1. $pp \to t\bar{t}$: Generate 10000 events of top pair production at the 14 TeV LHC using `Pythia8`, with ISR and FSR on, and do the following:

   (a) Plot the lepton multiplicity. Also plot the p_T, η, ϕ of the first two leptons. Explain the behaviour you see.

   (b) Cluster the jets using `anti-kt` algorithm, with jet radius $R = 0.5$. Plot the jet multiplicity, and p_T, η, ϕ of the first four jets. Also plot the angular separation ΔR in the $\eta - \phi$ plane between the first two jets. Explain what you see.

   (c) Reconstruct the dijet invariant mass. Also reconstruct the three-jet invariant mass. Explain both the plots.

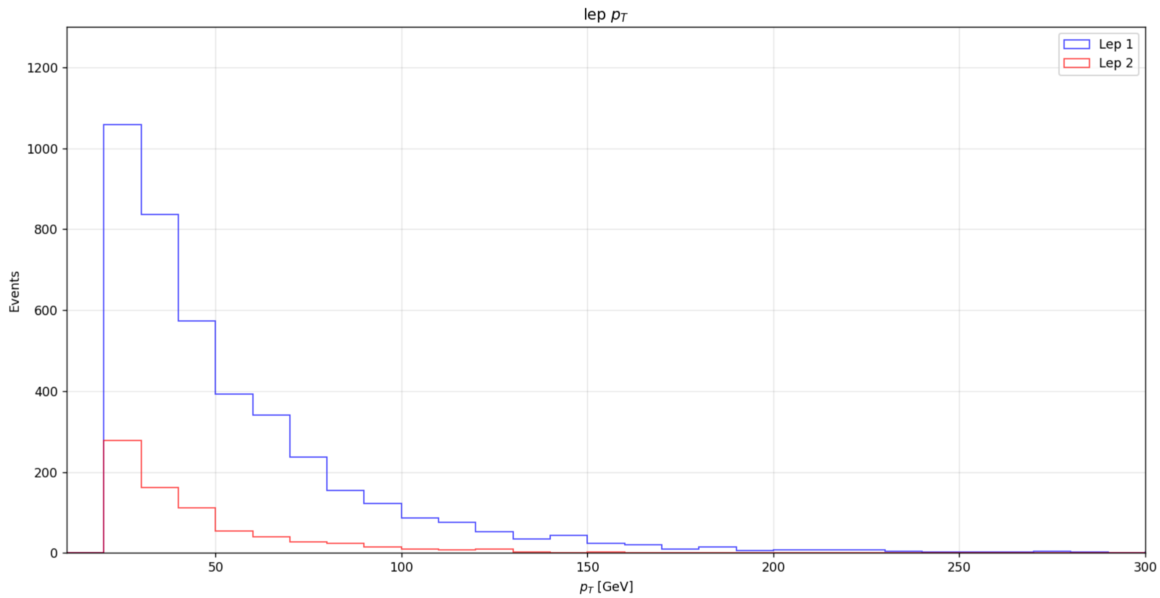
   (d) Plot the jet multiplicities with $R = 0.2, 0.5, 1.0$ in the same plot and compare.

   (e) Plot the missing transverse momentum p_T^{miss}: (i) without any forced decay, and (ii) forcing the $W^\pm$ bosons to decay leptonically. Compare the plots and explain.

   (f) Keeping the forced leptonic decay of $W^\pm$ bosons, tag the neutrinos and vector-sum their p_T. Plot this sum alongside the p_T^{miss} plot you got from the forced decay. What do you see? explain. (As mentioned in the tutorial, neutrinos cannot be tagged at the detector, but this is a good check for understanding p_T^{miss}.)

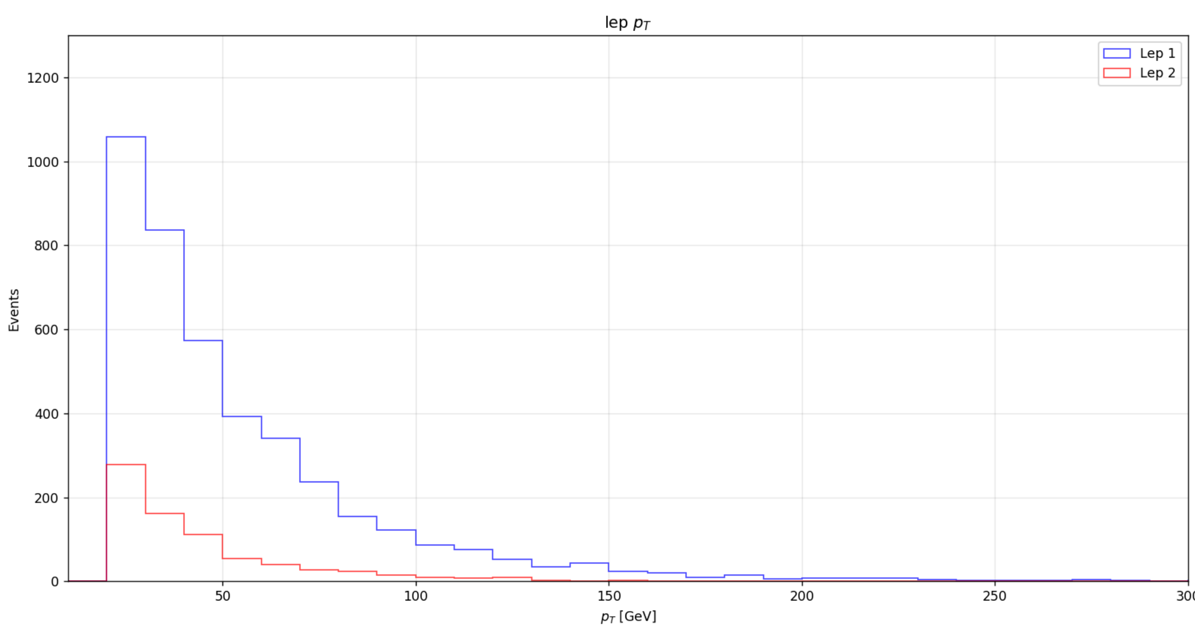
a) Lepton pT plot

- R = 0.4



The plot begins at 20 GeV due to the minimum pT cut of 20 GeV. The peak in the pT is seen in the range 40-45 GeV. Since top quark decays to W and b and further W decays leptonically, the pT peak is around half the mass of W i.e 40 GeV.

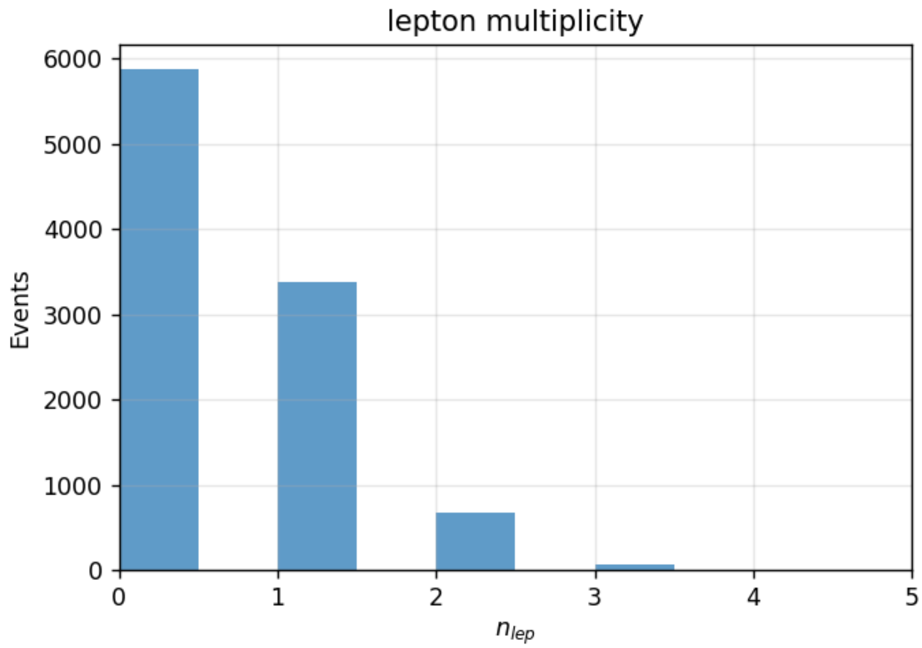
- R = 0.5



Similar plot and explanation applies to this case also.

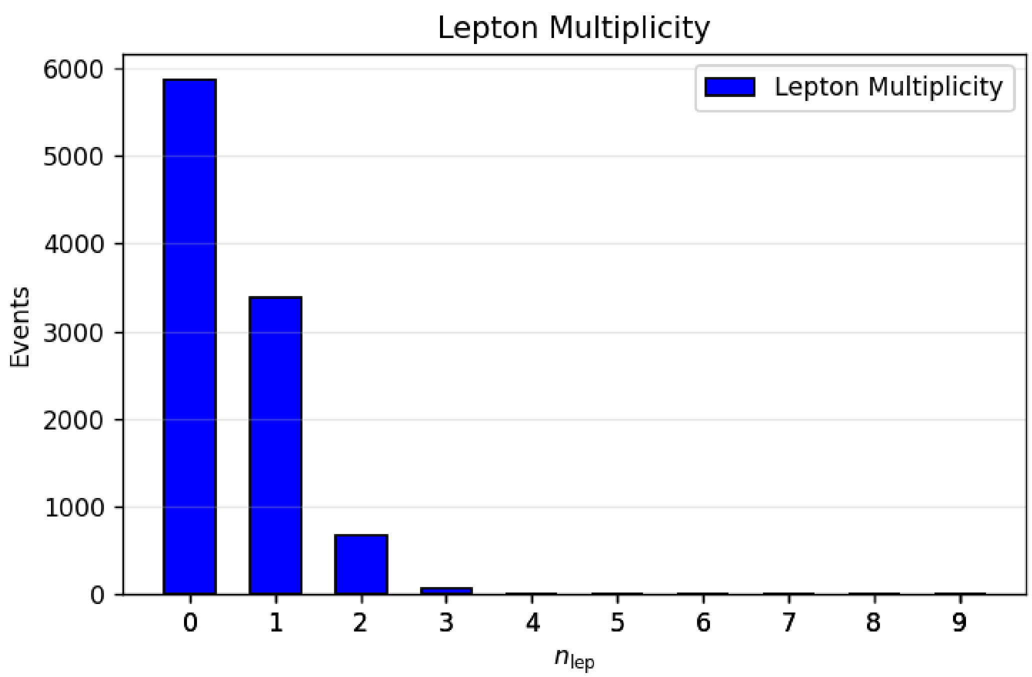
b) Lepton multiplicity

- R = 0.4



For t tbar production, hadronic decays dominate followed by semi-leptonic and leptonic decays. The same can be seen from the lepton multiplicity plots. When W decays leptonically, we can get two leptons from W+ and W- respectively. That explains the lepton multiplicity,

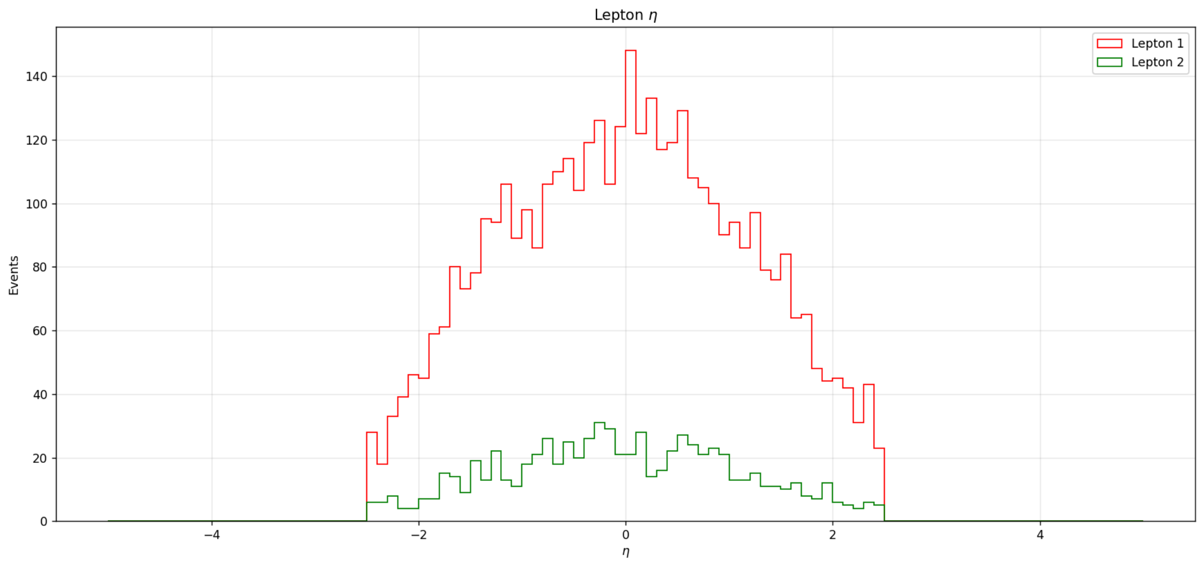
- R = 0.5



Lepton multiplicity and its explanation for the R = 0.5 case follows same as R =0.4 case.

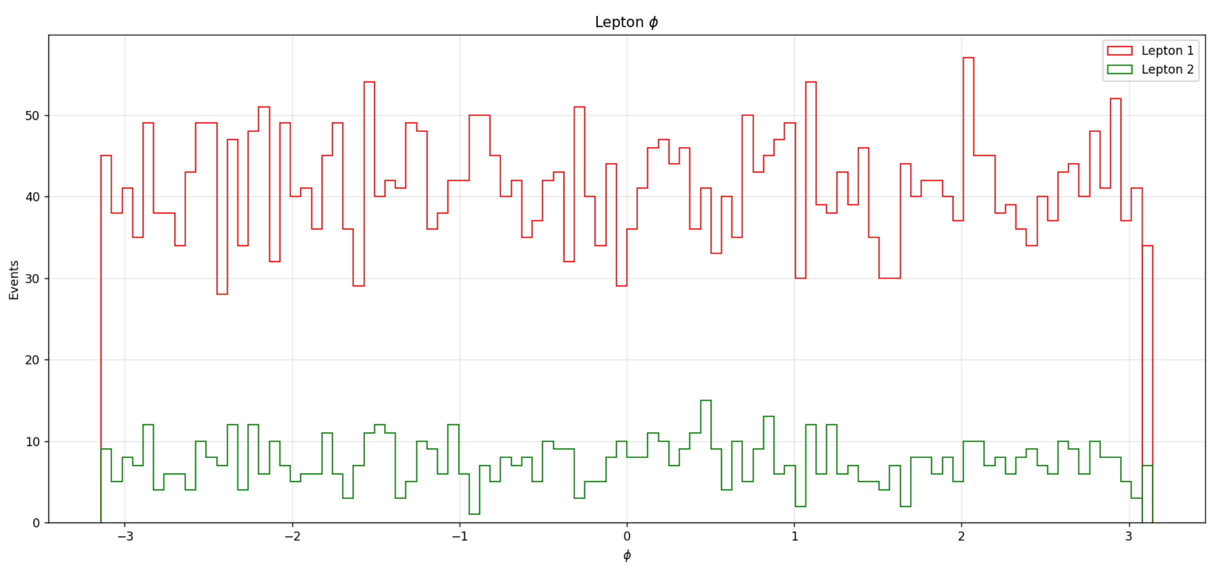
c) Lepton Eta

- R = 0.5



We see that the lepton eta plot for lepton 1 has a peak at 0 with a little spread. Hence these are produced symmetrically in the central region which is the expected outcome from the pp -> t t_bar process.

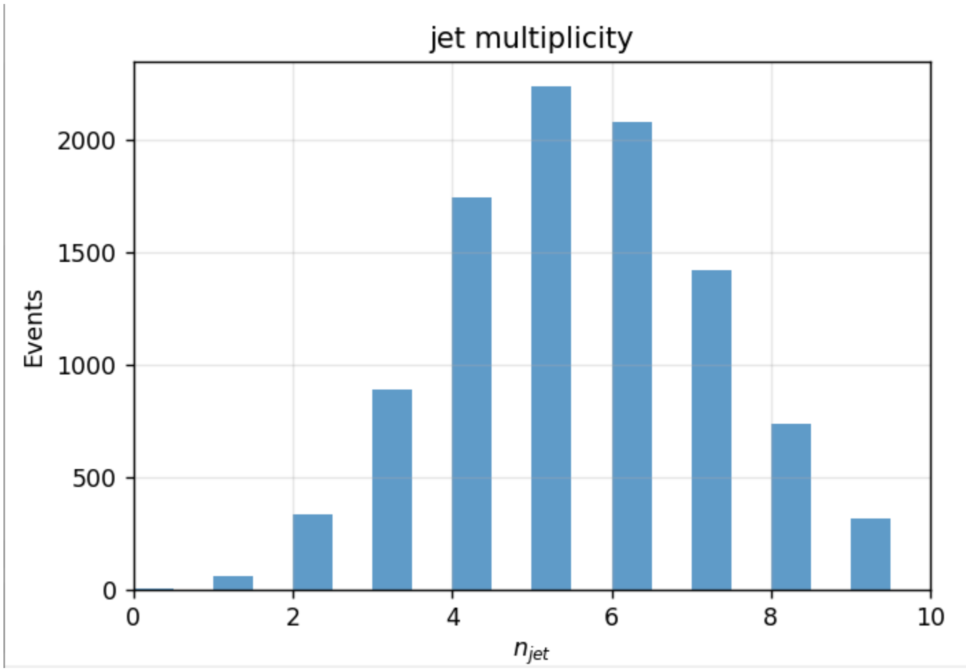
d) Lepton Phi



The uniform distribution in lepton points to isotropy with no preferred direction for leptons which is the expected result.

e) Jet multiplicity

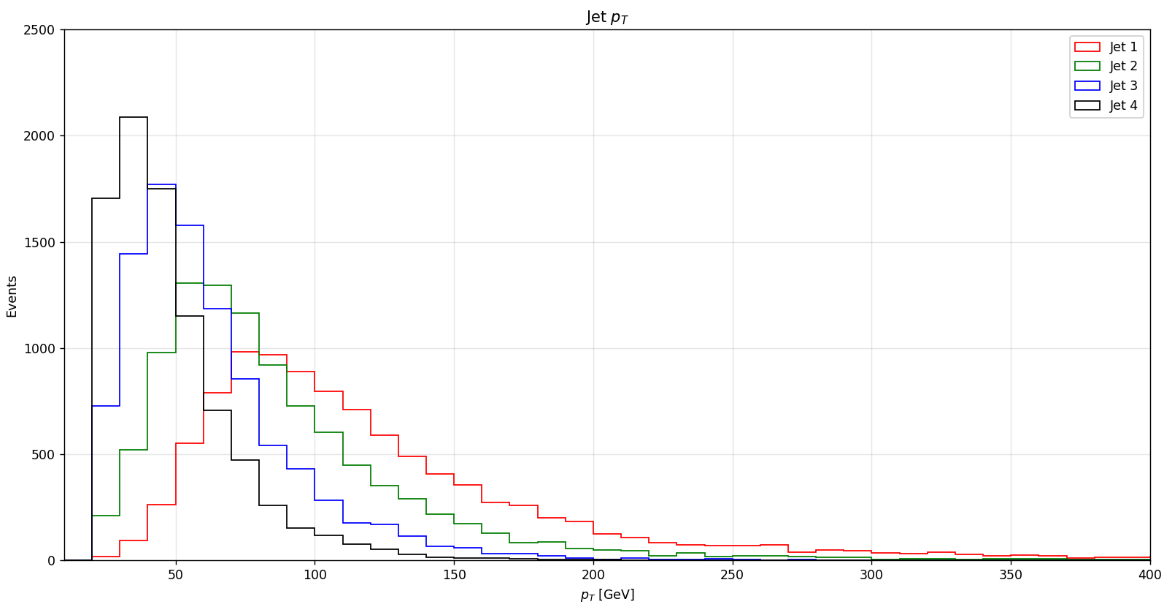
- R = 0.5



Top quark dominantly goes to W and b quark. W can decay hydronically to give 2 quark jets. So for fully hadronic decays, one can expect 1 jet from b quark and 2 jets from W for each top quark. Hence one expects 6 jets in total which can be seen from the plot. Since ISR/FSR is on, other gluon induced jets are added and hence total number of jets reaches upto 9. For semi-leptonic decays, one expects 4 jets i.e 2 b jets and 2 quark jets from W and for fully leptonic decays we only expect two b jets.

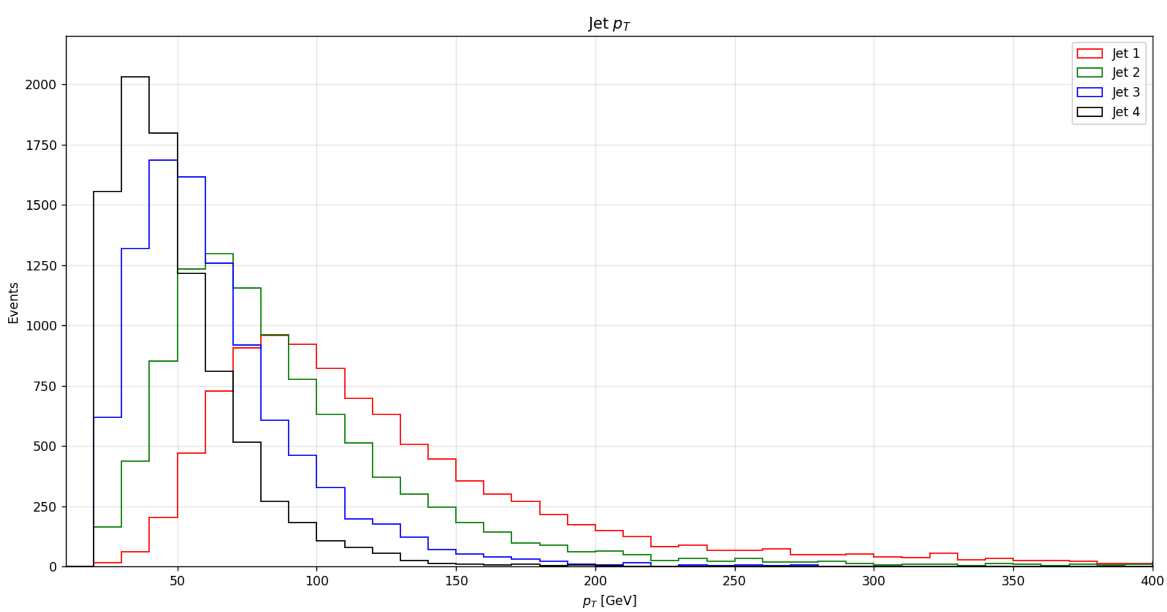
f) Jet pT

- R = 0.4



This is jet pT constructed for the first four jets for two different jet radii. Jet 1 pT peaks around 70-80 GeV range. Jet 2 pT peaks around 50-70 GeV range. Jet 3 pT peaks around 40-50 GeV range and Jet 4 pT peaks around 30-40 GeV range. Mass of top quark is around 172 GeV. Hence we see that leading and the first subleading jet has a pT peak around half the mass of top quark. Since top goes to b and W and W further can decay hadronically, the other subleading jets have mass peaks around 40 GeV range which is half the mass of the W. This explains the jet pT distribution.

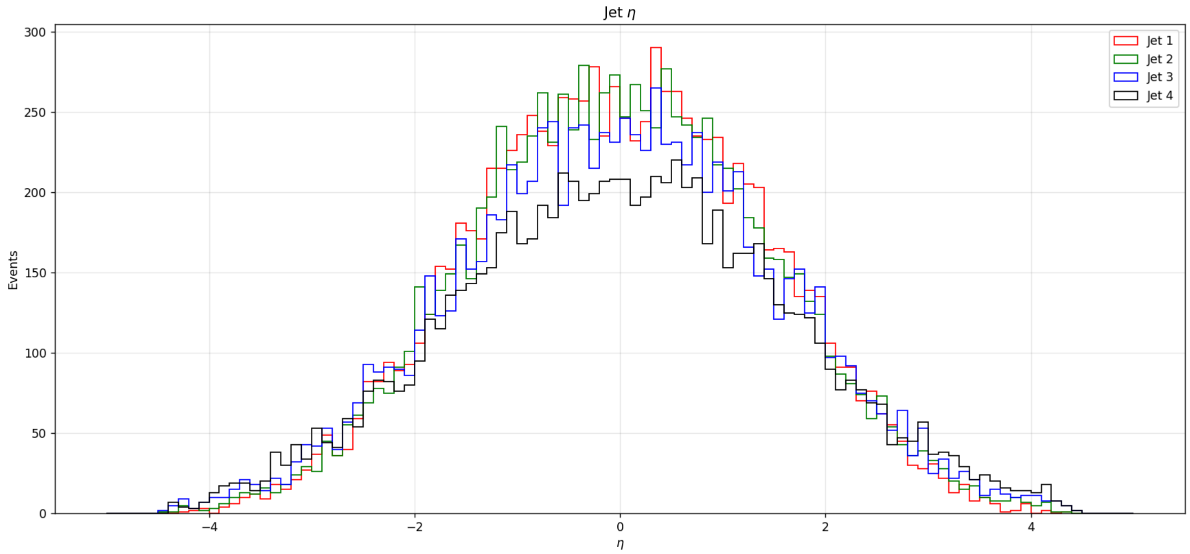
- R = 0.5



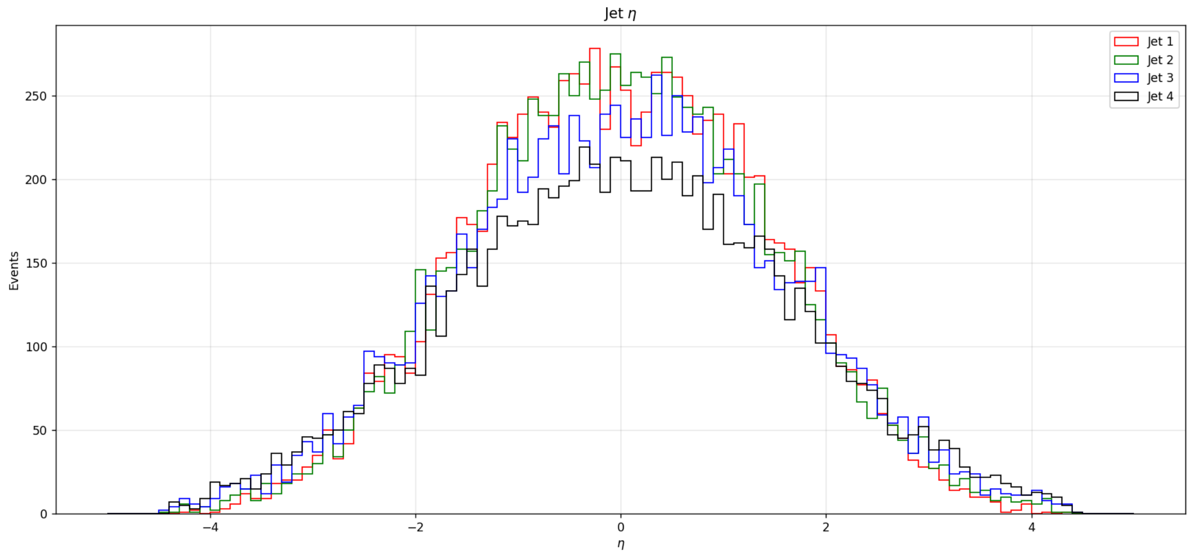
In case of R=0.5,we see that the peaks follow in the same manner as the R = 0.4 case and can be explained in the same way.

g) Jet Eta

- R = 0.4



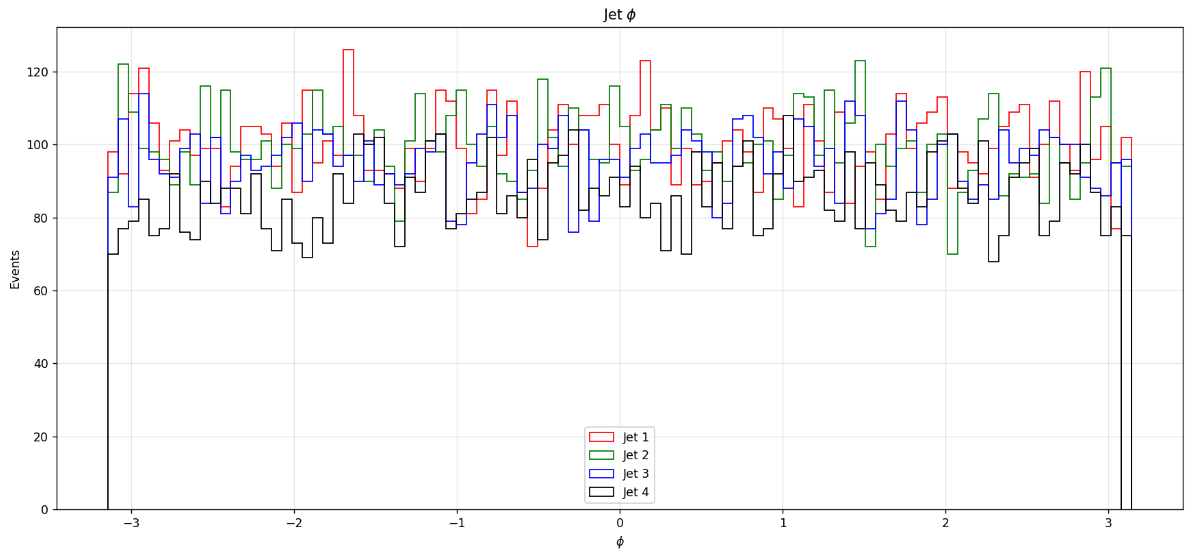
- R = 0.5



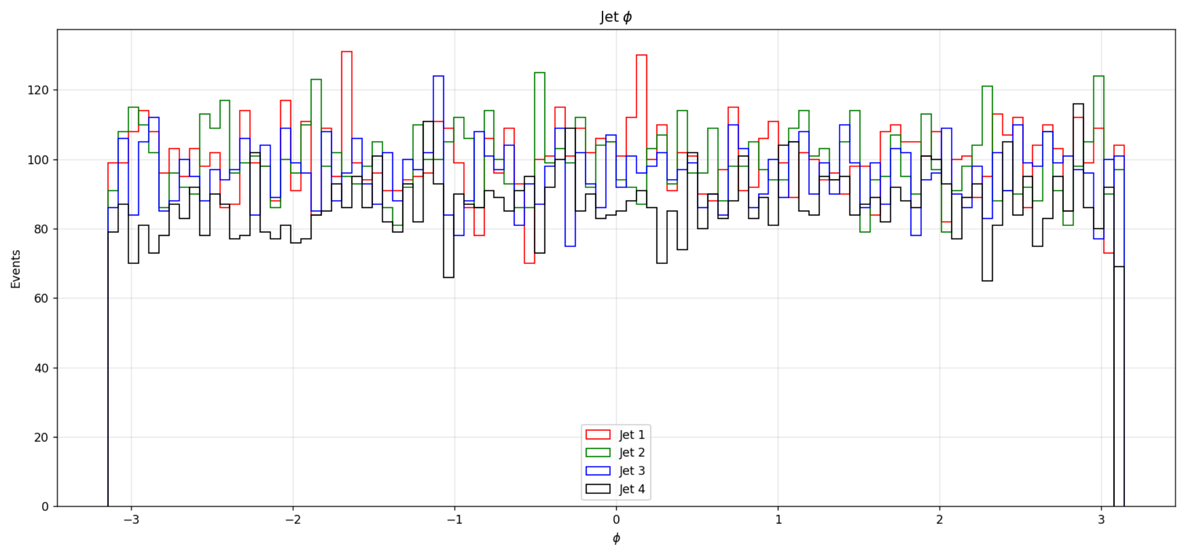
The plot shows that for all the jets, the peak is around eta=0. This suggests that all the jets are produced centrally in the collision in the barrel region of the detector. The symmetry of the plot suggests that there is no forward backward preference. There is no significant difference between the two jet radii cases.

h) Jet Phi

- R 0.4



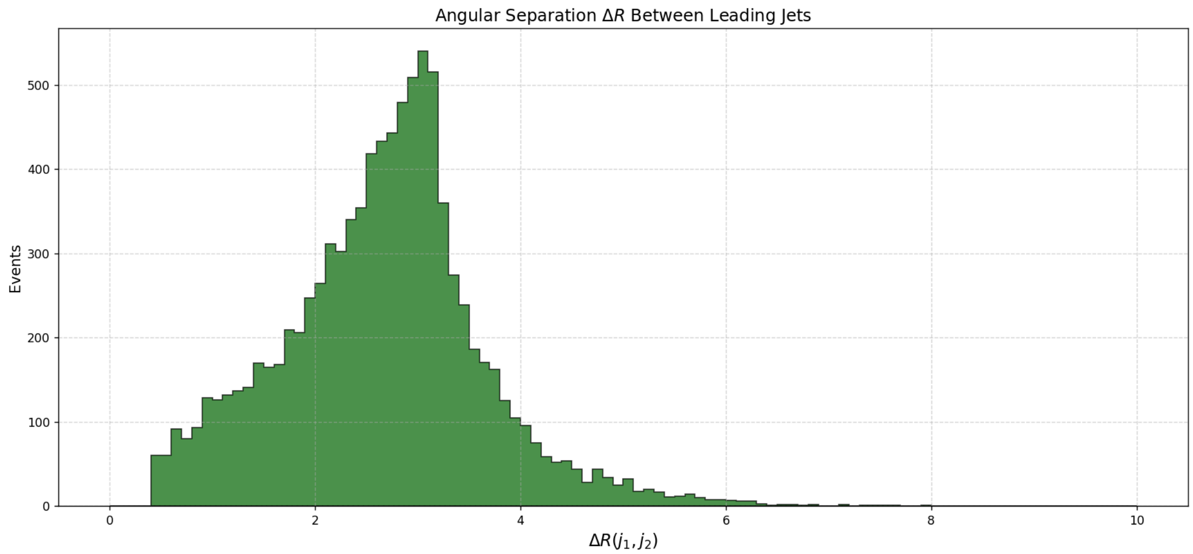
- R = 0.5



The uniform distribution of phi suggests that no angular direction is preferred and the jets are isotropic in angular direction. The distribution is not wrapped around [-π,+π] but instead [0,+π].

i) Angular separation

- $R = 0.4$

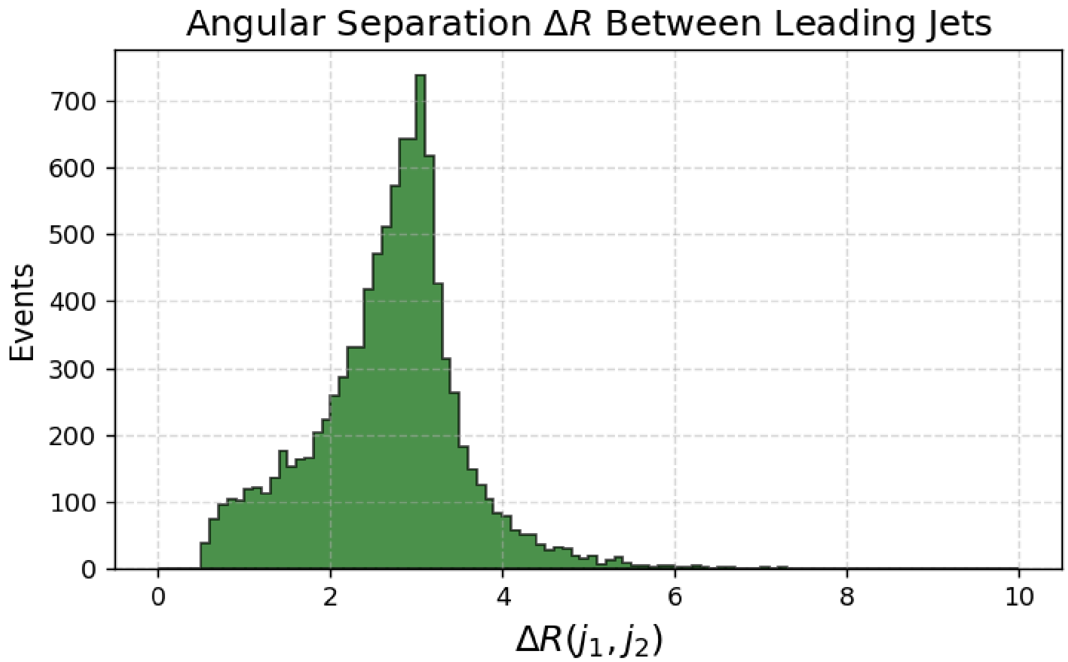


One can calculate the angular separation between the jets as;

$$\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$$

In the plot we see a peak at around 2.5-3 and sharp drops along the right side and a more gradual drop along the left side. This can tell us that the two leading jets are not widely separated. These jets likely come from the b quarks. There are very few events with wide jet separation. Since we are also working at $\sqrt{s} = 14\ TeV$, the jets are likely to be boosted and hence clustered closer.

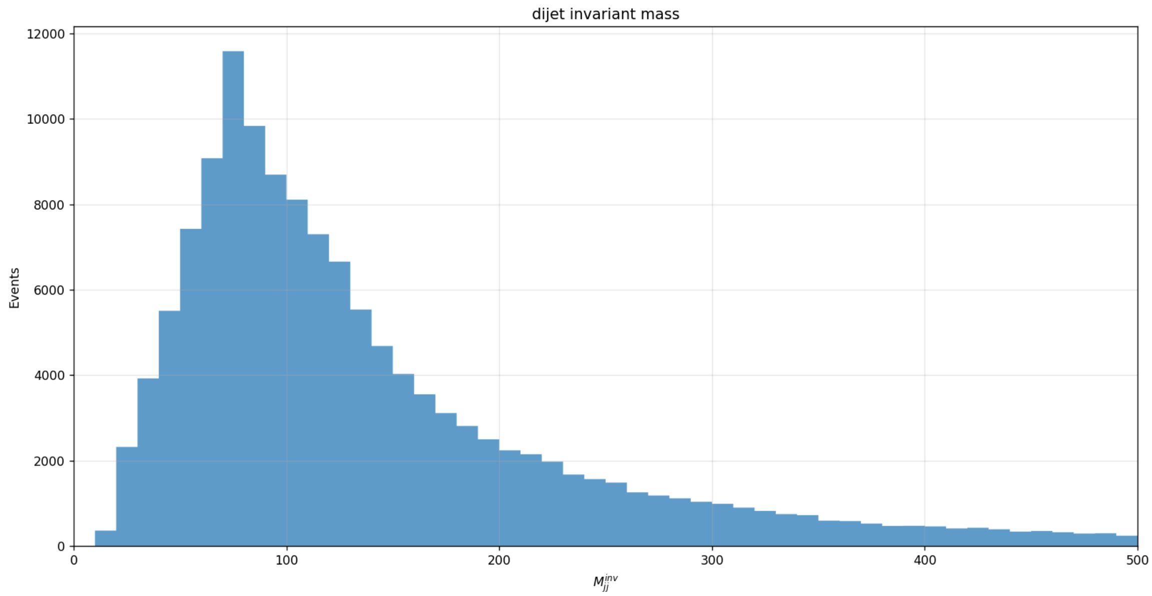
- R = 0.5



No significant difference can be seen in this case and the R = 0.4 case. The peak is still observed at around 2.5-3.

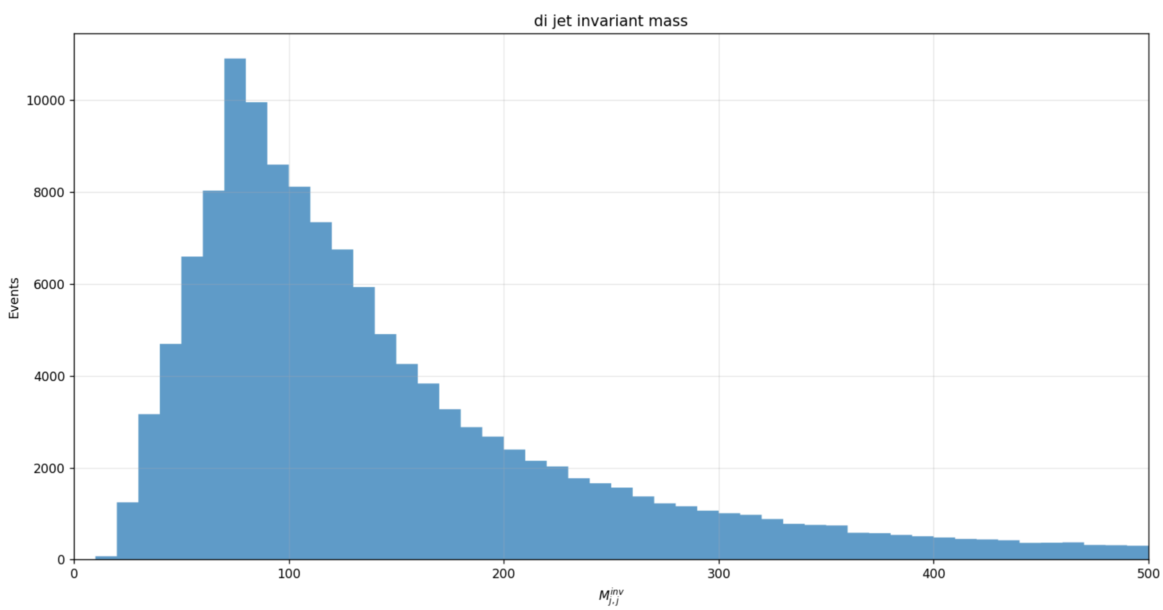
j) Dijet invariant mass

- R = 0.4



The peak in the dijet invariant mass is around 70-80 GeV. Since top decays to bottom and a W and W can further decay hadronically, reconstructing the mass from hadronic decays given that the jets are coming from W, should give us the W mass which seems to be consistent with the plot.

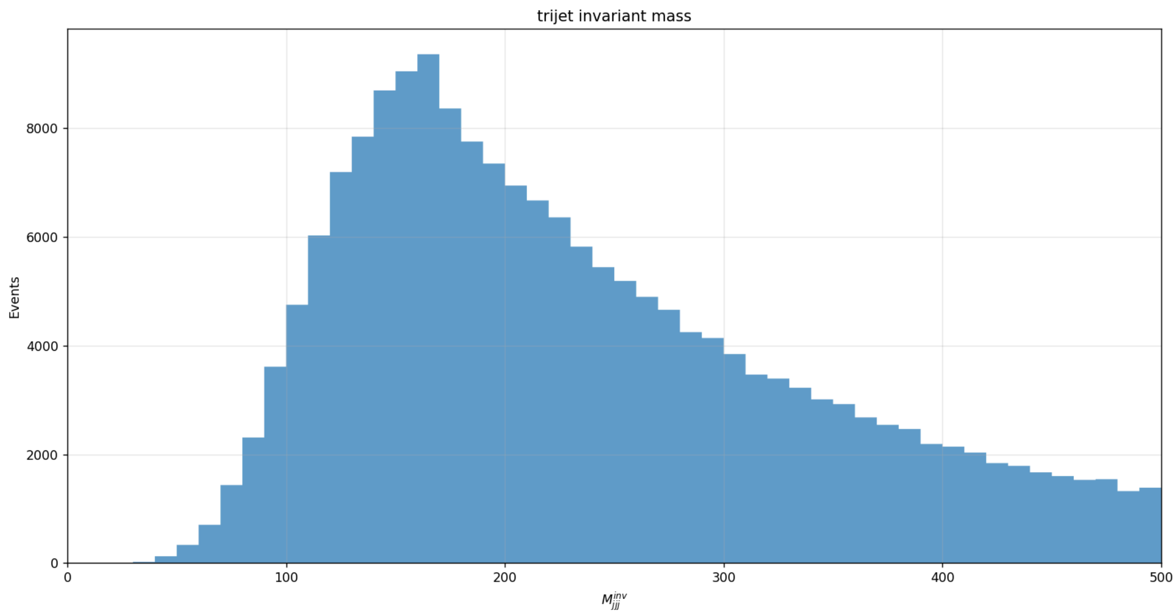
- R = 0.5



Exactly similar dijet invariant mass is constructed for R = 0.5 also and the explanation follows from the R = 0.4 case.

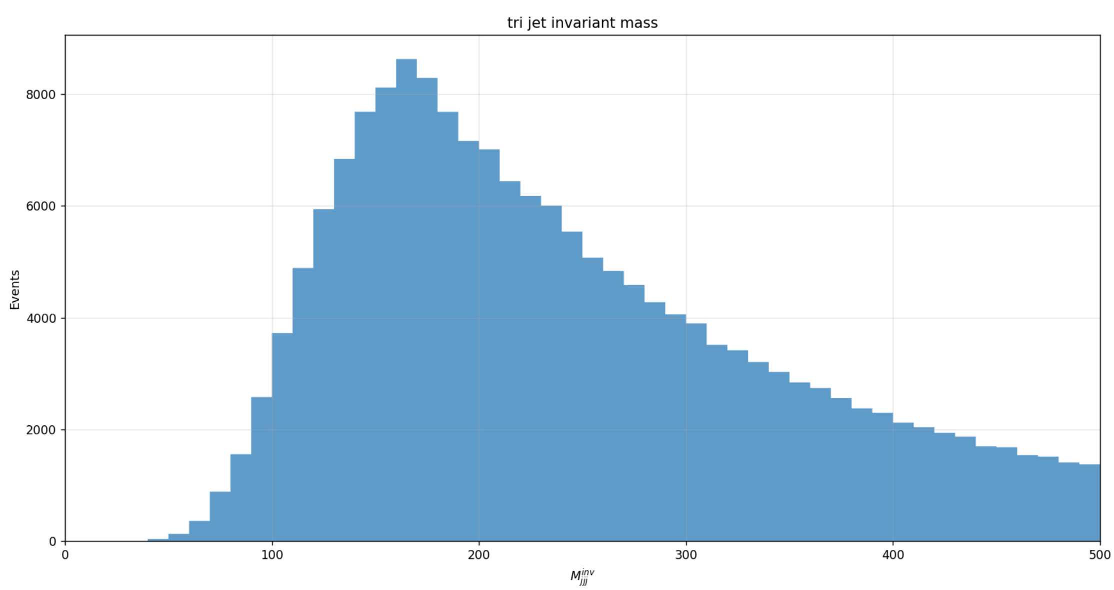
k) Trijet invariant mass

- R = 0.4



For the trijet invariant mass the peak is seen around 170-180 GeV. The mass of top quark is ~172 GeV. Hence the invariant mass of b jet and jets from W decay can reconstruct the mass of the top quark. This seems consistent with the plot.

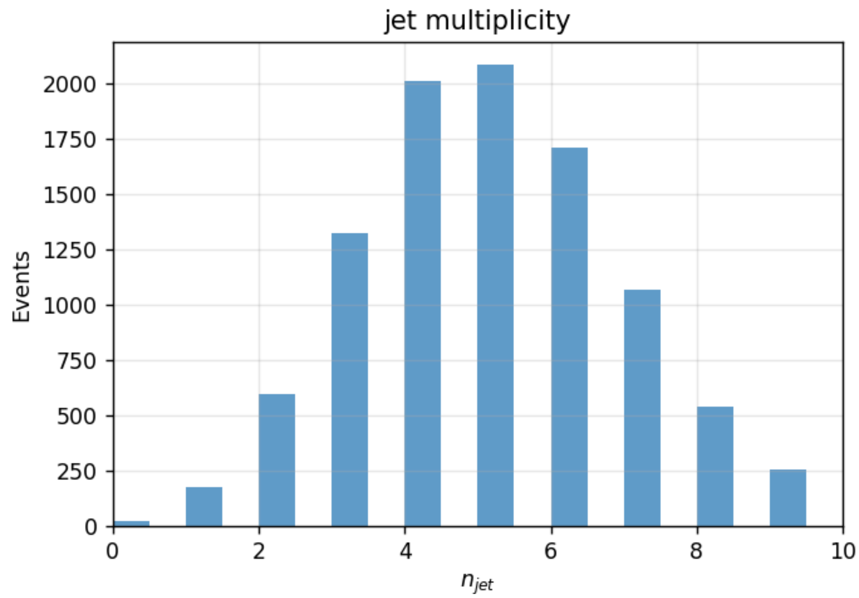
- R = 0.5



In this case also we see a peak at the mass of the top quark which is reconstructed from the first three jets with one jet b jet and two jets from the W hadronic decay.

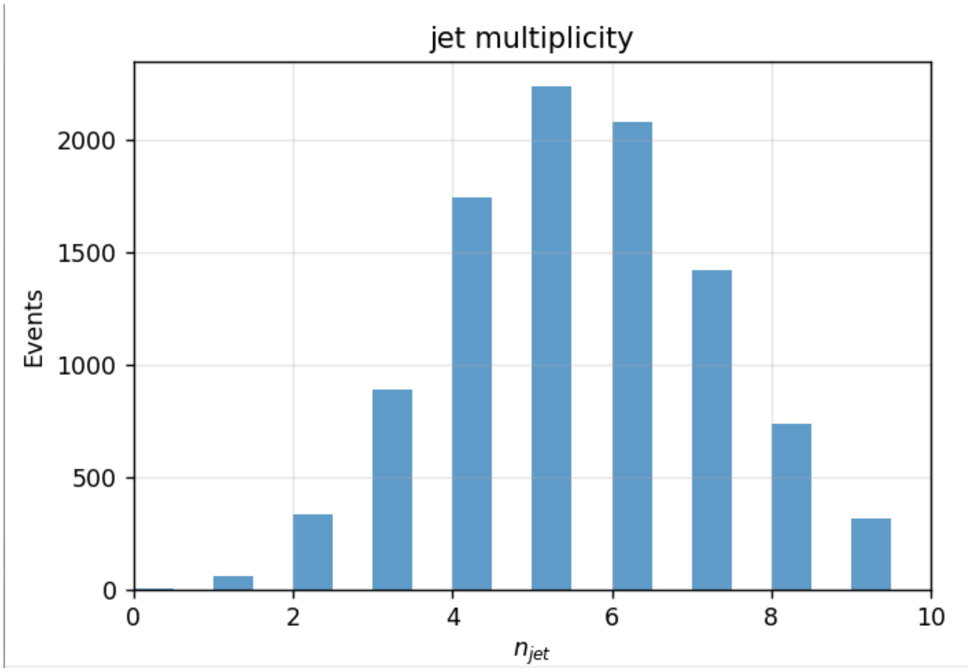
l) Jet multiplicity with ISR/FSR on

- R = 0.2



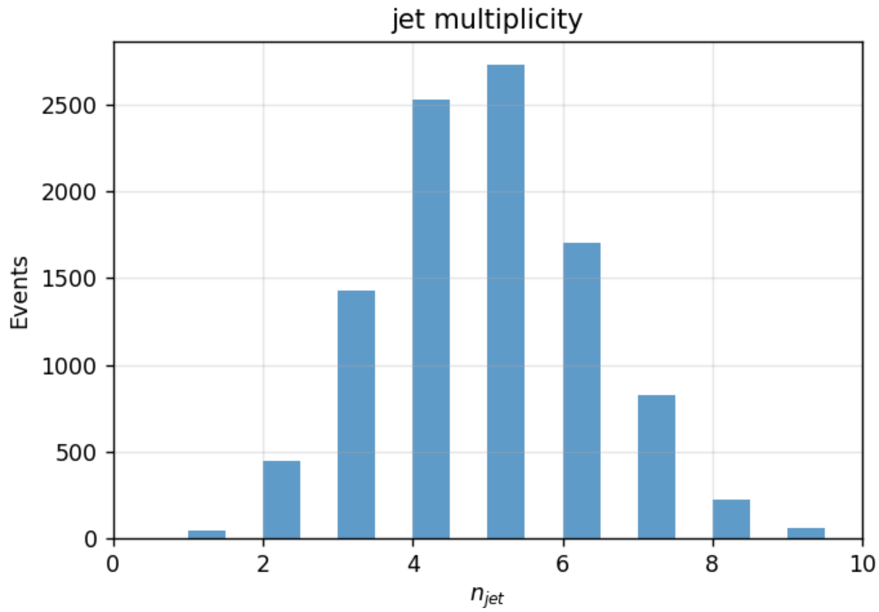
For the above jet multiplicity we observe that the peak is around 4-5 jets. 4 jets are expected from semileptonic decays of W. Due to ISR/FSR one can have more gluon induced jets. Important thing to notice is that there are some amount of events, though very less, with 0 jets.

- R = 0.5



Similar to the above case, the peak appears at 5.One important difference that can be noted is that for all jet multiplies to the left of 5 jets, events have reduced in number and the entries to the right of 5 jets have increased slightly. 0 jet events have almost vanished.

- R = 1

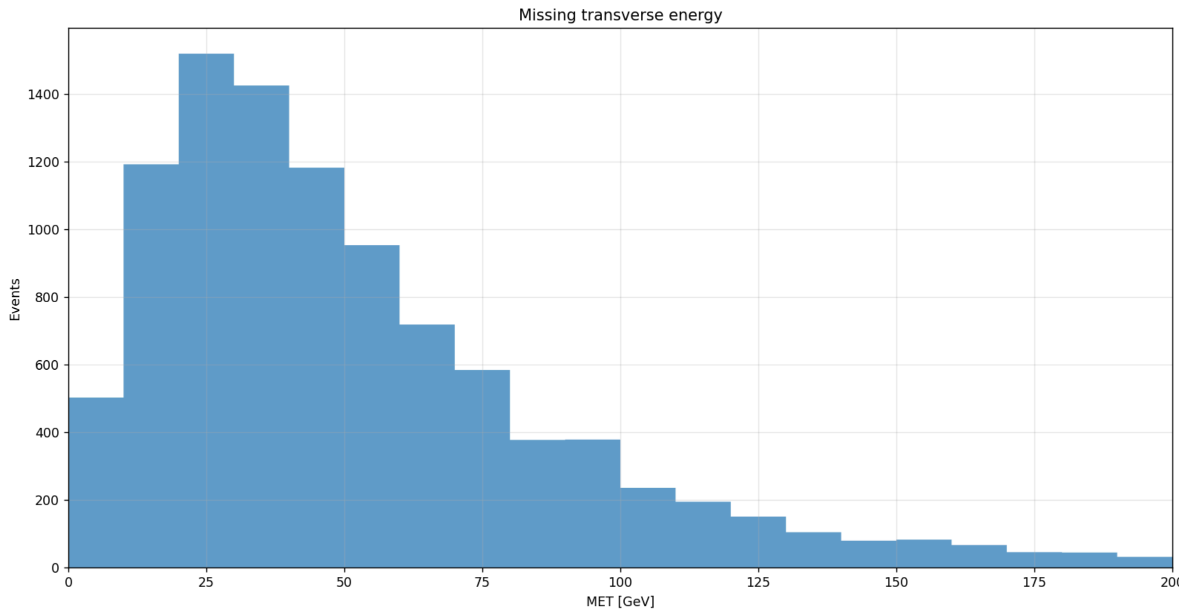


For this case, the structure of the plot seems fairly similar to the first case with the difference of 0 events for n_{jet}=0.

One conclusion that may be drawn from the above comparison is that for quite large or quite small values of R, there may be similar structure of jet multiplicities. If we look at moderate jet radius values such as 0.4,0.5 etc. we may get to see some changes in multiplicities structure.

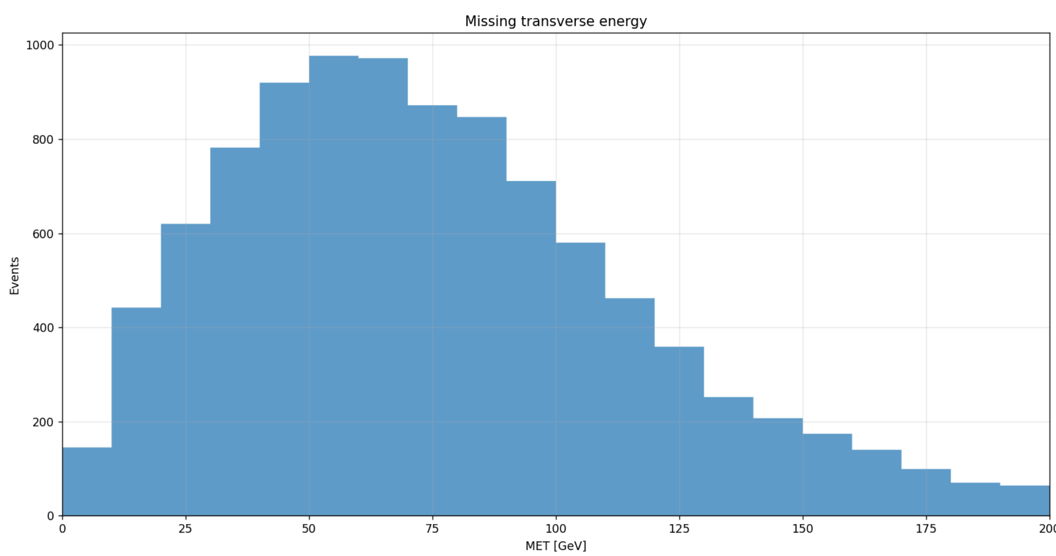
m) MTE/pT_(miss)

- MTE without forced W decay



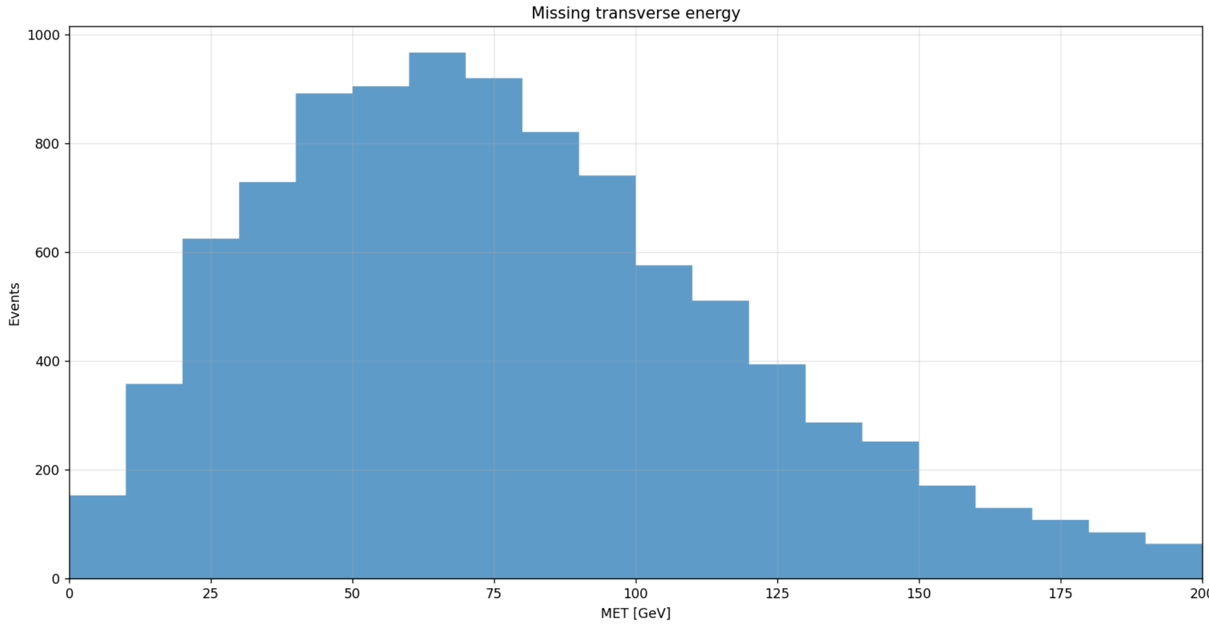
The peak for MTE is seen at around 25-30 GeV. Since the W can decay leptonically as well as hadronically, one would in ideal case expect the MTE to peak around half the mass of W i.e at around 40 GeV. But due to recoil from the aftereffects of ISR/FSR the peak comes out at the value slightly lesser than 40 GeV. The plot seems to be in agreement with expectation.

- MTE with forced W decay



We see that with the forced W decay, the peak has shifted and became broader, likely due to ISR/FSR. With the forced decay, one would expect that the MTE peaks at half the mass of W i.e at about 40 GeV. In the above case the peak is slightly shifted to 50 GeV.

- Neutrino tagged MTE

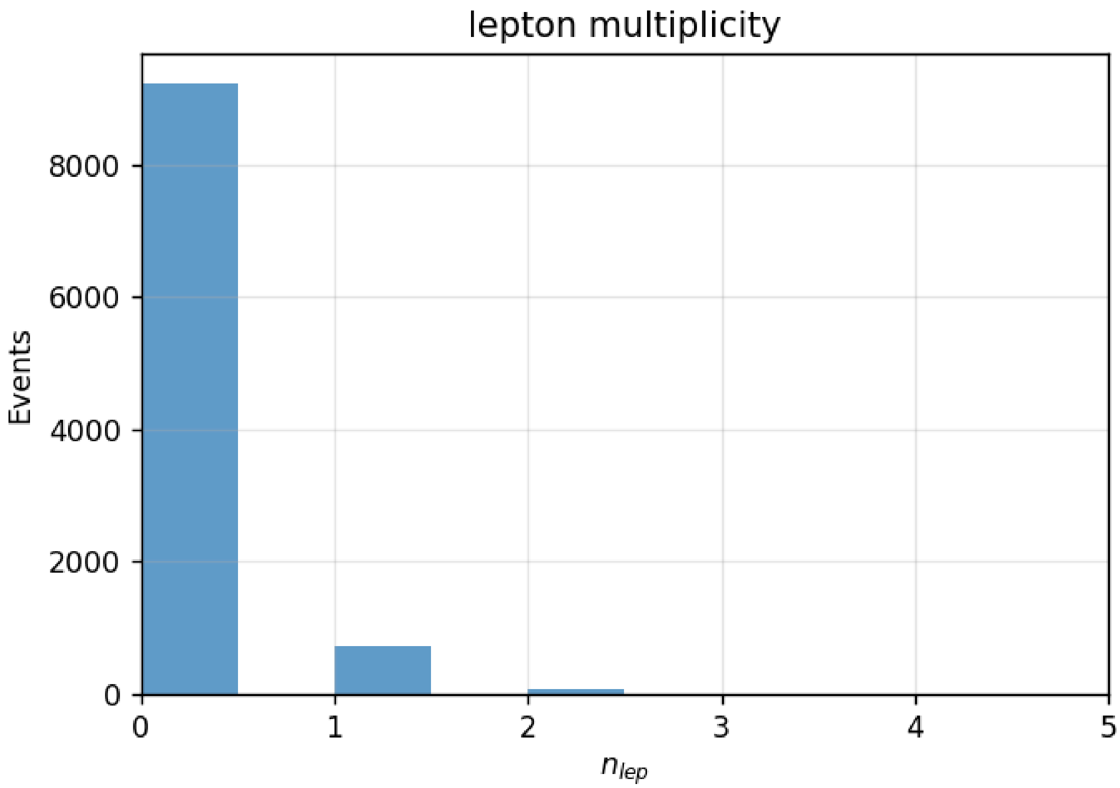


The tagged neutrino pT(miss) looks similar to the forced decay MTE. This strongly suggests that MTE calculated from the forced decay is a good proxy for neutrinos. This is since in detectors, neutrinos cannot be tagged and hence one needs to infer their existence from the MTE plot. Thus its expected that the neutrino-tagged pT(miss) be similar to MTE calculated from the forced W decay.

2. $\underline{gg \to h}$: Generate 10000 events of gluon-gluon fusion to Higgs boson production at the 14 TeV LHC using `Pythia8`, and do the following:

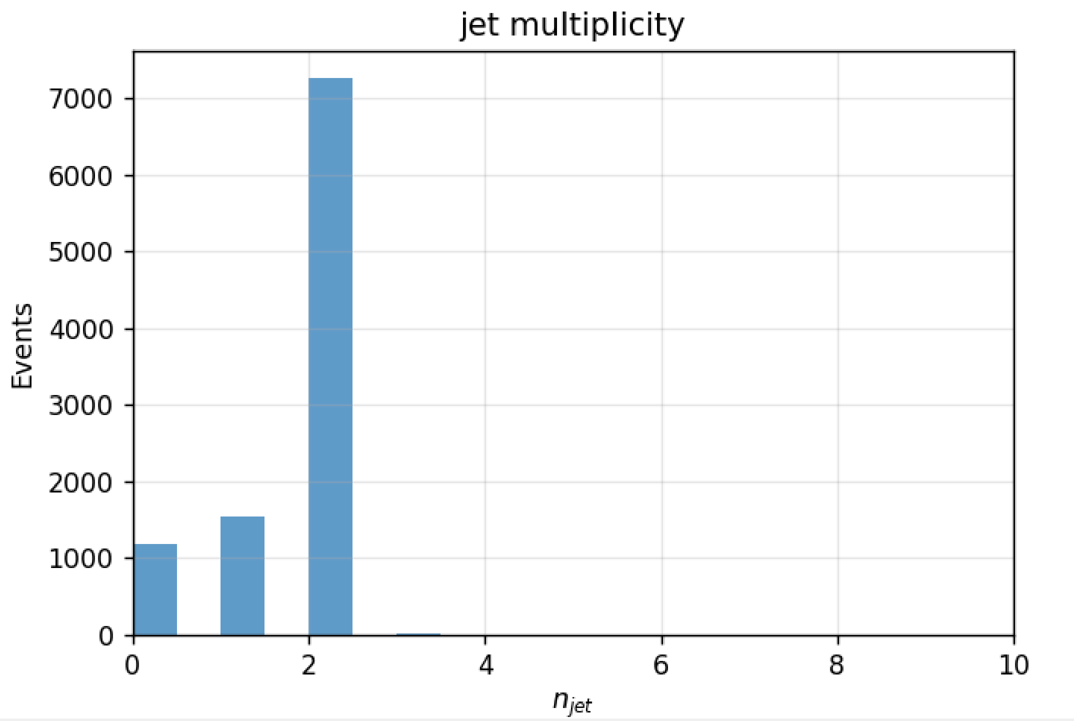
    - First, keep ISR/FSR off. Plot the lepton multiplicity, jet multiplicity, and p_T, η, ϕ of the first two jets. Explain the behaviour and sources of the leptons/jets.
    - Keeping ISR/FSR off, plot the dijet invariant mass. Explain the peak(s) in the distribution.
    - Now, turn on the ISR/FSR and plot the jet multiplicity, and the dijet invariant mass again. Explain the changes in the plots.
    - Keep ISR/FSR on. Force the Higgs boson to decay into two Z-bosons, and subsequently force the Z-boson decay into leptons only. Plot the lepton multiplicity, and reconstruct the four-lepton invariant mass. Explain the plots.
    - Finally, find out a way to reconstruct the Higgs boson invariant mass from its decay into two photons i.e. $h \to \gamma\gamma$. Explain the plots.

a) **Lepton multiplicity (ISR/FSR off)**



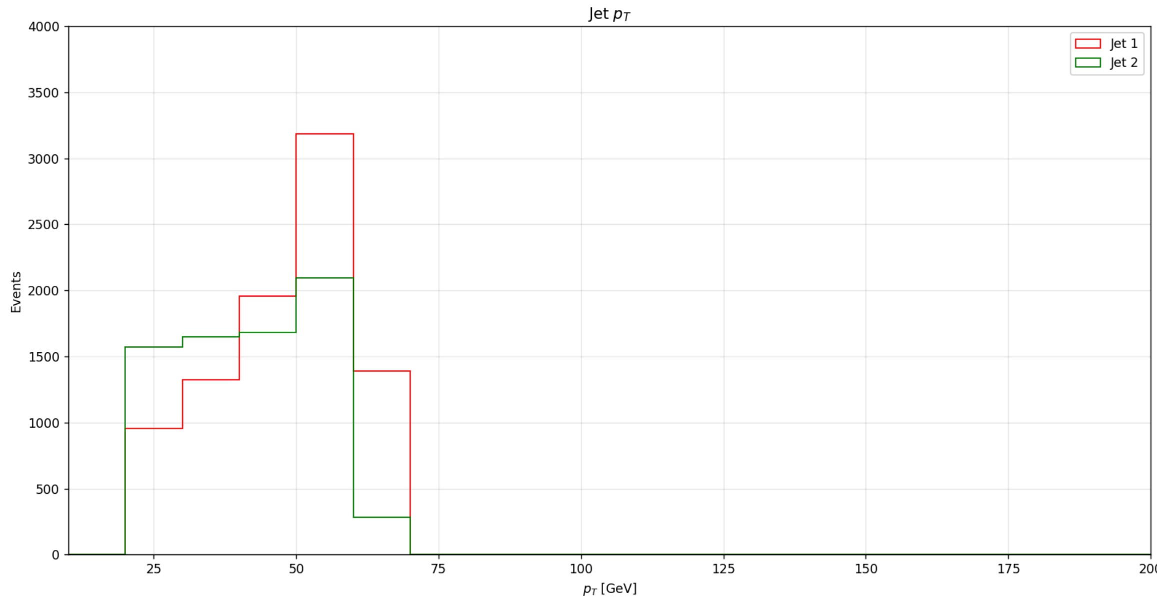
For h production from gg, as expected there is dominance of 0 lepton events. This is since h can further go to b b_bar, W+ W-, ZZ etc as dominant mondes. W can decay leptonically or hadronically and hence one may expect some leptons to be present in some events. Similar logic with Z. Hence we n_{lep}=1 and n_{lep}=2 events. Important point to notice is that ISR/FSR is off for this case.

b) Jet multiplicity (ISR/FSR off)



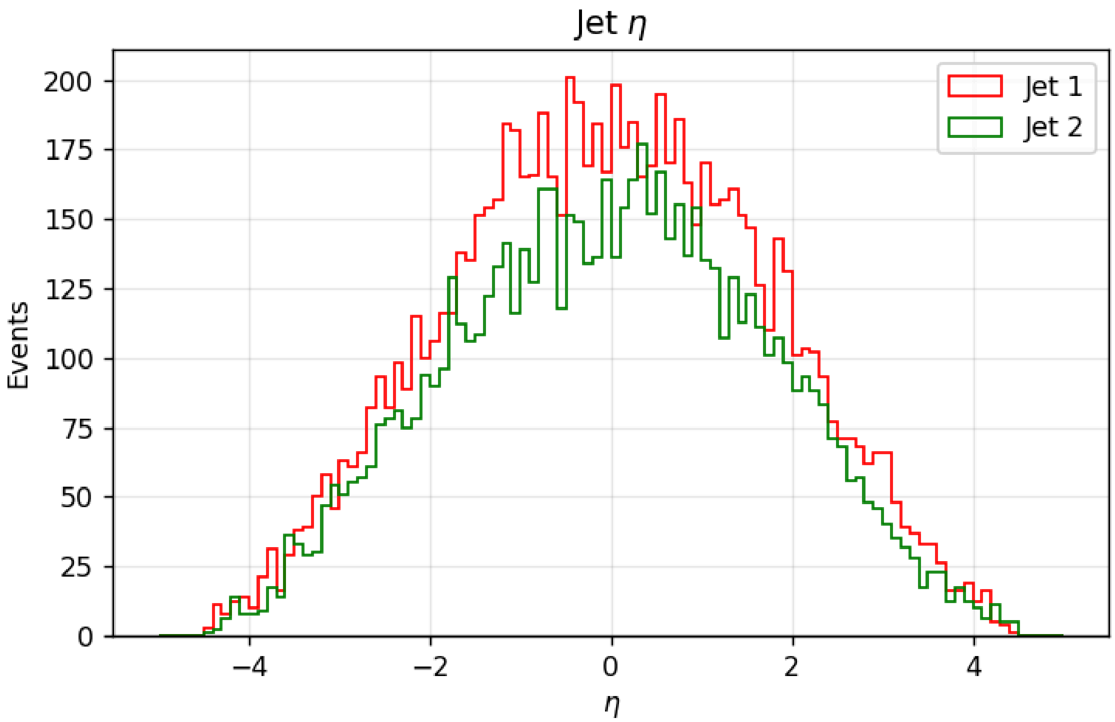
With ISR/FSR on, one may expect the jet multiplicity to go upto 8 jets or 9 jets.But in this case, ISR/FSR is off and hence as mentioned in the previous part, from the b b_bar channel one expects 2 jets. From the W+ W- channel, if W decays hadronically, then one expects 4 jets. We see in the plot that are tiny number of 3 jet events. Since we are operating at 14 TeV centre of mass energy, the jets are likely highly boosted hence peaking around n_{jet}=2. The we expect such a behavior.

c) Jet pT (ISR/FSR off)



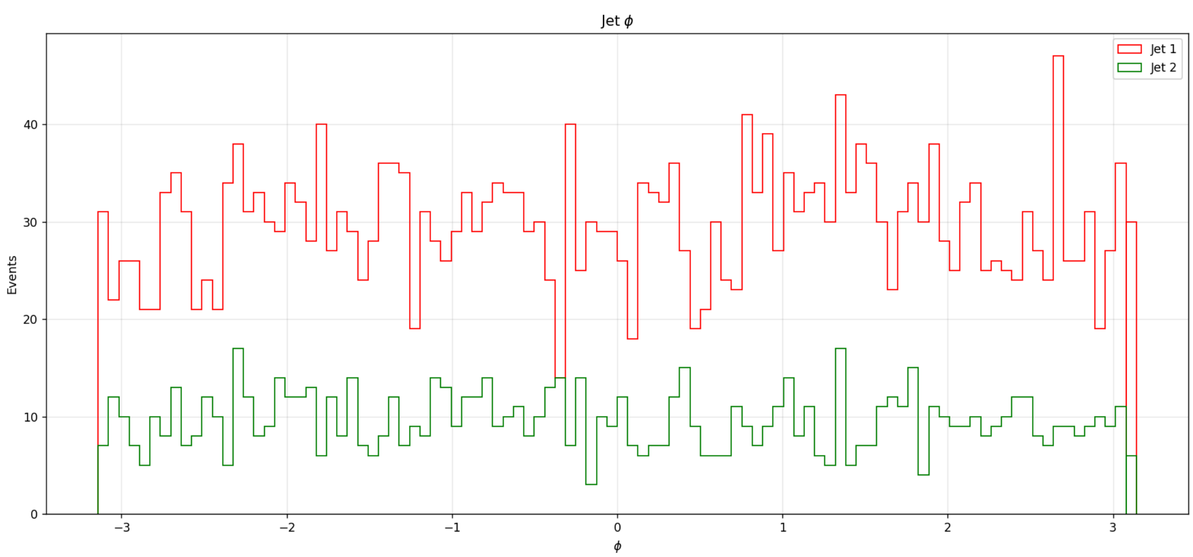
This plot represents the jet pT for the first two jets when ISR/FSR is off. The peak is around 60-70 GeV. This is about half the mass of higgs and in the b b_bar channel of higgs decay one would expect such a behavior.

d) <u>Jet eta (ISR/FSR off)</u>



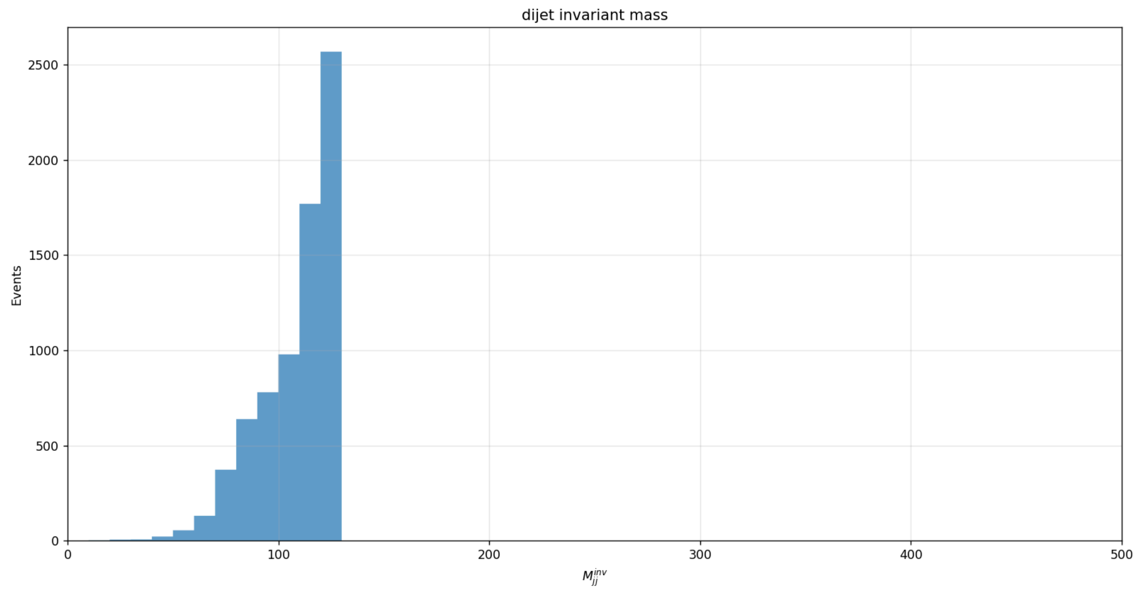
The peak for eta plot of the first two jets is seen at 0 indicating no strong preference forwarded or backwarded jets as expected.

e) <u>Jet Phi (ISR/FSR off)</u>



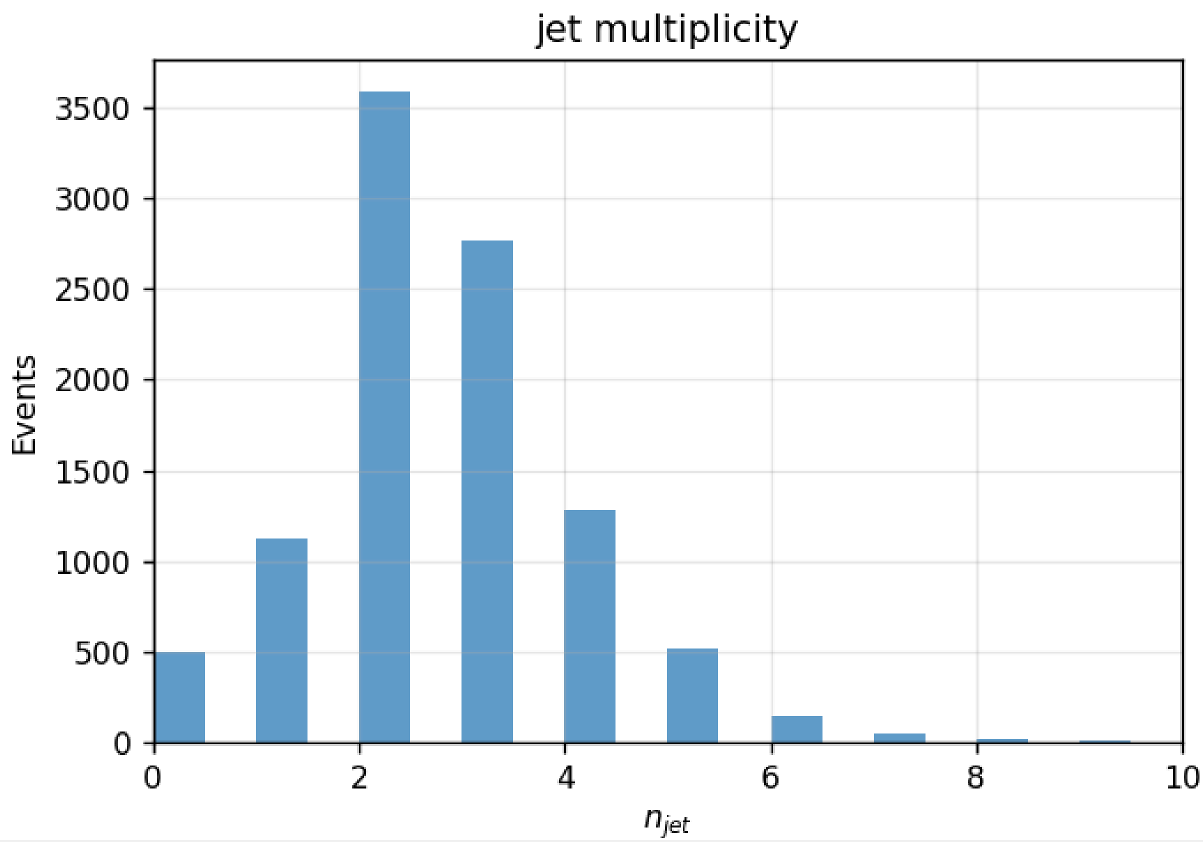
A uniform distribution in Phi is indicative of no preference for any angular direction and isotropic nature of jet angular orientation.

f) Dijet invariant mass (ISR/FSR off)



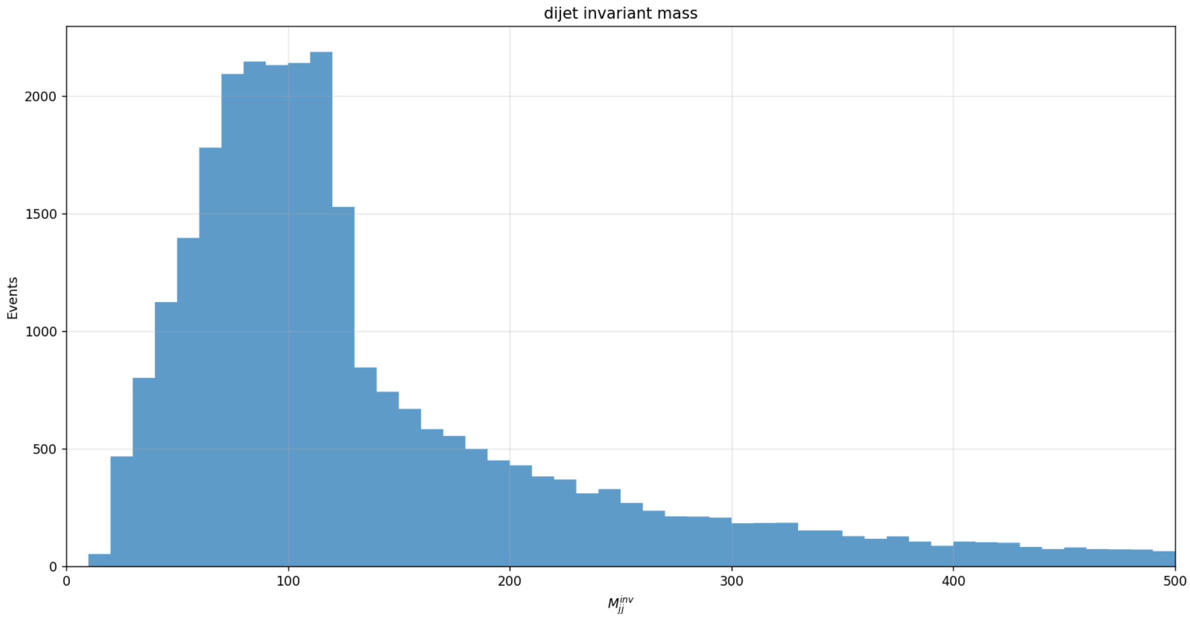
With the ISR/FSR off, one would expect a sharp peak at the higgs mass of 125 GeV. Plot represents the same. We see a sharp peak at the higgs mass since after gg produces higgs, it can go through different decay channels and those are responsible for giving jets. Hence the invariant mass of the two leading jets should reconstruct higgs mass as is seen in the plot. No tail behavior is seen in the plot.

g) Jet multiplicity (ISR/FSR on)



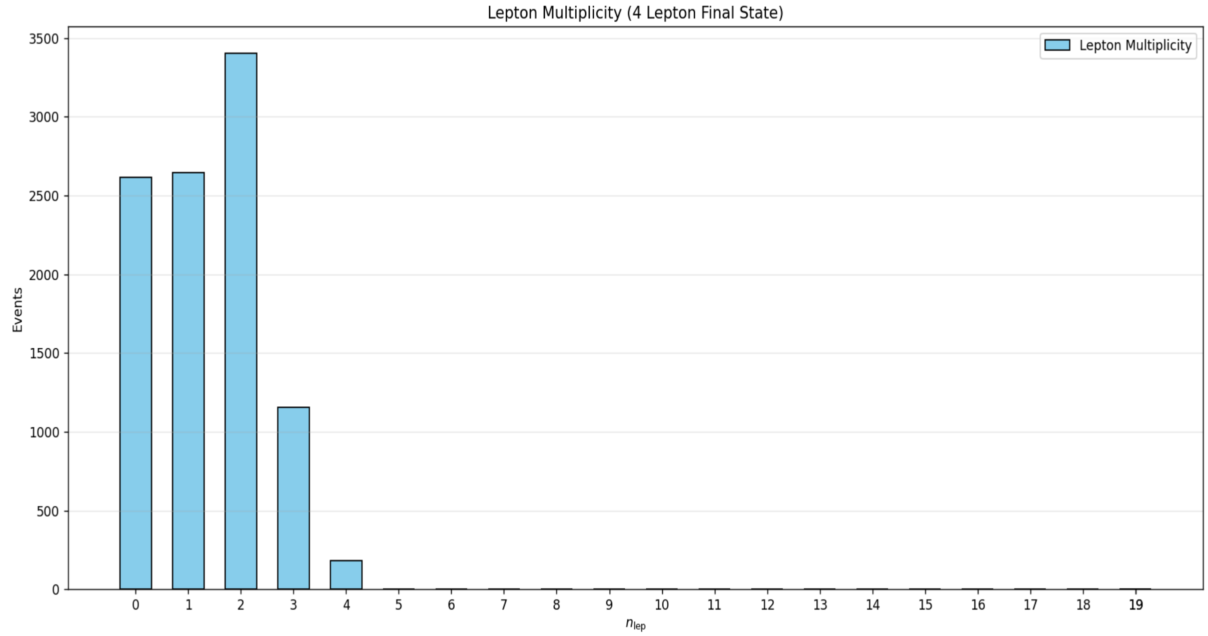
As was mentioned in the point (b) above, with ISR/FSR on, we see a lot more jets with highest number reaching upto 8-9 jets. The peak is still seen at 2 as expected. The difference is that now we see 3 and 4 jet events also. 4 jet events likely arise from the W+ W- and ZZ channel decay channel of higgs with both W's/Z's decaying hadronically. The 2 jet events can arise from b b_bar. The higher jet multiplicity is likely a result of gluon induced jets.

h) Dijet invariant mass (ISR/FSR on)



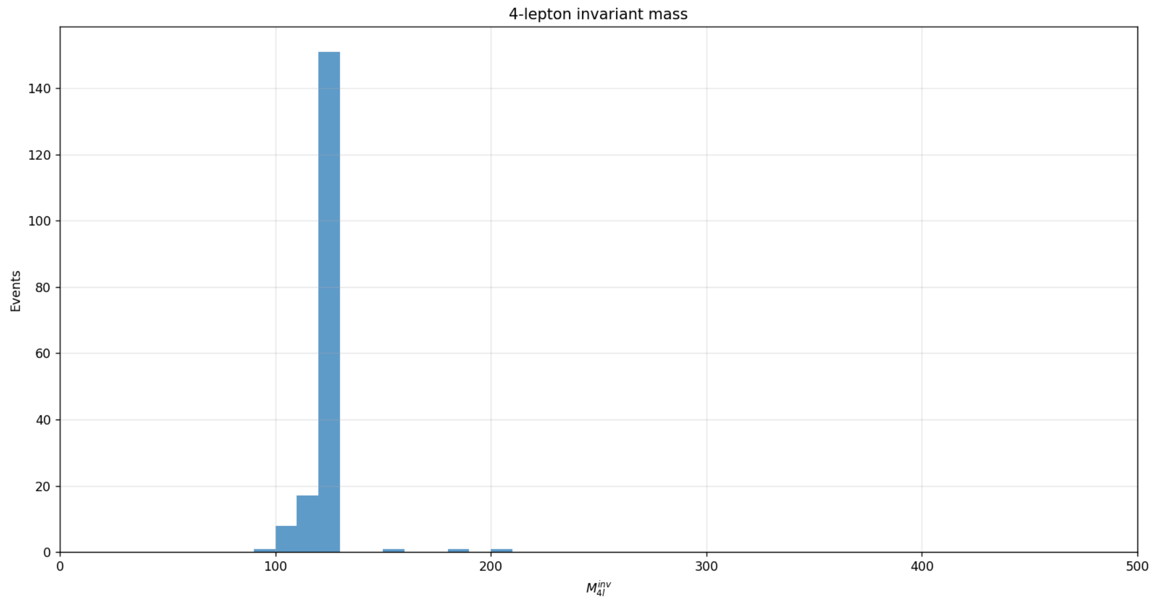
Comparing to the ISR/FSR off case, the dijet invariant mass peak has broadened and slightly shifted leftwards i.e lesser than 125 GeV. The peak can be still seen around the 120 GeV but broadened between 100-130 GeV.This effect is anticipated with likely gluon interactions in pp collisions. We also see a gradually falling off tail behavior for this case. That behavior was absent in the ISR/FSR off case.

g) Lepton multiplicity with forcing Z to decay leptonically



The letonic decay of Z can produce e+ e-, mu+ mu- or tau+ tau-. Even neutrino decays are possible which manifest as MET. Thus one expects to have 4 leptons from the ZZ decay channel of higgs. In the plot we see some 4 lepton events but the peak is at 2 lepton events. I think this may be due to boosted frame and lepton isolation caveat. If we isolate leptons one should see highest number of 4 lepton events. The above plot may also have effects from ISR/FSR induced interactions.

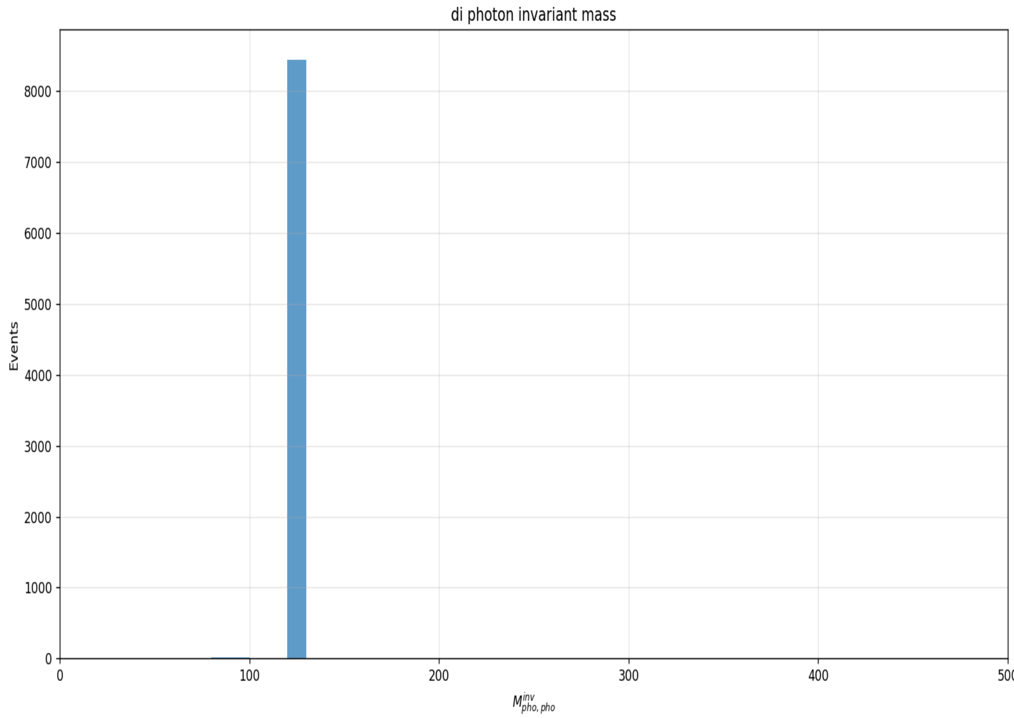
h) Four lepton invariant mass



As anticipated the four leptons from h->zz->4l should reconstruct the higgs mass which is what we see from the plot. Majority of the 4l events reconstruct the higgs mass accurately. However, we see that there are quite less number events in the bin. I have tried plotting it multiple times but I get the same plot. I think this is likely due to inaccurate binning of lepton events. However as expected we are able to reconstruct higgs mass.

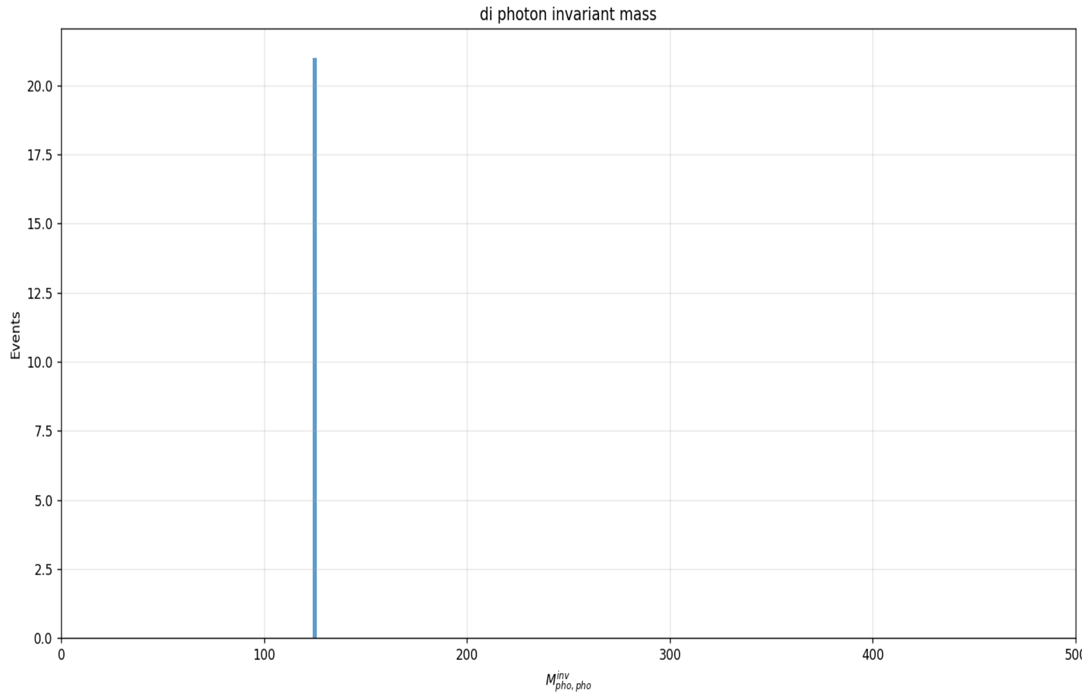
i) Diphoton channel

- Forcing the h -> ɣ ɣ



As expected after forcing the higgs to decay through diphoton channel and reconstructing the diphoton invariant mass, we get the peak at higgs mass exactly. This is completely expected.

- Not forcing the h -> ɣ ɣ



Even after not forcing the decay, one can reconstruct the higgs mass from the diphoton channel. But due to the lower branching fraction, we see very less number of diphoton events.

Codes:

- Codes used for Q1

```
#include "Pythia8/Pythia.h"
#include "Pythia8Plugins/FastJet3.h"
#include "fastjet/PseudoJet.hh"
#include "fastjet/ClusterSequence.hh"
#include <iostream>
#include <sstream>
#include <fstream>
#include <cmath>
#include <numeric>

using namespace Pythia8;
using namespace std;
using namespace fastjet;



int main(){

int nEvent=10000;
double lumi=1000.;

double R        = 0.4;
double pTMin    = 20.0;
double etaMax   = 4.5;
double etaMaxLep = 2.5;

Pythia pythia;
Event& event = pythia.event;

pythia.readString("Beams:frameType=1");
pythia.readString("Beams:idA = 2212");
pythia.readString("Beams:idB = 2212");
pythia.readString("Beams:eCM = 14000.");

pythia.readString("Top:gg2ttbar=on");
pythia.readString("Top:qqbar2ttbar=on");
```

```
pythia.readString("PartonLevel:FSR = on");
pythia.readString("PartonLevel:ISR = on");
pythia.readString("PartonLevel:MPI = off");
pythia.readString("HadronLevel:Hadronize = on");
pythia.readString("HadronLevel:Decay = on");

pythia.readString("SigmaProcess:renormScale2 = 4");
pythia.readString("SigmaProcess:factorScale2 = 4");

pythia.readString("Next:numberCount = 100 ");


//pythia.readString("24:onIfMatch = 11 -12");
//pythia.readString("24:onIfMatch = 13 -14");
//pythia.readString("24:onIfMatch = 15 -16");
//pythia.readString("24:onIfMatch = -11 12");
//pythia.readString("24:onIfMatch = -13 14");
//pythia.readString("24:onIfMatch = -15 16");


fastjet::JetDefinition jetDef(fastjet::antikt_algorithm, R);
std::vector <fastjet::PseudoJet> fjInputs;

pythia.init();

// Histograms
Hist jetmul("jet multiplicity", 20,0.,10.);
Hist lepmul("lepton multiplicity", 20, 0., 10.);
Hist leppt1("lep 1 pt", 200, 0.,2000.);
Hist leppt2("lep 2 pt", 200, 0.,2000.);
Hist lepeta1("lep 1 eta", 100, -5., 5.);
Hist lepeta2("lep 2 eta", 100, -5., 5.);
Hist lepphi1("lep 1 phi", 100, -M_PI, M_PI);
Hist lepphi2("lep 2 phi", 100, -M_PI, M_PI);
Hist jetpt1("jet 1 pt", 200, 0.,2000.);
Hist jetpt2("jet 2 pt", 200, 0.,2000.);
Hist jetpt3("jet 3 pt", 200, 0.,2000.);
Hist jetpt4("jet 4 pt", 200, 0.,2000.);
Hist jeteta1("jet 1 eta", 100, -5., 5.);
```

```
Hist jeteta2("jet 2 eta", 100, -5., 5.);
Hist jeteta3("jet 3 eta", 100, -5., 5.);
Hist jeteta4("jet 4 eta", 100, -5., 5.);
Hist jetphi1("jet 1 phi", 100, -M_PI, M_PI);
Hist jetphi2("jet 2 phi", 100, -M_PI, M_PI);
Hist jetphi3("jet 3 phi", 100, -M_PI, M_PI);
Hist jetphi4("jet 4 phi", 100, -M_PI, M_PI);
Hist deltaR("delta R between jet1 and jet2", 100, 0., 10.);
Hist Mjj("dijet invariant mass", 300, 0., 3000.);
Hist Mjjj("three-jet invariant mass", 300, 0., 3000.);
Hist MET("missing pT", 300, 0., 3000.);
Hist MET_nu("neutrino vector sum pT", 300, 0., 3000.);

for (int iEvent = 0; iEvent < nEvent; ++iEvent) {
    pythia.next();
    if (iEvent < 10) {
        pythia.process.list();
        event.list();
    }

    int nlep = 0;
    double lep_pt[10] = {0.};
    double lep_px[10] = {0.};
    double lep_py[10] = {0.};
    double lep_pz[10] = {0.};
    double lep_e[10] = {0.};
    int lepIndex[10] = {0};

    double nu_px = 0.;
    double nu_py = 0.;

    for (int il=0; il<event.size(); ++il){
        if (event[il].isFinal()) {
            if (abs(event[il].eta()) < etaMaxLep) {
                if (event[il].idAbs() == 11 || event[il].idAbs() == 13){
                    if (event[il].pT() > 20.0){
                        lep_pt[nlep]=event[il].pT();
                        lep_px[nlep]=event[il].px();
                        lep_py[nlep]=event[il].py();
                        lep_pz[nlep]=event[il].pz();
```

```
                        lep_e[nlep]=event[il].e();
                        lepIndex[nlep] = il;
                        ++nlep;
                    }
                }
            }
            if (event[il].idAbs() == 12 || event[il].idAbs() == 14 ||
event[il].idAbs() == 16){
                nu_px += event[il].px();
                nu_py += event[il].py();
            }
        }
    }

    lepmul.fill(nlep);
    if (nlep >= 1){
        leppt1.fill(lep_pt[0]);
        lepeta1.fill(event[lepIndex[0]].eta());
        lepphi1.fill(event[lepIndex[0]].phi());
    }
    if (nlep >= 2){
        leppt2.fill(lep_pt[1]);
        lepeta2.fill(event[lepIndex[1]].eta());
        lepphi2.fill(event[lepIndex[1]].phi());
    }

    int njet = 0;
    fjInputs.resize(0);
    for (int ifj=0; ifj<event.size(); ++ifj){
        if (event[ifj].isFinal() && abs(event[ifj].eta()) < etaMax){
            if (event[ifj].isHadron() || event[ifj].idAbs()==22){
                PseudoJet particleTemp = event[ifj];
                fjInputs.push_back( particleTemp );
            }
        }
    }

    vector <PseudoJet> inclusiveJets, sortedJets;
    ClusterSequence clustSeq(fjInputs, jetDef);
    inclusiveJets = clustSeq.inclusive_jets(pTMin);
```

```
    sortedJets    = sorted_by_pt(inclusiveJets);
    njet = sortedJets.size();
    jetmul.fill(njet);

    double j_pt[20] = {0.};
    double j_px[20]={0.};
    double j_py[20]={0.};
    double j_pz[20]={0.};
    double j_e[20]={0.};

    for (int ij=0; ij<njet; ++ij){
        j_pt[ij] = sortedJets[ij].pt();
        j_px[ij]=sortedJets[ij].px();
        j_py[ij]=sortedJets[ij].py();
        j_pz[ij]=sortedJets[ij].pz();
        j_e[ij]=sortedJets[ij].e();
    }

    if (njet >= 1) {
        jetpt1.fill(j_pt[0]);
        jeteta1.fill(sortedJets[0].eta());
        jetphi1.fill(wrapPhi(sortedJets[0].phi()));
    }
    if (njet >= 2) {
        jetpt2.fill(j_pt[1]);
        jeteta2.fill(sortedJets[1].eta());
        jetphi2.fill(wrapPhi(sortedJets[1].phi()));
        double dEta = sortedJets[0].eta() - sortedJets[1].eta();
        double dPhi = fabs(wrapPhi(sortedJets[0].phi()) -
wrapPhi(sortedJets[1].phi()));
        if (dPhi > M_PI) dPhi = 2 * M_PI - dPhi;
        double dR = sqrt(dEta*dEta + dPhi*dPhi);
        deltaR.fill(dR);
    }
    if (njet >= 3) {
        jetpt3.fill(j_pt[2]);
        jeteta3.fill(sortedJets[2].eta());
        jetphi3.fill(wrapPhi(sortedJets[2].phi()));
    }
    if (njet >= 4) {
```

```
        jetpt4.fill(j_pt[3]);
        jeteta4.fill(sortedJets[3].eta());
        jetphi4.fill(wrapPhi(sortedJets[3].phi()));
    }

    if (njet >= 2){
        for (int i=0; i<njet; ++i){
            for (int j=i+1; j<njet; ++j){
                double pxsum = j_px[i] + j_px[j];
                double pysum = j_py[i] + j_py[j];
                double pzsum = j_pz[i] + j_pz[j];
                double esum = j_e[i] + j_e[j];
                double mjj =
sqrt(pow2(esum)-pow2(pxsum)-pow2(pysum)-pow2(pzsum));
                Mjj.fill(mjj);
            }
        }
    }

    if (njet >= 3){
        for (int i=0; i<njet; ++i){
            for (int j=i+1; j<njet; ++j){
                for (int k=j+1; k<njet; ++k){
                    double px = j_px[i] + j_px[j] + j_px[k];
                    double py = j_py[i] + j_py[j] + j_py[k];
                    double pz = j_pz[i] + j_pz[j] + j_pz[k];
                    double E  = j_e[i]  + j_e[j]  + j_e[k];
                    double m3j = sqrt(pow2(E) - pow2(px) - pow2(py) -
pow2(pz));
                    Mjjj.fill(m3j);
                }
            }
        }
    }

    double pxvis = 0.;
    double pyvis = 0.;
    for (int i=0; i<nlep; ++i){
        pxvis += lep_px[i];
        pyvis += lep_py[i];
```

```cpp
    }
    for (int i=0; i<njet; ++i){
        pxvis += j_px[i];
        pyvis += j_py[i];
    }
    MET.fill(sqrt(pow2(pxvis)+pow2(pyvis)));
    MET_nu.fill(sqrt(pow2(nu_px) + pow2(nu_py)));
}

HistPlot hpl("CPP_Assi1");

hpl.frame("lep_mul", "lepton multiplicity", "$n_l$"); hpl.add(lepmul);
hpl.plot(true);
hpl.frame("lep_pt", "lepton pT", "$p_T$"); hpl.add(leppt1);
hpl.add(leppt2); hpl.plot(true);
hpl.frame("lep_eta", "lepton eta", "$\\eta$"); hpl.add(lepeta1);
hpl.add(lepeta2); hpl.plot(true);
hpl.frame("lep_phi", "lepton phi", "$\\phi$"); hpl.add(lepphi1);
hpl.add(lepphi2); hpl.plot(true);

hpl.frame("jet_mul", "jet multiplicity", "$n_j$"); hpl.add(jetmul);
hpl.plot(true);
hpl.frame("jet_pt", "jet pT", "$p_T$"); hpl.add(jetpt1); hpl.add(jetpt2);
hpl.add(jetpt3); hpl.add(jetpt4); hpl.plot(true);
hpl.frame("jet_eta", "jet eta", "$\\eta$"); hpl.add(jeteta1);
hpl.add(jeteta2); hpl.add(jeteta3); hpl.add(jeteta4); hpl.plot(true);
hpl.frame("jet_phi", "jet phi", "$\\phi$"); hpl.add(jetphi1);
hpl.add(jetphi2); hpl.add(jetphi3); hpl.add(jetphi4); hpl.plot(true);
hpl.frame("deltaR", "Delta R between jets", "$\\Delta R$");
hpl.add(deltaR); hpl.plot(true);

hpl.frame("jet_mjj", "dijet invariant mass", "$M_{jj}$"); hpl.add(Mjj);
hpl.plot(true);
hpl.frame("Mjjj", "three-jet invariant mass", "$M_{jjj}$"); hpl.add(Mjjj);
hpl.plot(true);
hpl.frame("met", "MET", "MET"); hpl.add(MET); hpl.plot(true);
hpl.frame("met_nu", "MET from neutrinos", "MET (neutrino sum)");
hpl.add(MET_nu); hpl.plot(true);

pythia.stat();
```

```
return 0;
}
```

- Codes used for Q2

```
#include "Pythia8/Pythia.h"
#include "Pythia8Plugins/FastJet3.h"
#include "fastjet/PseudoJet.hh"
#include "fastjet/ClusterSequence.hh"
#include <iostream>
#include <sstream>
#include <fstream>
#include <cmath>
#include <numeric>

using namespace Pythia8;
using namespace std;
using namespace fastjet;

int main(){

int nEvent=10000;
double lumi=1000.;

double R        = 0.5;
double pTMin    = 20.0;
double etaMax   = 4.5;
double etaMaxLep = 2.5;

Pythia pythia;
Event& event = pythia.event;

pythia.readString("Beams:frameType=1");
pythia.readString("Beams:idA = 2212");
pythia.readString("Beams:idB = 2212");
pythia.readString("Beams:eCM = 14000.");

pythia.readString("HiggsSM:gg2H=on");

pythia.readString("PartonLevel:FSR = on");
pythia.readString("PartonLevel:ISR = on");
```

```
pythia.readString("PartonLevel:MPI = off");

pythia.readString("HadronLevel:Hadronize = on");
pythia.readString("HadronLevel:Decay = on");

pythia.readString("SigmaProcess:renormScale2 = 4");
pythia.readString("SigmaProcess:factorScale2 = 4");

pythia.readString("Next:numberCount = 100 ");

pythia.readString("24:onMode = off");
pythia.readString("24:onIfMatch = 11 -12");
pythia.readString("24:onIfMatch = 13 -14");
pythia.readString("24:onIfMatch = 15 -16");
pythia.readString("24:onIfMatch = -11 12");
pythia.readString("24:onIfMatch = -13 14");
pythia.readString("24:onIfMatch = -15 16");

fastjet::JetDefinition jetDef(fastjet::antikt_algorithm, R);
std::vector <fastjet::PseudoJet> fjInputs;

pythia.init();

Hist jetmul("jet multiplicity", 20,0.,10.);
Hist lepmul("lepton multiplicity", 20, 0., 10.);

Hist leppt1("lep 1 pt", 200, 0.,2000.);
Hist leppt2("lep 2 pt", 200, 0.,2000.);
Hist lepeta1("lep 1 eta", 100, -5., 5.);
Hist lepeta2("lep 2 eta", 100, -5., 5.);
Hist lepphi1("lep 1 phi", 100, -M_PI, M_PI);
Hist lepphi2("lep 2 phi", 100, -M_PI, M_PI);

Hist jetpt1("jet 1 pt", 200, 0.,2000.);
Hist jetpt2("jet 2 pt", 200, 0.,2000.);
Hist jetpt3("jet 3 pt", 200, 0.,2000.);
Hist jetpt4("jet 4 pt", 200, 0.,2000.);
Hist jeteta1("jet 1 eta", 100, -5., 5.);
Hist jeteta2("jet 2 eta", 100, -5., 5.);
Hist jeteta3("jet 3 eta", 100, -5., 5.);
```

```
Hist jeteta4("jet 4 eta", 100, -5., 5.);
Hist jetphi1("jet 1 phi", 100, -M_PI, M_PI);
Hist jetphi2("jet 2 phi", 100, -M_PI, M_PI);
Hist jetphi3("jet 3 phi", 100, -M_PI, M_PI);
Hist jetphi4("jet 4 phi", 100, -M_PI, M_PI);

Hist deltaR("delta R between jet1 and jet2", 100, 0., 10.);
Hist Mjj("dijet invariant mass", 300, 0., 3000.);
Hist Mjjj("three-jet invariant mass", 300, 0., 3000.);
Hist MET("missing pT", 300, 0., 3000.);


pythia.readString("25:onMode = off");
pythia.readString("25:onIfMatch = 23 23");
pythia.readString("23:onMode = off");
pythia.readString("23:onIfMatch = 11 -11");
pythia.readString("23:onIfMatch = 13 -13");
pythia.readString("23:onIfMatch = 15 -15");

Hist fourlep_mass("4-lepton invariant mass", 200, 0., 2000.);
Hist lepmul4l("lepton multiplicity (all leptons)", 20, 0., 20);


for (int iEvent = 0; iEvent < nEvent; ++iEvent) {
    pythia.next();

    if (iEvent < 10) {
        pythia.process.list();
        event.list();
    }

    int nlep = 0;
    double lep_pt[10] = {0.};
    double lep_px[10] = {0.};
    double lep_py[10] = {0.};
    double lep_pz[10] = {0.};
    double lep_e[10] = {0.};

    for (int il=0; il<event.size(); ++il){
        if (event[il].isFinal() && abs(event[il].eta()) < etaMaxLep) {
```

```
        if (event[il].idAbs() == 11 || event[il].idAbs() == 13){
            if (event[il].pT() > 20.0){

                lep_pt[nlep]=event[il].pT();
                lep_px[nlep]=event[il].px();
                lep_py[nlep]=event[il].py();
                lep_pz[nlep]=event[il].pz();
                lep_e[nlep]=event[il].e();

                ++nlep;
            }
        }
    }
}

lepmul.fill(nlep);
lepmul4l.fill(nlep);

if (nlep >= 1){
    leppt1.fill(lep_pt[0]);
    lepeta1.fill(event[0].eta());
    lepphi1.fill(event[0].phi());
}
if (nlep >= 2){
    leppt2.fill(lep_pt[1]);
    lepeta2.fill(event[1].eta());
    lepphi2.fill(event[1].phi());
}


if (nlep >= 4){
    double px = lep_px[0] + lep_px[1] + lep_px[2] + lep_px[3];
    double py = lep_py[0] + lep_py[1] + lep_py[2] + lep_py[3];
    double pz = lep_pz[0] + lep_pz[1] + lep_pz[2] + lep_pz[3];
    double E  = lep_e[0] + lep_e[1] + lep_e[2] + lep_e[3];
    double m4l = sqrt(pow2(E) - pow2(px) - pow2(py) - pow2(pz));
    fourlep_mass.fill(m4l);
}
```

```
HistPlot hpl("CPP_Assi1");

hpl.frame("lep_mul", "lepton multiplicity", "$n_l$"); hpl.add(lepmul);
hpl.plot(true);
hpl.frame("lep_mul_4l", "lepton multiplicity (all leptons)", "$n_l$");
hpl.add(lepmul4l); hpl.plot(true);
hpl.frame("lep_pt", "lepton pT", "$p_T$"); hpl.add(leppt1);
hpl.add(leppt2); hpl.plot(true);
hpl.frame("lep_eta", "lepton eta", "$\\eta$"); hpl.add(lepeta1);
hpl.add(lepeta2); hpl.plot(true);
hpl.frame("lep_phi", "lepton phi", "$\\phi$"); hpl.add(lepphi1);
hpl.add(lepphi2); hpl.plot(true);

hpl.frame("jet_mul", "jet multiplicity", "$n_j$"); hpl.add(jetmul);
hpl.plot(true);
hpl.frame("jet_pt", "jet pT", "$p_T$"); hpl.add(jetpt1); hpl.add(jetpt2);
hpl.add(jetpt3); hpl.add(jetpt4); hpl.plot(true);
hpl.frame("jet_eta", "jet eta", "$\\eta$"); hpl.add(jeteta1);
hpl.add(jeteta2); hpl.add(jeteta3); hpl.add(jeteta4); hpl.plot(true);
hpl.frame("jet_phi", "jet phi", "$\\phi$"); hpl.add(jetphi1);
hpl.add(jetphi2); hpl.add(jetphi3); hpl.add(jetphi4); hpl.plot(true);
hpl.frame("deltaR", "Delta R between jets", "$\\Delta R$");
hpl.add(deltaR); hpl.plot(true);

hpl.frame("jet_mjj", "dijet invariant mass", "$M_{jj}$"); hpl.add(Mjj);
hpl.plot(true);
hpl.frame("Mjjj", "three-jet invariant mass", "$M_{jjj}$"); hpl.add(Mjjj);
hpl.plot(true);
hpl.frame("met", "MET", "MET"); hpl.add(MET); hpl.plot(true);

hpl.frame("m4l", "4-lepton invariant mass", "$M_{4l}$");
hpl.add(fourlep_mass); hpl.plot(true);

pythia.stat();
return 0;
}
```

- Code for higgs to diphoton

```
#include "Pythia8/Pythia.h"
#include "Pythia8Plugins/FastJet3.h"
#include <iostream>
#include <cmath>

using namespace Pythia8;

int main() {
    int nEvent = 10000;
    double etaMax = 4.5;

    Pythia pythia;
    Event& event = pythia.event;


    pythia.readString("Beams:frameType = 1");
    pythia.readString("Beams:idA = 2212");
    pythia.readString("Beams:idB = 2212");
    pythia.readString("Beams:eCM = 14000.");


    pythia.readString("HiggsSM:gg2H = on");
    pythia.readString("25:onMode = off");
    pythia.readString("25:onIfMatch = 22 22");

    pythia.init();

    Hist nphotons("photon multiplicity", 20, 0., 20);
    Hist dipho_mass("diphoton invariant mass", 200, 0., 2000.);

    for (int iEvent = 0; iEvent < nEvent; ++iEvent) {
        if (!pythia.next()) continue;

        int npho = 0;
        double pho_px[10] = {0.}, pho_py[10] = {0.}, pho_pz[10] = {0.},
pho_e[10] = {0.};

        for (int i = 0; i < event.size(); ++i) {
            if (event[i].isFinal() && event[i].id() == 22 &&
```

```
                event[i].pT() > 20.0 && abs(event[i].eta()) < etaMax) {

                if (npho < 10) {
                    pho_px[npho] = event[i].px();
                    pho_py[npho] = event[i].py();
                    pho_pz[npho] = event[i].pz();
                    pho_e[npho]  = event[i].e();
                    ++npho;
                }
            }
        }

        nphotons.fill(npho);

        if (npho >= 2) {
            double px = pho_px[0] + pho_px[1];
            double py = pho_py[0] + pho_py[1];
            double pz = pho_pz[0] + pho_pz[1];
            double E  = pho_e[0] + pho_e[1];
            double mgg = sqrt(pow2(E) - pow2(px) - pow2(py) - pow2(pz));
            dipho_mass.fill(mgg);
        }
    }


    HistPlot hpl("diphotonPlots");
    hpl.frame("nphotons", "photon multiplicity", "$n_\\gamma$");
hpl.add(nphotons); hpl.plot(true);
    hpl.frame("dipho_mass", "diphoton invariant mass",
"$M_{\\gamma\\gamma}$"); hpl.add(dipho_mass); hpl.plot(true);

    pythia.stat();
    return 0;
}
```