

Advanced Heavy Flavor Tagging Using a Neural Network

H. Bachacou, P. Lujan, M. McFarlane, W. Yao

(Lawrence Berkeley National Laboratory)

Abstract

This note describes the application of neural networks to heavy flavor tagging using the existing taggers SecVtx, JetProb, and other jet kinematic variables. Emphasis is on increasing the b -tagging efficiency while reducing contamination from c and l jets. Among other options, the tagger provides 90% b efficiency while rejecting more than half of charm and light quark jets for SecVtx-tagged events. The performance of the improved b -tagged has been measured using the inclusive jet and inclusive electron data sample and finds to be in good agreement with Monte Carlo expectation.

1 Introduction

Run 2 has inspired several b -tagging algorithms: SecVtx identifies secondary vertices from the decay of long lifetime hadrons [1]; JetProb evaluates a jet lifetime probability based on the impact parameter of the tracks within the jet [2]; Soft Lepton Taggers identify leptons coming from the semileptonic decay of b hadrons [3]. Recent work has combined these algorithms by adding individual tags, but not aggregating information into a more sophisticated tagger. Presented herein is an attempt to combine the information provided by each algorithm, plus other kinematic quantities, to improve tagging performance. This new, neural network [4] based tagger distinguishes bottom (b) from charm (c) and light flavor (l) quark jets.

In the future we hope this tagger plus other tagger(s) also in progress, for instance, Karlsruhe's [5], plus the existing tagging methods will be combined into one larger tagger offering performance beyond each tool alone.

2 Data Samples

The networks are trained and tested using b , c and l jets from $t\bar{t}$ Monte Carlo samples: `ttoph1`, `ttopel`, `ttop*h`. Partons from the top and from the W decay are matched to the closest jet with $E_T > 15$ GeV at Level 4, $|\eta| < 2$ and $\Delta R < 0.4$. This results in a series of files of various classifications (untagged, tagged, positive, negative, etc.) of b , c and l jets ready for neural network use. The performance has measured using the various inclusive jet data and Monte Carlo samples: `gjt20d`, `gjt{1-3}0h` and `{j,g}qcd2f`, as well as the 8 GeV inclusive electron samples that both jets are b -tagged by SecVtx (Data: `blpc0d`, `blpc0h`; MC: `btop5a-8a`).

3 Properties of Jets with a Positive SecVtx Tag

We train our networks on jets positively tagged by SecVtx: the secondary vertex provides good handles to discriminate between b , c , and l jets. Figures 1 through 3 show the η , E_T and p_T distributions of these b , c , and l jets, Figures 4 through 19 show the sixteen variables accessed by the neural networks, which consist of 8 SecVtx tag related variables plus 8 untagged variables listed below. They are chosen primarily based on the characteristics of b quark decay that has higher track multiplicity, larger invariant mass, longer lifetime, and harder fragmentation function than charm and light quark. The mass-energy-momenta quantities, track quantities and decay length significance are good discriminates for b jets. The p_T ratio quantities are useful in identifying l jet, while c are more difficult to segregate; pseudo- $c\tau$ and χ^2 are their best discriminators. We also investigated the net performance by adding additional variables, but the improvement seems negligible.

Figures 4 to 11 show the properties of the SecVtx tag:

- Figure 4: Number of tracks
- Figure 5: Transverse decay length
- Figure 6: Transverse decay length significance
- Figure 7: χ^2
- Figure 8: Pseudo- $c\tau$ ($= L_{2D} \times M_{\text{SecVtx}}/p_T^{\text{SecVtx}}$)

- Figure 9: Vertex Mass
- Figure 10: Vertex p_T divided by the sum of the p_T of the good tracks in the jet
- Figure 11: Pass 1 or 2

We also investigated variables independent of the SecVtx vertex, Figures 12 through 19:

- Figure 12: Reconstructed mass of Pass 1 tracks
- Figure 13: Reconstructed mass of Pass 2 tracks
- Figure 14: Number of Pass 1 tracks
- Figure 15: Number of Pass 2 tracks
- Figure 16: p_T of Pass 1 tracks divided by jet p_T
- Figure 17: p_T of Pass 2 tracks divided by jet p_T
- Figure 18: Number of good tracks
- Figure 19: JetProb probability

4 The Neural Net

We use the JETNET [7] interface provided by Ohio State University [8], available at: http://www.physics.ohio-state.edu/~catutza/root_to_jetnet/rtj.html.

We distinguish two cases:

- The jet already contains a positive or negative SecVtx tag. In this case we focus on trying to separate b jets from charm and light.
- The jet is “taggable” (at least two good tracks), but does not contain a (positive) SecVtx tag. Here we will recover some b -tagging efficiency by separating light flavor from heavy flavor jets. This work is in progress.

The tagger employs two neural networks in series: one which is trained to separate b from l jets, and the other, b from c . Jets must satisfy one output cut for each network to be accepted by the b -tagger. A large pure sample of each species, derived from $t\bar{t}$ Monte Carlo, are selected via HEPG matching within a cone of 0.4, for the network training.

The b - l network is trained with 3600 tagged l jets, of which 1000 are negatively tagged and the balance positive. It is trained with 1000 epochs, ten hidden nodes and one output node. Jet parameters which are uniquely negative for negatively tagged events are converted to positive values. Training on both positively and negatively tagged l jets allows the network to respond better to both.

There are various attributes of the neural network generation subject to user manipulation, including: training epochs, hidden nodes, learning momentum, temperature, pruning, and others. Fortunately, these are found not to introduce instabilities when varied over reasonable ranges. Figure 20 shows the network output assignments for all three jets, tested upon separate testing samples of 200000 b jets, 12500 c jets and 3600 light jets (JETNET is more effective when trained with equal size samples, not so when testing). Figure 21 shows the efficiency versus rejection of the network in separating equal numbers of b from l .

The b - c network is trained using a sample of 12500 b and c positively tagged jets. The same sixteen parameters are active, though, this next work uses twelve hidden nodes. The

network was tested upon the aforementioned samples, shown in Figure 22. Figure 23 shows the efficiency versus rejection of the network in separating b from c .

Once the network is in place, a large independent pure sample of each species, derived from $t\bar{t}$ Monte Carlo, selected via HEPG matching within a cone of 0.4, is run through the tagger for each combination of cuts: 33 for each network, 1089 combinations overall. The best results for a given b efficiency, in terms of average c and l rejection and balance thereof are recorded for use, some of which are described in Table 1. One can alternatively select settings that do not balance the rejection of c and l in favor of a biased rejection rate. This would be useful, for instance, if one were studying a sample with one background larger than the other. Appendix 7 shows the network settings that yield b efficiencies from 75% to 95%.

Figure 24 illustrates the simultaneous distributions of each jet over both networks. The top left, top right and bottom left plots show the designations given to b , l and c jets respectively by the b - l network on the x -axis and the b - c network on the y -axis. The remaining plot shows the distribution of b jets relative to the c and l , that is, $b/(b+c+l)$, but bears no quantitative influence upon the analysis. Viewing this plot, we feel comfortable in our decision to impose one cut per network rather than a more complicated two dimensional cut.

Figure 25 shows the scatter plot of b efficiency vs the simple average rejection of c and l for many different network setting. Figure 26 shows the variation of c and l rejections for a given b efficiency.

We also checked the b -tagger performances using the b , c and l jets, derived from Dijet QCD and $WH \rightarrow l\nu b\bar{b}$, selected via HEPG matching within a cone 0.4. The results are consistent each other in a wider range of kinematic for a given set of cuts, as shown in Table 1. There are still some residual difference probably due to imperfect matching at HEPG level or different kinematic distributions. We will treat these as a part of systematic as described later.

Currently, this study emphasizes the operating point of 90% b efficiency for positive tags. Later, we will explore neural networks for untagged jets, and will verify the systematics for the case of 80% efficiency.

b efficiency			c rejection		l rejection		NN- bl cut	NN- bc cut
wh	ttbar	qcd2f	ttbar	qcd2f	ttbar	qcd2f		
0.95	0.94	0.956	0.300	0.220	0.520	0.620	0.152	0.152
0.90	0.89	0.922	0.484	0.410	0.600	0.710	0.182	0.242
0.85	0.84	0.870	0.607	0.531	0.717	0.810	0.273	0.303
0.80	0.79	0.830	0.694	0.611	0.776	0.865	0.333	0.364

Table 1: c and l Rejection for a given b Efficiency:

5 Validation of b -tag Efficiency and Scale Factors

We investigate the Monte Carlo modeling of input variables using double tagged inclusive electron samples, which provide an excellent control sample of pure b . Plots of the input

parameters for away and electron jets, data and Monte Carlo, are displayed in Figures 27 through 34. The agreement is excellent, which gives us confidence of the b tagger. We measured the b tag efficiencies using double tagged inclusive electron samples. The tagger is applied to both e(lectron)-jets and a(way)-jets in data and Monte Carlo, shown in Table 2. Since e jets are more pure than a jets, even after double tags, we take the central value from e jets and the difference between two as its systematic uncertainty: $SF = 0.97 \pm 0.01$. Note that this is additional SF on top of the regular SecVtx b -tagger scale factor, which has 8% of uncertainties. Figures 35 through 38 demonstrate the network response, as well as acceptance as a function of E_T .

	Gen 6	Gen 5		
	Data	Monte Carlo	SF=Data/MC	
e jet	0.880 ± 0.012	0.874 ± 0.005	0.898 ± 0.006	0.973 ± 0.009
a jet	0.874 ± 0.012	0.855 ± 0.005	0.889 ± 0.006	0.964 ± 0.009

Table 2: Comparison of away and electron jets, data and Monte Carlo.

6 Validation and Mistag Rate with Jet Data.

Figures 39 through 57 show the distributions of SecVtx tagged jet from Jet 50 data (gjt20d) and Jet 40 Monte Carlo (jqcd2f). We lower the E_T cut to 40 GeV for the Monte Carlo to provide more statistics without altering the distributions. Figures 58 and 59 show their network outputs for positive and negative events respectively. There are more b 's in Monte Carlo in the positive tags, which is consistent with the expectation of more mistags in the data. The agreement on the negative tags is perfect since they are dominated by the light quarks.

Data and Monte Carlo agree regarding the rejection of negatively tagged light jets, and the various samples are consistent. Averaging the rejection rates for the data samples and assigning the differences, listed in Table 3, to the systematic yields 0.67 ± 0.03 as the rejection factor for negative tags. Studying the negatively tagged data provides better understanding of mistags, for which studying Monte Carlo alone would be inadequate.

Neg. Tagged l jets	Total	Passing NN cuts	Rejection Factor
gjt20d	9292	3110	0.665 ± 0.005
gjt10h	1968	666	0.662 ± 0.011
gjt20h	4572	1391	0.696 ± 0.007
gjt30h	8867	2725	0.693 ± 0.005
jqcd2f	957	312	0.674 ± 0.015
ttopkl	2724	824	0.698 ± 0.009
			0.67 ± 0.03

Table 3: Rejection factors for negatively tagged l jets

In order to check the rejection factor for positive l jets, we select pure light jets from `qcdf2f` sample by requiring no real heavy flavor jets in the events, while for the $t\bar{t}$ Monte Carlo, we select the light quark jets from W decay without heavy flavor within a cone of 0.7. Then we compare the rejection factors with positive and negative tags and find that they are satisfactorily symmetric. See Table 4. We take the average as the rejection factor for positive tags, and the difference as the systematic uncertainty: 0.65 ± 0.05 . Note that the systematic assigned here is additional to SecVtx mistag matrix systematic uncertainties including the positive negative asymmetry.

	Positive	Negative
gqcd2f	0.707 ± 0.013	0.709 ± 0.017
ttopkl	0.595 ± 0.012	0.700 ± 0.017
Σ	0.65 ± 0.05	

Table 4: Positively tagged l jet rejection from Monte Carlo.

Following that, we also investigated the possibility of reconstructing a rejection matrix using the negatively tagged light jets from the jet data sample `gjt20d` passing the b -tagger. The matrix was parameterized according to E_T and N_{tracks} , or E_T and pseudo $c\tau$. The results are shown in Table 5. The matrix was tested by passing jet data through it and comparing the recovered distributions of the individual parameters to their originals. The results were mixed based on Figures 60 through 62, which seems to lack a significant improvement over quoting a single rejection for a given cut.

7 Conclusion

We describe the application of neural networks to heavy flavor tagging using the existing taggers SecVtx, JetProb, and other jet kinematic variables. Emphasis is on increasing the b -tagging efficiency while reducing contamination from c and l jets. Among other options, the tagger provides 90% b efficiency while rejecting more than half of charm and light quark jets for SecVtx-tagged events. The performance of the improved b -tagged has been measured using the inclusive jet and inclusive electron data sample and finds to be in good agreement with Monte Carlo expectation.

In the near future, we will create another neural network tagger for application to untagged jets, and are optimistic about recovering b jets therefrom. Also, we will expand the systematic studies to cover the 80% b efficiency case. Later, we hope our efforts will be an integral component of the planned compound tagger.

References

- [1] C. Neu *et al.*, “SecVtx Optimization Studies for 5.3.3 Analysis”, CDF Note 7578.
- [2] A. Sukhanov *et al.*, “Efficiency of Jet Probability Heavy Flavor Tagger Used for 2005 Winter Conferences”, CDF Note 7444.

- [3] L. Cerrito and A. Taffard, “A Soft Muon Tagger for Run II: Summer-04 version”, CDF Note 7122.
- [4] <http://www.statsoftinc.com/textbook/stneunet.html>
- [5] M. Feindt *et al.*, “A Neural Network b Tagger for Single Top Analyses”, CDF Note 7816.
- [6] S. Lai and P. Sinervo, “A Combined Heavy Flavor Tagger for Gen5 Using SecVtx and JetProb Algorithms”, CDF Note 7437.
- [7] C. Peterson *et al.*, “JETNET 3.0 — A versatile artificial neural network package”; Computer Physics Communications v. 81 (1994) pp. 185-220
- [8] C. Ciobanu *et al.*, “A ROOT Interface to JETNET”, CDF Note 5434.

Given 90% b efficiency:

N_{tracks}	E_T :	≤ 45	45-75	75-105	105-400
≤ 4		0.656 ± 0.025	0.671 ± 0.0253	0.647 ± 0.082	0
4-6		0.422 ± 0.0143	0.413 ± 0.013	0.449 ± 0.0373	0.379 ± 0.0637
6-8		0.361 ± 0.0117	0.331 ± 0.00957	0.308 ± 0.0232	0.357 ± 0.0523
8-10		0.284 ± 0.012	0.272 ± 0.00882	0.333 ± 0.0222	0.333 ± 0.041
≥ 10		0.287 ± 0.0136	0.279 ± 0.0078	0.290 ± 0.0153	0.305 ± 0.0265
$c\tau$					
0-0.04		0.487 ± 0.0101	0.437 ± 0.00724	0.446 ± 0.0158	0.456 ± 0.029
0.04-0.08		0.364 ± 0.0108	0.296 ± 0.00788	0.262 ± 0.0181	0.222 ± 0.031
≥ 0.08		0.120 ± 0.00889	0.0957 ± 0.0066	0.0944 ± 0.0154	0.135 ± 0.0335

Given 80% b efficiency:

N_{tracks}	E_T :	≤ 45	45-75	75-105	105-400
≤ 4		0.35 ± 0.0251	0.347 ± 0.0256	0.353 ± 0.082	0
4-6		0.232 ± 0.0122	0.232 ± 0.0112	0.258 ± 0.0328	0.241 ± 0.0562
6-8		0.16 ± 0.00895	0.147 ± 0.00719	0.152 ± 0.018	0.143 ± 0.0382
8-10		0.112 ± 0.00841	0.108 ± 0.00614	0.133 ± 0.016	0.152 ± 0.0312
≥ 10		0.118 ± 0.00973	$0.095 \pm .$	0.1 ± 0.0102	0.113 ± 0.0182
$c\tau$					
0-0.04		0.236 ± 0.00862	0.199 ± 0.00583	0.191 ± 0.0125	0.19 ± 0.0229
0.04-0.08		0.167 ± 0.00837	0.122 ± 0.00565	0.116 ± 0.0132	0.1 ± 0.0224
≥ 0.08		$0.0419 \pm .$	$0.05490.0242 \pm 0.00345$	0.0278 ± 0.00866	0.0577 ± 0.0229

Table 5: Acceptance of negatively tagged l jets for given E_T and N_{tracks} or $c\tau$. For 90% b efficiency, the total acceptance rate for l is 0.334 ± 0.00349 ; for 80% b efficiency, the total acceptance is 0.148 ± 0.00262 .

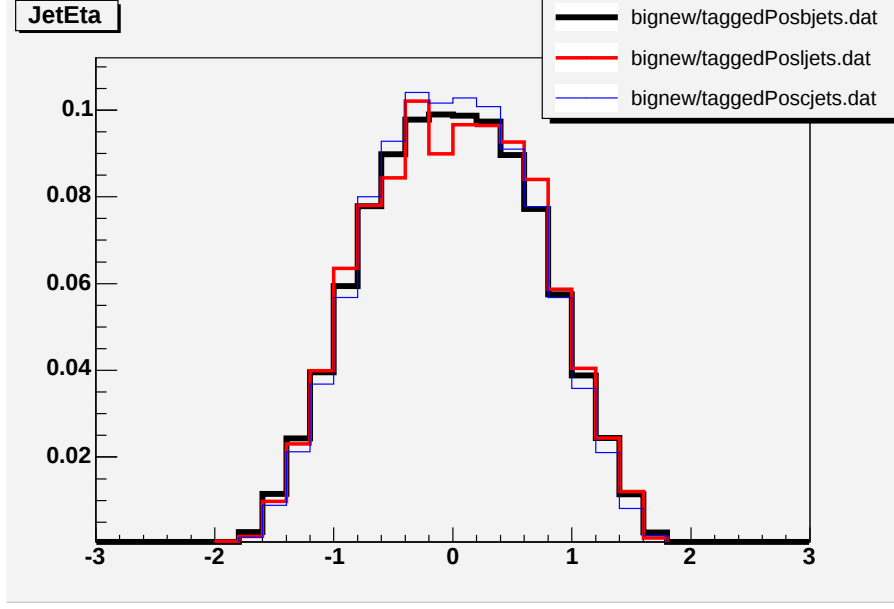


Figure 1: Pseudorapidity of SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

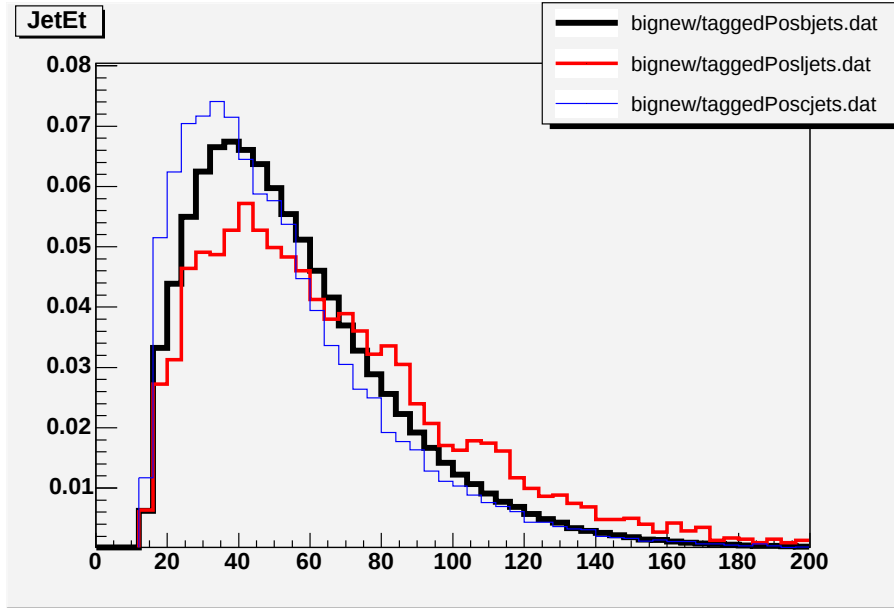


Figure 2: Transverse energy of SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

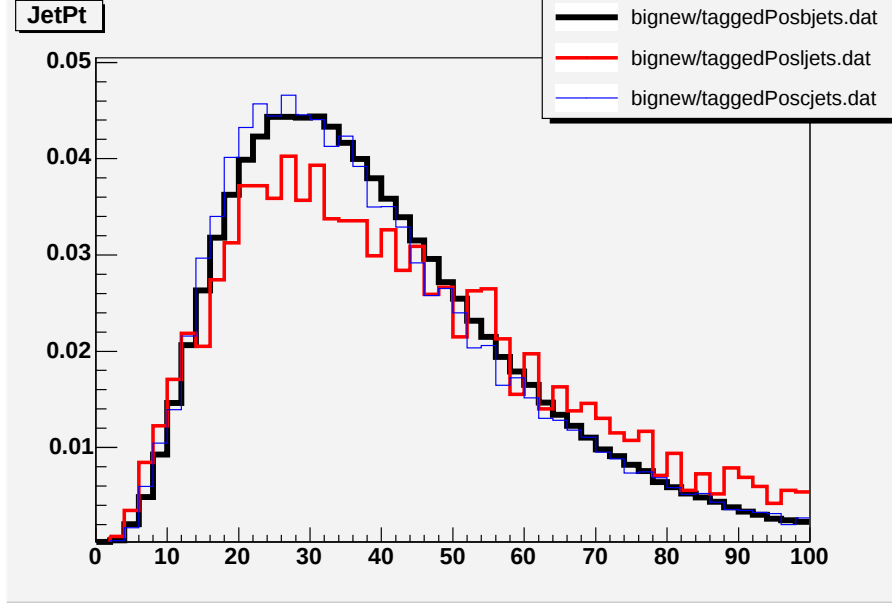


Figure 3: Transverse momentum of the SecVtx secondary vertex in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

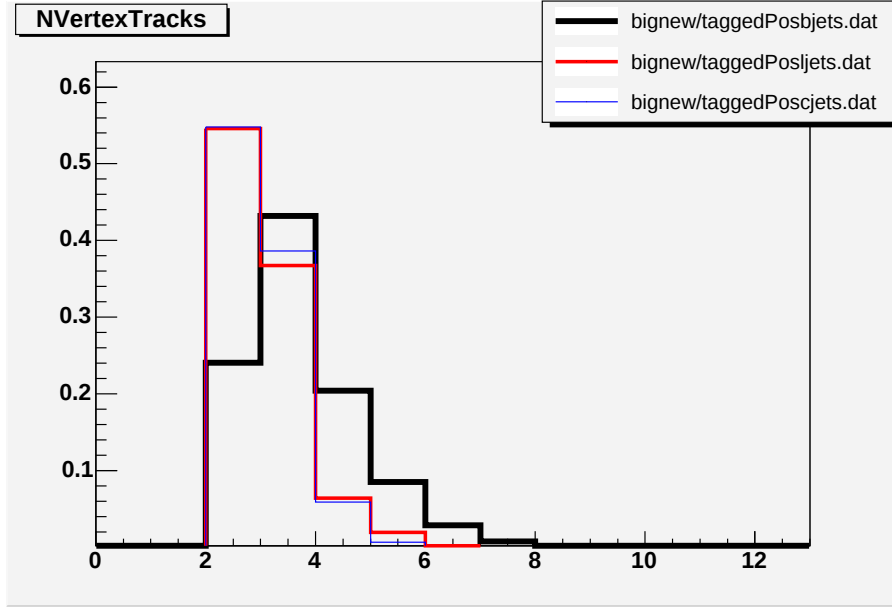


Figure 4: Number of tracks in the SecVtx secondary vertex in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

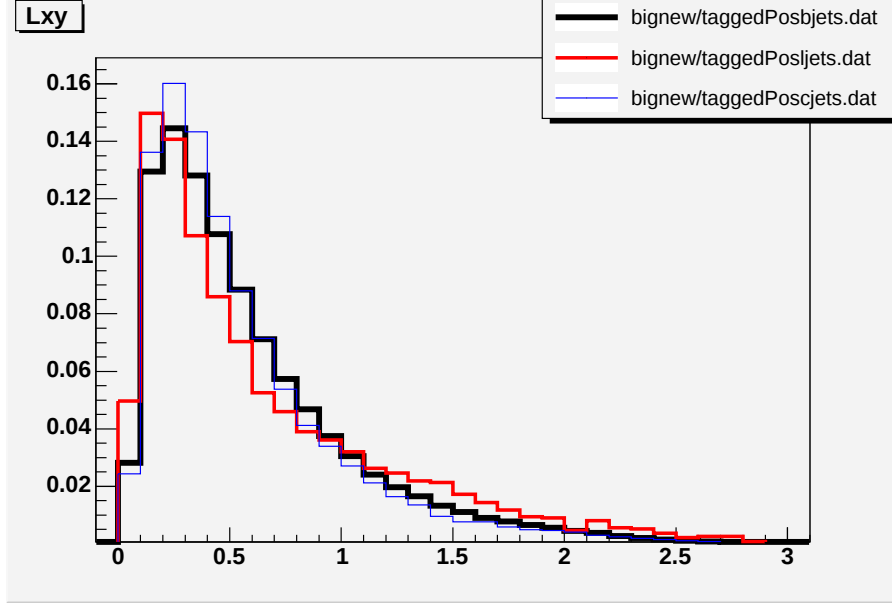


Figure 5: Transverse decay length of the SecVtx secondary vertex in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

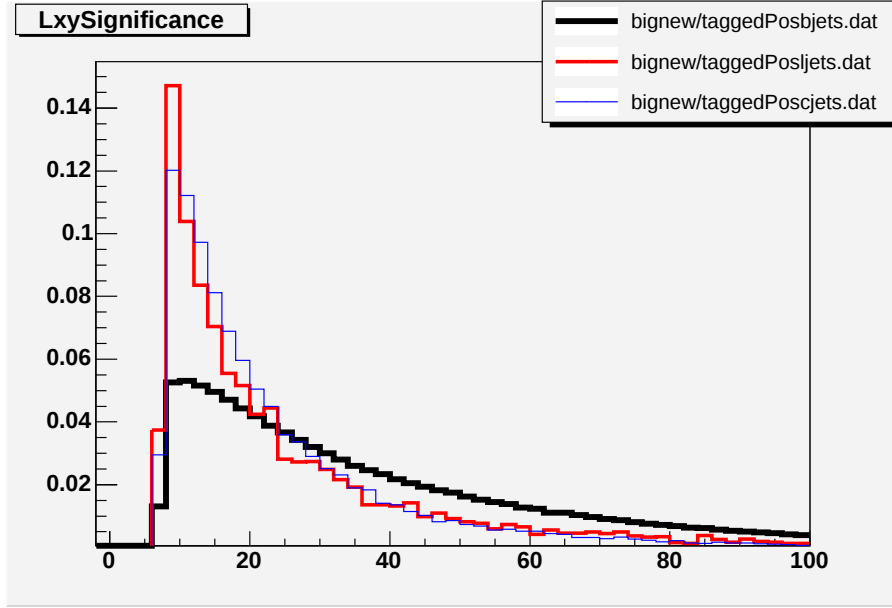


Figure 6: Transverse decay length significance of the SecVtx secondary vertex in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

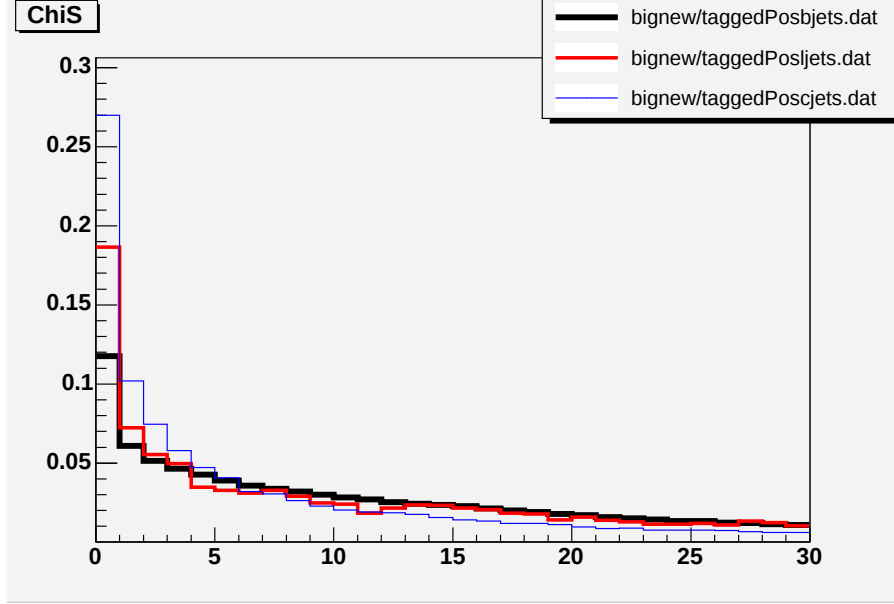


Figure 7: χ^2 in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

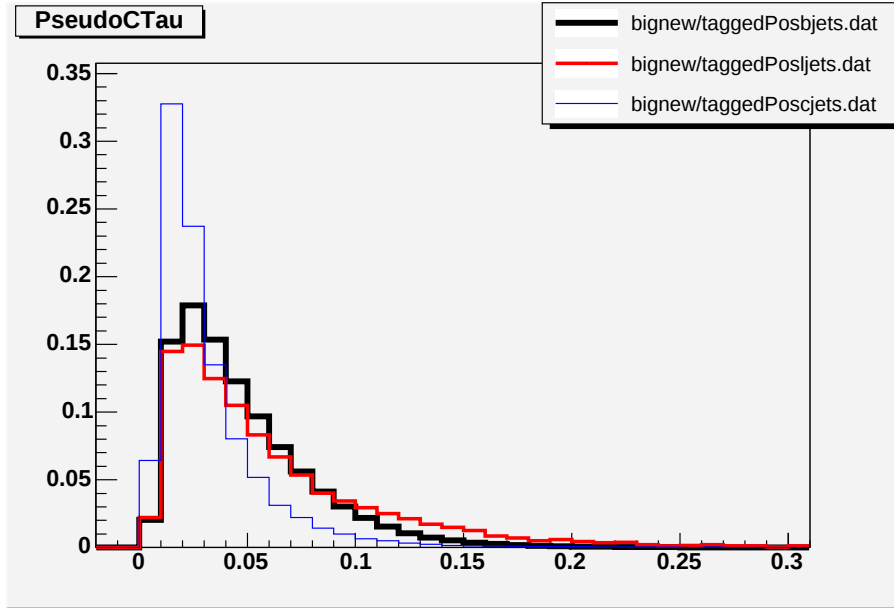


Figure 8: Pseudo-ctau of the SecVtx secondary vertex in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

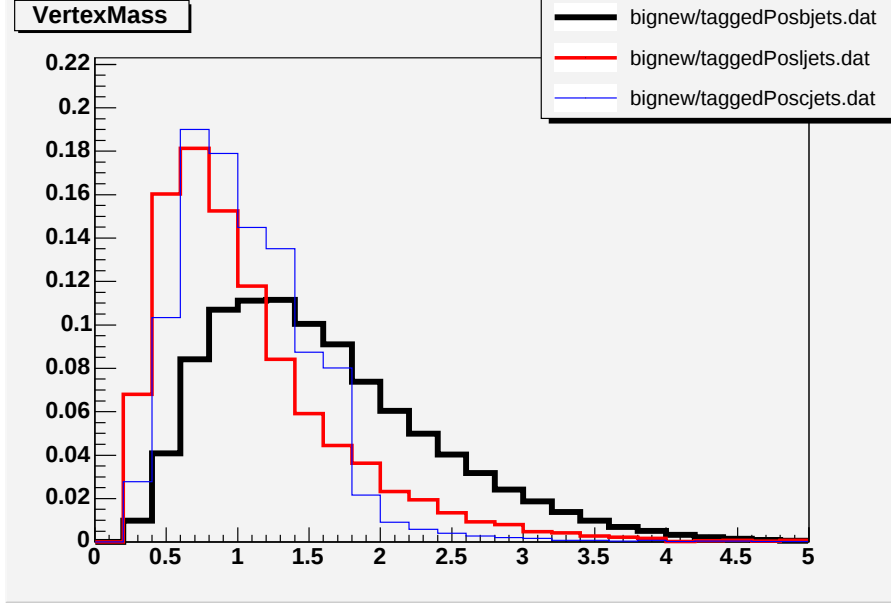


Figure 9: Vertex Mass of SevVtx-tagged jets. $t\bar{t}$ Monte Carlo.

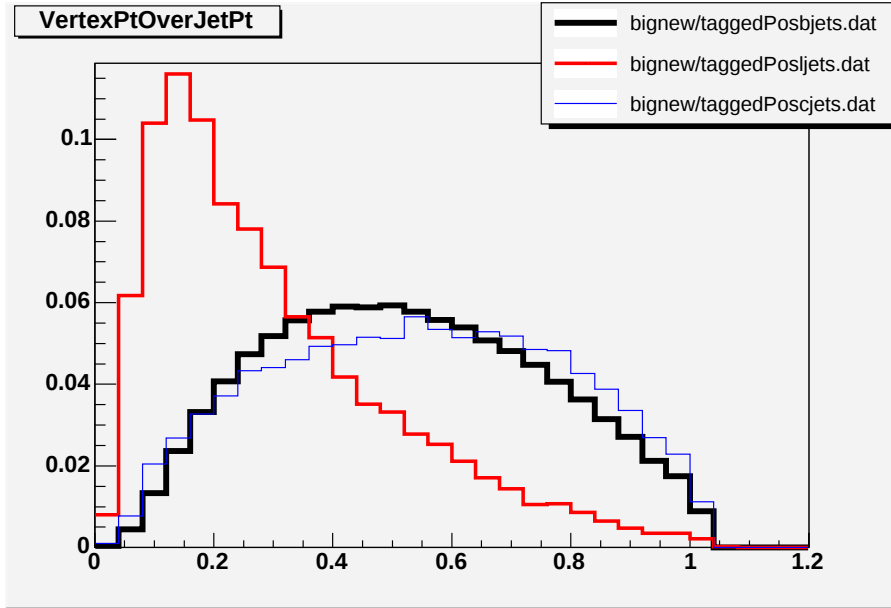


Figure 10: Vertex p_T divided by the sum of the p_T of the good tracks in the jet. $t\bar{t}$ Monte Carlo.

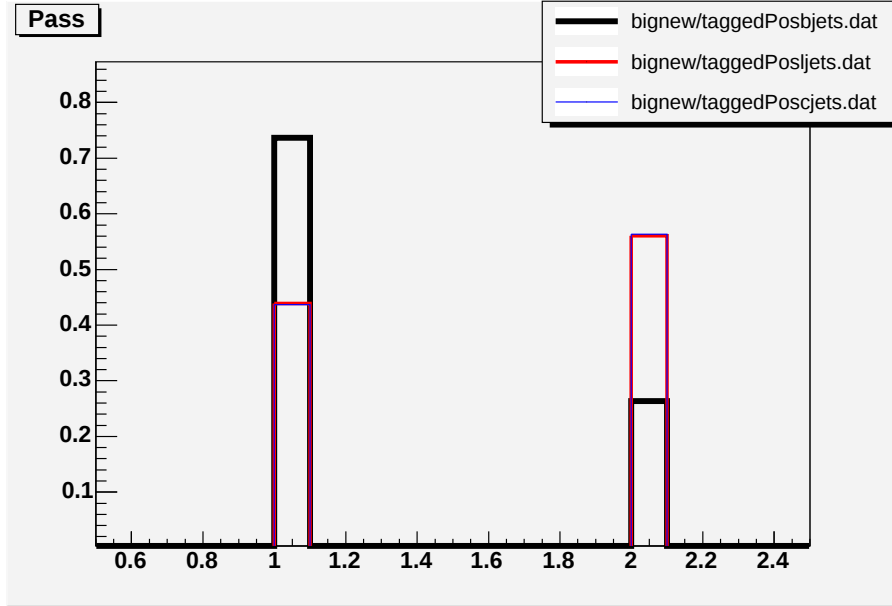


Figure 11: Pass 1 or 2 for SevVtx-tagged jets. $t\bar{t}$ Monte Carlo.

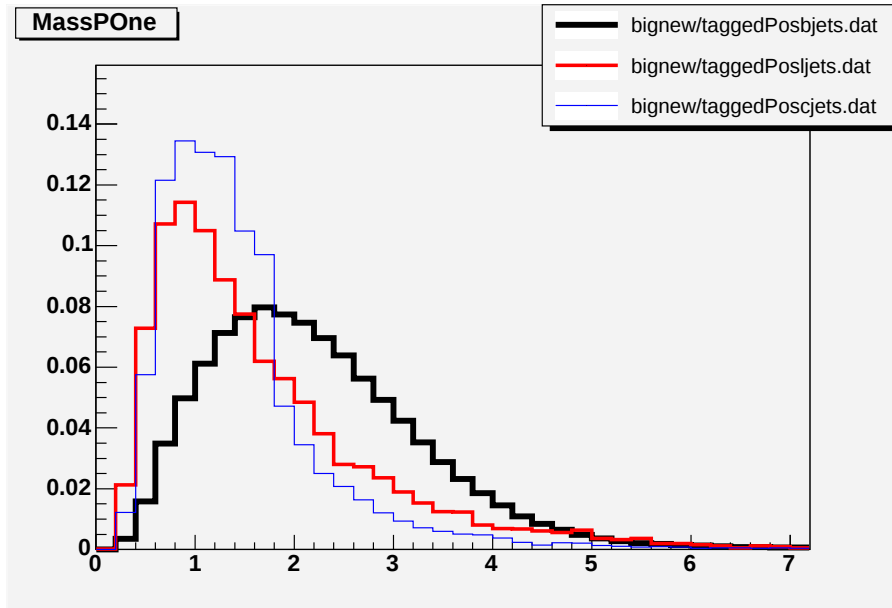


Figure 12: Reconstructed mass of Pass 1 tracks in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

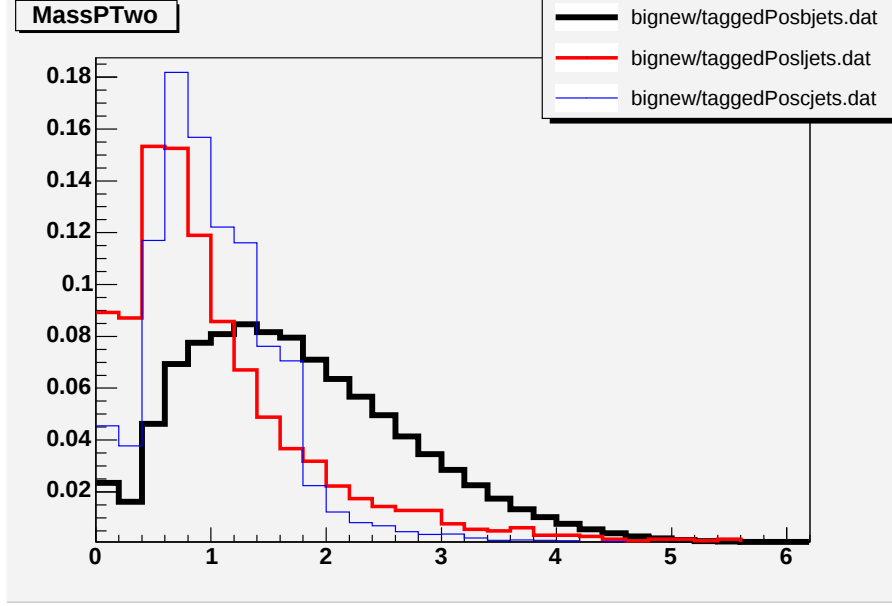


Figure 13: Reconstructed mass of Pass 2 tracks in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

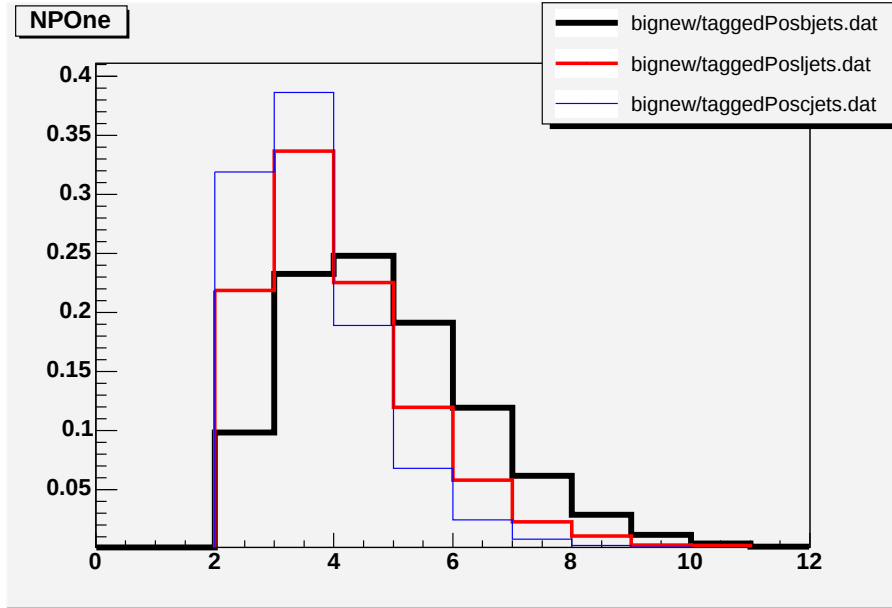


Figure 14: Number of Pass 1 tracks in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

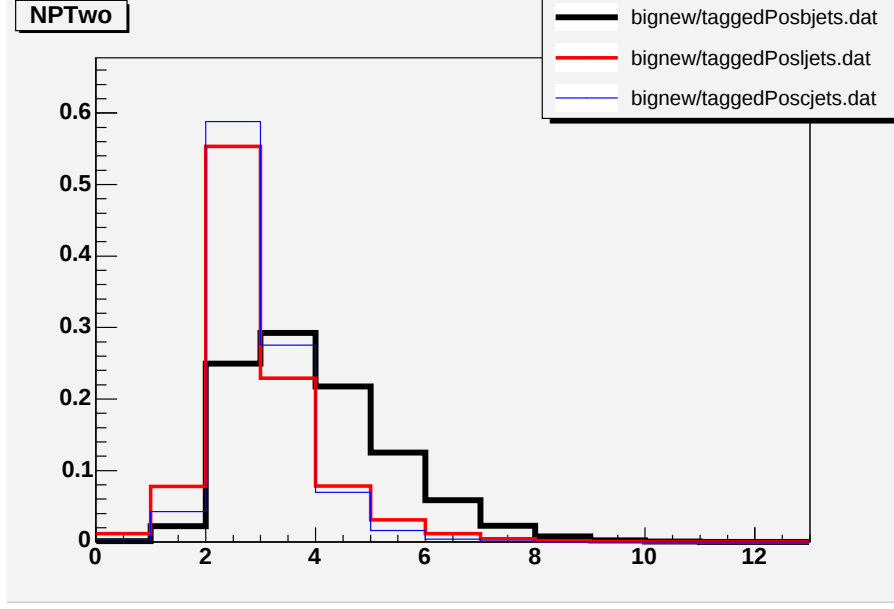


Figure 15: Number of Pass 2 tracks in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

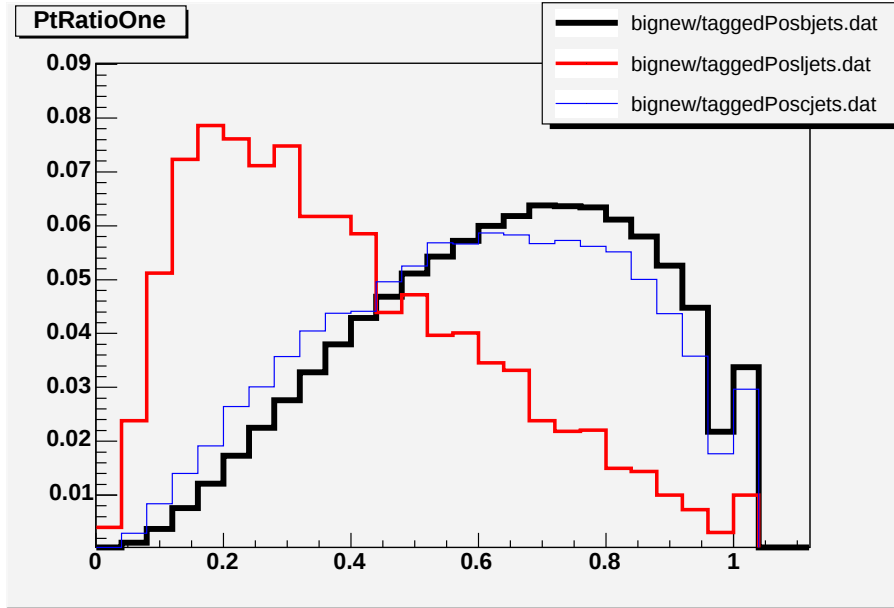


Figure 16: p_T of Pass 1 tracks divided by jet p_T . $t\bar{t}$ Monte Carlo.

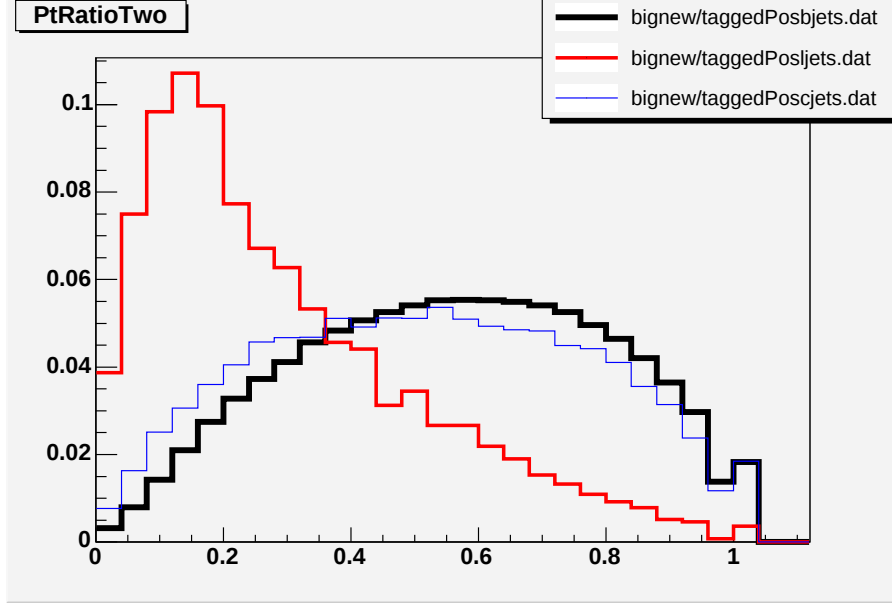


Figure 17: p_T of Pass 2 tracks divided by jet p_T . $t\bar{t}$ Monte Carlo.

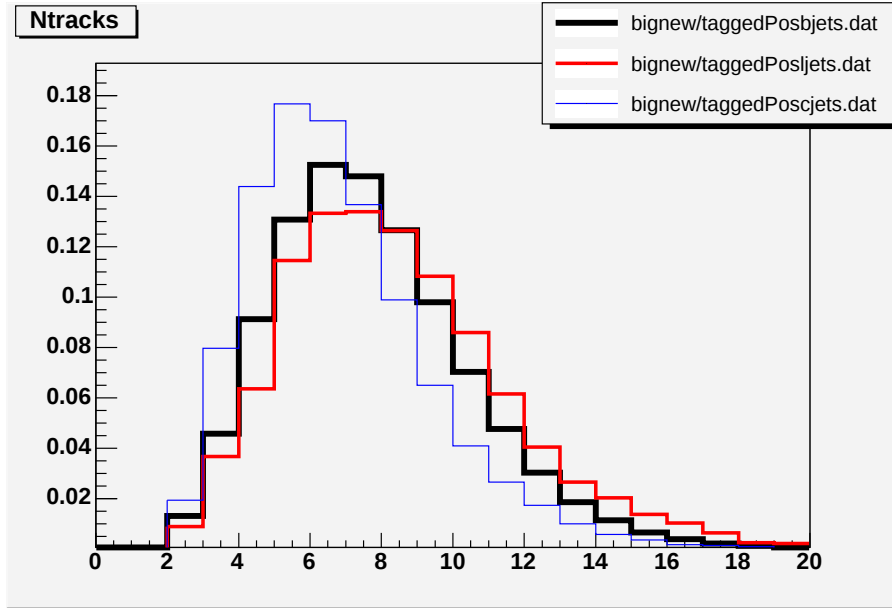


Figure 18: Number of good tracks in SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

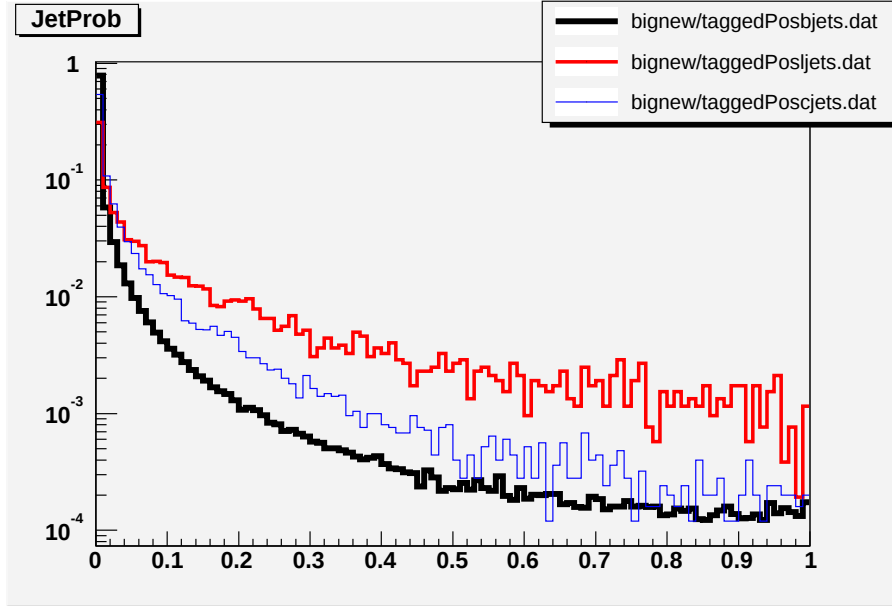


Figure 19: JetProb probability associated with SecVtx-tagged jets. $t\bar{t}$ Monte Carlo.

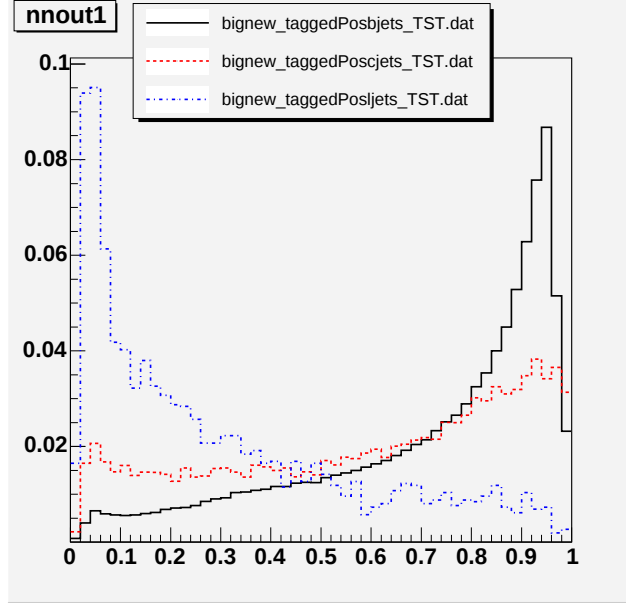


Figure 20: b - l network output for b , c and l jets. $t\bar{t}$ Monte Carlo.

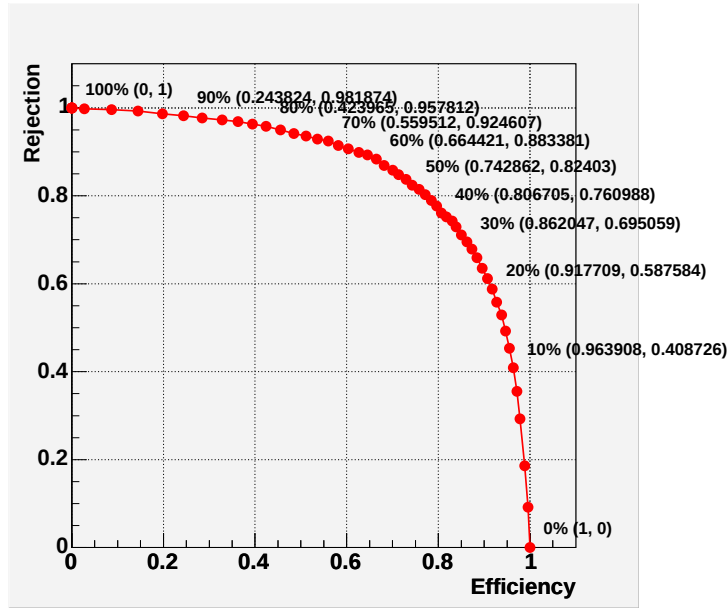


Figure 21: b - l efficiency versus rejection for b , c and l jets. $t\bar{t}$ Monte Carlo.

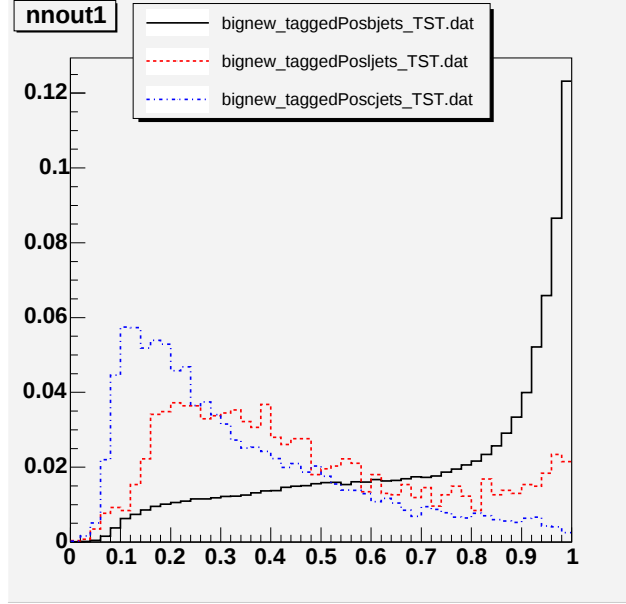


Figure 22: b - c network output for b , c and l jets. $t\bar{t}$ Monte Carlo.

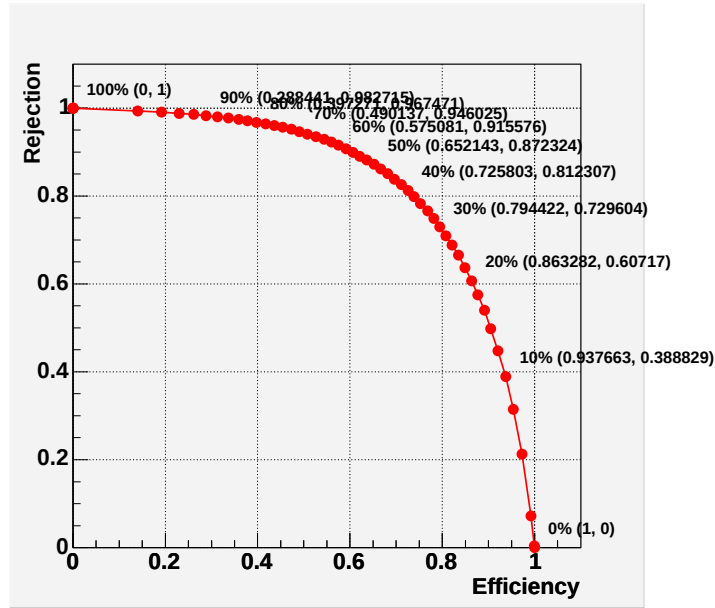


Figure 23: b - c efficiency versus rejection output for b , c and l jets. $t\bar{t}$ Monte Carlo.

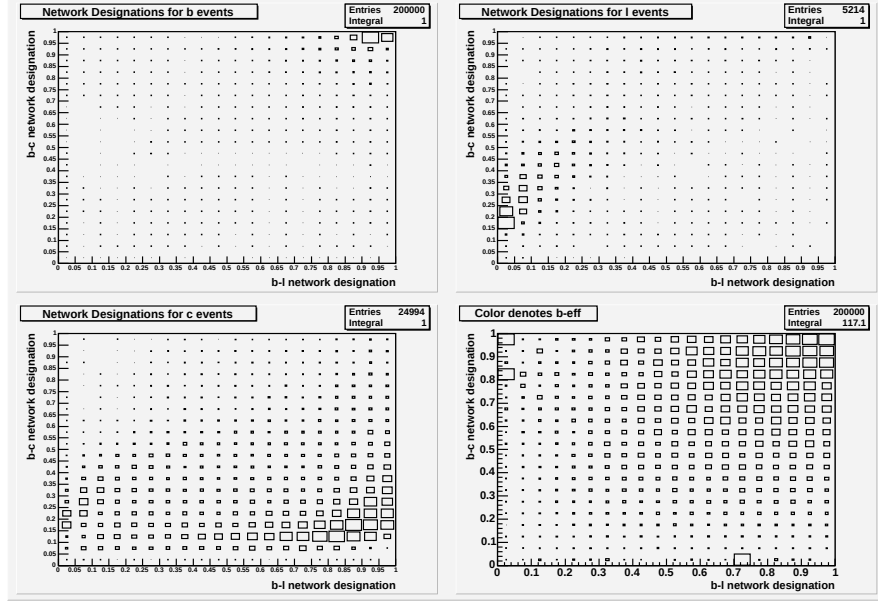


Figure 24: top left (b jets), top right (l jets), bottom left (c jets): scatter plot of designations from each network— b - l network on the x -axis, b - c network on the y -axis. Bottom right: b distribution relative to c and l ($b/(b + c + l)$). $t\bar{t}$ Monte Carlo.

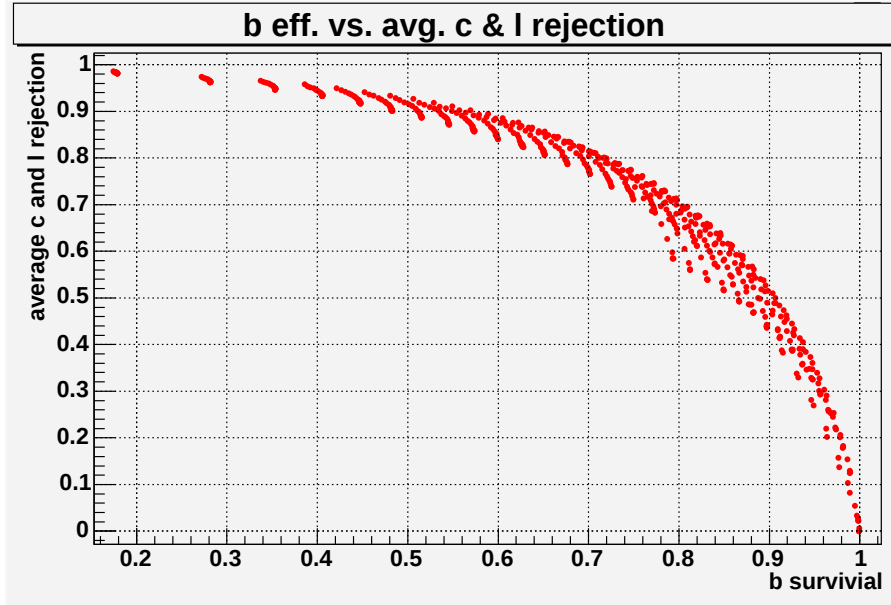


Figure 25: Scatter plot of b efficiencies and average rejection of c and l as a result of the tagger.

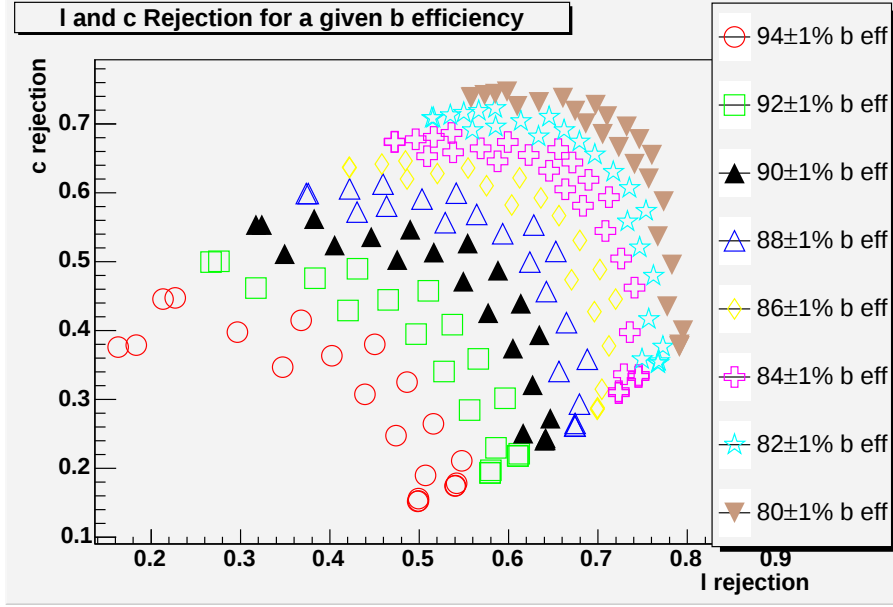


Figure 26: Scatter plot of c and l rejection for a given b efficiency (color) as a result of the tagger.

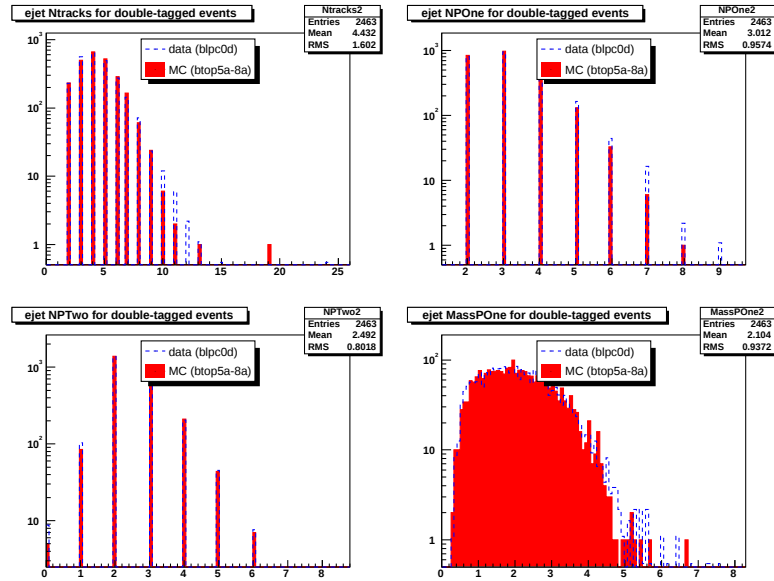


Figure 27: Input parameters for e jets. Dashed line: data, $blpc0d$, Fill: Monte Carlo: $b_{top\{5-8\}a}$

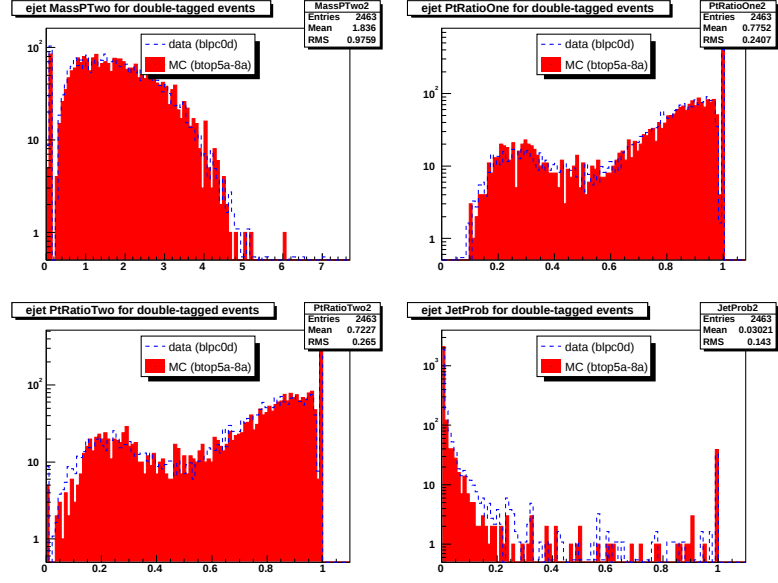


Figure 28: Input parameters for e jets. Dashed line: data, blpc0d , Fill: Monte Carlo: $\text{btop}\{5-8\}\text{a}$

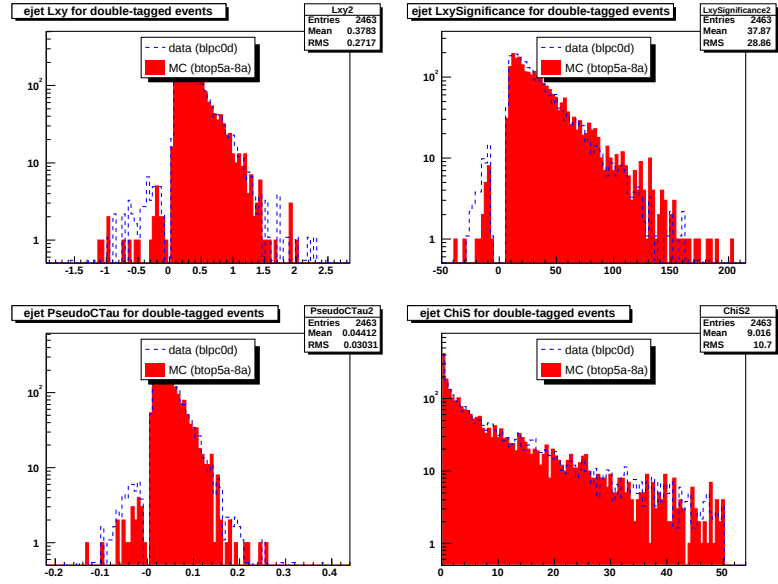


Figure 29: Input parameters for e jets. Dashed line: data, blpc0d , Fill: Monte Carlo: $\text{btop}\{5-8\}\text{a}$

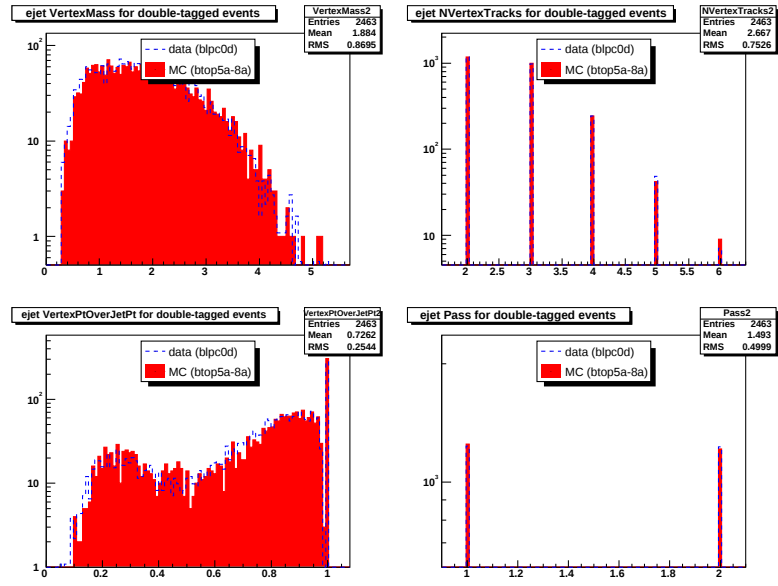


Figure 30: Input parameters for e jets. Dashed line: data, b1pc0d, Fill: Monte Carlo: btop{5-8}a

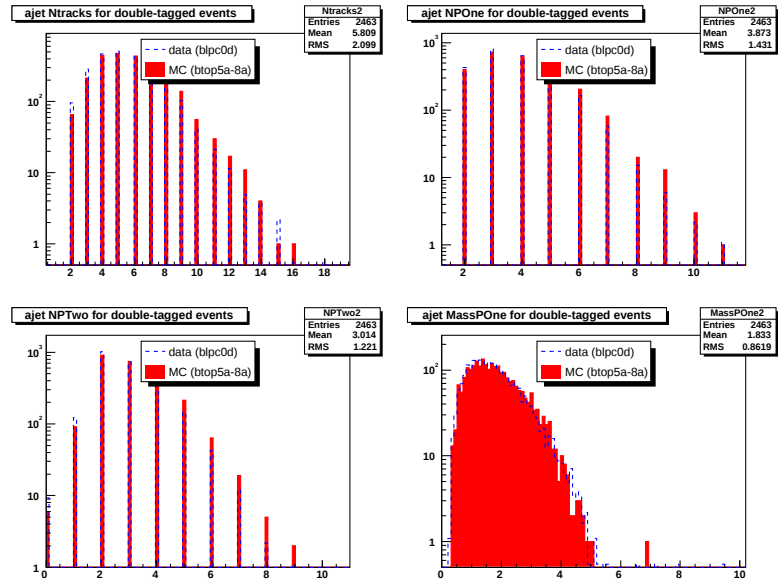


Figure 31: Input parameters for a jets. Dashed line: data, b1pc0d, Fill: Monte Carlo: btop{5-8}a

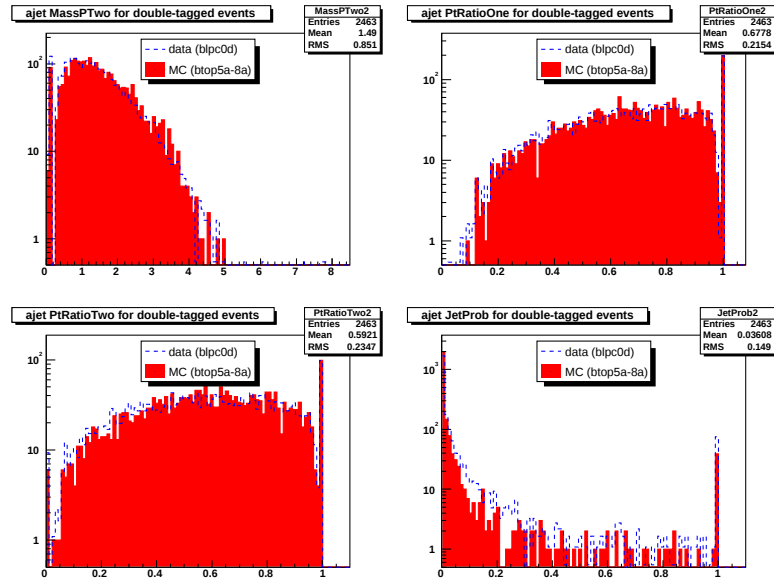


Figure 32: Input parameters for a jets. Dashed line: data, b1pc0d, Fill: Monte Carlo: btop{5-8}a

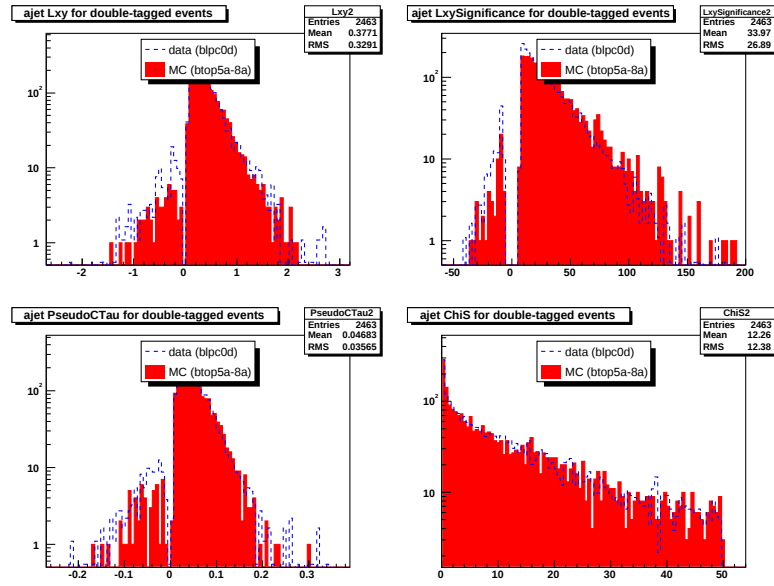


Figure 33: Input parameters for a jets. Dashed line: data, b1pc0d, Fill: Monte Carlo: btop{5-8}a

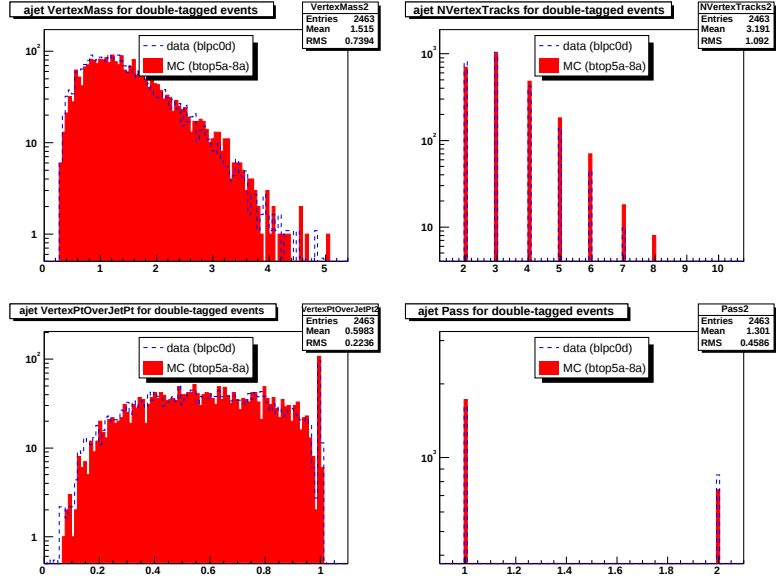


Figure 34: Input parameters for a jets. Dashed line: data, $b1pc0d$, Fill: Monte Carlo: $b_{top\{5-8\}a}$

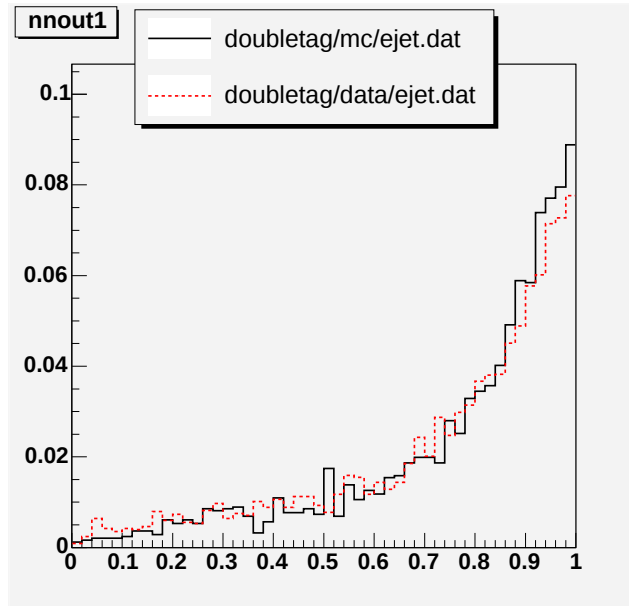


Figure 35: Electron jet $b-l$ network response.

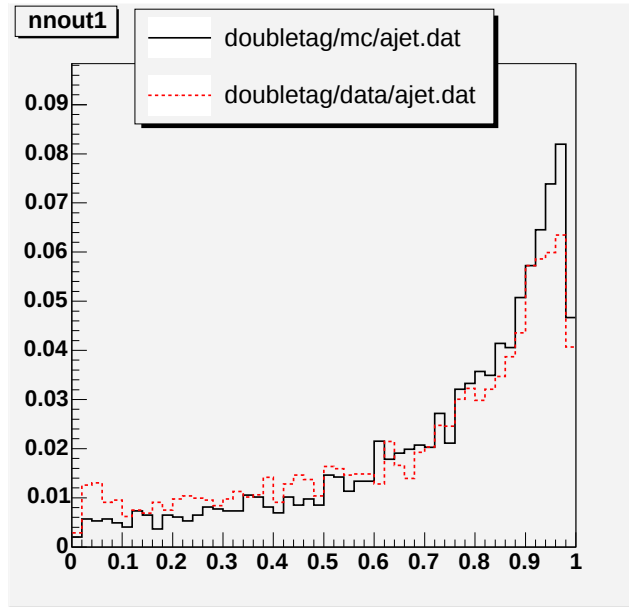


Figure 36: Away jet b - l network response.

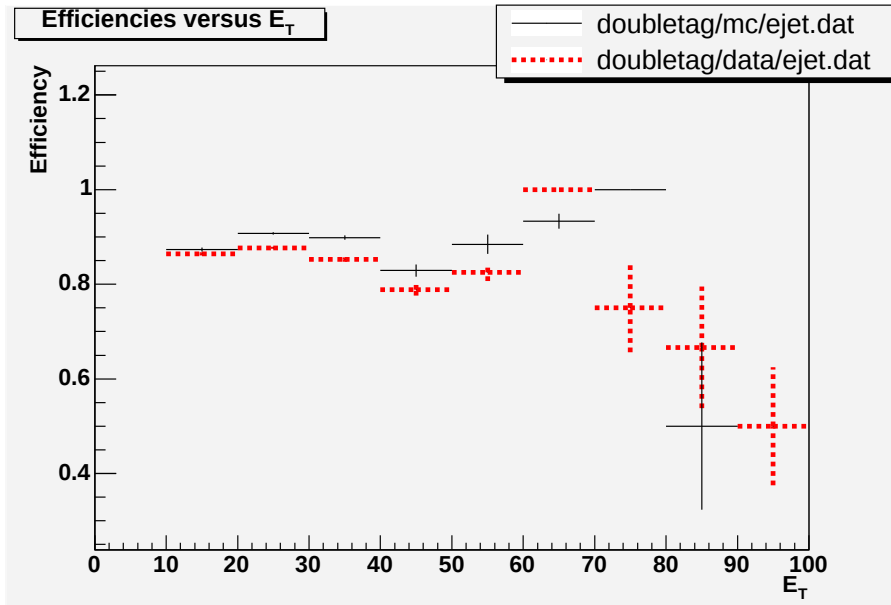


Figure 37: Acceptance of electron jets as a function of E_T

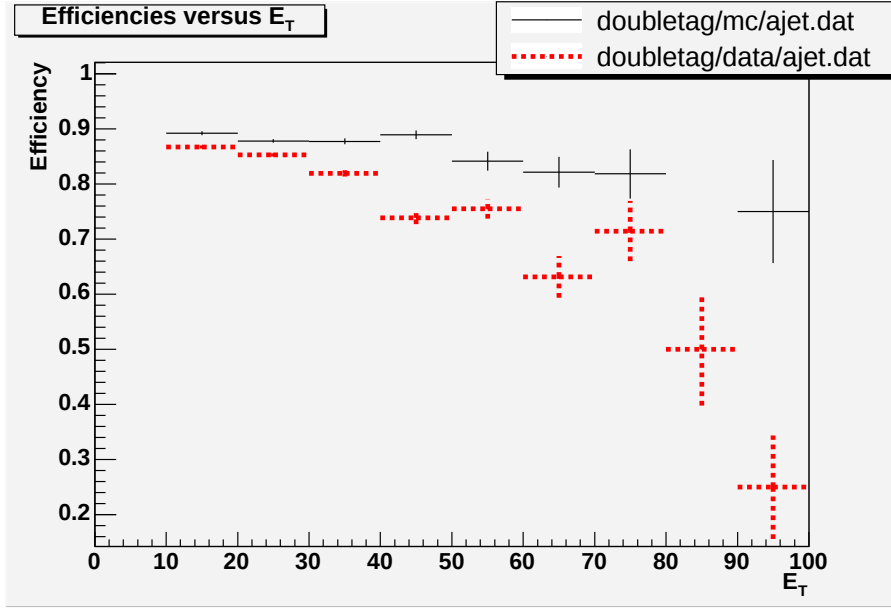


Figure 38: Acceptance of away jets as a function of E_T

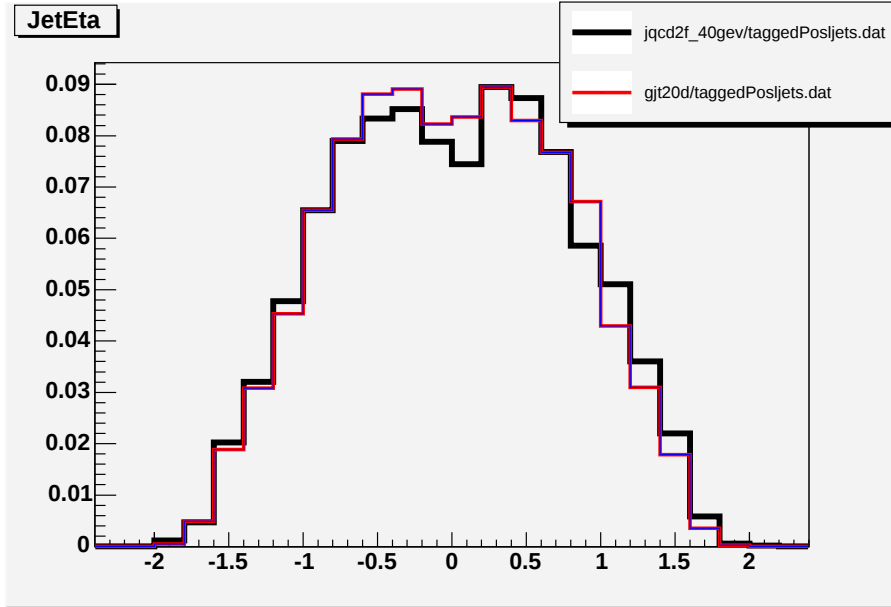


Figure 39: Pseudorapidity of SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

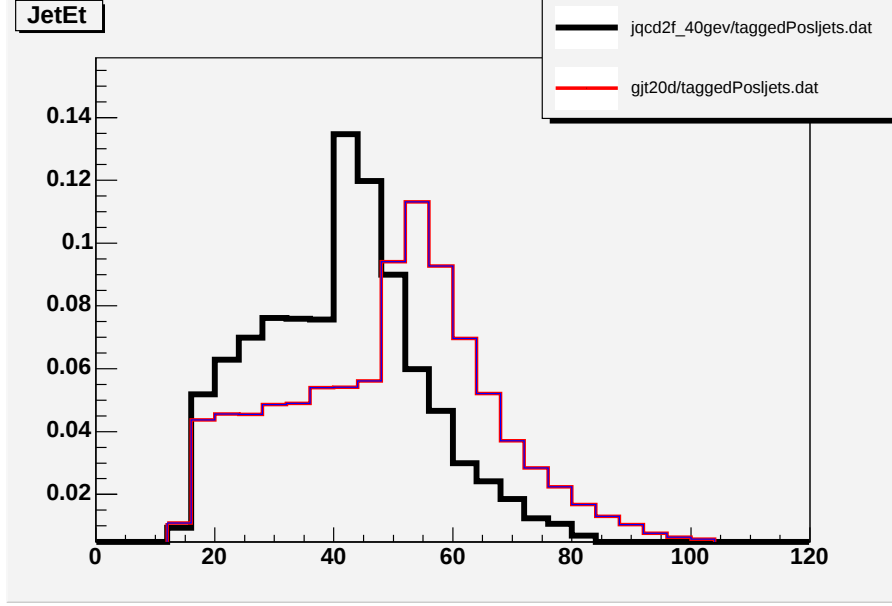


Figure 40: Transverse energy of SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

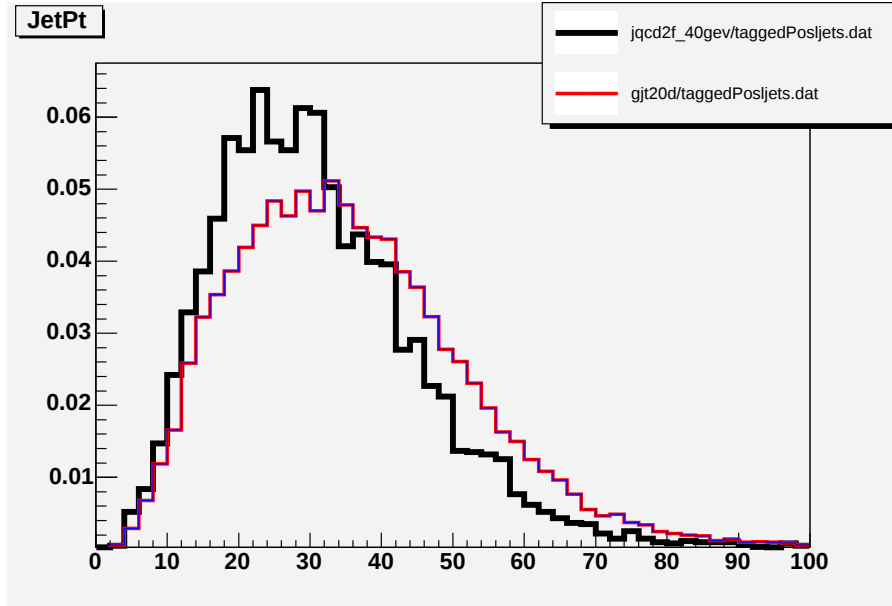


Figure 41: Transverse momentum of the SecVtx secondary vertex in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

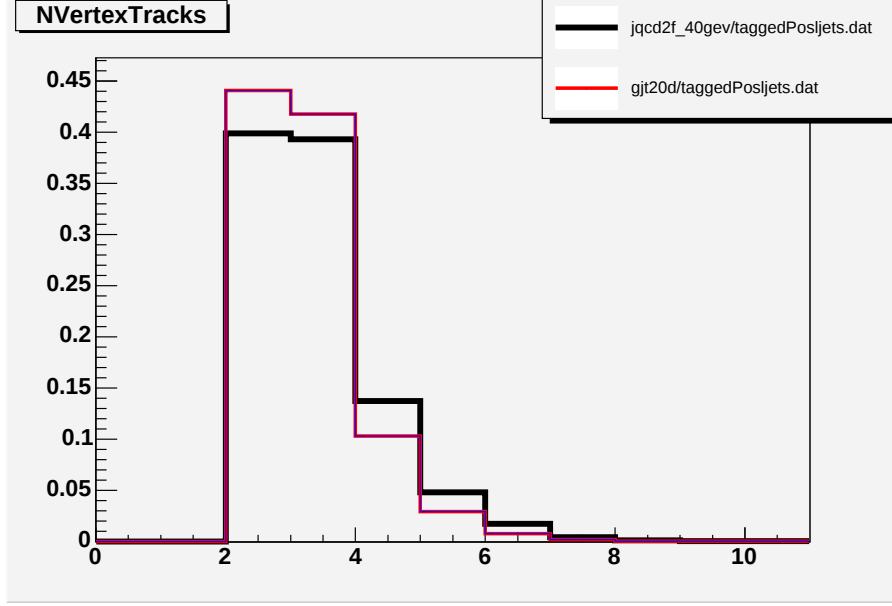


Figure 42: Number of tracks in the SecVtx secondary vertex in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

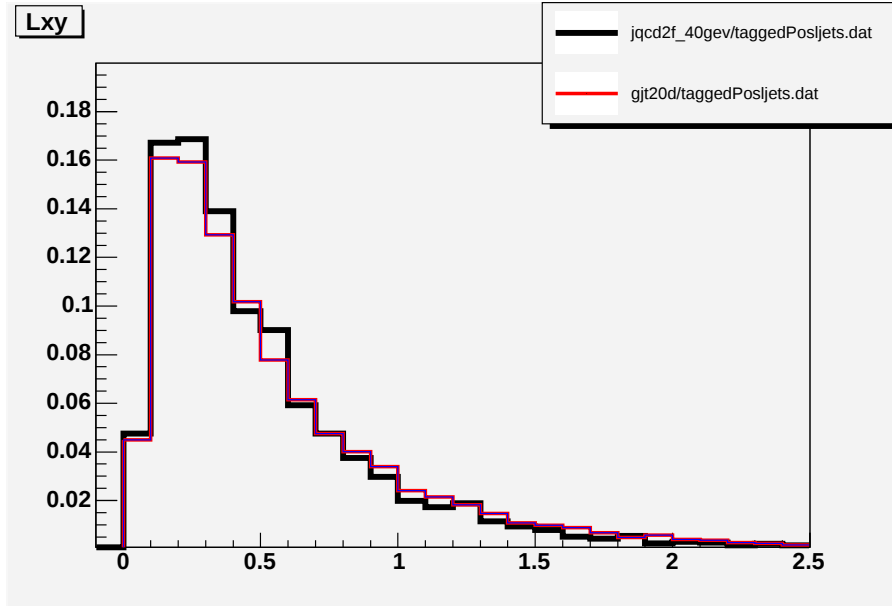


Figure 43: Transverse decay length of the SecVtx secondary vertex in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

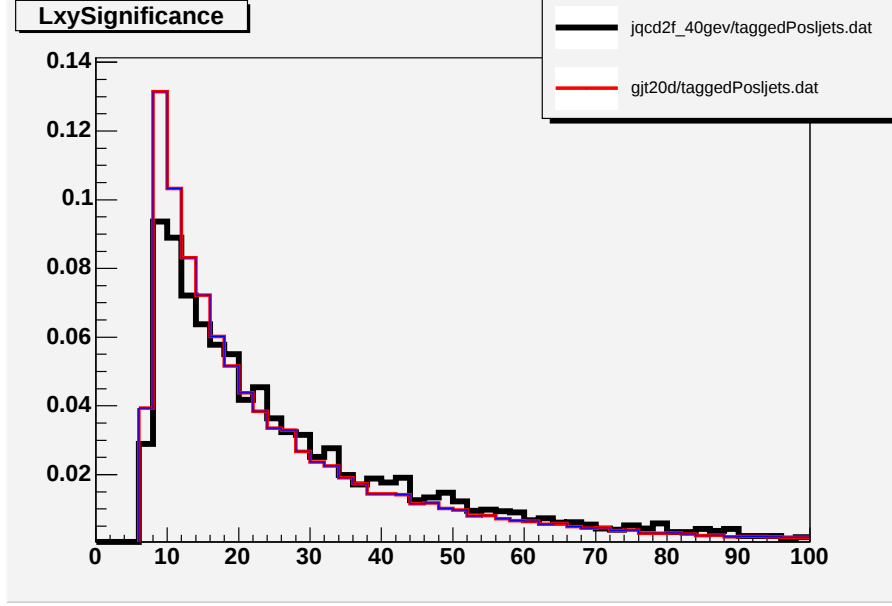


Figure 44: Transverse decay length significance of the SecVtx secondary vertex in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

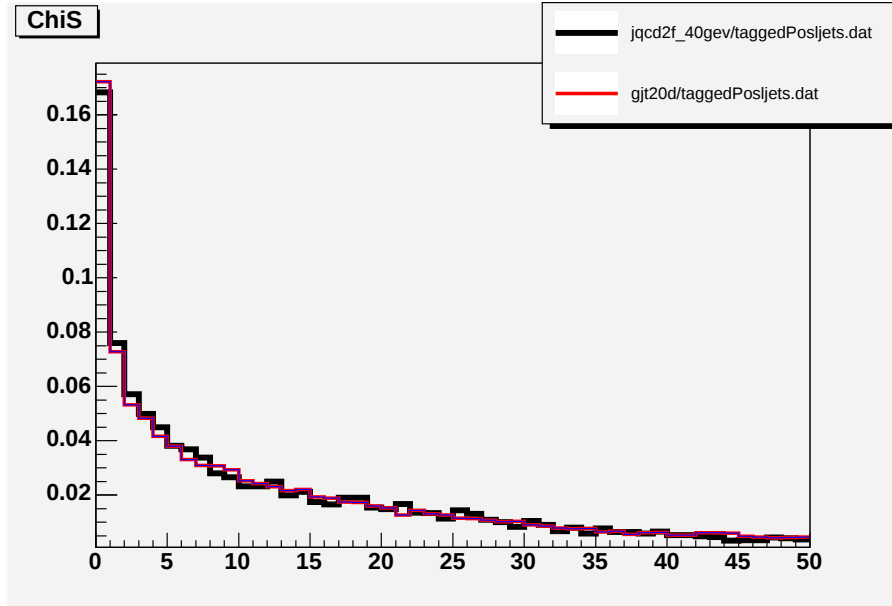


Figure 45: χ^2 in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

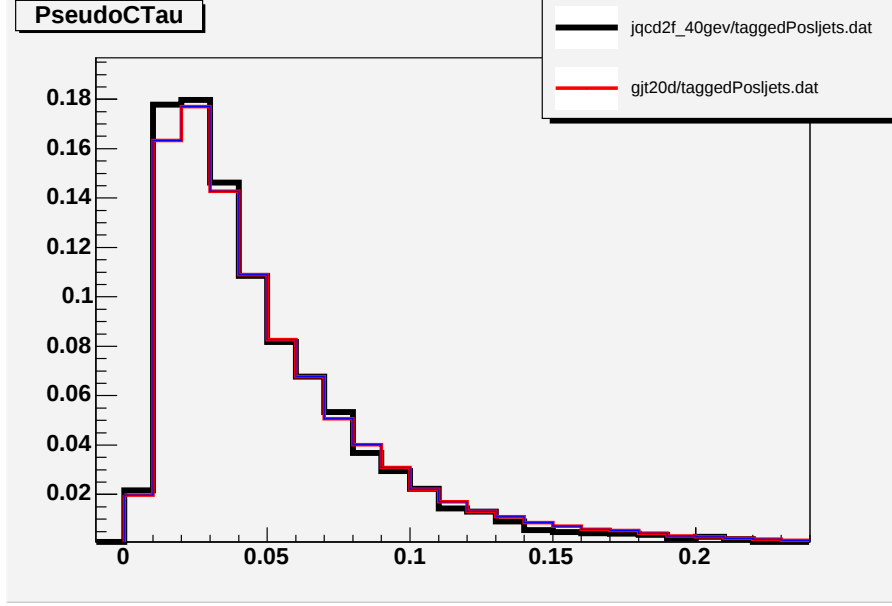


Figure 46: Pseudo-ctau of the SecVtx secondary vertex in SecVtx-tagged jets. Jet 40 Monte Carlo (jgcd2f; thick line), Jet 50 data (gjt20d, thin line).

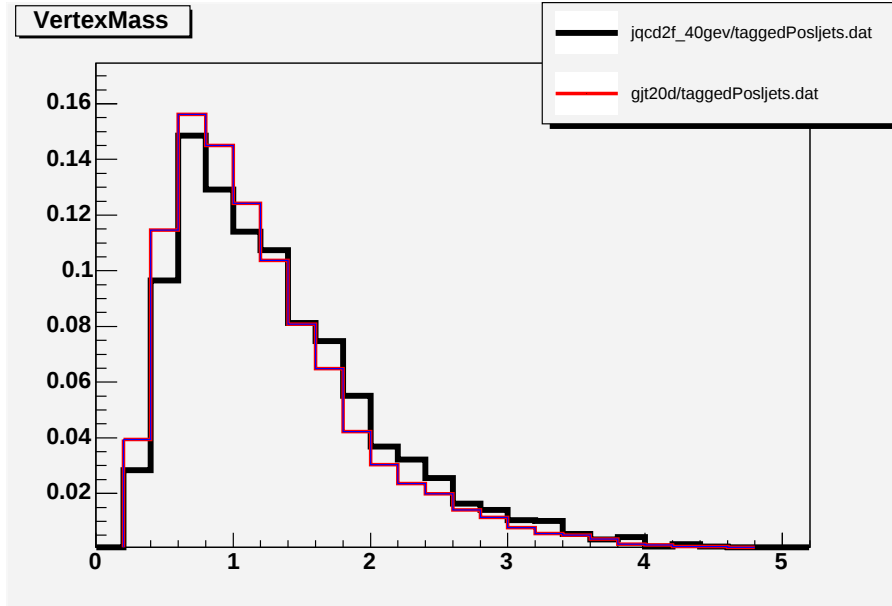


Figure 47: Vertex Mass of SevVtx-tagged jets. Jet 40 Monte Carlo (jgcd2f; thick line), Jet 50 data (gjt20d, thin line).

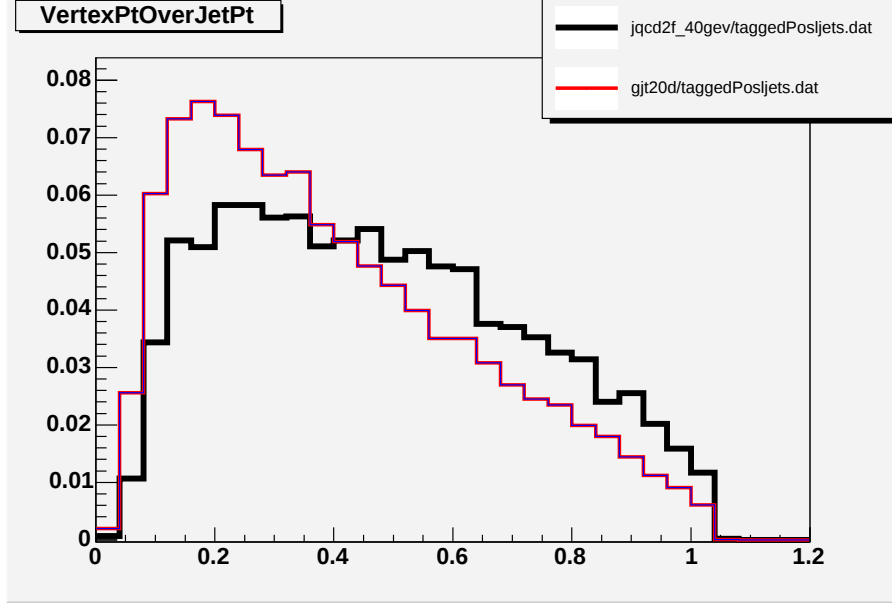


Figure 48: Vertex p_T divided by the sum of the p_T of the good tracks in the jet. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

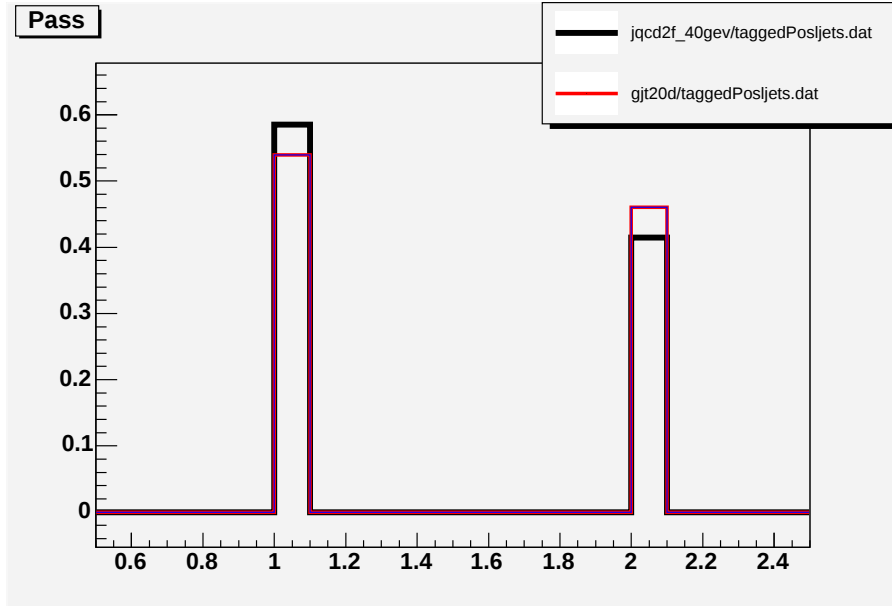


Figure 49: Pass 1 or 2 for SevVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

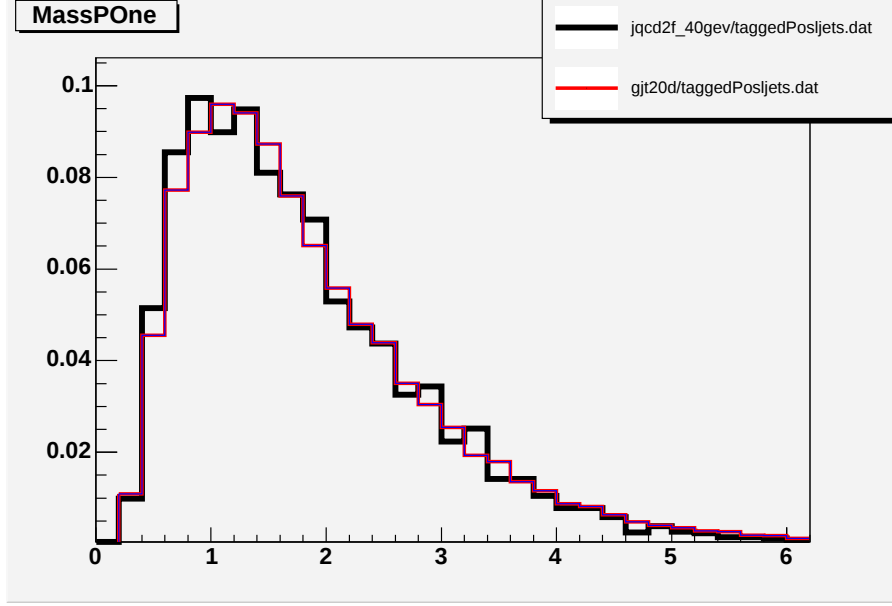


Figure 50: Reconstructed mass of Pass 1 tracks in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

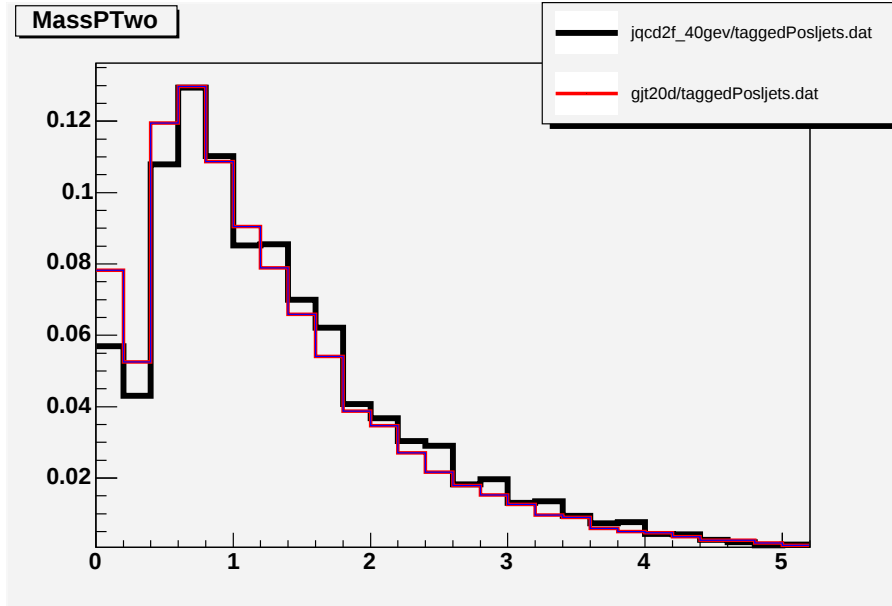


Figure 51: Reconstructed mass of Pass 2 tracks in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

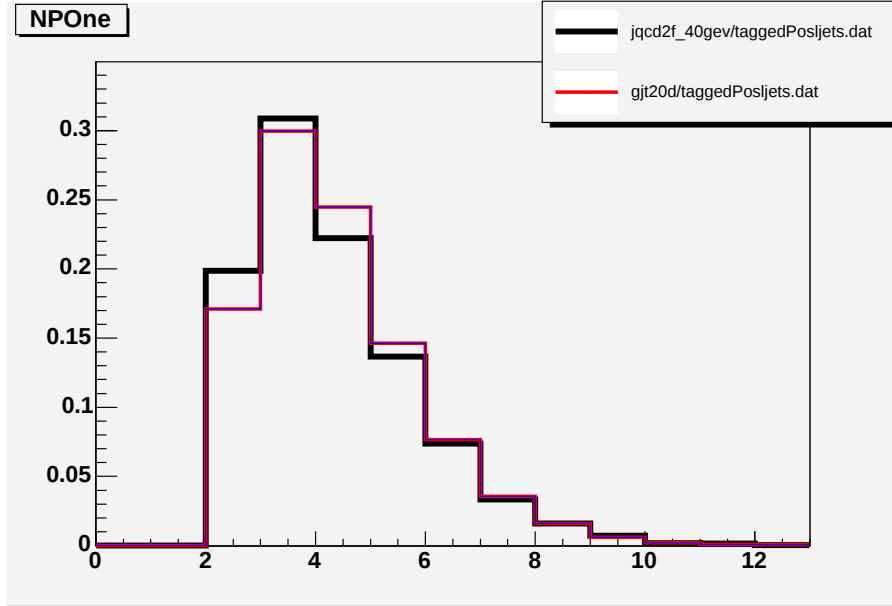


Figure 52: Number of Pass 1 tracks in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

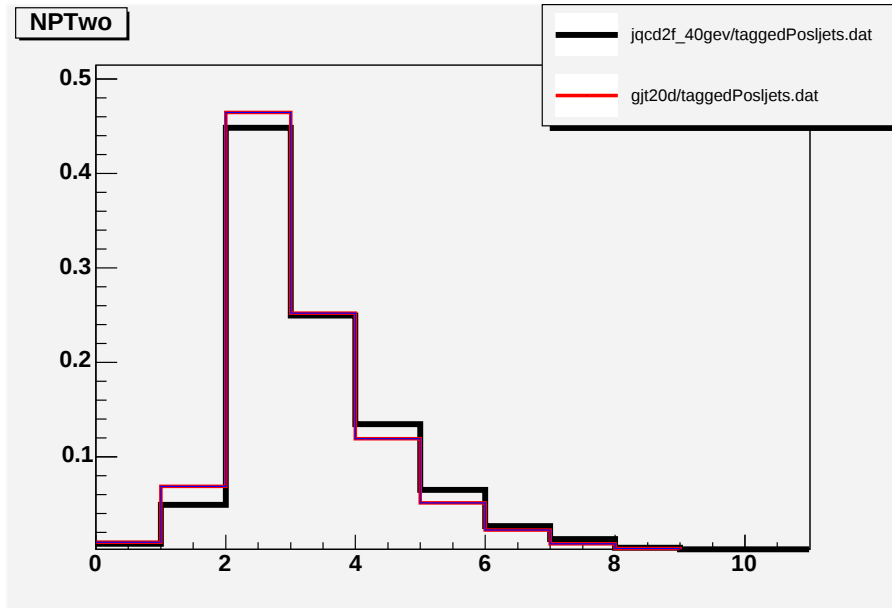


Figure 53: Number of Pass 2 tracks in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

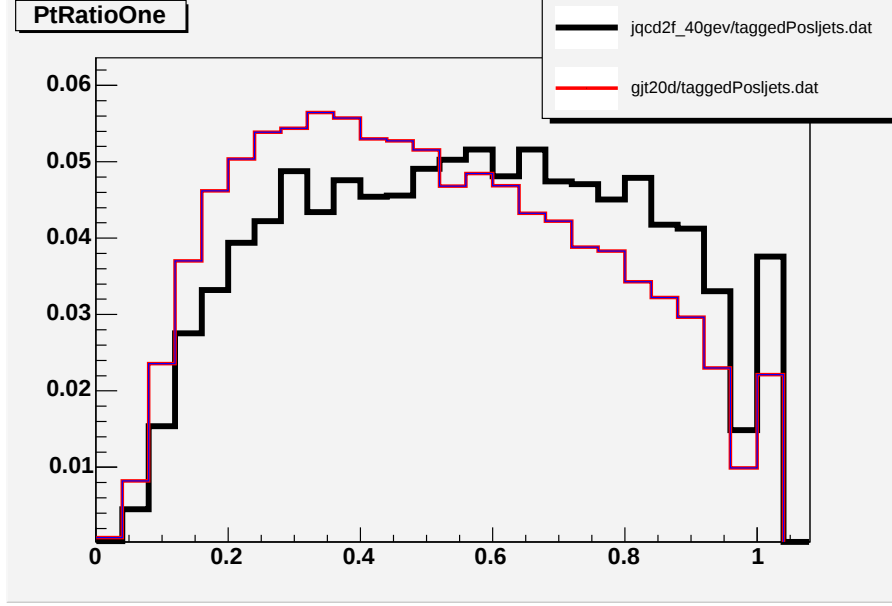


Figure 54: p_T of Pass 1 tracks divided by jet p_T . Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

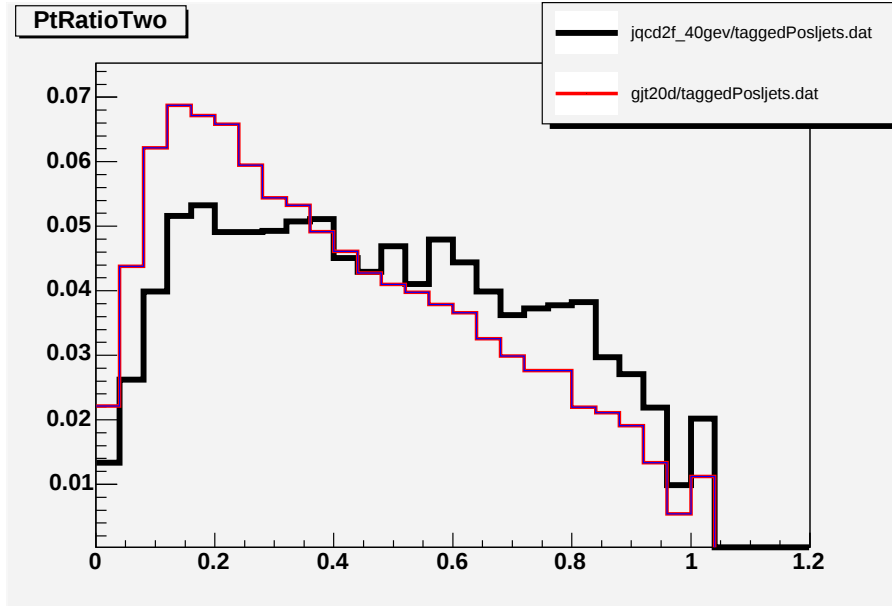


Figure 55: p_T of Pass 2 tracks divided by jet p_T . Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

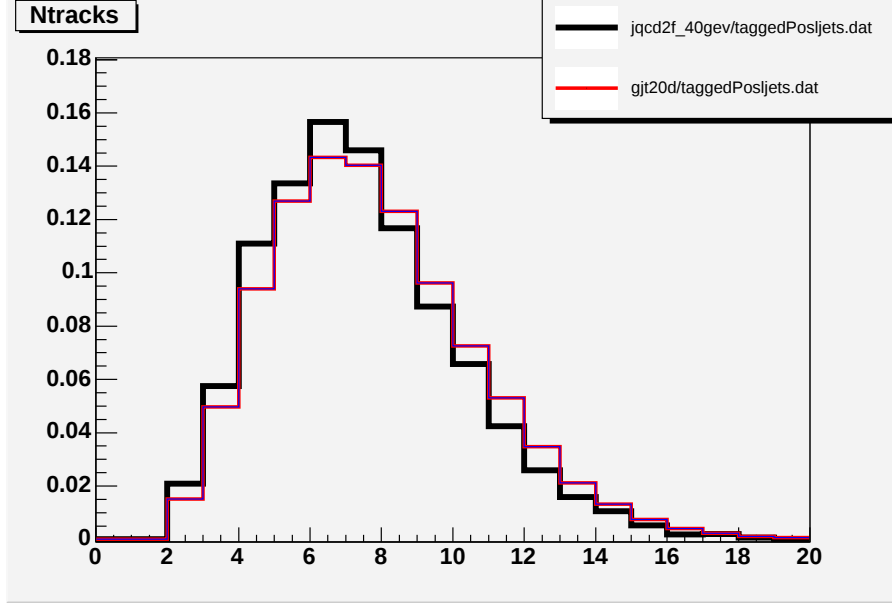


Figure 56: Number of good tracks in SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

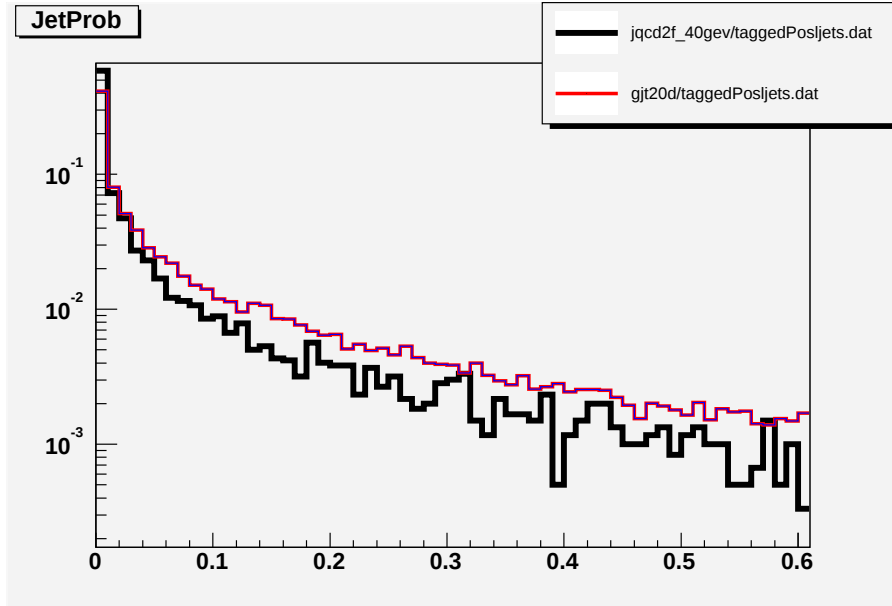


Figure 57: JetProb probability associated with SecVtx-tagged jets. Jet 40 Monte Carlo (jqcd2f; thick line), Jet 50 data (gjt20d, thin line).

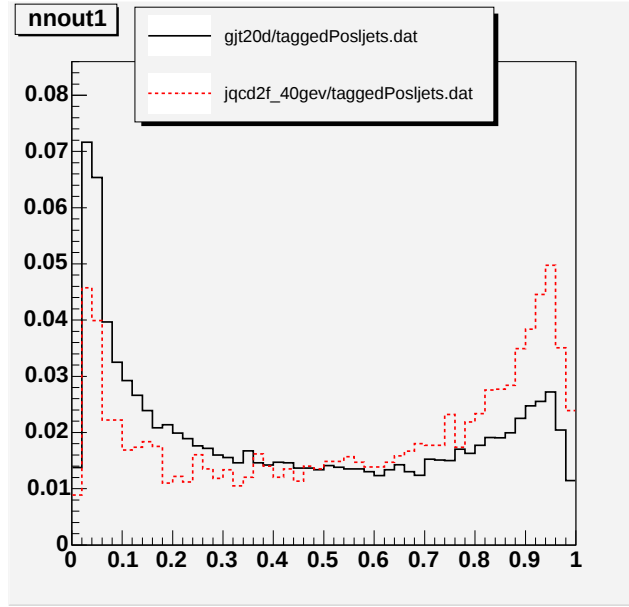


Figure 58: Jet data and Monte Carlo b - l network response: Positive tagged.

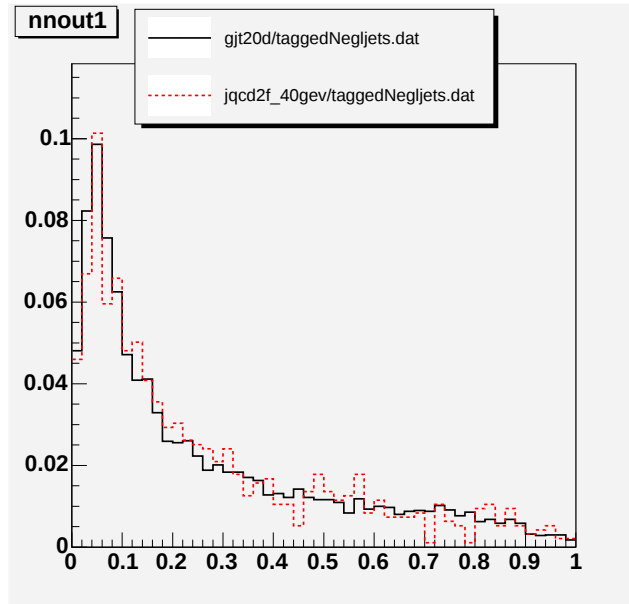


Figure 59: Jet data and Monte Carlo b - l network response: Negatively tagged.

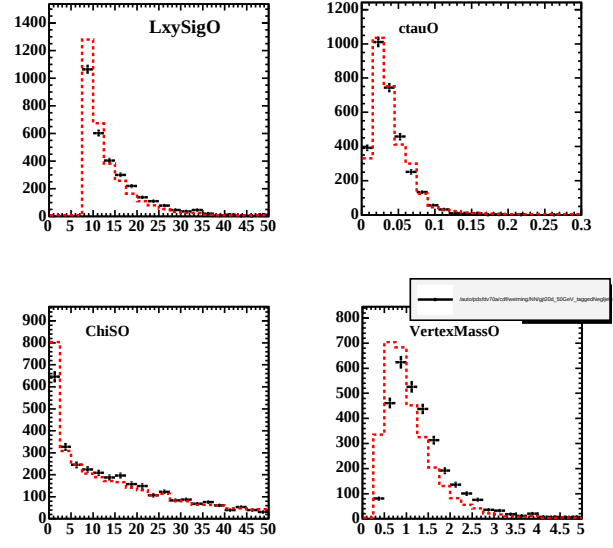


Figure 60: Jet data parameters, as predicted by the E_T - $c\tau$, 90% b efficiency matrix in Table 5 (solid line) compared to the actual distribution (dotted line)

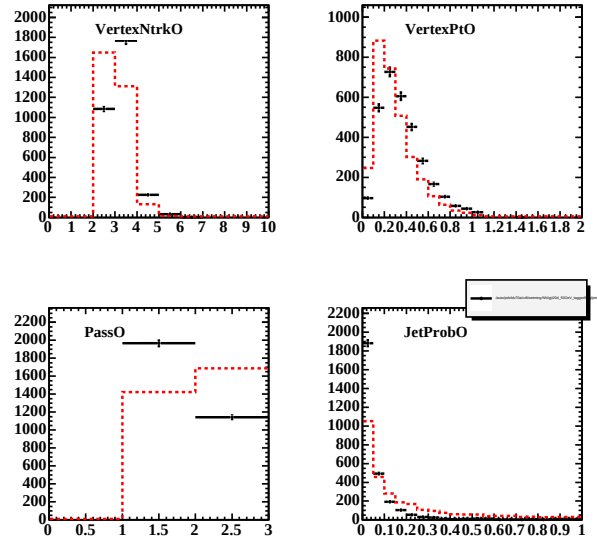


Figure 61: Jet data parameters, as predicted by the E_T - $c\tau$, 90% b efficiency matrix in Table 5 (solid line) compared to the actual distribution (dotted line)

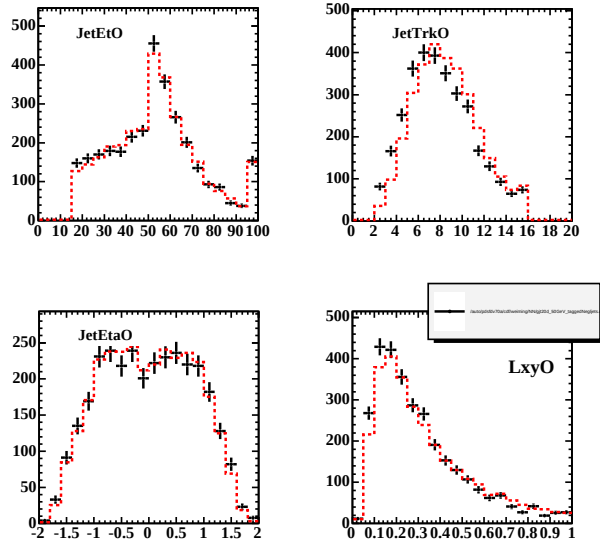


Figure 62: Jet data parameters, as predicted by the E_T - $c\tau$, 90% b efficiency matrix in Table 5 (solid line) compared to the actual distribution (dotted line)

A Appendix: Code

<http://www-cdf.lbl.gov/~weiming/nnBtagLklhd.cc>

```
float Afunction(float INPUT);
bool nnBtagLklhd(int& indx, float& nnbc, float& nnbl);
void NNoutBL(float* pattern, float* output);
void NNoutBC(float* pattern, float* output);

// Implementation follows:

float Afunction(float INPUT){return(1.0/(1.0 + exp(-2.0*INPUT)));}

// nnBtagLklhd(int& indx, float& nnbc, float& nnbl)
// indx is jet index in the SECVTX_COLL, nnbc is nn output from b vs c and
// nnbl is nn output from b vs l: the operation point is
// nnbc>0.242 && nnbl>0.182, which gives 90% of b efficiency and more than
// 50% of rejection of charm and light flavors.

bool nnBtagLklhd(int& indx, float& nnbc, float& nnbl){
    const float fbc(0.242);
    const float fbl(0.182);
    const float PiMass(0.1396);

    const int INPUTNODES=19;
    float Patt[INPUTNODES];
    const int SECVTX_COLL = 3;
    float OutputBL[] = {0.0};
    float OutputBC[] = {0.0};

    int np1Pos(0);
    int np2Pos(0);
    float px(0.);
    float py(0.);
    float pPos[4] = {0};
    float p2Pos[4] = {0};
    nnbl = -1;
    nnbc = -1;
    jet* jt = (jet*) jetA->At(indx);
    if(jt->CollType==SECVTX_COLL){ // cone 0.4 collection
for(int i=0;i<tjassocA->GetEntries();i++)
{
    tjassoc* tj = (tjassoc*) tjassocA->At(i);
    if(tj->jetid == indx)
```

```

{
    secvtxtrack* tr = (secvtxtrack*) svtxtrA->At(tj->trkid);
    if(fabs(tr->sPt)<1000)
    {
        float tpx((fabs(tr->sPt))*cos(tr->tPhi0));
        float tpy((fabs(tr->sPt))*sin(tr->tPhi0));
        float tpz(tr->tCot * fabs(tr->sPt));
        float tp(sqrt(tr->sPt*tr->sPt + tpz*tpz));
        px += tpx;
        py += tpy;
        if(tj->acode & 0x04) // pass 1 track
        {
            ++np1Pos;
            pPos[0] += sqrt(PiMass*PiMass + tp*tp);
            pPos[1] += tpx;
            pPos[2] += tpy;
            pPos[3] += tpz;
        }
        if(tj->acode & 0x08) // pass 2 track
        {
            ++np2Pos;
            p2Pos[0] += sqrt(PiMass*PiMass + tp*tp);
            p2Pos[1] += tpx;
            p2Pos[2] += tpy;
            p2Pos[3] += tpz;
        }
    }
}

float pt(sqrt(px*px+py*py));
float pt1Pos(sqrt(pPos[1]*pPos[1] + pPos[2]*pPos[2]));
float pt2Pos(sqrt(p2Pos[1]*p2Pos[1] + p2Pos[2]*p2Pos[2]));
float mass1Pos(0);
float mass2Pos(0);
float m12Pos(pPos[0]*pPos[0] - pPos[1]*pPos[1] -
pPos[2]*pPos[2] - pPos[3]*pPos[3]);
float m22Pos(p2Pos[0]*p2Pos[0] - p2Pos[1]*p2Pos[1] -
p2Pos[2]*p2Pos[2] - p2Pos[3]*p2Pos[3]);
if(m12Pos>0)mass1Pos=sqrt(m12Pos);
if(m22Pos>0)mass2Pos=sqrt(m22Pos);

Patt[0] = 999;//jt->ijet_etc[indx] ;
Patt[1] = 999;//jt->Eta ;
Patt[2] = 999;//pt ;
Patt[3] = jt->secvNtrk ;

```

```

Patt[4] = np1Pos ;
Patt[5] = np2Pos ;
Patt[6] = mass1Pos ;
Patt[7] = mass2Pos ;
Patt[8] = pt1Pos/pt ;
Patt[9] = pt2Pos/pt ;
Patt[10] = jt->jpbRPpos ;
Patt[11] = jt->secvL2d ;
Patt[12] = jt->secvL2d/jt->secvDl2d ;
Patt[13] = jt->secvL2d*jt->secvMass/jt->secvPt ;
Patt[14] = jt->secvChisq ;
Patt[15] = jt->secvMass ;
Patt[16] = jt->secvNvtrk ;
Patt[17] = jt->secvPt/pt ;!
Patt[18] = jt->secvPass ;
if(jt->secvTag==1){
    Patt[10] = jt->jpbRPneg;
    Patt[11] = fabs(Patt[11]) ;
    Patt[12] = fabs(Patt[12]);
    Patt[13] = fabs(Patt[13]);
}

NNoutBL(Patt,OutputBL);
NNoutBC(Patt,OutputBC);
nnbl = OutputBL[0];
nnbc = OutputBC[0];
//std::cout<<"OutputBL = "<<nnbl<<"\tOutputBC = "<<nnbc<<std::endl;
}
return (nnbl>fbl && nnbc>fbc);
}

void NNoutBC(float* pattern, float* output){
    int MaxIndim = 19;
    int Hidden = 12;
    int Outdim = 1;
    int ivalid = 0;
    float I_SUM = 0.0;
    float* H_0_SUM;
    H_0_SUM = new float[Outdim];
    // Input scales
    float scale[] = {1000.000000,10.000000,10000.000000,100.000000,100.000000
,100.000000,100.000000,100.000000,10.000000,10.000000
,10.000000,10.000000,1000.000000,1.000000,100.000000
,100.000000,100.000000,100.000000,10.000000};

```



```

,-1.919782,-1.919782};
    float H_O_Weight[] = {1.189701,-0.000942,-0.000942,1.057716,-0.000942
,-0.000942,-2.273642,-1.084268,1.420793,-0.846352
,-0.000942,-0.000942};
    float O_Thresh[] = {-0.028199};

    // Loop over output, hidden, and input nodes
    // to produce the neural net output
    //=====
    for (int out = 0; out<Outdim; out++){
H_O_SUM[out] = 0.0;
output[out] = 0.0;
    }
    for (int h = 0; h<Hidden; h++){
I_SUM = 0.;
ivalid = 0;
for (int i = 0; i<MaxIndim; i++){
    if(veto[i]){
        I_SUM += I_H_Weight[Hidden*ivalid+h] * pattern[i]/scale[i];
        ivalid++;
    }
}
I_SUM += H_Thresh[h];
for (int out = 0; out<Outdim; out++)
H_O_SUM[out] += H_O_Weight[Outdim*h+out]*Afunction(I_SUM);
    }
    for (int out = 0; out<Outdim; out++){
H_O_SUM[out] += O_Thresh[out];
output[out] = Afunction(H_O_SUM[out]);
    }
    delete [] H_O_SUM;
    return;
}

void NNoutBL(float* pattern, float* output){
    int MaxIndim = 19;
    int Hidden = 10;
    int Outdim = 1;
    int ivalid = 0;
    float I_SUM = 0.0;
    float* H_O_SUM;
    H_O_SUM = new float[Outdim];
    // Input scales
    float scale[] = {1000.000000,10.000000,10000.000000,100.000000,100.000000
,100.000000,100.000000,100.000000,10.000000,10.000000

```

```

,10.000000,10.000000,1000.000000,1.000000,100.000000
,100.000000,100.000000,100.000000,10.000000};
    // veto input patterns (0 == DO NOT USE)
    int veto[] = {0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1};
    // NN node weights and thresholds
    float I_H_Weight[] = {5.851072,-0.004697,-0.004699,-6.246035,-29.960289
,-0.004699,-0.004699,-0.004699,-8.175874,-0.004699
,-9.200892,-0.002473,-0.002473,8.811533,-0.135304
,-0.002473,-0.002473,-0.002473,0.187696,-0.002473
,-21.076284,-0.001853,-0.001854,9.993639,8.742924
,-0.001854,-0.001854,-0.001854,0.004845,-0.001854
,-0.054626,-0.001280,-0.001281,-22.885159,-7.268606
,-0.001281,-0.001281,-0.001281,7.883206,-0.001281
,-21.610380,-0.000979,-0.000979,-0.073648,0.253985
,-0.000979,-0.000979,-0.000979,-0.400699,-0.000979
,0.325248,-0.003385,-0.003386,5.357043,4.758751
,-0.003386,-0.003386,-0.003386,-0.122629,-0.003386
,-21.372839,-0.002838,-0.002839,7.855556,-0.093732
,-0.002839,-0.002839,-0.002839,6.905274,-0.002839
,17.638954,-0.000361,-0.000361,-74.662621,18.574038
,-0.000361,-0.000361,-0.000361,13.070364,-0.000361
,11.423511,-0.003454,-0.003455,4.655292,-4.692923
,-0.003455,-0.003455,-0.003455,-12.802147,-0.003455
,-18.424038,-0.001930,-0.001931,5.530622,1.103595
,-0.001931,-0.001931,-0.001931,-27.852163,-0.001931
,6.873257,-0.003076,-0.003077,-14.284559,-2.193805
,-0.003077,-0.003077,-0.003077,0.444641,-0.003077
,1.460633,-0.007985,-0.007987,0.261129,-2.757265
,-0.007987,-0.007988,-0.007988,0.218031,-0.007987
,-12.456869,-0.000859,-0.000859,13.988883,5.972669
,-0.000859,-0.000859,-0.000859,-0.010599,-0.000859
,-21.599915,-0.001732,-0.001733,12.576066,1.546479
,-0.001733,-0.001733,-0.001733,-14.802036,-0.001733
,0.174729,-0.000224,-0.000224,-6.964306,-0.204981
,-0.000224,-0.000224,-0.000224,0.278097,-0.000224
,2.836518,-0.009161,-0.009164,0.195712,-12.697416
,-0.009164,-0.009164,-0.009164,0.844316,-0.009164};
    float H_Thresh[] = {-0.000326,-1.968528,-1.968460,-0.837309,2.010411
,-1.968460,-1.968454,-1.968456,0.648109,-1.968458};
    float H_0_Weight[] = {-1.288710,-0.005778,-0.005779,1.970117,2.654743
,-0.005779,-0.005779,-0.005779,-1.185528,-0.005779};
    float O_Thresh[] = {-0.347765};

    // Loop over output, hidden, and input nodes
    // to produce the neural net output

```

```

//=====
for (int out = 0; out<Outdim; out++){
H_O_SUM[out] = 0.0;
output[out] = 0.0;
}
for (int h = 0; h<Hidden; h++){
I_SUM = 0.;
ivalid = 0;
for (int i = 0; i<MaxIndim; i++){
if(veto[i]){
I_SUM += I_H_Weight[Hidden*ivalid+h] * pattern[i]/scale[i];
ivalid++;
}
}
I_SUM += H_Thresh[h];
for (int out = 0; out<Outdim; out++)
H_O_SUM[out] += H_O_Weight[Outdim*h+out]*Afunction(I_SUM);
}
for (int out = 0; out<Outdim; out++){
H_O_SUM[out] += O_Thresh[out];
output[out] = Afunction(H_O_SUM[out]);
}
delete [] H_O_SUM;
return;
}

```

B Appendix: Network Settings

b efficiency	c efficiency	l efficiency	c l average	(NN bl , NN bc)
0.955	0.672	0.727	0.699	(0.061, 0.182)
0.953	0.710	0.612	0.661	(0.121, 0.152)
0.954	0.830	0.536	0.683	(0.182, 0.091)
0.948	0.625	0.836	0.731	(0.000, 0.212)
0.947	0.622	0.816	0.719	(0.030, 0.212)
0.947	0.653	0.652	0.653	(0.091, 0.182)
0.946	0.692	0.561	0.627	(0.152, 0.152)
0.947	0.753	0.526	0.639	(0.182, 0.121)
0.947	0.848	0.501	0.675	(0.212, 0.000)
0.947	0.848	0.501	0.674	(0.212, 0.030)
0.947	0.844	0.500	0.672	(0.212, 0.061)
0.941	0.602	0.704	0.653	(0.061, 0.212)
0.940	0.636	0.597	0.617	(0.121, 0.182)
0.945	0.810	0.494	0.652	(0.212, 0.091)
0.937	0.676	0.513	0.594	(0.182, 0.152)
0.938	0.735	0.484	0.610	(0.212, 0.121)
0.937	0.826	0.460	0.643	(0.242, 0.000)
0.937	0.826	0.459	0.643	(0.242, 0.030)
0.937	0.822	0.459	0.640	(0.242, 0.061)
0.932	0.555	0.786	0.671	(0.000, 0.242)
0.931	0.553	0.773	0.663	(0.030, 0.242)
0.933	0.586	0.633	0.609	(0.091, 0.212)
0.933	0.621	0.549	0.585	(0.152, 0.182)
0.934	0.790	0.453	0.621	(0.242, 0.091)
0.925	0.538	0.682	0.610	(0.061, 0.242)
0.926	0.570	0.580	0.575	(0.121, 0.212)
0.928	0.659	0.473	0.566	(0.212, 0.152)
0.928	0.716	0.444	0.580	(0.242, 0.121)
0.925	0.806	0.421	0.613	(0.273, 0.000)
0.925	0.806	0.420	0.613	(0.273, 0.030)
0.925	0.606	0.503	0.555	(0.182, 0.182)
0.925	0.802	0.419	0.611	(0.273, 0.061)
0.923	0.771	0.414	0.592	(0.273, 0.091)
0.919	0.525	0.617	0.571	(0.091, 0.242)
0.920	0.555	0.534	0.545	(0.152, 0.212)
0.916	0.591	0.463	0.527	(0.212, 0.182)
0.918	0.641	0.433	0.537	(0.242, 0.152)
0.916	0.699	0.405	0.552	(0.273, 0.121)

Table 6: The set of b , c and l efficiencies and corresponding network settings.

b efficiency	c efficiency	l efficiency	c l average	(NN bl , NN bc)
0.914	0.501	0.733	0.617	(0.000, 0.273)
0.914	0.499	0.725	0.612	(0.030, 0.273)
0.912	0.510	0.569	0.540	(0.121, 0.242)
0.912	0.542	0.489	0.516	(0.182, 0.212)
0.912	0.783	0.390	0.586	(0.303, 0.000)
0.912	0.782	0.389	0.586	(0.303, 0.030)
0.912	0.779	0.388	0.584	(0.303, 0.061)
0.909	0.488	0.651	0.570	(0.061, 0.273)
0.906	0.497	0.525	0.511	(0.152, 0.242)
0.906	0.575	0.423	0.499	(0.242, 0.182)
0.907	0.626	0.395	0.511	(0.273, 0.152)
0.910	0.749	0.383	0.566	(0.303, 0.091)
0.904	0.477	0.595	0.536	(0.091, 0.273)
0.903	0.528	0.450	0.489	(0.212, 0.212)
0.903	0.679	0.374	0.526	(0.303, 0.121)
0.897	0.447	0.682	0.565	(0.000, 0.303)
0.896	0.446	0.677	0.561	(0.030, 0.303)
0.898	0.465	0.553	0.509	(0.121, 0.273)
0.898	0.486	0.483	0.485	(0.182, 0.242)
0.895	0.560	0.387	0.474	(0.273, 0.182)
0.897	0.759	0.359	0.559	(0.333, 0.000)
0.897	0.759	0.359	0.559	(0.333, 0.030)
0.897	0.756	0.359	0.557	(0.333, 0.061)
0.895	0.727	0.353	0.540	(0.333, 0.091)
0.892	0.438	0.617	0.528	(0.061, 0.303)
0.892	0.453	0.510	0.482	(0.152, 0.273)
0.890	0.473	0.445	0.459	(0.212, 0.242)
0.894	0.514	0.411	0.462	(0.242, 0.212)
0.894	0.607	0.365	0.486	(0.303, 0.152)
0.888	0.429	0.569	0.499	(0.091, 0.303)
0.889	0.659	0.344	0.502	(0.333, 0.121)
0.883	0.419	0.536	0.478	(0.121, 0.303)
0.884	0.443	0.471	0.457	(0.182, 0.273)
0.881	0.460	0.406	0.433	(0.242, 0.242)
0.883	0.500	0.377	0.439	(0.273, 0.212)
0.883	0.543	0.357	0.450	(0.303, 0.182)
0.880	0.588	0.336	0.462	(0.333, 0.152)
0.883	0.738	0.326	0.532	(0.364, 0.000)
0.883	0.738	0.326	0.532	(0.364, 0.030)
0.883	0.735	0.325	0.530	(0.364, 0.061)
0.881	0.708	0.320	0.514	(0.364, 0.091)

Table 7: The set of b , c and l efficiencies and corresponding network settings.

b efficiency	c efficiency	l efficiency	$c\ l$ average	(NNbl, NNbc)
0.878	0.402	0.627	0.514	(0.000, 0.333)
0.878	0.401	0.625	0.513	(0.030, 0.333)
0.877	0.409	0.496	0.453	(0.152, 0.303)
0.877	0.431	0.435	0.433	(0.212, 0.273)
0.875	0.394	0.579	0.487	(0.061, 0.333)
0.871	0.388	0.541	0.464	(0.091, 0.333)
0.871	0.401	0.459	0.430	(0.182, 0.303)
0.871	0.447	0.372	0.410	(0.273, 0.242)
0.871	0.485	0.347	0.416	(0.303, 0.212)
0.875	0.642	0.312	0.477	(0.364, 0.121)
0.866	0.380	0.513	0.447	(0.121, 0.333)
0.868	0.419	0.397	0.408	(0.242, 0.273)
0.869	0.527	0.329	0.428	(0.333, 0.182)
0.866	0.572	0.304	0.438	(0.364, 0.152)
0.867	0.714	0.301	0.507	(0.394, 0.000)
0.867	0.714	0.301	0.507	(0.394, 0.030)
0.866	0.711	0.300	0.506	(0.394, 0.061)
0.862	0.372	0.480	0.426	(0.152, 0.333)
0.863	0.390	0.425	0.407	(0.212, 0.303)
0.865	0.686	0.295	0.490	(0.394, 0.091)
0.860	0.364	0.578	0.471	(0.000, 0.364)
0.859	0.364	0.577	0.470	(0.030, 0.364)
0.857	0.358	0.543	0.451	(0.061, 0.364)
0.856	0.365	0.445	0.405	(0.182, 0.333)
0.858	0.408	0.363	0.386	(0.273, 0.273)
0.859	0.433	0.344	0.388	(0.303, 0.242)
0.858	0.469	0.320	0.395	(0.333, 0.212)
0.855	0.512	0.298	0.405	(0.364, 0.182)
0.859	0.622	0.288	0.455	(0.394, 0.121)
0.854	0.353	0.516	0.435	(0.091, 0.364)
0.855	0.379	0.388	0.383	(0.242, 0.303)
0.850	0.554	0.281	0.417	(0.394, 0.152)
0.850	0.348	0.492	0.420	(0.121, 0.364)
0.846	0.341	0.463	0.402	(0.152, 0.364)
0.849	0.355	0.412	0.384	(0.212, 0.333)
0.845	0.368	0.355	0.362	(0.273, 0.303)
0.847	0.395	0.337	0.366	(0.303, 0.273)
0.846	0.419	0.317	0.368	(0.333, 0.242)
0.849	0.691	0.276	0.484	(0.424, 0.000)
0.849	0.691	0.276	0.484	(0.424, 0.030)
0.849	0.688	0.276	0.482	(0.424, 0.061)
0.848	0.664	0.271	0.468	(0.424, 0.091)

Table 8: The set of b , c and l efficiencies and corresponding network settings.

b efficiency	c efficiency	l efficiency	c l average	(NNbl, NNbc)
0.840	0.335	0.432	0.384	(0.182, 0.364)
0.841	0.345	0.377	0.361	(0.242, 0.333)
0.844	0.456	0.291	0.374	(0.364, 0.212)
0.842	0.603	0.265	0.434	(0.424, 0.121)
0.839	0.327	0.528	0.427	(0.000, 0.394)
0.839	0.327	0.527	0.427	(0.030, 0.394)
0.837	0.322	0.503	0.413	(0.061, 0.394)
0.840	0.496	0.275	0.385	(0.394, 0.182)
0.835	0.319	0.484	0.401	(0.091, 0.394)
0.832	0.314	0.464	0.389	(0.121, 0.394)
0.834	0.327	0.401	0.364	(0.212, 0.364)
0.832	0.336	0.345	0.340	(0.273, 0.333)
0.834	0.357	0.329	0.343	(0.303, 0.303)
0.834	0.382	0.311	0.346	(0.333, 0.273)
0.833	0.407	0.288	0.347	(0.364, 0.242)
0.834	0.537	0.259	0.398	(0.424, 0.152)
0.831	0.668	0.255	0.461	(0.455, 0.000)
0.831	0.667	0.255	0.461	(0.455, 0.030)
0.831	0.665	0.255	0.460	(0.455, 0.061)
0.828	0.309	0.441	0.375	(0.152, 0.394)
0.827	0.318	0.367	0.343	(0.242, 0.364)
0.829	0.442	0.268	0.355	(0.394, 0.212)
0.830	0.642	0.250	0.446	(0.455, 0.091)
0.824	0.304	0.414	0.359	(0.182, 0.394)
0.821	0.326	0.320	0.323	(0.303, 0.333)
0.822	0.345	0.304	0.324	(0.333, 0.303)
0.821	0.370	0.282	0.326	(0.364, 0.273)
0.824	0.480	0.254	0.367	(0.424, 0.182)
0.824	0.583	0.244	0.413	(0.455, 0.121)
0.819	0.292	0.485	0.388	(0.000, 0.424)
0.819	0.292	0.485	0.388	(0.030, 0.424)
0.817	0.288	0.465	0.377	(0.061, 0.424)
0.815	0.285	0.450	0.367	(0.091, 0.424)
0.818	0.297	0.386	0.341	(0.212, 0.394)
0.819	0.310	0.337	0.324	(0.273, 0.364)
0.818	0.393	0.265	0.329	(0.394, 0.242)
0.816	0.520	0.238	0.379	(0.455, 0.152)
0.813	0.281	0.433	0.357	(0.121, 0.424)
0.810	0.278	0.414	0.346	(0.152, 0.424)
0.812	0.289	0.354	0.322	(0.242, 0.394)

Table 9: The set of b , c and l efficiencies and corresponding network settings.

b efficiency	c efficiency	l efficiency	c l average	(NNbl, NNbc)
0.813	0.427	0.247	0.337	(0.424, 0.212)
0.813	0.648	0.232	0.440	(0.485, 0.000)
0.813	0.647	0.232	0.440	(0.485, 0.030)
0.813	0.645	0.232	0.438	(0.485, 0.061)
0.811	0.623	0.227	0.425	(0.485, 0.091)
0.806	0.274	0.391	0.332	(0.182, 0.424)
0.809	0.300	0.314	0.307	(0.303, 0.364)
0.809	0.315	0.295	0.305	(0.333, 0.333)
0.809	0.334	0.276	0.305	(0.364, 0.303)
0.807	0.358	0.260	0.309	(0.394, 0.273)
0.807	0.464	0.234	0.349	(0.455, 0.182)
0.806	0.566	0.222	0.394	(0.485, 0.121)
0.802	0.268	0.367	0.318	(0.212, 0.424)
0.804	0.282	0.326	0.304	(0.273, 0.394)
0.803	0.380	0.244	0.312	(0.424, 0.242)
0.797	0.261	0.442	0.352	(0.000, 0.455)
0.797	0.261	0.442	0.351	(0.030, 0.455)
0.795	0.258	0.426	0.342	(0.061, 0.455)
0.796	0.262	0.339	0.301	(0.242, 0.424)
0.797	0.289	0.289	0.289	(0.333, 0.364)
0.797	0.305	0.267	0.286	(0.364, 0.333)
0.796	0.323	0.253	0.288	(0.394, 0.303)
0.796	0.412	0.227	0.320	(0.455, 0.212)
0.799	0.505	0.217	0.361	(0.485, 0.152)
0.794	0.256	0.414	0.335	(0.091, 0.455)
0.792	0.254	0.402	0.328	(0.121, 0.455)
0.795	0.272	0.303	0.287	(0.303, 0.394)
0.792	0.346	0.239	0.292	(0.424, 0.273)
0.794	0.623	0.209	0.416	(0.515, 0.000)
0.794	0.623	0.209	0.416	(0.515, 0.030)
0.794	0.621	0.208	0.414	(0.515, 0.061)
0.792	0.601	0.204	0.402	(0.515, 0.091)
0.790	0.251	0.388	0.320	(0.152, 0.455)
0.787	0.249	0.370	0.310	(0.182, 0.455)
0.789	0.256	0.314	0.285	(0.273, 0.424)
0.785	0.281	0.262	0.271	(0.364, 0.364)
0.786	0.366	0.224	0.295	(0.455, 0.242)
0.789	0.450	0.213	0.332	(0.485, 0.182)
0.787	0.546	0.201	0.373	(0.515, 0.121)

Table 10: The set of b , c and l efficiencies and corresponding network settings.

b efficiency	c efficiency	l efficiency	c l average	(NN bl , NN bc)
0.783	0.244	0.351	0.297	(0.212, 0.455)
0.780	0.248	0.294	0.271	(0.303, 0.424)
0.784	0.263	0.279	0.271	(0.333, 0.394)
0.784	0.294	0.245	0.270	(0.394, 0.333)
0.781	0.311	0.235	0.273	(0.424, 0.303)
0.780	0.487	0.196	0.342	(0.515, 0.152)
0.778	0.239	0.325	0.282	(0.242, 0.455)
0.775	0.333	0.219	0.276	(0.455, 0.273)
0.779	0.400	0.207	0.303	(0.485, 0.212)
0.774	0.232	0.403	0.317	(0.000, 0.485)
0.774	0.232	0.402	0.317	(0.030, 0.485)
0.772	0.230	0.389	0.309	(0.061, 0.485)
0.771	0.228	0.378	0.303	(0.091, 0.485)
0.772	0.234	0.304	0.269	(0.273, 0.455)
0.772	0.255	0.253	0.254	(0.364, 0.394)
0.772	0.270	0.240	0.255	(0.394, 0.364)
0.771	0.435	0.192	0.313	(0.515, 0.182)
0.770	0.226	0.368	0.297	(0.121, 0.485)
0.768	0.225	0.358	0.291	(0.152, 0.485)
0.766	0.223	0.345	0.284	(0.182, 0.485)
0.770	0.239	0.271	0.255	(0.333, 0.424)
0.769	0.283	0.227	0.255	(0.424, 0.333)
0.769	0.355	0.204	0.279	(0.485, 0.242)
0.763	0.219	0.330	0.275	(0.212, 0.485)
0.764	0.227	0.285	0.256	(0.303, 0.455)
0.765	0.300	0.215	0.257	(0.455, 0.303)
0.761	0.386	0.187	0.286	(0.515, 0.212)
0.759	0.215	0.307	0.261	(0.242, 0.485)
0.759	0.232	0.247	0.240	(0.364, 0.424)
0.760	0.246	0.232	0.239	(0.394, 0.394)
0.758	0.260	0.222	0.241	(0.424, 0.364)
0.759	0.322	0.199	0.261	(0.485, 0.273)
0.754	0.211	0.288	0.249	(0.273, 0.485)
0.755	0.219	0.264	0.242	(0.333, 0.455)
0.754	0.273	0.208	0.240	(0.455, 0.333)
0.752	0.341	0.184	0.263	(0.515, 0.242)
0.750	0.203	0.375	0.289	(0.000, 0.515)
0.750	0.203	0.374	0.289	(0.030, 0.515)
0.748	0.202	0.362	0.282	(0.061, 0.515)
0.747	0.200	0.354	0.277	(0.091, 0.515)
0.746	0.199	0.345	0.272	(0.121, 0.515)
0.747	0.205	0.272	0.238	(0.303, 0.485)
0.747	0.224	0.226	0.225	(0.394, 0.424)
0.746	0.237	0.214	0.226	(0.424, 0.394)
0.749	0.291	0.195	0.243	(0.485, 0.303)

Table 11: The set of b , c and l efficiencies and corresponding network settings.