Pileup Subtraction and Jet Energy Prediction Using Machine Learning

Vein S Kong SKONG@UMICH.EDU
Stanford University

Jiakun Li JIAKUN@STANFORD.EDU
Stanford University

Yujia Zhang YUJIAZ@STANFORD.EDU
Stanford University

Abstract

In the Large Hardron Collider (LHC), multiple proton-proton collisions cause pileup in reconstructing energy information for a single primary collision (jet). This project aims to select the most important features and create a model to accurately estimate jet energy. Different machine learning methods were explored, including linear regression, support vector regression and decision tree. The best result is obtained by linear regression with predictive features and the performance is improved significantly from the baseline method.

1. Introduction

With the development of the Large Hadron Collider (LHC), we are able to collide particles in a highly contained and powerful environment. This allows us to observe the activity of proton-proton collisions which produces jets, caused by the formation of hadrons from quarks and gluons.

These jets give an indication of how much energy is produced on each collision. In the LHC, each event consists of multiple collisions, which cause "pileup" particle interferences, making it hard to get the precise information from the jets. Currently it's around 20 collisions per event. In the future, there may be up to 200 collisions per event making denoising of the data an important aspect for future LHC research. There currently are some ways to reduce the noise, but the method of mean centering the signal-noise ratio only works well in a few conditions. Is there a novel machine learning approach that can learn and identify irrelevant particles and remove their influence?

The objective of this project is to create a more accurate model to estimate the true jet energy from the pileup noise and identify the most predictive features in pileup subtraction. Machine learning techniques such as linear regression, support vector regression and decision trees algorithms are tested and compared. All the regression models outperform the current area-based correction approach.

2. Related work

Matteo Cacciari and Gavin P. Salam (Matteo Cacciari, 2008) proposed an area-based approach to subtract pileup effects. A jets truth momentum is given by subtracting the product of jet area and pileup momentum per unit area from the measured momentum. However, this approach suffers from inaccuracy by assuming pileup particles are uniformly distributed within each event and the pileup effect is only dependent on the jet area. (Peter Berta, 2014) and (Cacciari et al., 2015) et al has developed algorithms using particle level detail, but the approach is similar to (Matteo Cacciari, 2008), (Daniele Bertolini, 2014).

2.1. Key metric

We will use the mean and the variance of the absolute error as our metric.

$$\mu_{error} = \frac{1}{n}\sum_{1}^{n} |jpt_{pred}^{(i)} - tjpt^{(i)}| \tag{1}$$

$$\sigma_{error}^2 = \frac{1}{n}\sum_{1}^{n} (|jpt_{pred}^{(i)} - \mu_{error}|)^2 \tag{2}$$

where jpt_{pred} is the predicted jet energy and tjpt is the true jet energy.

2.2. Area-based approach

The current area-based method is defined as follows.

$$jpt_{pred} = jptnoarea - \rho * A \tag{3}$$

where jptnoarea denotes observed raw jet energy, ρ denotes average energy density and A denotes area of the jet, which is set at cirlce of 0.5 radius (Cacciari et al., 2008).

The following table compares the offsets before and after the area-based correction. It shows that the area-based method reduces the mean of the offsets closer to 0 but variance is still large

Stats	Before AB	After AB
Mean	31.34	7.66
Variance	209.48	56.23

Table 1. Key metric for area based approach

Figure 1 shows the distribution of the gaps between the estimated jet energy and the true energy before and after the area-based correction. The distribution after area-based method is shifted closer to the true jet energy but the two distributions are very similarly shaped, suggesting that the area-based method just subtracts a constant value.

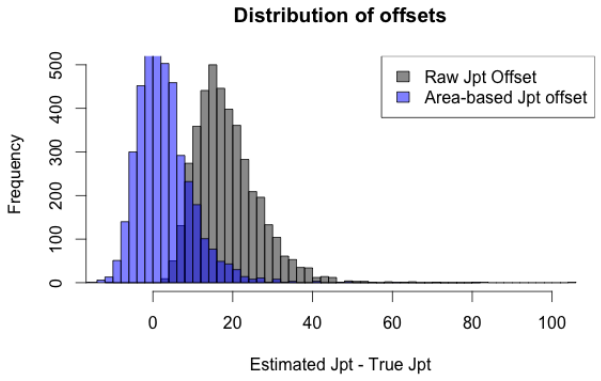

Figure 1. Distribution of offsets

2.3. Relative importance of features

"jptnoarea", which indicates the raw energy observed, is assumed to be the most important feature of our models as we are trying to predict the energy itself. To identify the other most relevant features, we ran a quick test on the offset of corrected jpt vs. one feature.

We can see that only sumtrkPU has a linear relationship with (jpt - tjpt). This means that when there are more charged pileup particles within the jet area, the area-based approach tends to underestimate the jet energy. This finding suggests sumtrkPU should be taken into account to attain better performance in estimating jet energy.

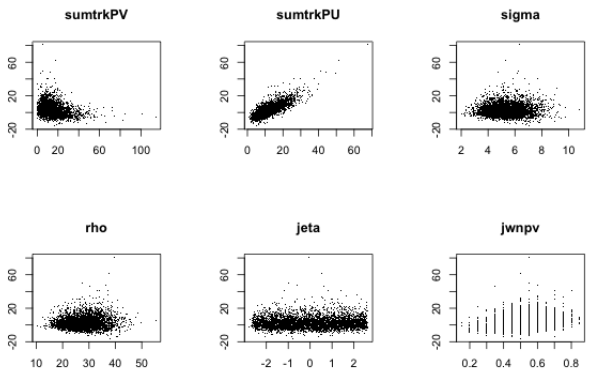

Figure 2. Prediction offset vs.key features

3. Dataset

We worked with the SLAC scientists who provided us with the dataset of 21850 observations. Each has a true jet energy, the number of collisions and 39 features. The sample is produced with the Monte Carlo generator Pythia 8, simulating proton-proton collisions at sqrt(s) equals 14 TeV, where s is the squared sum of the momenta of the colliding particles.

The effect of pileup from multiple collisions in one crossing of the collider beams is simulated by adding particles from additional Pythia 8 collisions. The number of additional pileup interactions per bunch crossing is varied between 20 and 50. The FastJet (v3.1.3) software is then used to build anti-kt 62 jets with R =0.4. Two different set of jets are reconstructed. The first set only uses particles from the hard scatter event, and defines the hard-scatter (or truth) jets in the event. The second set is constructed using all particles in the event, including both hard scatter and pileup. The jet area pileup subtraction correction is applied. This is the collection of reconstructed jets. Reconstructed jets are then associated with truth jets based on a momentum fraction criterion.

The pileup noise level is dictated by the number of proton-proton collisions (NPV). In physical studies the actual number of collisions is an estimated value obtained by counting the number of vertices reconstructed from charged particle tracks; thus using NPV as an input parameter can be an additional source of error in real application. Our training set was built out of 80% randomly sampled data for each number of collisions (NPV) and the rest was used as the testing set. Since our model is trained across

different NPVs, it should be able to operate without the knowledge of the number of collisions.

3.1. Definition of key features

jpt = Jet transverse momentum

jptnoarea = The raw observed jet transverse momentum including pileup energy contamination

sumtrkPV: Total pt of charged particles in the jet from the hard-scatter (signal vertex)

sumtrkPU: Total pt of charged particles in the jet from additional pileup vertices

rho: Event pt density

jeta: Jet psedorapidity, spatial coordinate describing the angle of a particle relative to the beam axis

sigma: RMS of rho (measure of how large the point-to-point fluctuations of pileup are in the event)

area: Jet area $\pi * R^2$, with R=0.4 (jets are roughly circles of radius R=0.4)

ptX: jpt computed in circles of radius X/100, corrected by rho * area

4. Training Models

We tried three regression models **linear regression**, **support vector regression** and **CART trees** and compared their performance.

4.1. Linear Regression

The linear regression assumes the true jet energy $tjpt$ to be a linear combination of the input features x_i with an intercept term.

$$tjpt_{=} \sum_{i=j}^{N} x_j \theta_j + \theta_0 \tag{4}$$

It learns the coefficients of different features by minimizing the sum of the square error across the training set. Out of the total 39 features, 9 features are collision properties and the other 30 features describe the jet energy distribution within the circles of different sizes. We did an exhaustive search on all features combinations for the first 9 features and applied forward searching on the remaining ones. The best feature combination for linear regression is given by jptnoarea, sumtrkPU, sumtrkPV and pt20.

Based on the feature selection result obtained from simple linear regression, we ran a locally-weighted regression model which assigns higher weights the training data points that are closer to the query point.

4.1.1. Regularized Linear Regression

We also worked on getting results for linear regression with parameterization using RIDGE and LASSO techniques. LASSO uses the L1 norm of the weight vector as the regularization term. The LASSO objective function is

$$\min_{\theta} \frac{1}{2m} \sum_{1}^{m} (y^{(i)} - x^{(i)}\theta) + \lambda \sum_{j=1}^{n} |\theta_j| \tag{5}$$

$$\text{subject to } \textstyle\sum_{j=1}^{n} |\theta_j| < C$$

The RIDGE is very similar to LASSO except it uses the L2 norm.

$$\min_{\theta} \frac{1}{2m} \sum_{1}^{m} (y^{(i)} - x^{(i)}\theta)^2 + \lambda \sum_{j=1}^{n} \theta_j^2 \tag{6}$$

$$\text{subject to } \textstyle\sum_{j=1}^{n} |\theta_j|^2 < C$$

The result of regularized linear regression is the same as the basic regression. The λ parameter for regularization was essentially 0. Since regularization penalizes overcomplicated models, it reconfirms that our model is not overfitting.

4.2. Support Vector Regression

Support Vector Regression (SVR) applies the kernel method to train the model in a high-dimensional feature space without incurring additional computational cost. We attempted support vector regression with Gaussian kernel and linear kernel to explore the influence of high-order features and the interaction between features.

4.3. CART tree

The regression tree is built out of the CART algorithm. In each step, the tree splits the dataset into two subsets and it always finds the best variable and the best position to split on that variable which minimizes the mean square error across the training set. Since the CART algorithm is always choosing the best split, it is resistant to irrelevant features. So we trained the decision tree model on all the features.

5. Results

Figure 3 summarizes the results for different models by plotting the two metric values in x and y axes. The top right dot corresponds to the baseline performance, which is the area-based approach. The blue cluster in the middle denotes the results obtained by linear regression with

jptnoarea and rho*area. In this case, linear regression outperforms the baseline method. This is because instead of assuming uniformly distributed pileup, linear regression learns about the relationship between average pileup density and total jet energy while constructing the model with training data. It takes that information and stores them in the coefficients as shown in Table 2. Also, note that jeta and sigma*sqrt(area) do not improve the linear regression performance, meaning detailed information about jet is not significant in predicting jet energy.

(Intercept)	jptnoarea	rho*area
6.5	0.75	1.02

Table 2. Area based parameters

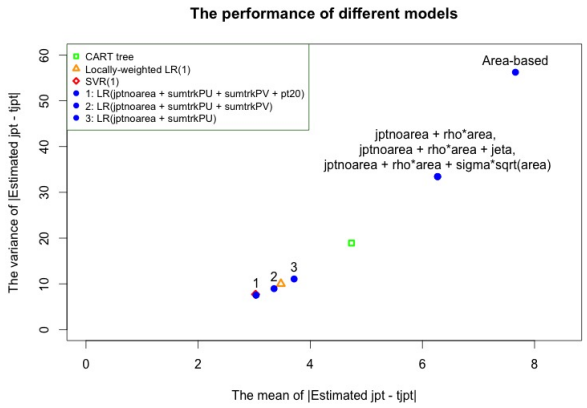

Figure 3. Performance Comparison for different models

The blue dot numbered 3 in the bottom left represents linear regression with jptnoarea and sumtrkPU, which has significant better performance than using rho*area. This is because sumtrkPU is the energy of charged particles in the pileup and the ratio between charged and neutral particles in an event is roughly 2:1. Therefore sumtrkPU helps predict the amount of pileup observed and has more relevance compared to rho*area. Taking sumtrkPV into account further improved the performance (blue dot 2). sumtrkPV denotes the amount of charged particles in the jet, thus gives us more information about jet energy.

The overall best performance is given by linear regression with jptnoarea, sumtrkPU, sumtrkPV and pt20 (blue dot 1). The coefficients for the four features are listed in Table 3.

(Intercept)	jptnoarea	sumtrkPU	sumtrkPV	pt20
1.7846111	0.5011870	-0.7851541	0.2814567	0.2785845

Table 3. Coefficents for the optimal linear regression model

Intuitively this makes sense, the estimate is based on jptnoarea and then subtract out the pileup from other jets by sumtrkPU and to compensate for the error from oversubtraction with sumtrkPV and pt20. ptX indicates the energy in certain radius and the available values range from 5 to 50. pt20 is the most important one because the medium radius balances between jet energy and pileup addition.

Locally weighted linear regression was also applied with the best feature set (yellow triangle). It does not outperform regular linear regression. Moreover, the training error decreased whereas the testing error increased, which indicates possible overfitting.

Support vector regression was also explored with Gaussian and linear kernels. The best performance is obtained with linear kernel and also coincides with linear regression result using feature set 1. This indicates that the relationship between features and jpt prediction is mainly linear. High order features and feature interactions do not play a significant role in jpt prediction.

The result for CART tree is shown by the green rectangle. It performs better than area-based approach and linear regression with jptnoarea and rho*area. Figure 4 ranks the top 10 most important features in building the tree. The feature of the highest importance is sumtrkPV and all the features that follow are the jet energy within different subareas. The decision tree uses the energy of charged particles in the jet area as the basis and reconstruct the true jet energy with the information within different radii. Unlike linear regression which discovers and utilizes the linear relationship between specific features and the jet energy, the decision tree performs prediction by fitting a piece-wise function of subarea information without the knowledge of the observed jet energy.

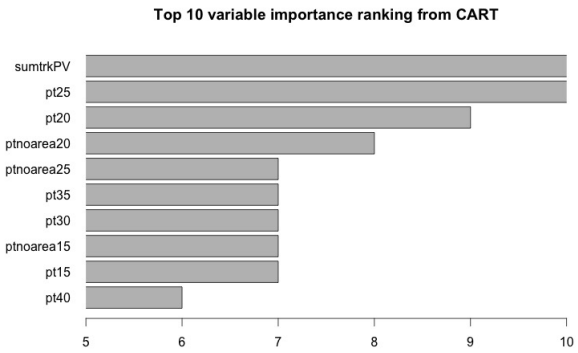

Figure 4. Top 10 variable importance ranking from Cart tree

To summarize the overall achieved performance, Figure 5 shows a comparison between the best obtained model and the baseline method (area-based approach). The red bar is a histogram plot of the prediction error based on the best

model. The mean and variance is summarized in Table 4. The trend curve for both distributions are constructed assuming normal distribution with the measured mean and variance. It is observed that compared to baseline, our model improved both mean and variance of the prediction.

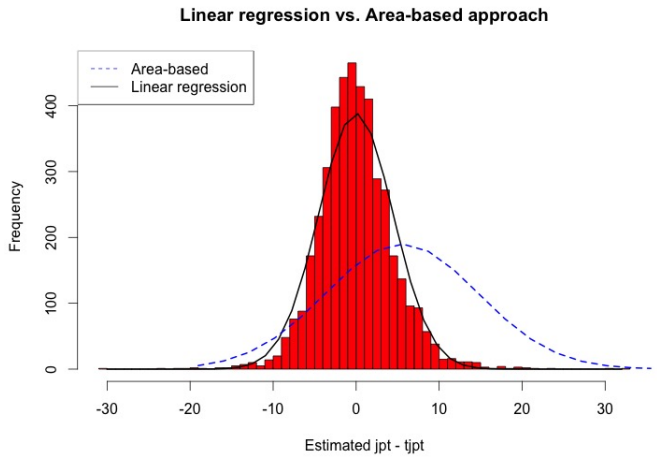

Figure 5. Linear Regression vs Area Based

Stats	Area-based	Linear Regression
Mean	7.66	3.03
Variance	56.23	7.51

Table 4. Key metrics comparison for area-based approach and optimal linear regression model

6. Conclusion and Future Work

We worked with simulation data and identified relevant features for jet energy prediction. Linear regression, support vector regression and decision tree algorithms were applied and linear regression produced the best result based on both mean and variance metrics.

Our work shows that there could be large improvements made to the current area based approach. Linear regression with the four features listed in Table 3 seems to give us great results. It seems like using machine learning techniques in pileup subtraction has great potential.

Future work would be to include running linear regression with a different dataset. The dataset will be generated where the jets are produced in association with a Higgs Boson so that we can test the improvement in a scenario where the precise estimation of the jet energy is crucial. We can also test our dataset on low energy jets to see how our model works with data with the values $20 < tjpt < 40$.

Acknowledgments

We would like to thank Professor Ariel Schwartzman and Dr. Francesco Rubbo for the guidance and support throughout the project.

References

Cacciari, Matteo, Rojo, Juan, Salam, Gavin P, and Soyez, Gregory. Quantifying the performance of jet definitions for kinematic reconstruction at the lhc. *Journal of High Energy Physics*, 2008(12):032, 2008.

Cacciari, Matteo, Salam, Gavin P, and Soyez, Gregory. Softkiller, a particle-level pileup removal method. *The European Physical Journal C*, 75(2):1–16, 2015.

Daniele Bertolini, Philip Harris, Matthew Low Nhan Tran. Pileup per particle identification. *JHEP 1410 (2014) 59*, 2014.

Matteo Cacciari, Gavin P. Salam. Pileup subtraction using jet areas. *Phys.Lett.B659*, pp. 119–126, 2008.

Peter Berta, Martin Spousta, David W. Miller Rupert Leitner. Particle-level pileup subtraction for jets and jet shapes. *JHEP 1406 (2014) 092*, 2014.