Jet tagging in $p - p$ collision at LHC using several approaches of Machine Learning.

ORAL

Maxime Munari

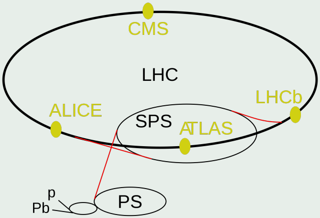
Université de Strasbourg

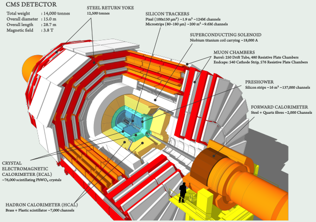
Supervisor: Gourab Saha,

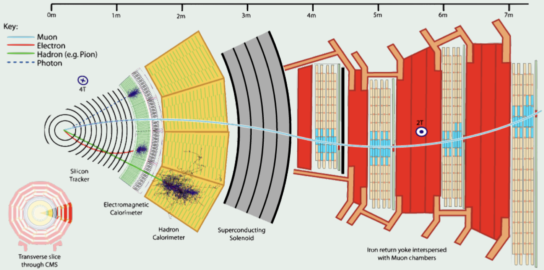
May, 2024

Contents

Compact Muon Solenoid

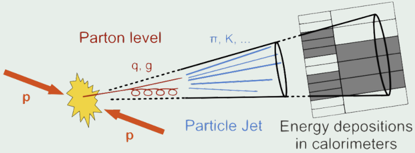






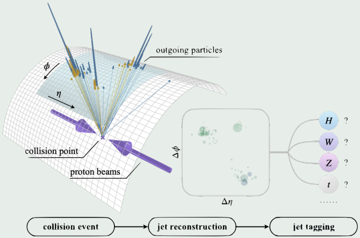
What is a Jet?

- *"Narrow cone of hadrons and other particles produced by the hadronization of a quark or gluon in a particle physics."* [twiki.cern.ch]
- A collection of partons that are clustered in a cone shape.



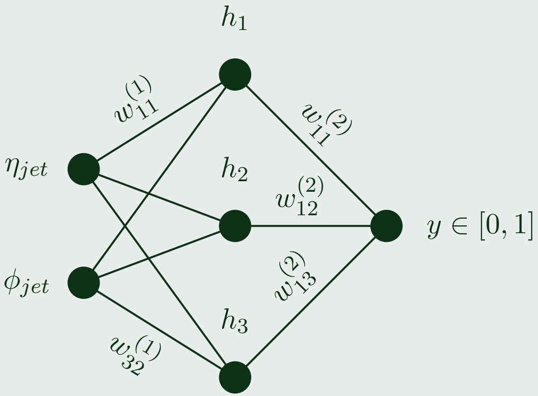
Université de Strasbourg

Jet Tagging



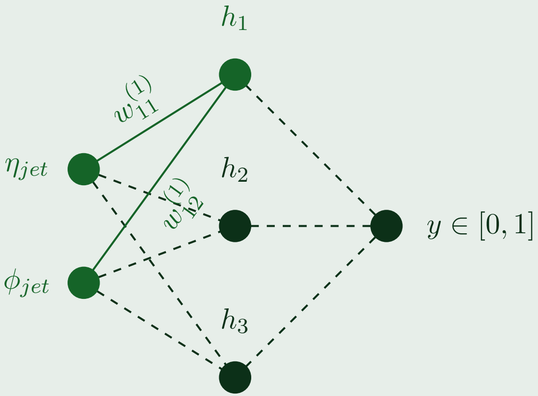
- Discriminate Quark and Gluon jets (tagging)
- Standard Model measurements
- Search for New Particles (BSM) decaying to Quarks
- Suppressing Gluon background

Nodes and Weights Representation



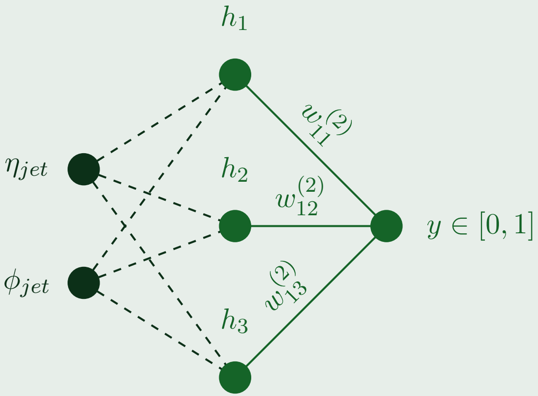
- Given (η, ϕ) what should y be?
- Adjust $\{w_{ij}\}$ to make the best predictions.

Update Nodes



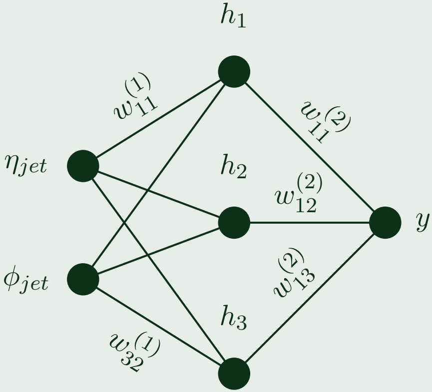
$$h_1 = \eta_{jet} w_{11}^{(1)} + \phi_{jet} w_{12}^{(1)}$$

Compute Output



$$y = h_1 w_{11}^{(2)} + h_2 w_{12}^{(2)} + h_3 w_{13}^{(2)}$$

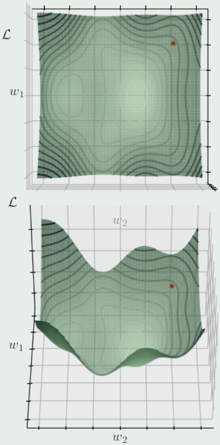
General Expression



$$x_i^{(l)} = \mathrm{ReLU}\left(\sum_{j=1}^{N} x_i^{(l-1)} w_{ij}^{(l)} + b_i^{(l)}\right)$$

with $\mathrm{ReLU}(x) = \max(0, x)$.

Gradient Descent



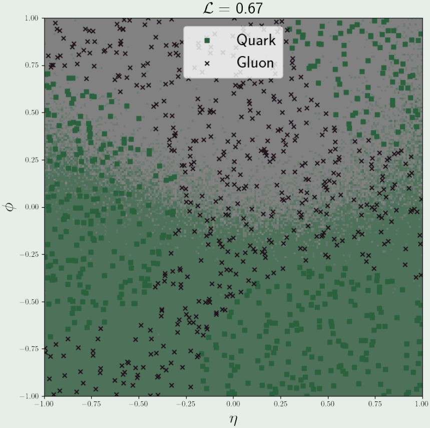
- Calculate the loss $\mathcal{L}$ of the prediction (e.g. MSE, Cross Entropy, ...):

$$\mathcal{L} = -\sum_{i=1}^{N} [y_\mathrm{t} \log(y_\mathrm{p}) + (1 - y_\mathrm{t}) \log(1 - y_\mathrm{p})]$$

- Update Weights and Biases using $\nabla\mathcal{L}$ until Convergence:

$$w_{ij}^{(l+1)} = w_{ij}^{(l)} - \alpha \frac{\partial \mathcal{L}(X, w^{(l)})}{\partial w_{ij}}$$

Example: Classification



- Jets inside detector's plane.
- **Data not physically accurate.**
- Accuracy: 0.968
- It is a **function**:

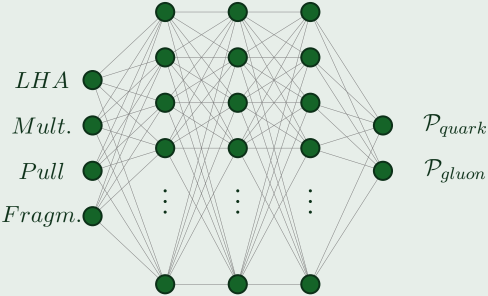
$$\text{Network}(\eta, \phi) = \{Q, G\}$$

Jets Dataset

Label	ϕ_{jet}	η_{jet}	$\dots$	p^T_{jet}	$\dots$	ϕ_{part}	ϕ_{part}	p^T_{part}	$\dots$

- Events Generation: PYTHIA8 $(Z(\rightarrow \nu\bar{\nu}) + \{g \text{ or } q\})$
- Reconstruction: Anti-k_T for $R = 0.4$
- COM Energy $\sqrt{s} = 14$ TeV

Deep Neural Network



- Input is a vector of variables computed using jet variables.
- Output is $\mathcal{P}$ of being a Quark or Gluon Jet.
- 3 hidden layers with 64 nodes each.
- Activation function: ReLU
- L2-Regularisation (Inputs Normalisation) + Dropout

Energy Flow Networks

Deep Sets

Manzil Zaheer[1,2], **Satwik Kottur**[1], **Siamak Ravanbhakhsh**[1],
Barnabás Póczos[1], **Ruslan Salakhutdinov**[1], **Alexander J Smola**[1,2]
[1] Carnegie Mellon University [2] Amazon Web Services
{manzilz,skottur,mravanba,bapoczos,rsalakhu,smola}@cs.cmu.edu

Apr 2018

Abstract

We study the problem of designing models for machine learning tasks defined on *sets*. In contrast to traditional approach of operating on fixed dimensional vectors, we consider objective functions defined on sets that are invariant to permutations. Such problems are widespread, ranging from estimation of population statistics [1],

Energy Flow Networks: Deep Sets for Particle Jets

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

Center for Theoretical Physics, Massachusetts Institute of Technology,
77 Massachusetts Avenue, Cambridge, MA 02139, U.S.A.

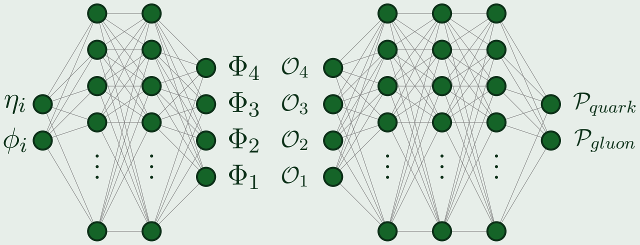
Department of Physics, Harvard University,
17 Oxford Street, Cambridge, MA 02138, U.S.A.

E-mail: pkomiske@mit.edu, metodiev@mit.edu, jthaler@mit.edu

h] 11 Jan 2019

- ML technics on sets.
- Inputs are Point Clouds.
- Permutation Invariance.
- Built from scratch.

Architecture



$$\Phi(\eta_i, \phi_i) \qquad F\left(\sum_{i=1}^{N} p_i^T \Phi(\eta_i, \phi_i)\right)$$

$$\mathbb{I} \to \mathbb{R}^l \qquad \mathbb{R}^l \to \mathbb{T}$$

- Input is a Point Cloud $\{(\eta_{\text{part}}, \phi_{\text{part}}), \forall \text{part} \in \text{jet}\}$
- Φ is per-particule mapping (latent space)
- F aggregates information and map to target space $\mathbb{T}$

ParticleNet (GNN)

Jet Tagging via Particle Clouds

Huilin Qu*
Department of Physics, University of California, Santa Barbara, California 93106, USA

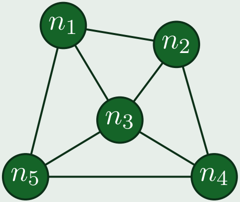
Loukas Gouskos†
CERN, CH-1211 Geneva 23, Switzerland

How to represent a jet is at the core of machine learning on jet physics. Inspired by the notion of point clouds, we propose a new approach that considers a jet as an unordered set of its constituent particles, effectively a "particle cloud". Such a particle cloud representation of jets is efficient in incorporating raw information of jets and also explicitly respects the permutation symmetry. Based on the particle cloud representation, we propose ParticleNet, a customized neural network architecture using Dynamic Graph Convolutional Neural Network for jet tagging problems. The ParticleNet architecture achieves state-of-the-art performance on two representative jet tagging benchmarks and is improved significantly over existing methods.

- Jets are *Particle Cloud.*
- Dyn.Graph Conv. Neural Network.
- Built from scratch.

Université de Strasbourg

Graph Neural Network



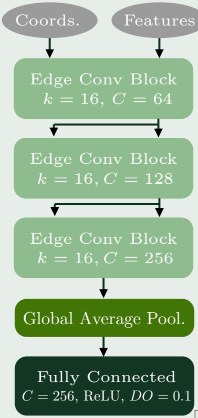
- Input is a graph (Nodes + Edges)
- A graph represents a jet $\leftrightarrow$ Each node represents a particle
- Each node has node features $\{\eta_i, \phi_i, \dots\}$
- Each node has k Neighbors (k-NN)

Particle Net

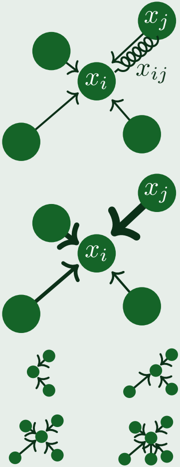
- Uses Graph information (Neighbours *Correlations*).
- EdgeConv (Conv. on PC):

$$x'_i = \mathop{\square}_{j=1}^{k} h_\theta(x_i, x_{i_j})$$

with h_θ a MLP.



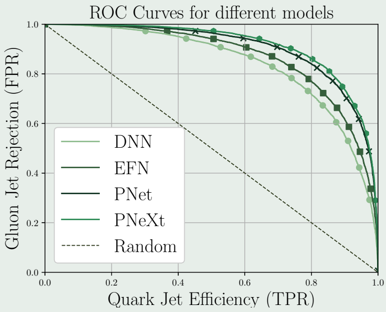
Particle NeXt



- Adding pairwise features (edge features)
- Add different importance to each neighbour
- Concatenate multiple k to travel more in depth into the graph.

Université de Strasbourg

Results



	Accuracy	AUC	Litt. AUC
DNN	0.781	0.836	0.839
EFN	0.797	0.862	0.873
PNet	0.822	0.903	0.911
PNeXt	0.829	0.910	N/A

Conclusion

- Explored jet physics.
- Built from scratch several architectures of Machine Learning.
- Successfully recovered results from literature.
- Future plans:
  - Explore Particle Transformers.
  - Tag on real data from Open Data (CERN).

Thank you!

Anti-k_t Algorithm

- The k_t and Cambridge/Aachen algorithms are defined as:

$$d_{ij} = \min(k_{ti}^{2p}, k_{tj}^{2p}) \frac{\Delta_{i,j}^2}{R^2}$$

with $\Delta_{i,j} = \sqrt{(\eta_i - \eta_j)^2 + (\phi_i - \phi_j)^2}$ and k_t the transverse momentum.

- The anti-k_t algorithm corresponds to $p = -1$.
- A particle belongs to the jet if $d_{ij} < \Delta R$.

Clustering Algortihms

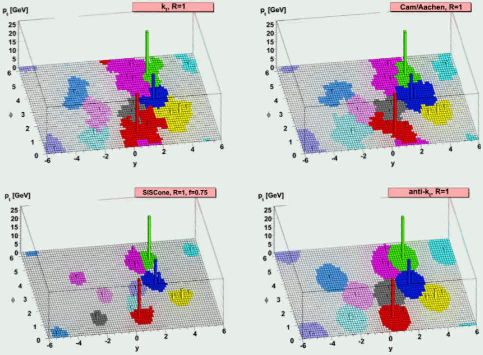


Figure: Comparison of different jet algorithms resp. k_t, Cam/Aachen, SISCone and anti-k_t. (Matteo Cacciari and Gavin P. Salam.)

Generalized Angles

The generalized angles are defined as:

$$\lambda_\beta^\kappa = \sum_i \left(\frac{p_{T,i}}{p_{T,\mathrm{jet}}}\right)^\kappa \left(\frac{\Delta R(i,j)}{R}\right)^\beta$$

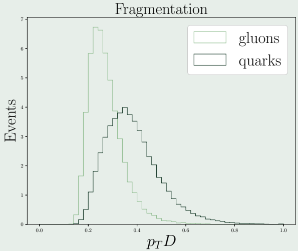
and characterize the distribution of energy within a jet.

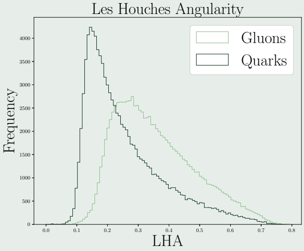
Some useful angles are defined for:

	LHA	Girth	Width	p_T^2	Mult.
κ	1	1	1	2	0
β	0.5	2	1	0	0

Université de Strasbourg

Generalized Angles: Examples





Université de Strasbourg

Receiver Operating Characteristic

- $\mathrm{TPR} = \dfrac{\mathrm{TP}}{\mathrm{TP+FN}}$
- $\mathrm{FPR} = \dfrac{\mathrm{FP}}{\mathrm{TN+FP}}$
- AUC is the degree of separability of the classes.