Jets in NanoAODplus

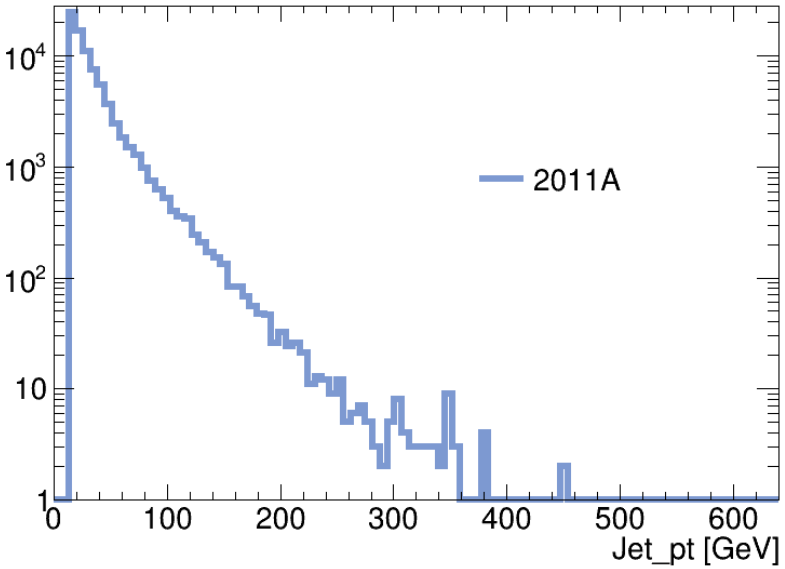
Armando Bermudez Martinez
06.05.2020

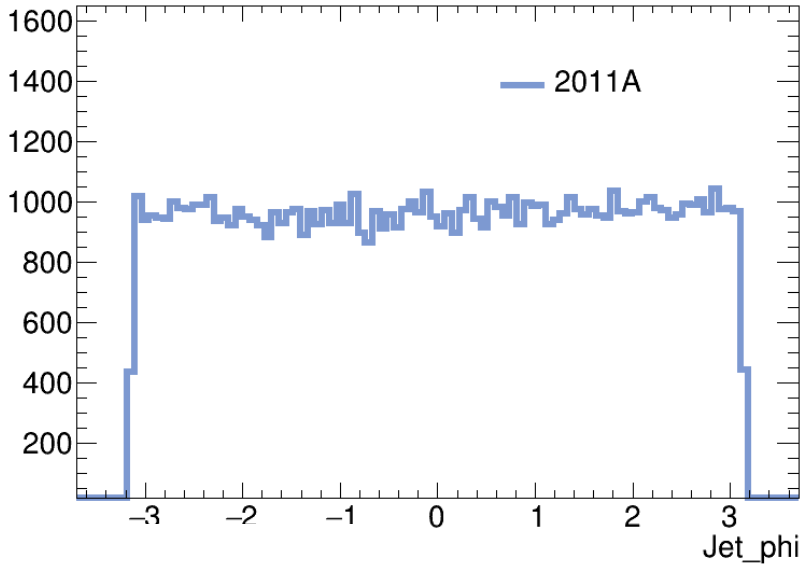
What has been implemented

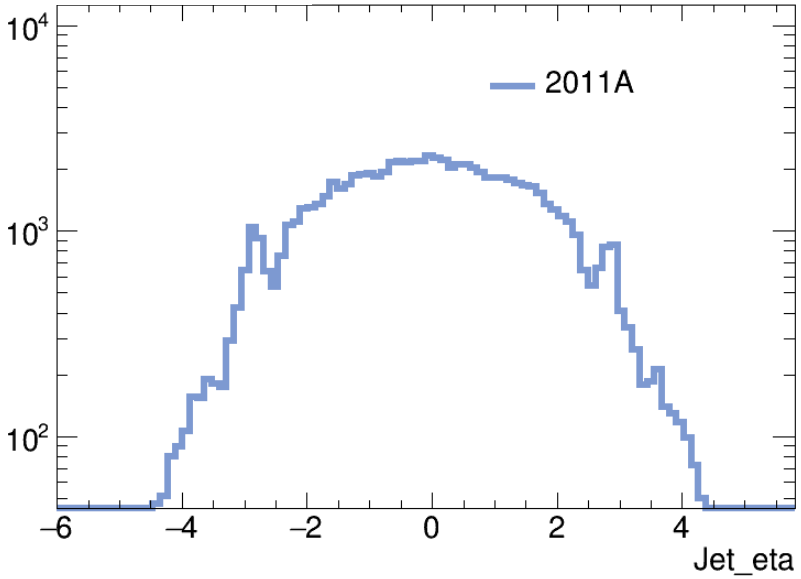
Jet [back to top]

Object property	Type	Description
Jet_area	Float_t	jet catchment area, for JECs
Jet_bRegCorr	Float_t	pt correction for b-jet energy regression
Jet_bRegRes	Float_t	res on pt corrected with b-jet regression
Jet_btagCMVA	Float_t	CMVA V2 btag discriminator
Jet_btagCSVV2	Float_t	pfCombinedInclusiveSecondaryVertexV2 b-tag discriminator (aka CSVV2)
Jet_btagDeepB	Float_t	DeepCSV b+bb tag discriminator
Jet_btagDeepC	Float_t	DeepCSV charm btag discriminator
Jet_btagDeepFlavB	Float_t	DeepFlavour b+bb+lepb tag discriminator
Jet_chEmEF	Float_t	charged Electromagnetic Energy Fraction
Jet_chHEF	Float_t	charged Hadron Energy Fraction
Jet_cleanmask	UChar_t	simple cleaning mask with priority to leptons
Jet_electronIdx1	Int_t(index to Electron)	index of first matching electron
Jet_electronIdx2	Int_t(index to Electron)	index of second matching electron
Jet_eta	Float_t	eta
Jet_genJetIdx	Int_t(index to Genjet)	index of matched gen jet
Jet_hadronFlavour	Int_t	flavour from hadron ghost clustering
Jet_jetId	Int_t	Jet ID flags bit1 is loose (always false in 2017 since it does not exist), bit2 is tight, bit3 is tightLepVeto
Jet_mass	Float_t	mass
Jet_muEF	Float_t	muon Energy Fraction
Jet_muonIdx1	Int_t(index to Muon)	index of first matching muon
Jet_muonIdx2	Int_t(index to Muon)	index of second matching muon
Jet_nConstituents	Int_t	Number of particles in the jet
Jet_nElectrons	Int_t	number of electrons in the jet
Jet_nMuons	Int_t	number of muons in the jet
Jet_neEmEF	Float_t	neutral Electromagnetic Energy Fraction
Jet_neHEF	Float_t	neutral Hadron Energy Fraction
Jet_partonFlavour	Int_t	flavour from parton matching
Jet_phi	Float_t	phi
Jet_pt	Float_t	pt
Jet_puId	Int_t	Pilup ID flags
Jet_qgl	Float_t	Quark vs Gluon likelihood discriminator
Jet_rawFactor	Float_t	1 - Factor to get back to raw pT
nJet	UInt_t	slimmedJets, i.e. ak4 PFJets CHS with JECs applied, after basic selection (pt > 15)

Jet distributions examples



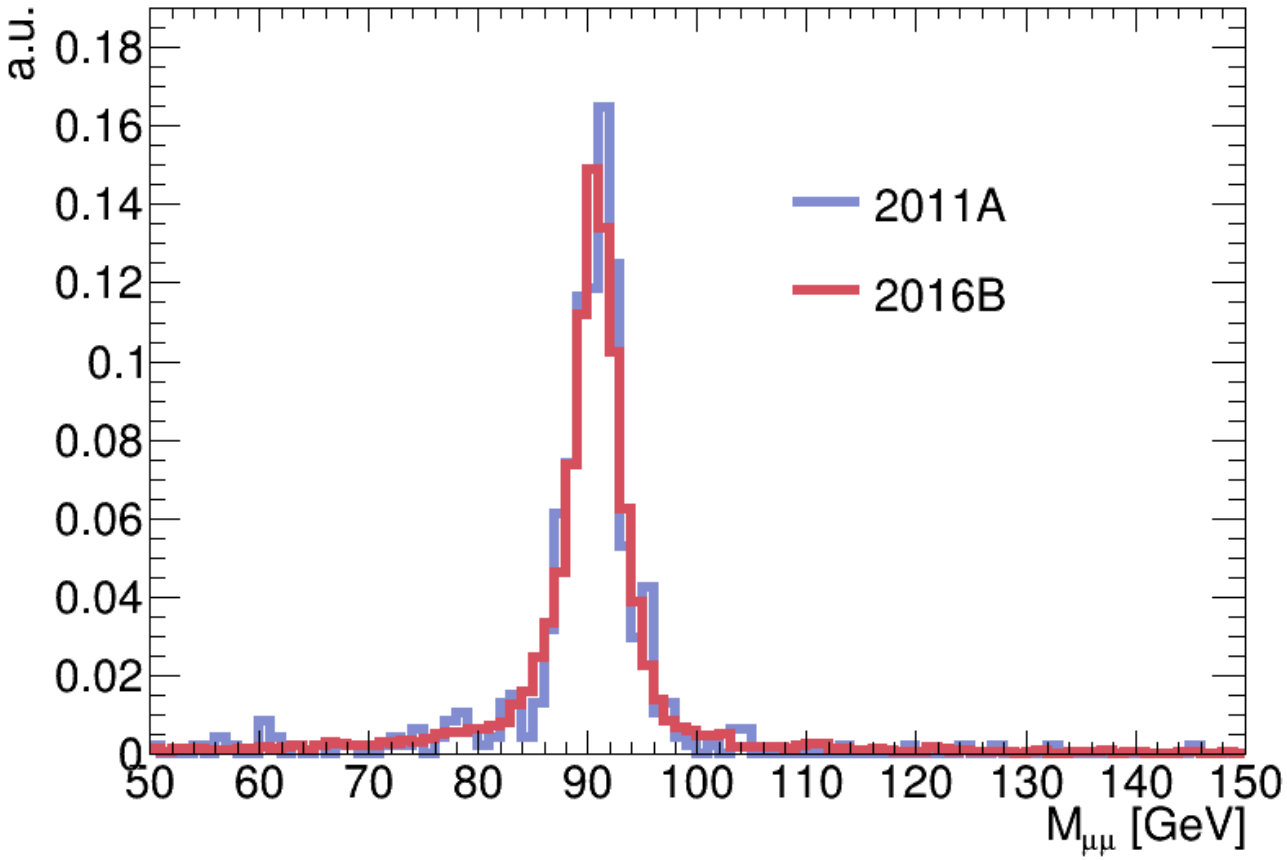




Data samples:
root://eospublic.cern.ch//eos/opendata/cms/Run2011A/SingleMu/AOD/12Oct2013-v1/10000/00209631-9C37-E311-BACA-002590494FDE.root

Steps towards validation

NanoAODplus 2011 SingleMu data sample has been successfully used in our Z+jets framework



Data samples:
root://eospublic.cern.ch//eos/opendata/cms/Run2011A/SingleMu/AOD/12Oct2013-v1/10000/00209631-9C37-E311-BACA-002590494FDE.root
root://eoscms.cern.ch//eos/cms/store/data/Run2016B_ver2/SingleMuon/NANOAOD/Nano25Oct2019_ver2-v1/40000/201D0DBF-CB50-314E-927E-8551DBFB5F64.root

Few technical details

- Look for your favorite data set (**http://opendata.cern.ch/**)
- Run NanoAODplus framework
- Use Tselector (**https://root.cern.ch/developing-tselector**) on the output to generate a skeleton for your analyzer:

ROOT provides the TTree::MakeSelector function to generate a skeleton class for a given TTree.

For example, if the file "treefile.root" contains the tree "T", a skeleton called "MySelector" can be generated in following way:

```
root [0] TFile *f = TFile::Open("treefile.root")
root [1] TTree *t = (TTree *) f->Get("T")
root [2] t->MakeSelector("MySelector")
root [3] .!ls MySelector*
MySelector.C  MySelector.h
root [4]
```

This skeleton class is a good a starting point for the analysis class. It is recommended that users follow this method.

- Run your analyzer over the nanoAODplus samples
- Compare your results to already published data if possible

Validation

Data set: Jets 2012 (8 TeV)

```
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/000F86EB-C390-E211-A01C-003048679168.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/001E25FA-4C91-E211-9E98-0025905964A6.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/00254F5A-D590-E211-81AF-00259059649C.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/002F0E57-2391-E211-9626-00261894397F.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/003290CC-DD90-E211-9E7A-003048678BAC.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/00832414-C590-E211-B0E9-00304867C04E.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/009012A3-D690-E211-8A68-003048679266.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/00B9DCA9-EB90-E211-9EF9-00304867BED8.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/00C85053-A190-E211-8B1C-0026189438E1.root
root://eospublic.cern.ch//eos/opendata/cms/Run2012C/JetMon/AOD/22Jan2013-v1/10000/0206D65B-F190-E211-AB50-003048679284.root
```

Jason file: Cert_190456-208686_8TeV_22Jan2013ReReco_Collisions12_JSON.txt

Event selection:

- Consider all jets in the event with a minimum p_{T} of 100 GeV and $|y| < 5$.
- Select events with at least two jets.
- The two leading jets, which define $\Delta\phi_{dijet}$, should have $|y| < 2.5$ with the p_{T} of the leading one to be greater than 200 GeV.
- Select events with $\frac{\not{E}_T}{\sum E_T} < 0.1$.

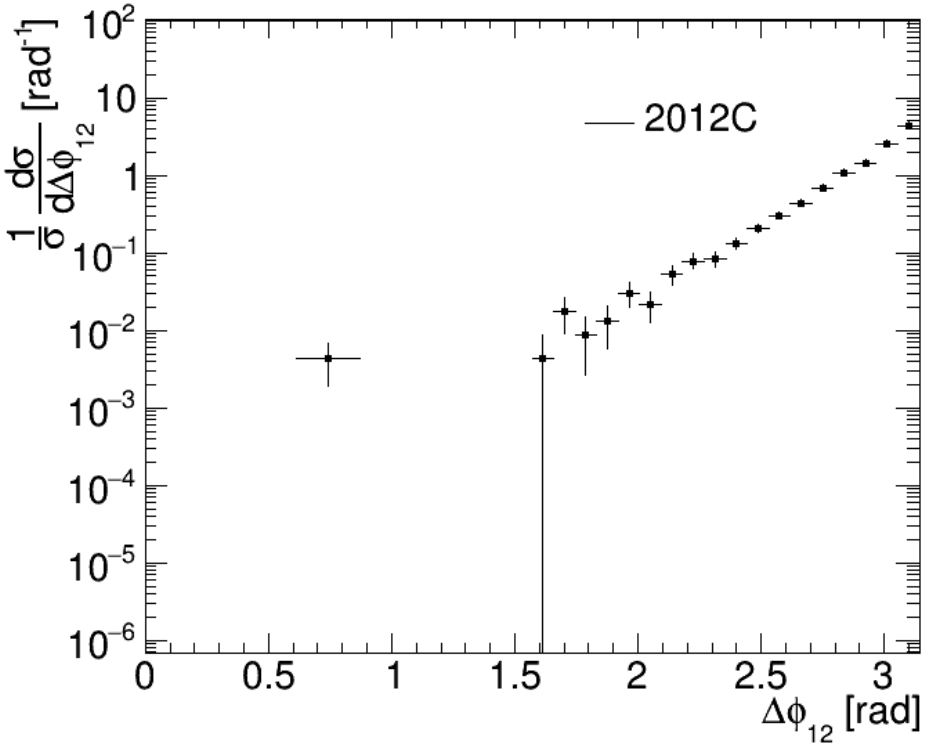
Validation (Dijet azimuthal separation at 8 TeV)

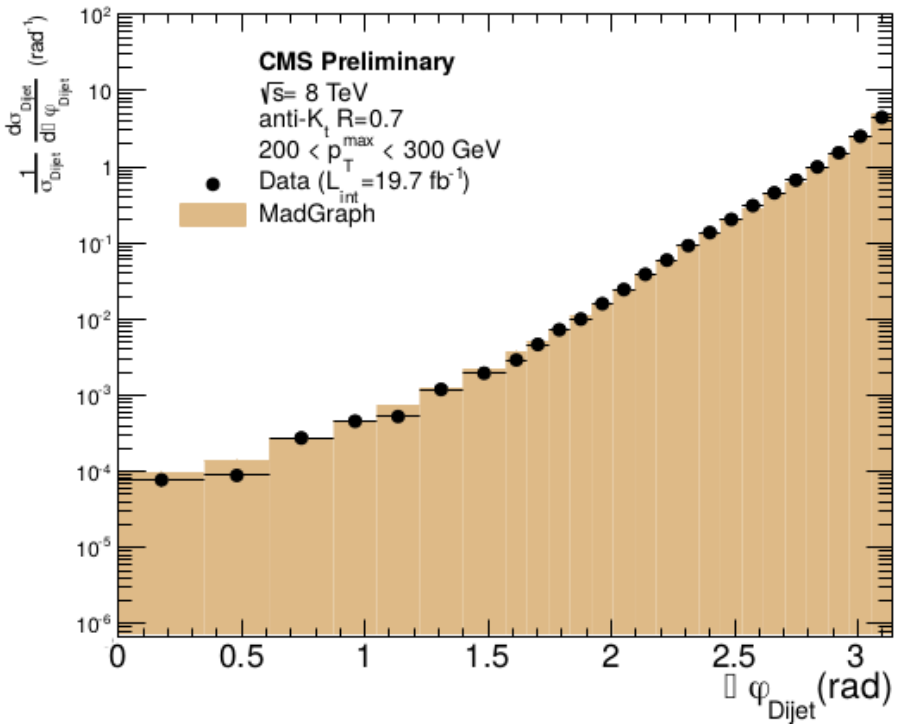
Trigger: HLT_PFJet140

200 < pT < 300 GeV

AN-2014/119

10.1140/epjc/s10052-016-4346-8





Looks promising by eye