Learning to Remove Pileup at the LHC with Jet Images

ACAT 2017

Eric M. Metodiev

Center for Theoretical Physics, Massachusetts Institute of Technology

Work with Patrick T. Komiske, EMM, Benjamin P. Nachman, Matthew D. Schwartz

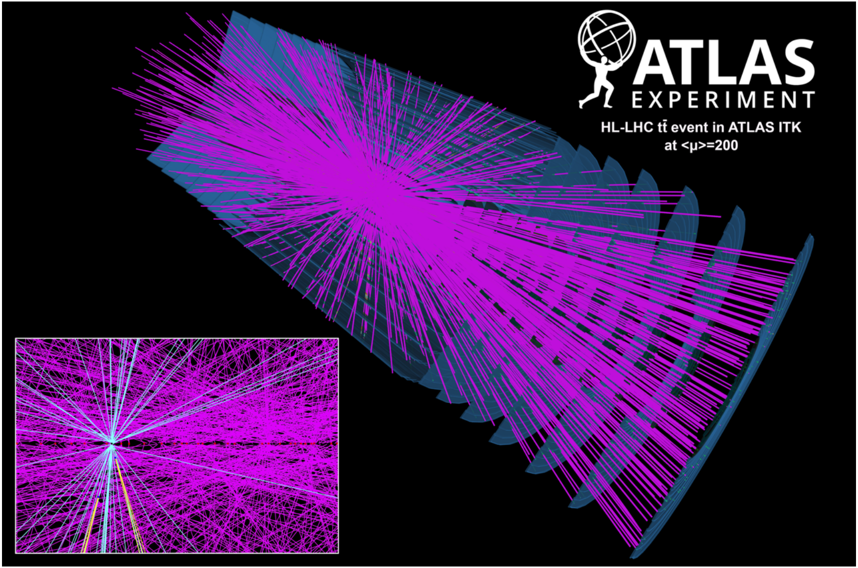
arXiv:1707.08600

August 22, 2017

Overview

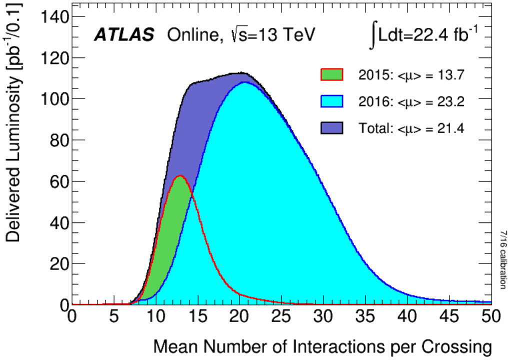
- Pileup
- Jet Images
- Pileup Mitigation with Machine Learning (PUMML)
- Performance and Robustness
- What is being learned?

Pileup



Pileup

- Pileup problem in context
  - Presently: $\sim$20 pileup vertices per bunch crossing
  - Run 3: $\sim$80 pileup vertices per bunch crossing
  - HL-LHC: $\sim$200 pileup vertices per bunch crossing



Machine Learning?

- How to input the information?
  - The spirit is to organize all of our available local information.
  - Have information on whether charged particles are pileup or not.
  - Need low-level inputs.

- What sort of architecture?
  - Use tools from modern machine learning.
  - Don't necessarily have to go "deep"

- What sort of loss function?

Mitigation Approaches

Pileup Per Particle Identification (PUPPI)

- Bertolini, Harris, Low, and Tran, arXiv:1407.6013
- Correct particle/calorimeter energies based on surrounding charged pileup distribution.

SoftKiller

- Cacciari, Salam, Soyez, arXiv:1407.0408
- Dynamically determined transverse momentum cut.

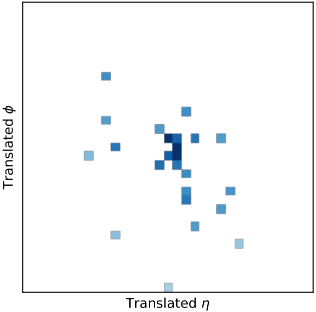
Jet Cleansing

- Krohn, Low, Schwartz, Wang, arXiv:1309.4777
- Rescaling subjet four-momenta using charged leading vertex/pileup information.

Used default parameters to give sense of performance.

Jet Images

- Treat the detector as a camera and energy deposits as pixel intensities.
  - Cogan, Kagan, Strauss, Schwartzman. arXiv:1407.5675
- Make use of the extensively developed computer vision technology, such as convolutional neural nets.
  - de Oliviera, Kagan, Mackey, Nachman, Schwartzman. arXiv:1511.05190



Modern ML in HEP

An overview of recent machine learning applications with jet images.

- Classification
  - W vs QCD jets. (de Oliviera, Kagan, Mackey, Nachman, Schwartzman. arXiv:1511.05190)
  - Top vs QCD jets. (Kasieczka, Plehn, Russell, Schell. arXiv:1701.08784)
  - Quark vs Gluon jets. (Komiske, EMM, Schwartz. arXiv:1612.01551)
  - And more...
- Generation
  - Generative model. (de Oliveira, Paganini, Nachman. arXiv:1701.05927)
- Regression
  - This work.

Our Model

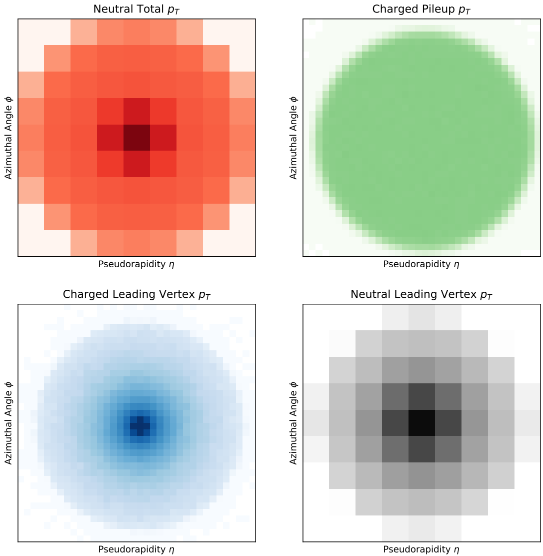
- Inputs: three-channel RGB "pileup image"
  - red $= p_T$ of all neutral particles
  - green $= p_T$ of charged PU particles
  - blue $= p_T$ of charged LV particles
- Output: single-channel neutral image
  - output $= p_T$ of neutral LV particles

Our Study

- Process
  - Leading vertex: 500GeV scalar to dijets with Pythia8
  - $R = 0.4$ anti-k_T jets in $|\eta| < 2$ with $p_T > 100$GeV.
  - Pileup: NPU=140 Poissonian of soft QCD events overlaid.

- Image parameters:
  - Charged jet image pixel resolution: $\Delta\eta \times \Delta\phi = 0.025 \times 0.025$
  - Neutral jet image pixel resolution: $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$
  - Jet image size 0.9×0.9
  - Leading vertex/pileup information for charged particles with $p_T > 500$MeV

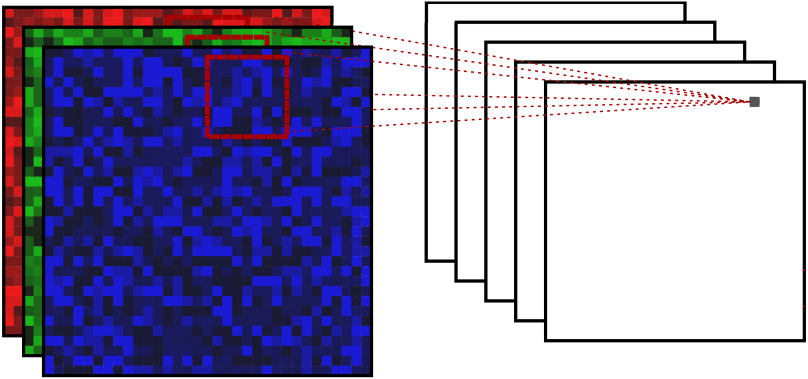
Pileup Images



Architecture

What sort of neural network layers should we use?

- Dense: Units connected to every input pixel with different weights
- Locally connected: Units connected to local input patches with different weights
- Convolutional: Units connected to local input patches with weight sharing

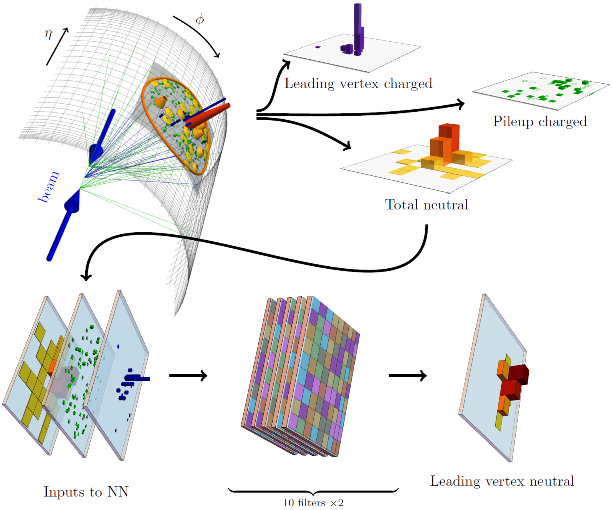


Architecture

- Architecture: Two convolutional layers
  - 6×6 filter sizes
  - 10 filters per layer
  - Only 4711 parameters

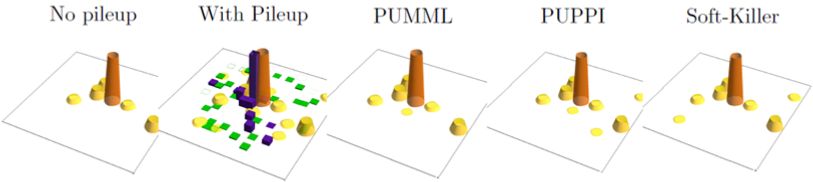
- Architecture is *local*:
  - Pileup removal of a pixel depends only on the information in a window around it
  - Can apply the trained model at the event-level, jet level, or on any specified region

PUMML Framework



Subtracted Jets

An example event with pileup and subtracted with each method.



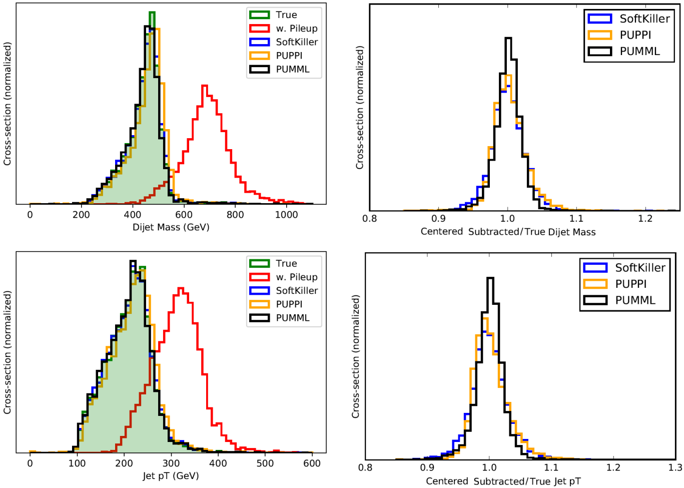
Loss function: Should we treat all p_T errors equally or penalize hard/soft errors more?

$$\ell = \left\langle \log\left(\frac{p_T^{\text{(pred)}} + \bar{p}}{p_T^{\text{(true)}} + \bar{p}}\right)^2 \right\rangle,$$

with $\bar{p} \to 0$ favoring soft pixels and $\bar{p} \to \infty$ favors all p_T equally.

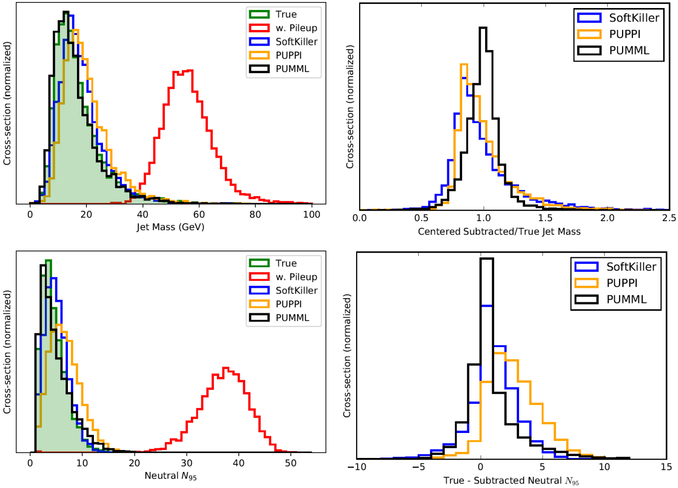
Subtracted Observables

Distributions before and after subtraction of jet p_T and dijet mass



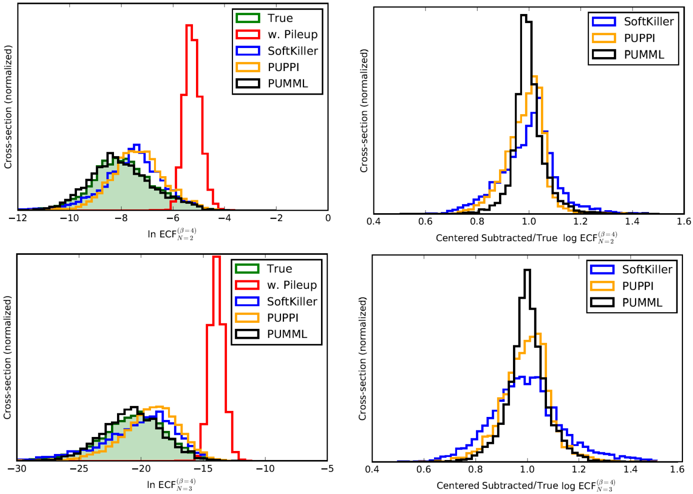
Subtracted Observables

Distributions before and after subtraction of jet mass and N_{95}.

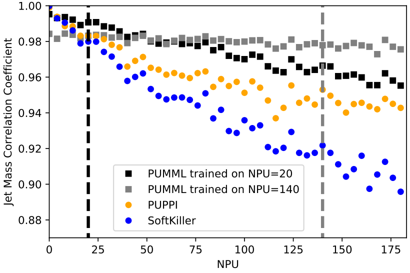


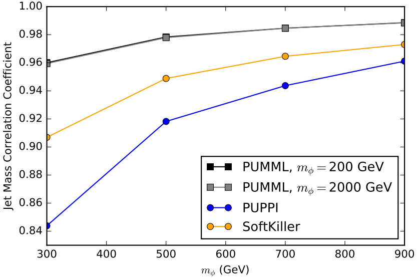
Subtracted Observables

Distributions before and after subtraction of two energy correlation functions.



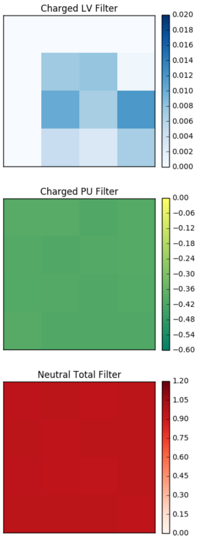
Model Robustness





- Study robustness to pileup by training and testing with different NPU.
- Study robustness to the process by training and testing with different m_ϕ.

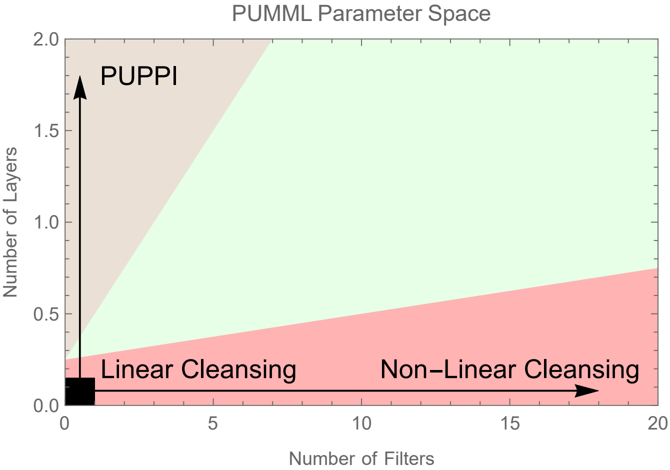
What is being learned?



- Train a single 4×4 filter and inspect it.
- Pixel-wise: $p_T^{N,LV} \approx p_T^{N,tot} - \frac{1}{2} p_T^{C,PU}$
- This is linear cleansing with $\bar{\gamma}_0 = 2/3$!

$$p_T^{N,LV} = p_T^{N,tot} + (1 - \frac{1}{\bar{\gamma}_0}) p_T^{C,PU}$$

What is being learned?



Learning from Data

- Training from simulation risks mis-modelling issues
- Prefer to train on data rather than simulation
  - Data overlay approach using minimum bias and zero-bias events already used by experimental groups in other contexts.
  - Promising for training PUMML directly with data for the relevant application.

Concluding Remarks

- We have developed an ML framework that successfully organizes all of the availabe local information to directly learn to mitigate pileup.
- Can use tools from modern machine learning without going “deep”.
- Pileup mitigation can be a good proving ground for modern machine learning techniques in high energy physics.

The End

Thank You!