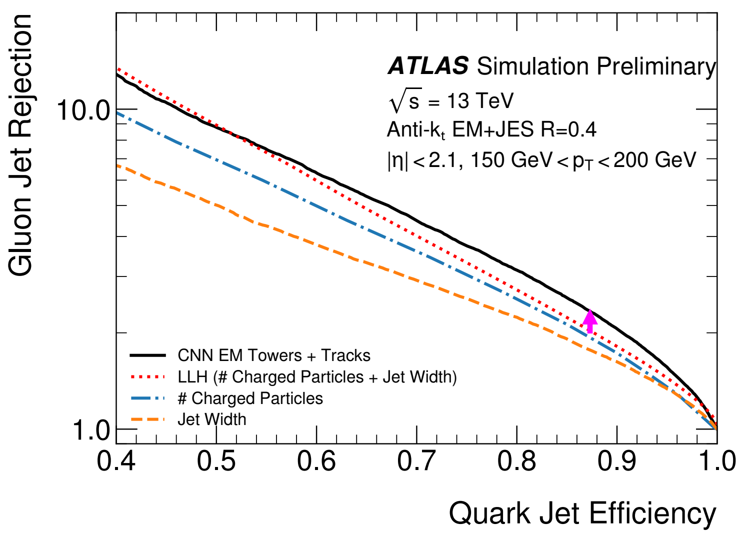
QCD-Aware RNNs for Jet Physics, 1702.00748

Neural Message Passing for Jet Physics, NIPS DLPS 107

# q / g Discrimination in ATLAS



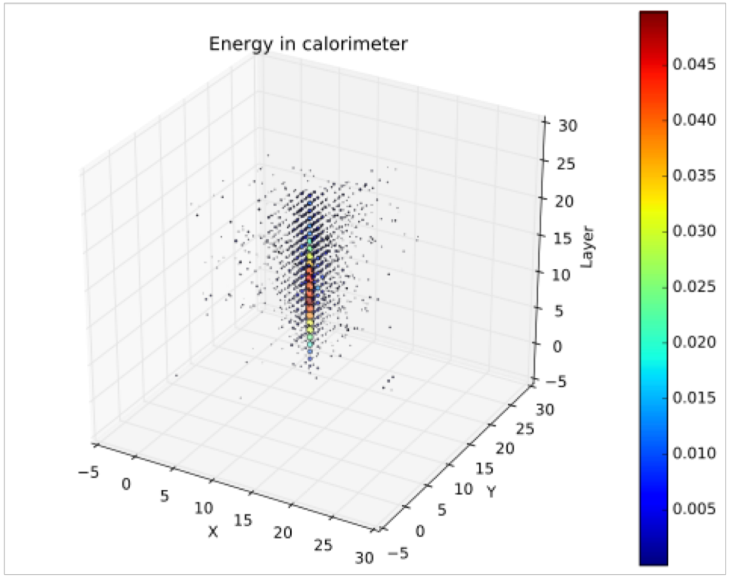

- 1st ATLAS result on ML imaging!
- CNN trained on EM towers + tracks **outperforms** classifiers based on # charged particles and jet width



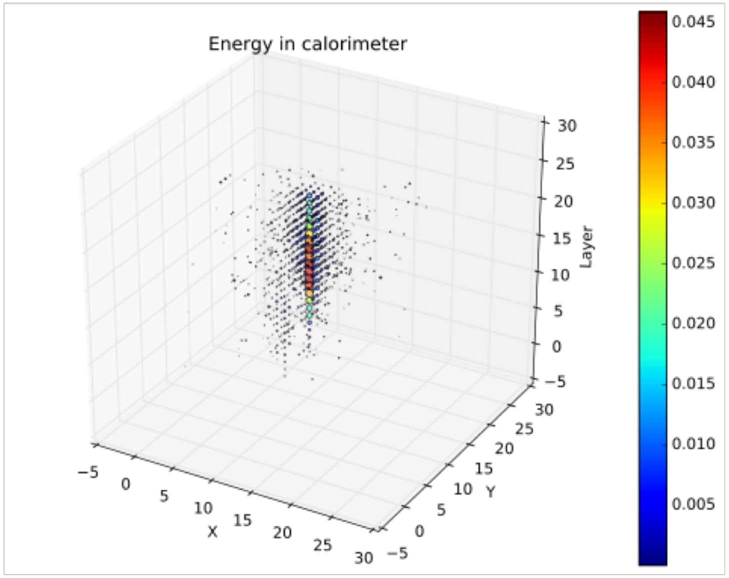

**Quark versus Gluon Jet Tagging Using Jet Images with the ATLAS Detector,** ATL-PHYS-PUB-2017-017

# ML Imaging for Particle ID

Example 60 GeV
photon shower




Example 60 GeV
$\pi_0 \rightarrow \gamma\gamma$ shower
(opening Angle < 0.01 rad)

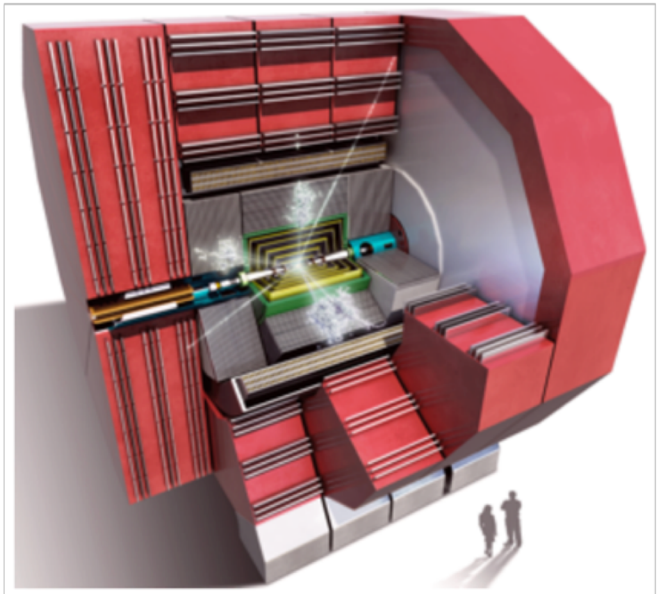


Q: can you name these celebrities?

A: they don't exist! Images generated by GANs

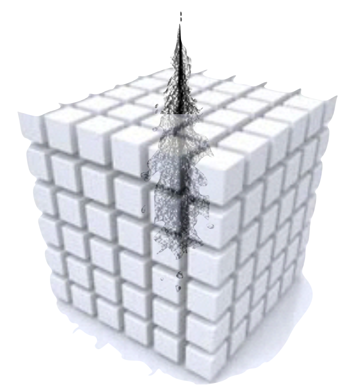
NVIDIA, "Progressive Growing of GANs for Improved Quality, Stability, and Variation," [ICLR 2018]

# Calorimetry with Machine Learning

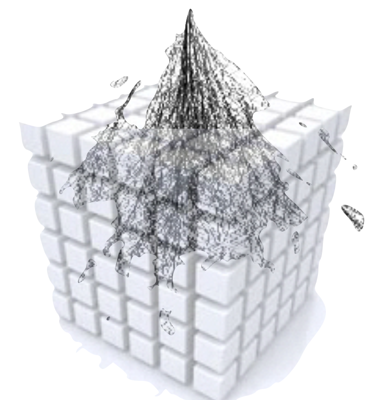


Large Collider Detector (LCD) for proposed CLIC machine

single particle showers in high-granularity 3D calorimeter



electromagnetic shower



hadronic shower

- **Classification**, **regression**, and **fast simulation** of single particles (e, $\pi^+$, $\gamma$, $\pi^0$) with high-granularity calorimeter
- Full Geant-based simulation of CLIC LCD detector
- Investigate improvements from cell-based DNNs and CNNs w.r.t. feature-based DNNs and BDTs

Calorimetry with Deep Learning: Particle Classification, Energy Regression, and Simulation for High-Energy Physics [pdf]
Benjamin Hooberman, Amir Farbin, Gulrukh Khattak, Vitória Pacela, Maurizio Pierini, Jean-Roch Vlimant, Maria Spiropulu, Wei Wei, Matt Zhang and Sofia Vallecorsa

Proceedings of the Deep Learning for Physical Sciences Workshop at Neural Information and Processing Systems (NeurIPS17)

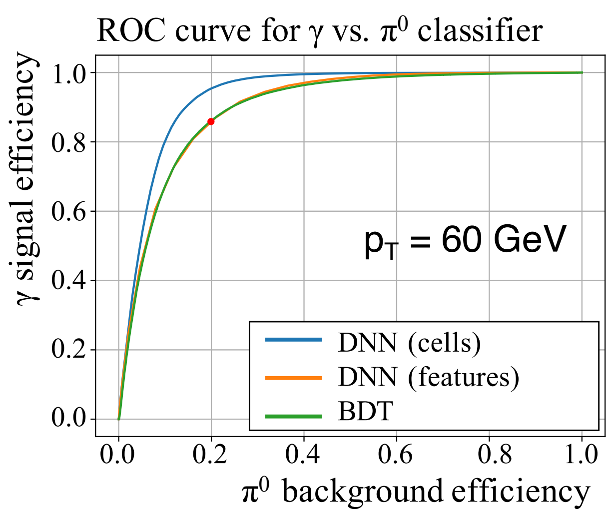
Calorimetry with Deep Learning: Particle Identification and Simulation for Collider Physics

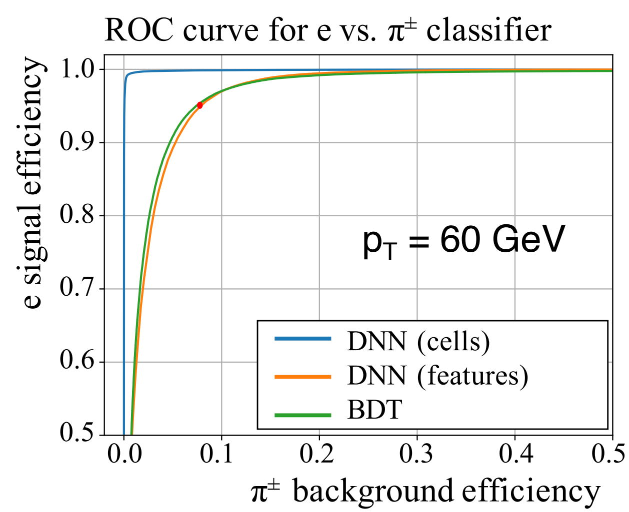
Dawit Belayneh[1], Federico Carminati[2], Amir Farbin[3], Benjamin Hooberman[4], Gulrukh Khattak[25], Miaoyuan Liu[6], Junze Liu[4], Dominick Olivito[7], Vitória Barin Pacela[8], Maurizio Pierini[2], Alexander Schwing[4], Maria Spiropulu[9], Sofia Vallecorsa[2], Jean-Roch Vlimant[9], Wei Wei[4], and Matt Zhang[a4]

to appear soon

# Classification Results

- Train classifiers for $\gamma$ **vs.** $\pi^0(\rightarrow \gamma\gamma)$ and **e vs.** $\pi^\pm$
  - Apply filters to select hadrons that mimic signal: for $\pi^0$ require $\theta(\gamma,\gamma)<0.1$, for $\pi^+$ require H/E < 1/40
- Similar performance from DNN trained on features vs. BDT trained on features
- **Significant improvement in performance from DNN trained on cells**



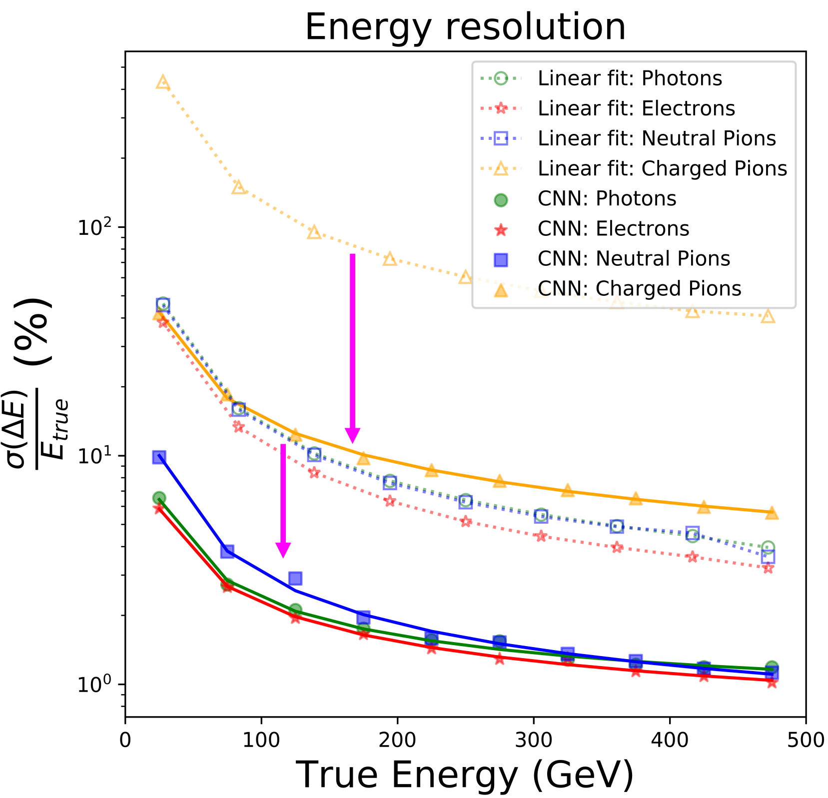




**caveats:**
- single particle showers at normal incidence only
- no noise or PU
- highly-granular calo with uniform cell cizes

| | $\gamma$ **vs.** $\pi^0$ | | | | $e$ **vs.** $\pi$ | | | |
|---|---|---|---|---|---|---|---|---|
| **Model** | **acc.** | **AUC** | $\Delta\epsilon_{\text{sig}}$ | $\Delta R_{\text{bkg}}$ | **acc.** | **AUC** | $\Delta\epsilon_{\text{sig}}$ | $\Delta R_{\text{bkg}}$ |
| BDT | 83.1% | 89.8% | - | - | 93.8% | 98.0% | - | - |
| DNN (features) | 82.8% | 90.2% | 0.9% | 0.95 | 93.6% | 98.0% | -0.1% | 0.95 |
| DNN (cells) | 87.2% | 93.5% | 9.4% | 1.63 | 99.4% | 99.9% | 4.9% | 151 |

# Regression Results

- Significant improvements in energy resolution w.r.t. linear discriminant based on total ECAL and HCAL energies




**Simple Linear Model**

| Particle Type | a | b | c |
|---|---|---|---|
| Photons | 55.5 | 1.85 | 1245 |
| Electrons | 42.3 | 1.51 | 1037 |
| Neutral pions | 55.3 | 1.71 | 1222 |
| Charged pions | 442 | 25 | 11706 |

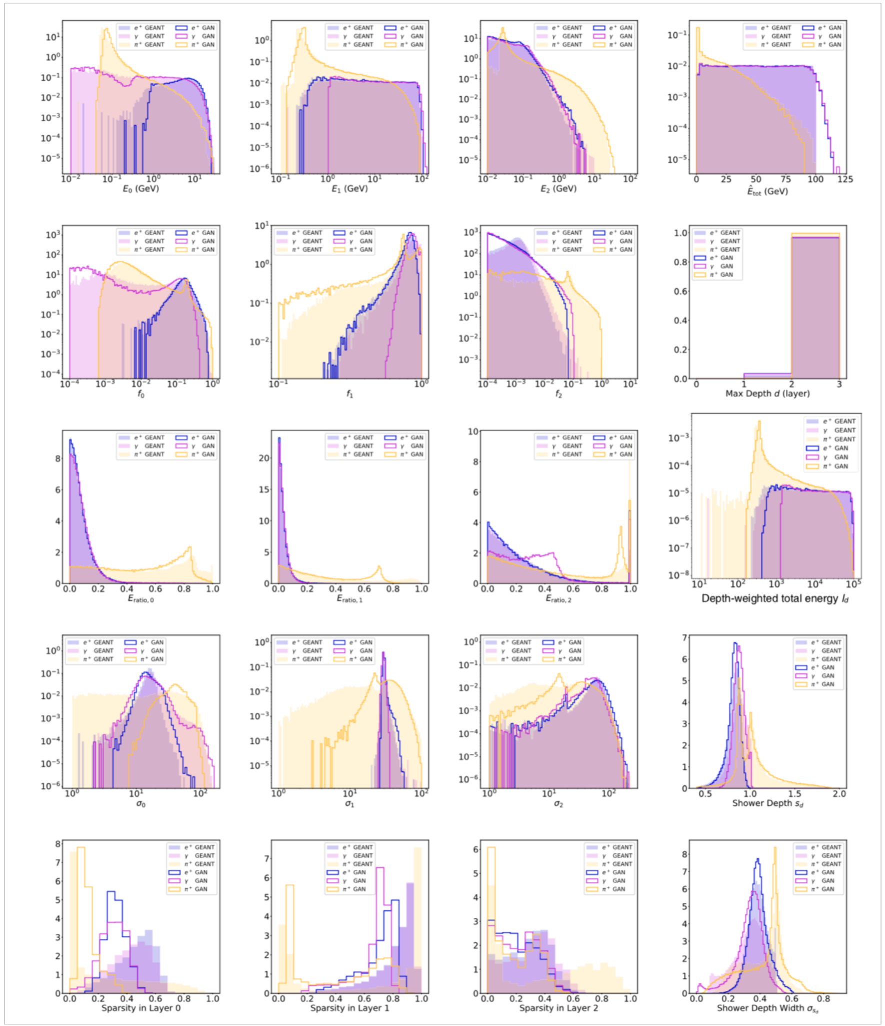
**CNN Model**

| Particle Type | a | b | c |
|---|---|---|---|
| Photons | 18.3 | 0.75 | 131 |
| Electrons | 18.7 | 0.574 | 111 |
| Neutral pions | 19.3 | 0.45 | 231 |
| Charged pions | 114 | 1.02 | 893 |

$$\frac{\sigma(\Delta E)}{E_{\text{true}}} = \frac{a}{\sqrt{E_{\text{true}}}} \oplus b \oplus \frac{c}{E_{\text{true}}}$$

**caveats:**
- single particle showers at normal incidence only
- no noise or PU
- highly-granular calo with uniform cell cizes

# Feature Modeling with GANs

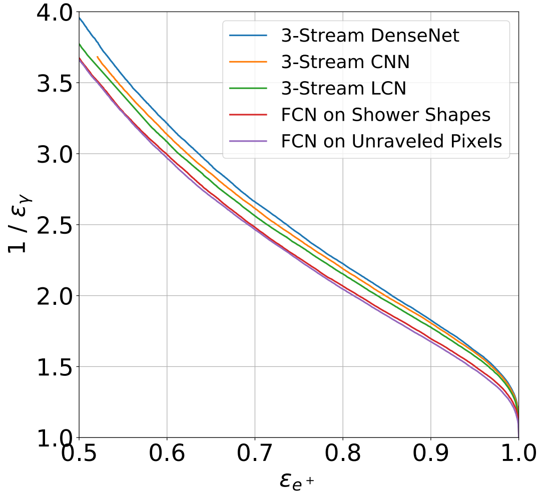


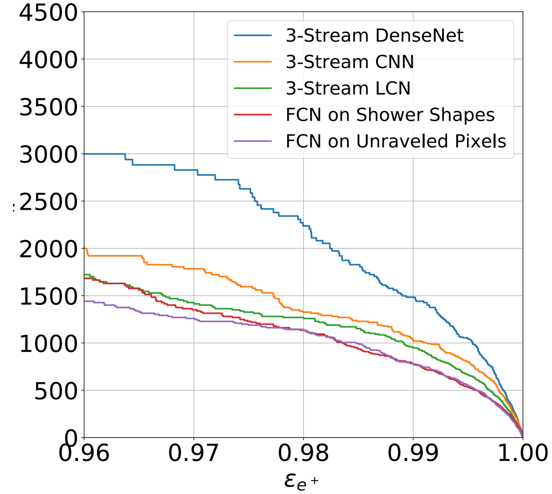
Paganini, de Oliveira, Nachman, CaloGAN for 3D particle showers, [PRD 97, 014021 (2018)]

- Validate GAN images by comparing features to Geant

# More on Particle ID

- Study e vs. $\gamma$ and e vs. $\pi^+$ classification using Geant4 model inspired by ATLAS calorimeter
- Compare performance for various network architectures for combining info from 3 LAR layers with different granularities
- Densely-connected NN provides best performance




Survey of Machine Learning Techniques for High Energy Electromagnetic Shower Classification, DLPS NIPS 2017 proceedings,
Michela Paganini, Luke de Oliveira and Benjamin Nachman

# Software & Computational Challenges

- Lots of available ML frameworks: TensorFlow/Keras (C++), PyTorch (python), etc.
- **Training** often performed with private python-based Keras+TensorFlow
  ⟶ few processes, single threads, little memory constraints, expendable jobs
- **Production** performed in custom C++ framework with ROOT-based I/O
  ⟶ many processes / threads, memory constraints, processes can't die
- *Deploying ML approaches at full scale with efficient multi-threading and memory usage is a key challenge*
- **MXNet:** a flexible & scalable library for ML https://mxnet.apache.org/
  - Acceleration libraries to fully exploit GPU and cloud computing capabilities
    - Device placement, multi-GPU training, automatic differentiation, optimized predefined layers
  - Speed-up of ~2-3× observed (for DeepAK8) [1]
  - Need to tackle issues of thread safety
- *Defining networks in format that is independent of ML framework is highly desirable*

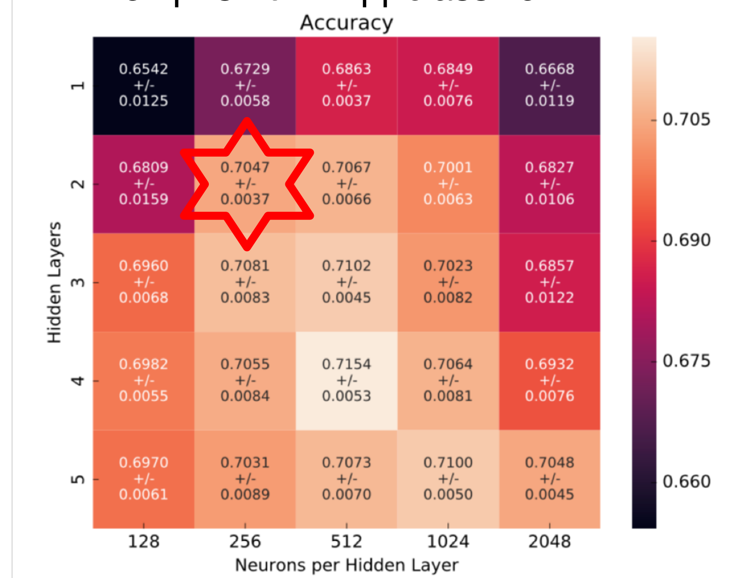
[1] see talk M. Verzetti, Fermilab ML for jets workshop, https://indico.cern.ch/event/745718/

# Hyperparameter Optimization with HPCs

- Optimizing network architecture & hyperparameters is **computationally expensive**
  - Different architectures (CNN, DNN, RNN, etc.), loss functions, gradient descent methods, etc.
  - Tuning hyperparameters: number of layers, neurons per layer, learning & dropout rates, etc.
- Scan over permutations and compare performance metrics (AUC, accuracy, etc.)
- Highly parallelizable task ⟶ optimal for HPC / **supercomputer!**
  - Good HEP use case for GPU-enabled machines

- **Blue Waters Supercomputer** at National Center for Supercomputing Applications
  - Largest Supercomputer on a university campus
  - Cray XE/XK hybrid machine with 2.3 GHz AMD 6276 *Interlagos* processors and NVIDIA GK110 (K20X) *Kepler* accelerators
  - **4228 GPU-enabled XK nodes with 25 TB memory**

DNN hyperparameter scan for $\gamma$ vs. $\pi^0 \longrightarrow \gamma\gamma$ classifier




Calorimetry with Deep Learning: Particle Classification, Energy Regression, and Simulation for High-Energy Physics, Belayneh, Farbin, BH, Khattak, Liu, Olivito, Pacela, Pierini, Schwing, Spiropulu, Vallecorsa, Vlimant, Wei, Zhang, to appear soon

# An Affordable PC with a Powerful GPU

PC with **GeForce 2080 Ti GPU** for $2,000
https://pcpartpicker.com/user/anilr2/saved/HYPYTW

| Component | Selection | Base | Promo | Shipping | Tax | Price | Where | |
|---|---|---|---|---|---|---|---|---|
| CPU | AMD Ryzen 7 2700X 3.7 GHz 8-Core Processor | $238.63 | — | FREE | — | $238.63 | OutletPC | Buy |
| Motherboard | Gigabyte X470 AORUS ULTRA GAMING ATX AM4 Motherboard | $135.99 | — | Prime | — | $135.99 | amazon.com | Buy |
| Memory | Corsair Vengeance LPX 16 GB (2 x 8 GB) DDR4-3000 Memory | $77.99 | — | Prime | — | $77.99 | amazon.com | Buy |
| Storage | Samsung 970 Pro 1 TB M.2-2280 NVME Solid State Drive | $299.99 | — | Prime | — | $299.99 | amazon.com | Buy |
| Video Card | Zotac GeForce RTX 2080 Ti 11 GB GAMING AMP Video Card | $1169.99 | — | Prime | — | $1169.99 | amazon.com | Buy |
| Case | Fractal Design Focus G ATX Mid Tower Case | $55.88 | — | Prime | — | $55.88 | amazon.com | Buy |
| Power Supply | EVGA SuperNOVA G3 650 W 80+ Gold Certified Fully Modular ATX Power Supply | $121.88 | -$20.00 [1] | FREE | — | $101.88 | OutletPC | Buy |
| | | | | | Base Total: | $2100.35 | | |
| | | | | | Mail-in Rebates: | -$20.00 | | |
| | | | | | Total: | $2080.35 | | |

# ML Imaging for Object Classification & Regression

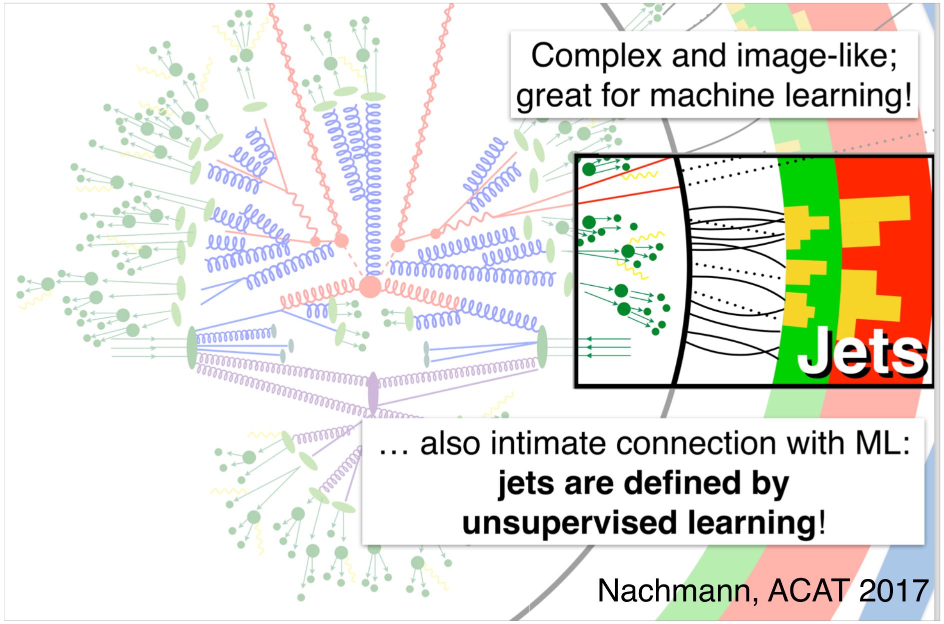
- **Lots of possible applications:**
- **Jets**
  - Jet substructure (W/top-tagging)
  - Quark vs. gluon discrimination
  - b/c-tagging
  - Measuring $E_T$, $\eta$, $\phi$
  - Pileup mitigation
- **Leptons**
  - Electron classification and $E_T$, $\eta$, $\phi$
  - Tau classification and $E_T$, $\eta$, $\phi$
- **Photons**
  - Photon classification and $E_T$, $\eta$, $\phi$
- **MET**
  - MET measurement

Jet-Images: Computer Vision Inspired Techniques for Jet Tagging, JHEP 02 (2015) 118
Jet Images – Deep Learning Edition, JHEP07 (2016) 069
Jet Substructure Classification in High-Energy Physics with Deep Neural Networks, PRD 93 (2016) 094034
Jet Constituents for Deep Neural Network Based Top Quark Tagging, arXiv:1704.02124
Parton Shower Uncertainties in Jet Substructure Analyses with DNNs, PRD 95, 014018 (2017)
How Much Information is in a Jet? JHEP 06, (2017) 073
Novel Jet Observables from Machine Learning, arXiv:1710.01305 [hep-ph]
Energy flow polynomials: A complete linear basis for jet substructure, arXiv:1712.07124 [hep-ph]
Jet Substructure at the LHC: A Review of Recent Advances in Theory and Machine Learning, arXiv:1709.04464 [hep-ph]
Deep-learning Top Taggers or The End of QCD? JHEP 05 (2017) 006
New Developments for Jet Substructure Reconstruction in CMS, CMS-DP-2017-027
ML Techniques for the Identification of Hadronic W Bosons and Top Quarks in ATLAS, ATLAS-PHYS-PUB-2017-004

Deep learning in color: towards automated quark/gluon jet discrimination, JHEP 01 (2016) 110
Quark versus Gluon Jet Tagging Using Jet Images with the ATLAS Detector, ATL-PHYS-PUB-2017-017

Pileup Mitigation with Machine Learning (PUMML), JHEP 12 (2017) 051

Calorimetry with Deep Learning: Particle Classification, Energy Regression, and Simulation for HEP, DLPS NIPS 2017 proceedings
Survey of ML Techniques for High Energy Electromagnetic Shower Classification, DLPS NIPS 2017 proceedings



- asd

# Neural Networks in a Nutshell

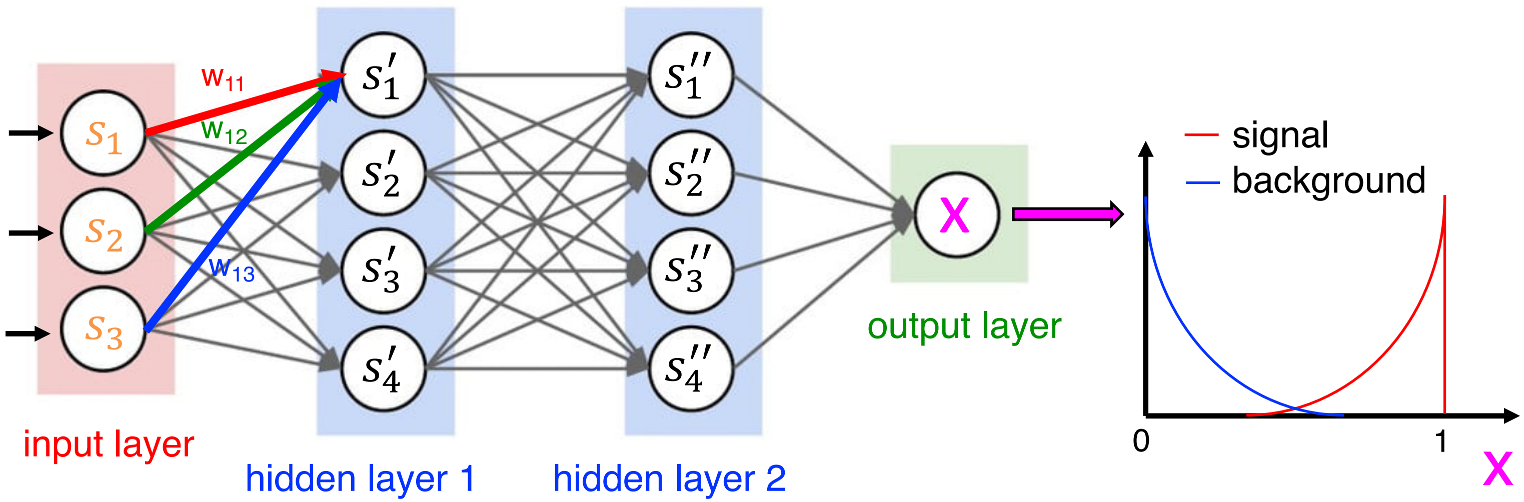
$$\text{loss} = \langle (f(\vec{s}) - 1)^2 \rangle_{\text{signal}} + \langle (f(\vec{s}) - 1)^2 \rangle_{\text{bkg}}$$

inputs can be…

**features**
($n_{hits}$, $E_{tot}$, etc.)

or

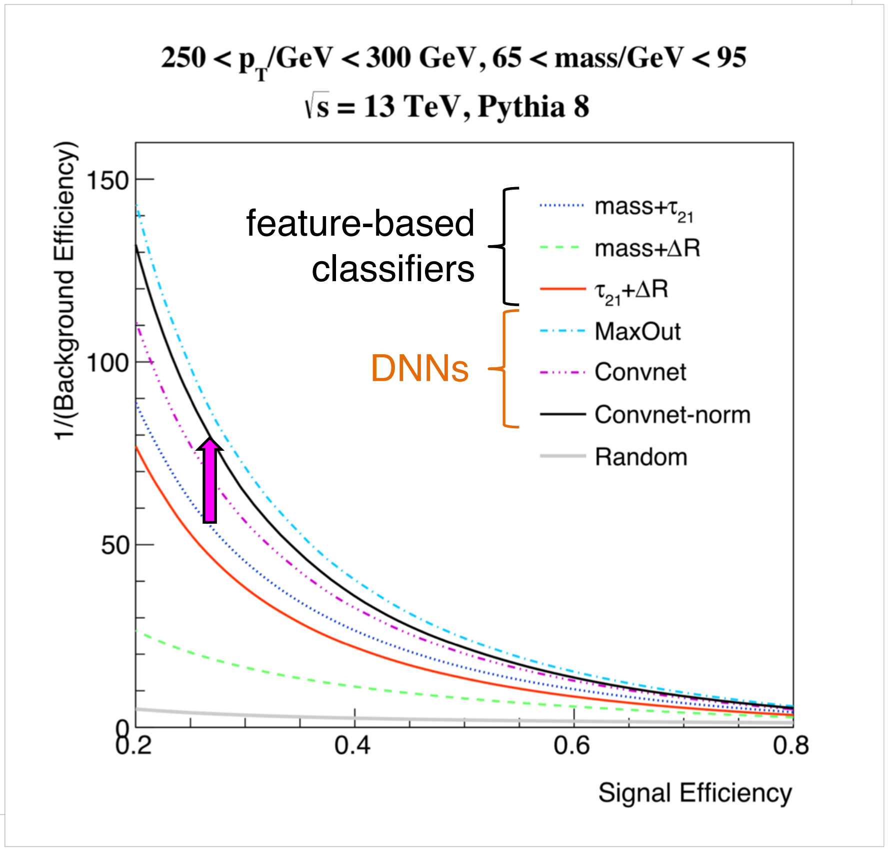
**low-level data**
(tracks, hits / clusters, particle flow candidates




$$s'_i = \Sigma\, w_{ij}\, s_j$$

- Multiple layers of nodes ("neurons") interconnected with variable weights
  - Train weights $W_{ij}$ by minimizing loss function, using backpropagation and steepest descent
- Universal Approximation Theorem: NNs can approximate any continuous function [1]

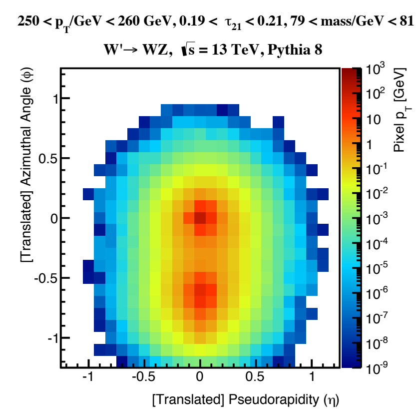
# Example of Jet Classification: W-tagging



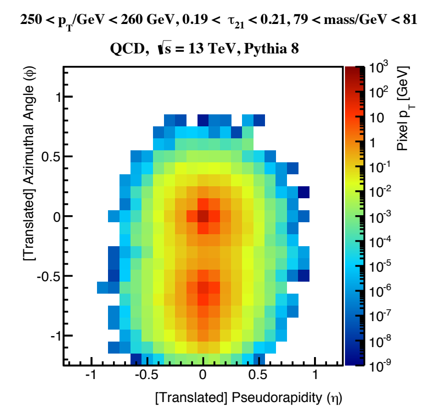

[1] de Oliveira, Kagan, Mackey, Nachman, Schwartzman, “Jet-images – deep learning edition,” JHEP 07:069, 2016

- Seminal work demonstrated W➞jj tagging using jet images [1]
- DNN outperforms feature-based classifiers ➞ there is additional info in the cells (beyond mass, $\tau_{21}$, $\Delta_R$)

**signal** (boosted W➞jj)



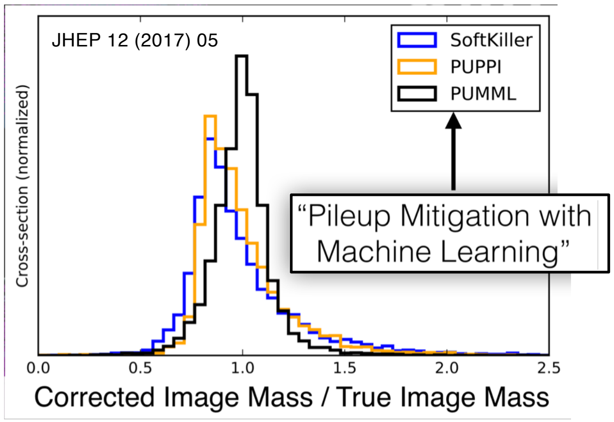

**background** (q / g jet)



# Jets: Selected Results

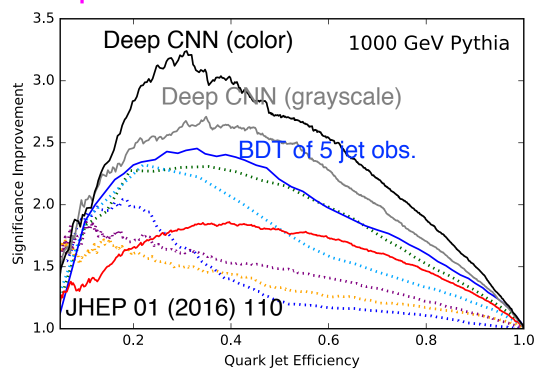
**Pileup Mitigation**

*parton-level




deep learning in color*:
**quark/gluon jet discrimination**

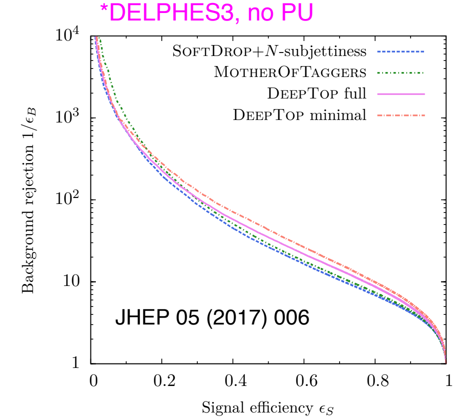
*parton-level




red= transverse momenta of charged particles
green= the transverse momenta of neutral particles
blue= charged particle multiplicity

**top tagging**

*DELPHES3, no PU




- DNNs and CNNs tend to outperform feature-based classifiers
- Caveats*: parton-level or fast simulation → no detector resolution, noise, PU, dead cells, material upstream of calorimeter, etc