

Contents

1	Introduction	1
1.1	Installing Herwig-7	1
2	Generating events:	1
2.1	The Partonic Level Events:	1
2.2	Showering:	2
3	Reading the root files:	3
4	Results:	3

1 Introduction

The main GitHub link for this work is <https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER>. The latest version of this document can always be found at “https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/blob/master/HERWIG_PART/README.pdf”. The sources for this document (including the figures in case you want to use them anywhere) can be found at “https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/tree/master/HERWIG_PART/Documentation”.

1.1 Installing Herwig-7

To install Herwig, please make sure you have a **proper** root installation (and source “`thisroot.sh`” if it is installed in non-standard location) as Herwig uses `libpyroot`. Download and install Herwig by running the script <https://herwig.hepforge.org/herwig-bootstrap>:

```
wget -c 'https://herwig.hepforge.org/herwig-bootstrap'
chmod +x './herwig-bootstrap'
'./herwig-bootstrap' './HerwigInstall'
```

This will install Herwig to the location ‘`./HerwigInstall`’ along with all of it’s dependencies (`boost`, `fastjet`, `HepMC`, etc). Full instructions for using this script is found in <https://herwig.hepforge.org/tutorials/installation/bootstrap.html>.

2 Generating events:

2.1 The Partonic Level Events:

The quickest way I could find to generate events in Herwig was by generating parton level events in MadGraph and shower using Herwig (I also got suggestions to adopt similar method for Pythia as this nicely isolates matrix element generation from showering and our main aim is the study effect of different showering and hadronization techniques on our jet substructure algorithms).

```
wget -c 'https://launchpad.net/mg5amcnlo/2.0/2.6.x/+download/MG5_aMC_v2.6.1.tar.gz';
tar -xf "MG5_aMC_v2.6.1.tar.gz";
cd "MG5_aMC_v2_6_1/";
./bin/mg5_aMC
```

This will start madgraph, now you can generate LHE files:

```
generate p p > z j
output zj
exit
```

Once you are done with above steps, MadGraph exits, now you will need to edit some configuration files:

```
cd zj/Cards/  
<editor> 'run_card.dat'
```

Where, <editor> stands for any text editor. You will need to change the following in the file:

```
0.0 = ptheavy ! minimum pt for at least one heavy final  
state
```

To the value desired, we will pick this to be 200 GeV.

```
200.0 = ptheavy ! minimum pt for at least one heavy final  
state
```

Next edit madspin_card_default.dat and change the line:

```
decay z > all all
```

to:

```
decay z > ta- ta+
```

now madgraph is set up and can be run, in the shell:

```
cd ../;  
./bin/generate_events
```

Once you get the MadGraph prompt:

```
4  
0  
0
```

Now note the “4” is important, you will need to run with MadSpin (to force $Z \rightarrow \tau\tau$ decays) once you are done, you will have the LHE files in:

```
./Events/run_01_decayed_1/unweighted_events.lhe.gz
```

you can extract this:

```
gzip -d "./Events/run_01_decayed_1/unweighted_events.lhe.gz"
```

The complete list of LHE files generated using the above methods can be downloaded from “<https://drive.google.com/file/d/1sVQSYUDPqSB0vgh84Y8II0ucBSMMr9rL/view?usp=sharing>”

2.2 Showering:

Move the LHE file to another directory where you want to do the showering:

```
mkdir ~/shower;  
cp "./Events/run_01_decayed_1/unweighted_events.lhe" ~/shower/1.lhe
```

now you will need to activate Herwig (note, <HerwigInstall> needs to be replaced by the correct directory):

```
source <HerwigInstall>/bin/activate
```

Once this is done, download the file “https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/blob/master/HERWIG_PART/LHEwithISR.in” to the current directory (~/.shower). In LHE.in, edit the line:

```
set LesHouchesReader:FileName unweighted_events.lhe
```

and change “unweighted_events.lhe” to the LHE file you want to shower. You can also edit the PDF choices if you want (note that you may need to download the appropriate pdf fits manually using `lhpdf` installed in the `<HerwigInstall>` directory).

note that the lines

```
read snippets/HepMC.in
set /Herwig/Analysis/HepMC:PrintEvent 100
```

is responsible for producing the HepMC file, you can then use Delphes on these files. We are done. Now run Herwig:

```
Herwig read LHE.in
Herwig run LHE.in
```

You can change underlying event, tunes, etc in the LHE.in file, explore <https://herwig.hepforge.org/tutorials/faq/shower.html> for details.

By default, Herwig does not force τ to decay hadronically. This can be forced, check the link: <https://herwig.hepforge.org/tutorials/faq/decay.html> to know how this can be done. Basically you would need:

```
set /Herwig/Particles/tau+/tau+>nu_taubar,nu_e,e+;:OnOff Off
set /Herwig/Particles/tau+/tau+>nu_taubar,nu_mu,mu+;:OnOff Off
set /Herwig/Particles/tau-/tau->nu_tau,nu_ebar,e-;:OnOff Off
set /Herwig/Particles/tau-/tau->nu_tau,nu_mubar,mu-;:OnOff Off
```

A sample card to run with (https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/blob/master/HERWIG_PART/LHEWithISR.in) and without (https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/blob/master/HERWIG_PART/LHENoISRMPI.in) mpi is available in the GitHub page.

Example for an extremely simple reader is available in <https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/tree/master/COMMON/DelphesReader>.

3 Reading the root files:

The main reader program can be found at “<https://github.com/aravindhv10/MUMBAIPUNETAUTAGGER/tree/master/COMMON/DelphesReader>”.

4 Results:

We use the following color convention for graphs through out the document:

$$\begin{aligned}
&pp \rightarrow jZ, (Z \rightarrow \tau\bar{\tau}), p_T^Z \geq 200 \text{ GeV} \\
&pp \rightarrow jZ, (Z \rightarrow \nu_\tau\bar{\nu}_\tau), p_T^Z \geq 200 \text{ GeV} \\
&pp \rightarrow jZ, (Z \rightarrow b\bar{b}), p_T^Z \geq 200 \text{ GeV} \\
&pp \rightarrow jZ, (Z \rightarrow \tau\bar{\tau}), p_T^Z \geq 0 \text{ GeV}
\end{aligned}$$

We present some simple results obtained using jet substructure variables on jets tagged using BDRS+Filtering methods (Note: we use the final 2 step filtered (BDRS+filtering) jet to evaluate variables and all of these results are for events with MPI enabled). Note that planar flow [Figure 5, Figure 6] seem to work extremely well.

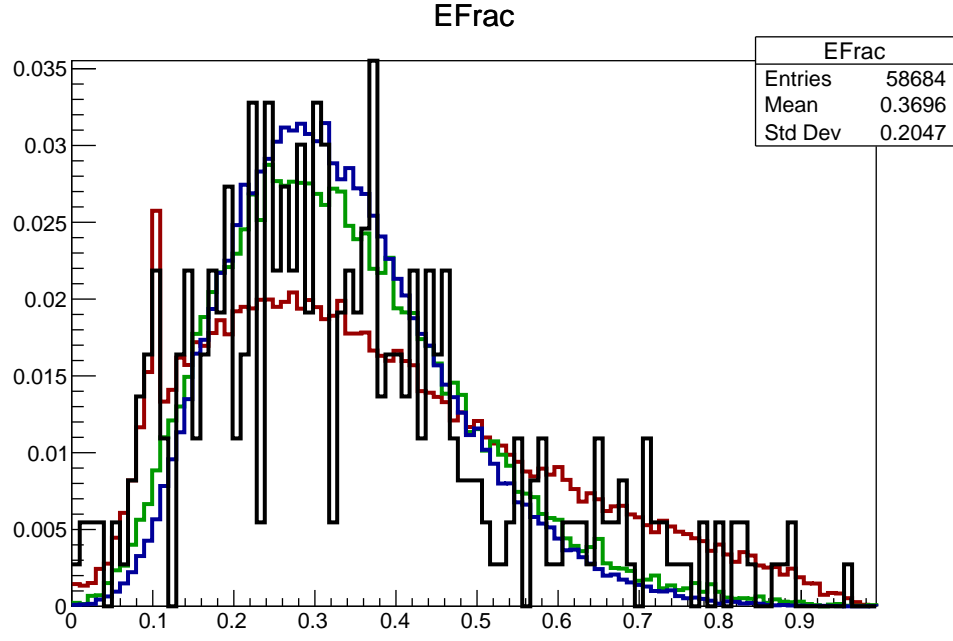


Figure 1: Electromagnetic energy fraction of the jet, normalized to unit area under the curve.

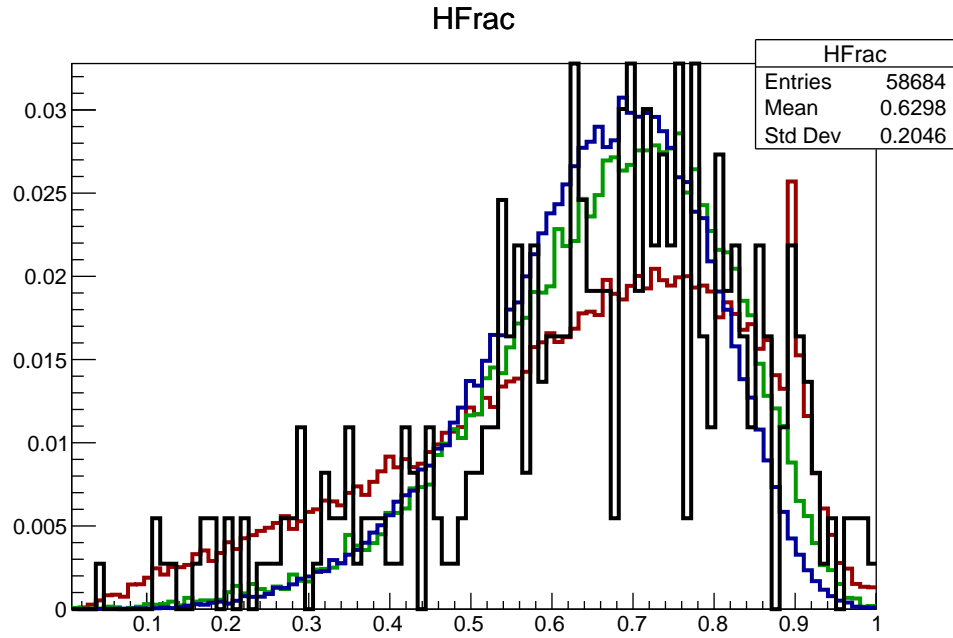


Figure 2: Hadronic energy fraction of the jet, normalized to unit area under the curve.

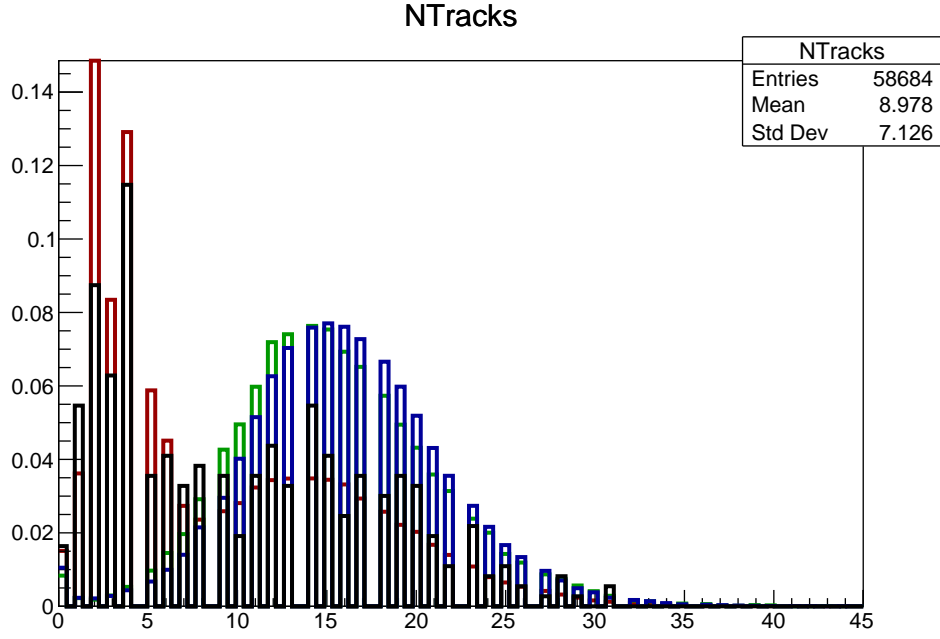


Figure 3: Number of tracks in the jet, normalized to unit area under the curve.

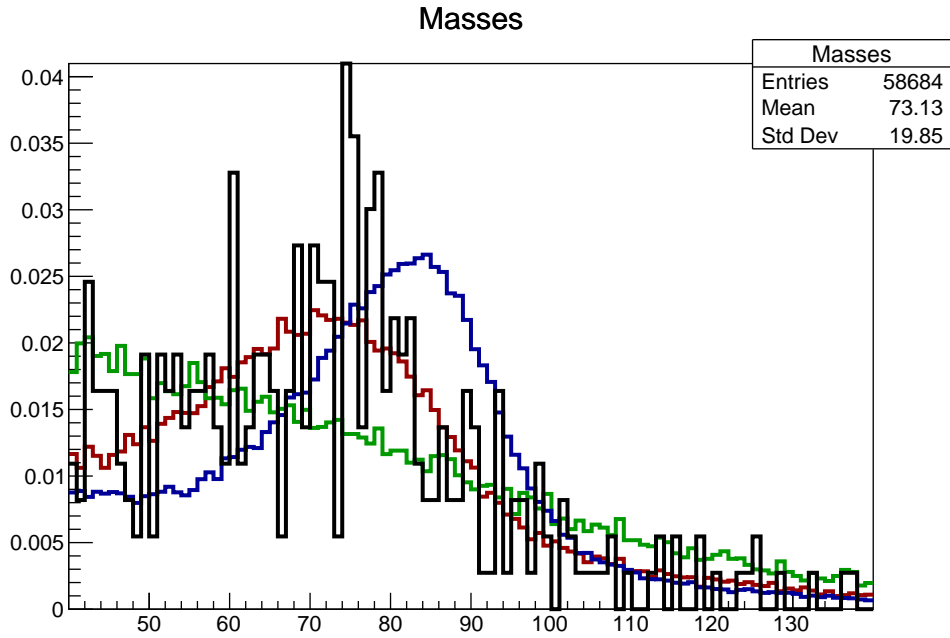


Figure 4: Masses of the jet after BDRS and filtering steps, normalized to unit area under the curve.

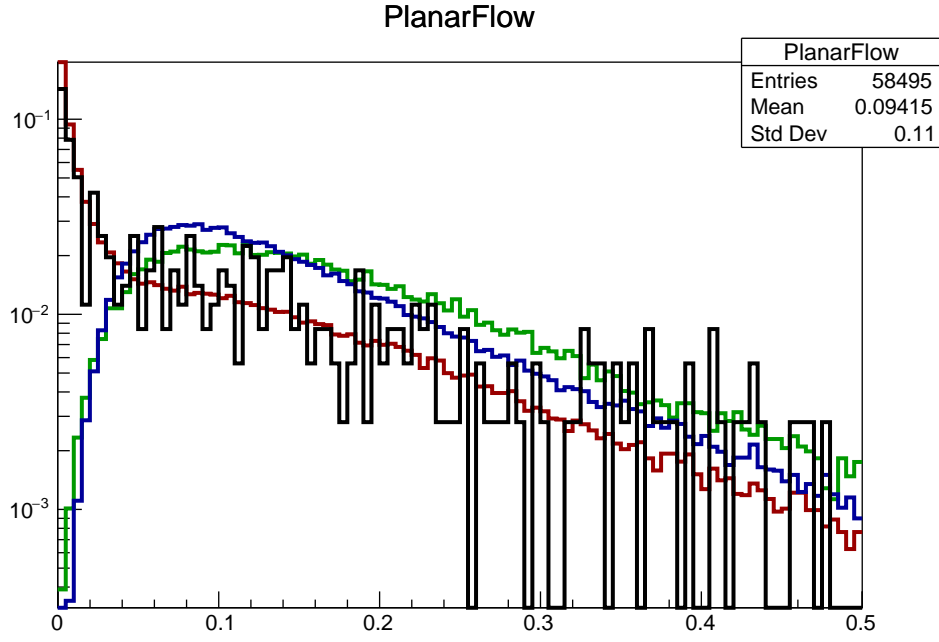


Figure 5: Planar Flow of the jet after BDRS and filtering steps, normalized to unit area under the curve.

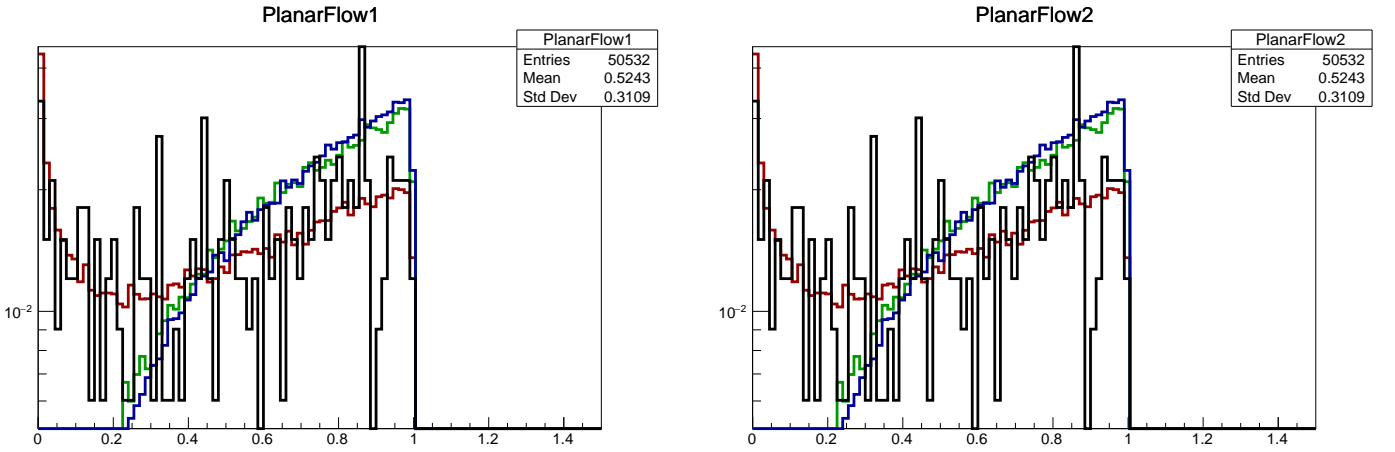


Figure 6: Planar Flow of the two subjects after BDRS and filtering steps, normalized to unit area under the curve.

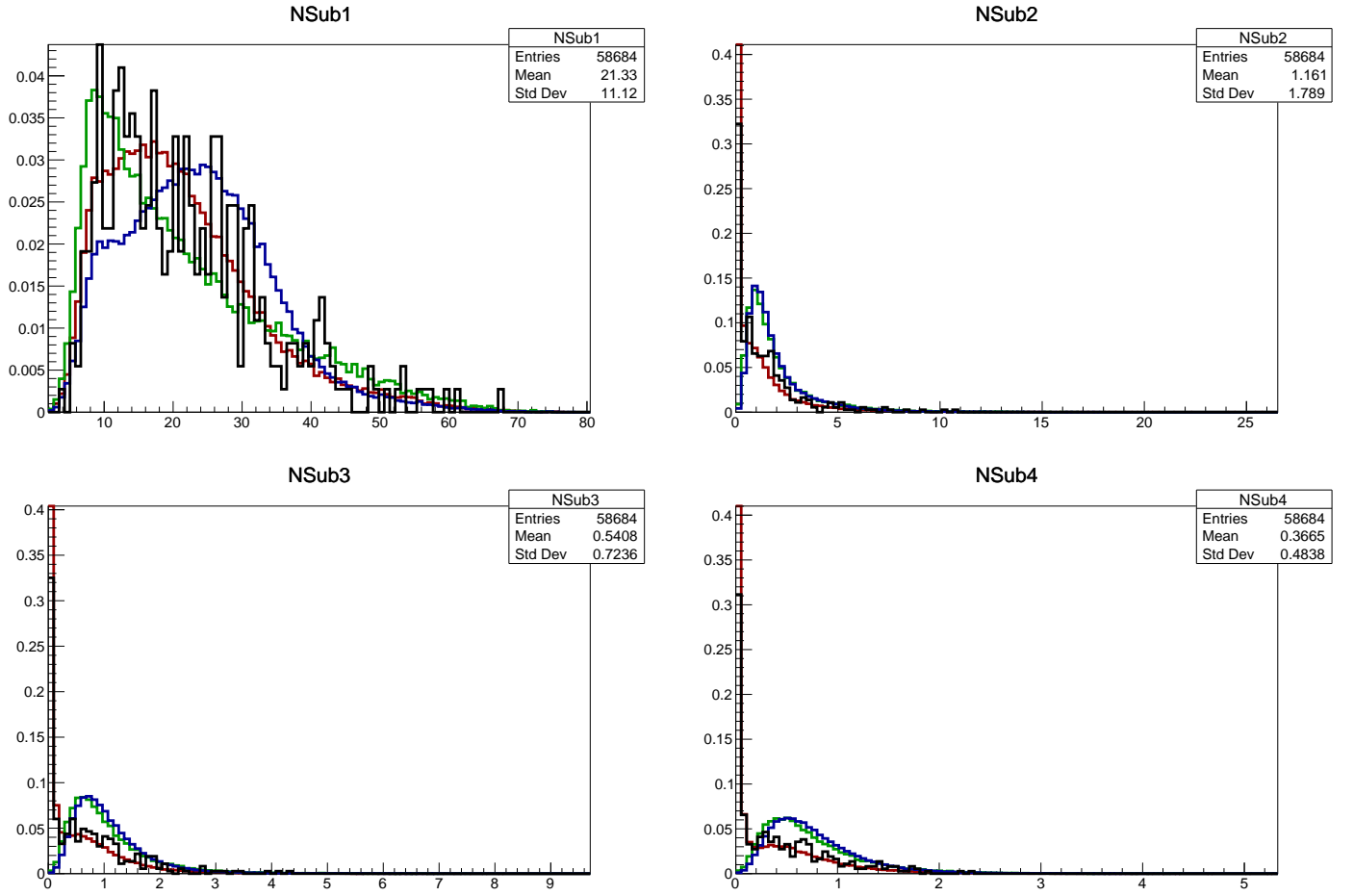


Figure 7: NSubJettiness observable, normalized to unit area under the curve.

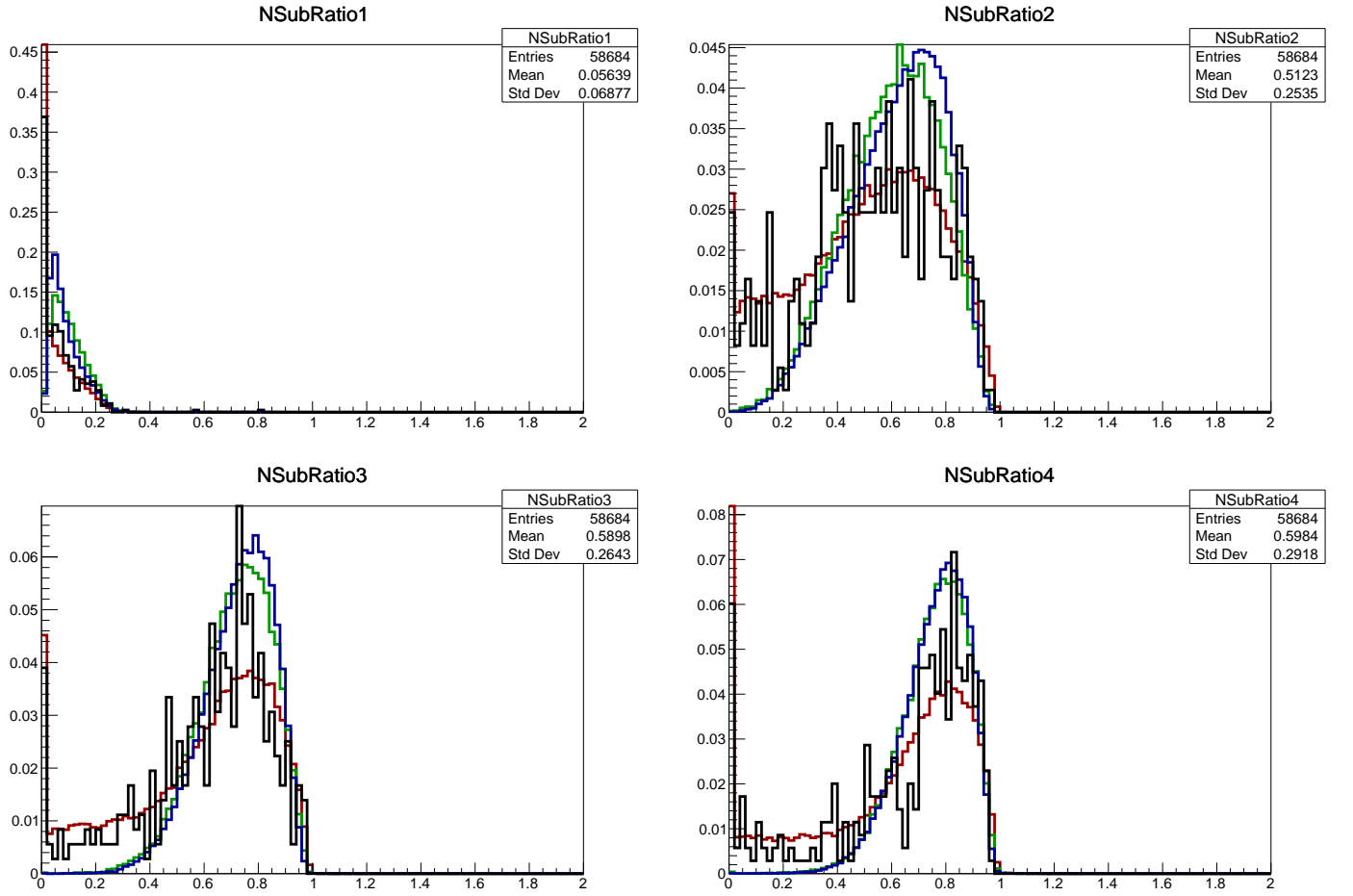


Figure 8: Ratio of NSubJettiness observable, normalized to unit area under the curve.