通过卷积神经网络方法对Durham喷注重建算法的改进

指导老师：白羽
史静远

简介

Durham算法：

Cluster间距离：

$$y_{ij} \equiv \frac{2\min\{E_i^2, E_j^2\}}{Q^2}(1-\cos\theta_{ij}),$$

重复合并最小距离的Cluster，直到得到指定的Cluster数目或是所有Cluster之间距离都大于所设的阈值

目前分析中没有指定距离阈值，而是根据研究的信号直接指定需要重建jet的多重数：

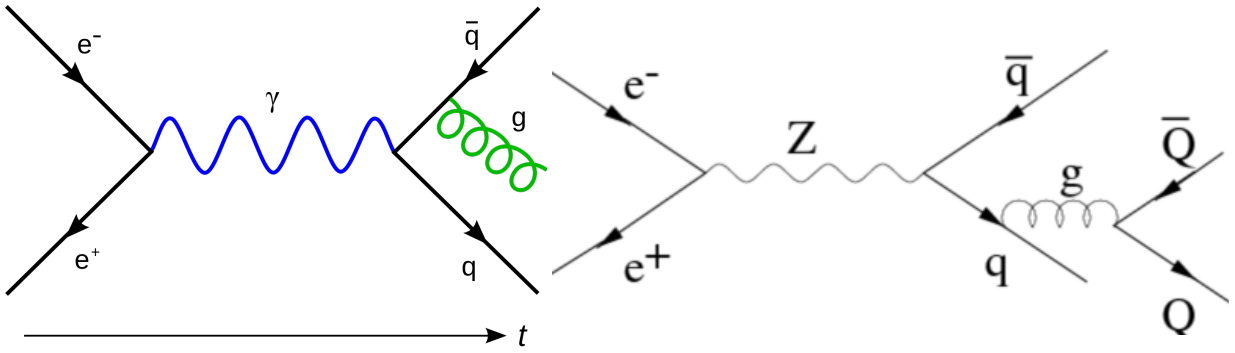
- 较好的质量分辨率
- 对jet多重数判断较差

目标：判断jet多重数。

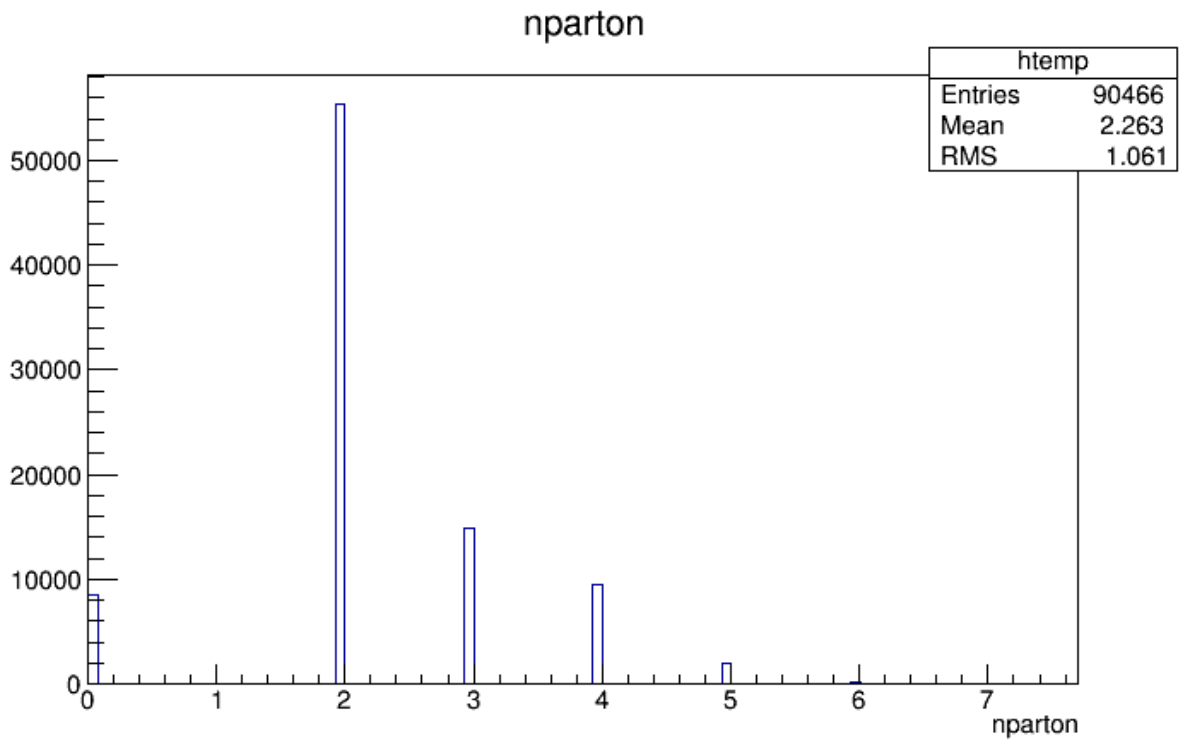
应用：使jet重建更加准确，Higgs强衰变过程的研究，QCD过程的研究等。

得到Jet数目

考虑到胶子劈裂和胶子辐射的情况，
从MC Truth信息中提取出部分子的数目



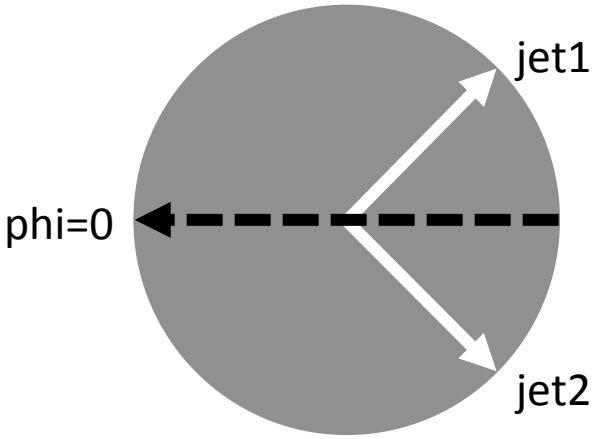
胶子辐射　　胶子劈裂




```
if(p_gluon.DeltaR(p_gluon_brothers) > 0.7 && p_gluon.Pt()>15.0 && (p_gluon.E()/p_gluon_mother.E())>0.2)

                                    if(p_quark1.E()/(p_quark1.E()+p_quark2.E())>0.25 && p_quark2.E()/(p_quark1.E()+p_quark2.E())>0.25 &&
( (p_quark1.DeltaR(p_quark2)>0.7 && p_quark1.E()>10.0 && p_quark2.E()>10.0)||(p_quark1.DeltaR(p_quark2)>0.5 && p_quark1.E()>30.0 && p_quark2.E(
```

手动设定参数得到jet数

预处理



对phi的预处理

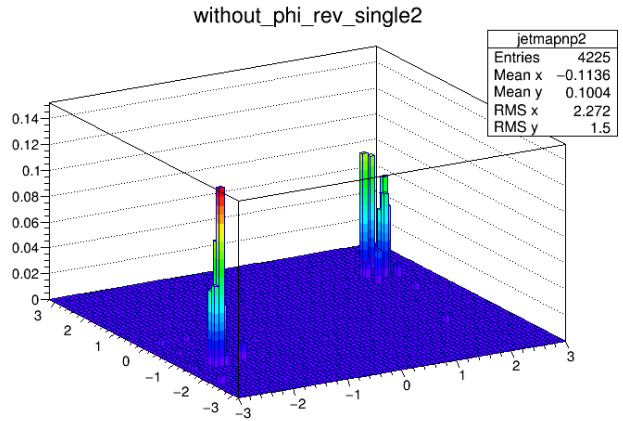
```
float delta_phi = phi1 - phi2;
float phi=0;
if(delta_phi>TMath::Pi() || delta_phi<-TMath::Pi()) phi = (phi1+phi2)/2.0;
else phi = (phi1+phi2)/2.0 + TMath::Pi();

double eta = (p_jet1.Eta()+p_jet2.Eta())/2.0;
```

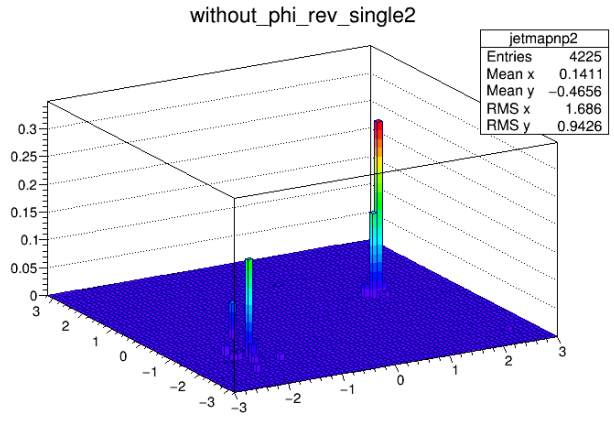
```
double deta = pfo_jet_C.at(i).Eta();
double dphi = pfo_jet_C.at(i).Phi();
double detanew = deta -eta;
double dphinew = dphi-phi;
if(dphinew>TMath::Pi()) dphinew = -2*TMath::Pi()+dphinew;//
if(dphinew<-TMath::Pi()) dphinew = dphinew +2*TMath::Pi();
```

具体代码实现

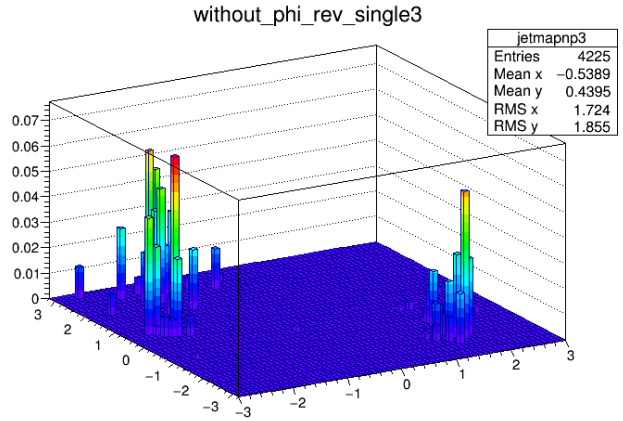
对eta的预处理则是各object减去eta的平均值，
使得两jet的eta异号

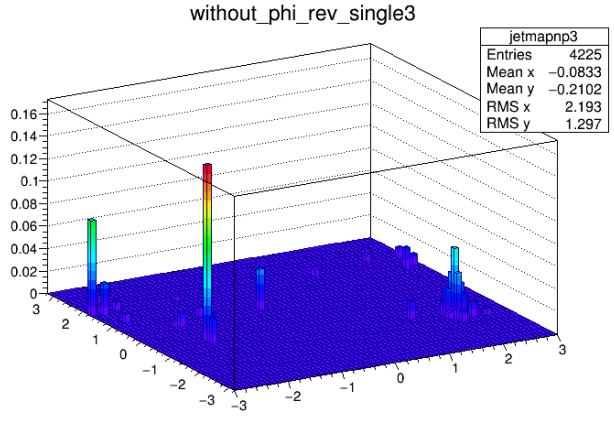
喷注图像



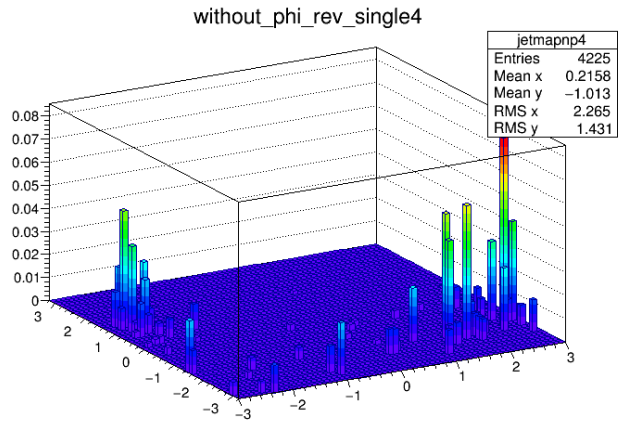


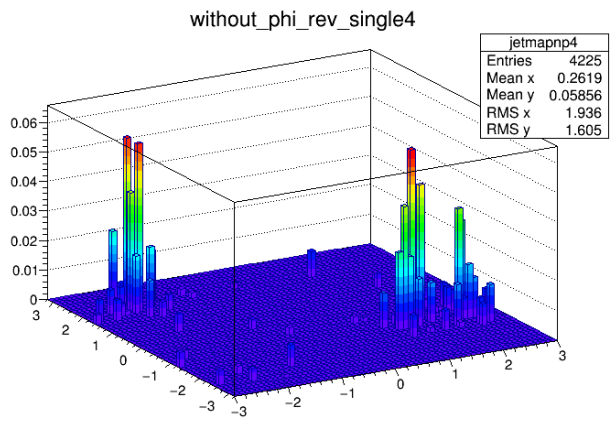
2 jets





3 jets



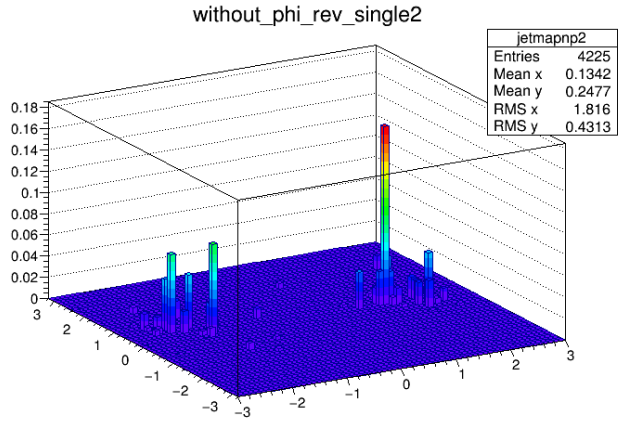


4 jets

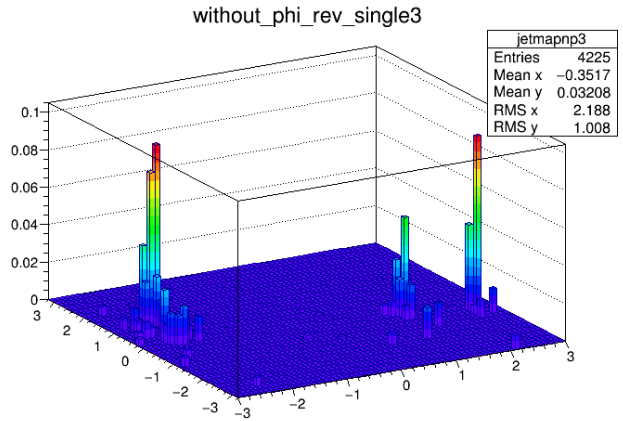
H->bb

H->cc

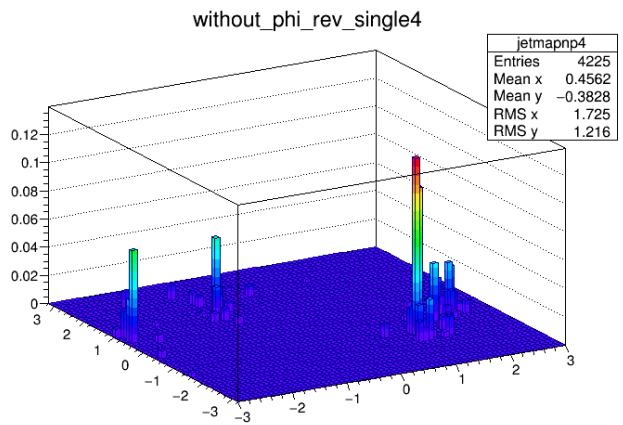
喷注图像



2 jets



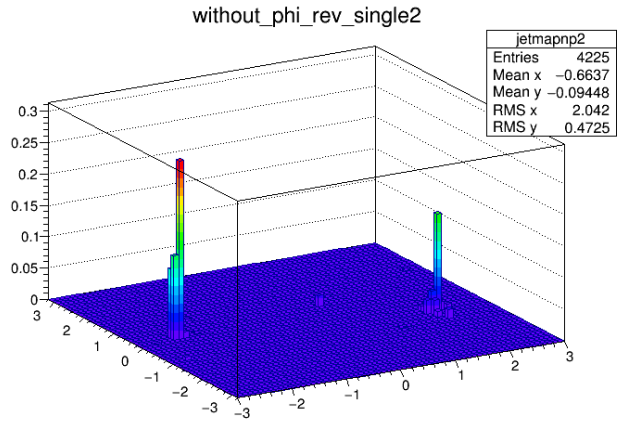
3 jets

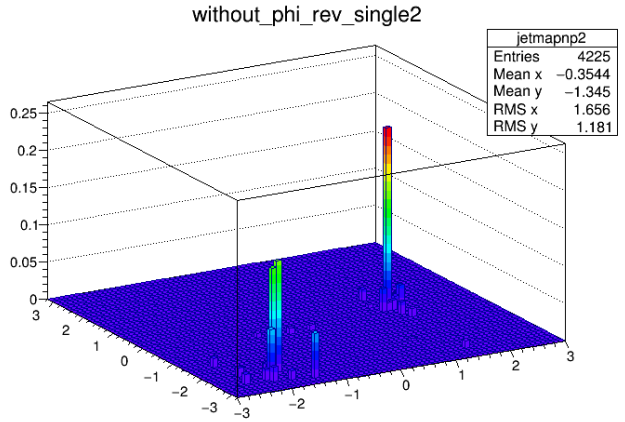


4 jets

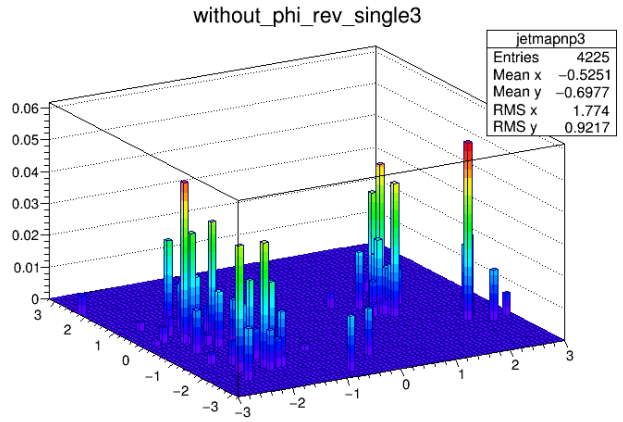
H->gg

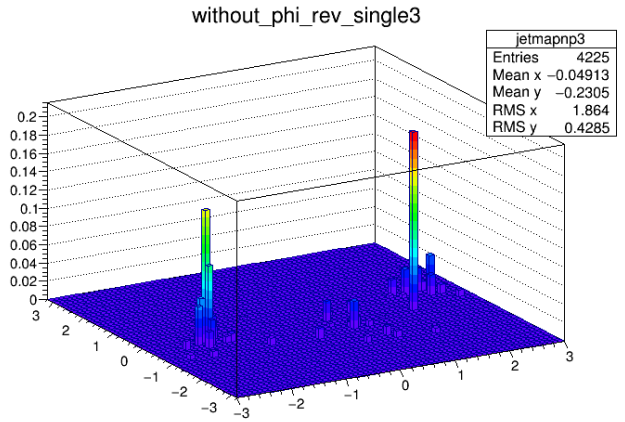
喷注图像



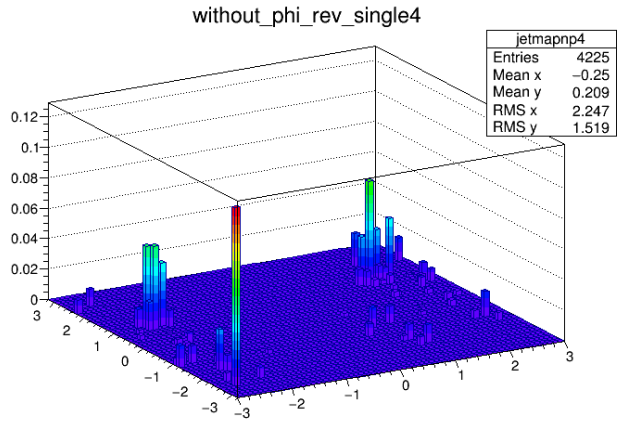


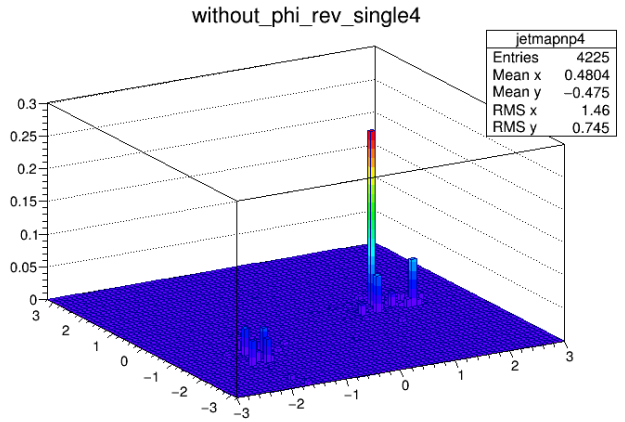
2 jets





3 jets





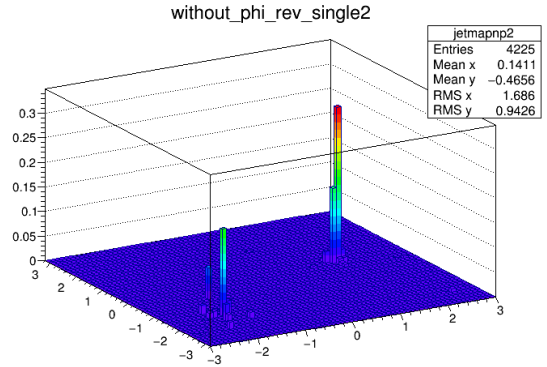
4 jets

H->ZZ

H->WW

卷积神经网络（CNNs）

喷注图像83000个

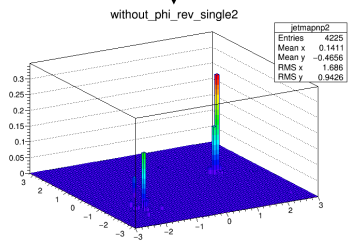


65*65

预处理

卷积核数：64
核尺寸：10*10

池化尺寸：4*4

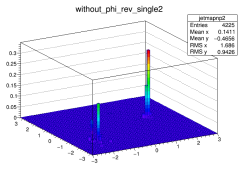


14*14

卷积核数：64
核尺寸：5*5

池化尺寸：2*2

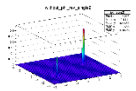
5*5



卷积核数：64
核尺寸：4*4

池化尺寸：2*2

1*1



Hidden layer

128 units
激活函数：tanh

Layer 3

Logistic Regression

0 1 2 3 4+

结果分析

```
Validation minibatch losses[10] = 0.162000
Validation minibatch losses[11] = 0.210000
Validation minibatch losses[12] = 0.178000
epoch 200, minibatch 140/140, validation error 19.923078 %
     epoch 200, minibatch 140/140, test error of best model 21.169232 %
Optimization complete.
Best validation score of 19.923078 % obtained at iteration 28000, with test perform
ance 21.169232 %
The code for file convolutional_mlpNoJet_withcut_NumJet_2.py ran for 52.28m
```

目前正确率在80%左右

- 下一步计划能够得到各个喷注数的错误率以及神经网络误判的类型
- 更加合理的parton数作为label

谢谢

thomas3030@foxmail.com