CDF/ANAL/HEAVYFLAVOR/CDFR/1914

12 February 1993

# Search for Top in Multi–Jet Events with a Neural Network: Part I

**A. Caner, B. Denby, J.L. Wyss**

***Fermilab***

**A. Castro, L. Stanco**

***Padova University and INFN, Padova***

**Abstract**

**A neural network classifier has been used to discriminate between Monte Carlo $t\bar{t}$ multi–jet events (generated with ISAJET and HERWIG) and real CDF multi–jet events from the 1988-89 run (considered as a pure background sample), using only the kinematical variables of the six leading jets. The classifier produces a probability variable which can be used to enrich samples in *top* . A comparison is made to the kinematical discriminant described in CDF 1871. The neural network is found to provide better a $S/B$ ratio for equal *top* efficiency. When applied to those events which were *selected* by the kinematical discriminant, cuts on network output can be used to significantly improve the $S/B$ ratio with very little loss of efficiency. In addition, for the lower values of *top* efficiency, this combined classifier is found to be superior to either method taken individually. $S/B$ ratios of order 1/10, with 5-10% absolute *top* efficiency are possible. The conclusion is that, with the application of known methods for improving even further the $S/B$ , including the SVX, the multi–jet channel will contribute significantly to the top search. An initial survey of data from the 1992/93 run shows that these data have properties similar to those of the 1988/89 data.**

# 1 Introduction

Although 44% of $t\bar{t}$ pairs decay hadronically, these multi–jet events have been thought to be of dubious utility in the top search due to extremely high backgrounds from generic QCD processes. As pointed out in [1], in the SUM_ET trigger data (summed transverse energies in all towers > 120 $GeV$) from the 1988/89 run ($\sim 4\ pb^{-1}$), the $S/B$ ratio for totally hadronic *top* ranges from $1/10^3$ to $1/10^4$. The top search in multi–jet events is difficult because the cross section for multijet QCD process is large compared to that for *top* and because the QCD events and *top* events, at the naïve level, resemble each other. However, the attributes of these two classes of events will certainly be quite different in detail, springing as they do from two distinctly different physical processes. The problem faced by the experimentalist is to determine how much of the underlying physical difference between the classes remains in the raw experimental variables which are available.

A number of studies have been done at CDF to try to find ways of reducing the background [2], [3], [4], [1], [5]. In particular, in [1], in which Monte Carlo $t\bar{t}$ is distinguished from real 1988/89 CDF SUM_ET data assumed to be pure background, it was found that, by requiring a topology with at least 6 jets, the $S/B$ ratio could be improved by about a factor of 100 by making simple kinematical cuts.

In the present analysis a feedforward neural network[1] was used to discriminate between Monte Carlo $t\bar{t}$ events (generated with ISAJET and HERWIG) and a sample of real CDF multi–jet data, using 18 input variables consisting of the $E_t$, $\eta$, and $\phi$ of each of the six leading jets in the event. The real multi–jet data, for the purposed of constructing the classifier, are assumed to be purely background (a discussion of this point appears in Appendix D). The real data used was the same multi–jet data sample used in [1] and is described briefly in section 2.

## 1.1 Multivariate Approaches

A cursory examination of the distributions of signal and background events in the kinematic variables shows substantial overlap (figure 1), although there are clearly differences between the two classes. Since no single one of these variable appears to be sufficient to provide good discrimination, it is reasonable to try multivariate approaches. In [6] and [3], a Fisher's linear discriminate was tried (NB: those analyses use different input variables from those used

[1] The type of classifier used is referred to in the pattern recognition literature as a *Multilayer Perceptron*. For simplicity, we shall simply refer to it as a 'neural network'.

here). In [1] the approach used was to create some new *ad hoc* variables using the 18 input variables and choose appropriate cuts on these variables by examining the distributions in them of the Monte Carlo *top* events and the real background events. [2]

## 1.2 Bayesian Probabilities

Another approach would be to divide the 18-dimensional input variable space into many bins and examine the population of signal and background events in each bin. The fraction of *top* events in each bin is then a measurement of the probability of an event falling in that bin to be a *top* event. This probability is an approximation of what is called the *Bayesian* probability. The true Bayesian probability is only obtained in the limit of an infinite statistical sample of both classes and an infinitely fine binning. A true Bayes classifier is an *optimal* classifier in the sense that it minimizes better than any other type of classifier the number of errors made in assigning events to one class or another. Although binned Bayesian classifiers have very good performance, they become impractical to construct when the dimensionality of the input space is large, due to the necessity to hand craft the binning to match the distribution of data in input space and the large amount of memory required. In practice, in most applications which estimate an optimal Bayes classifier, some kind of parametrization is used to simplify the implementation.

## 1.3 Neural Networks

Recently there have been numerous applications of neural networks to event classification in high energy physics (see [12] and references within for a summary) which demonstrate that in many instances their performance is superior to that which can be obtained with 'traditional' methods (see in particular [13] comparing various multivariate discriminants for B-tagging at LEP).

It can be shown that a feed forward neural network trained with a squared error cost function technique[3] produces a network output variable which approximates the Bayesian probability of an event to belong to a particular class [7]. A simple derivation of this, following [9], is given in the appendix. The use of a neural network parametrization has several advantages:

- The network output variable is itself an approximation of the Bayesian probability.

[2] That analysis also made use of non-kinematical variables. This point will be discussed again later.
[3] We shall discuss this training technique in more detail in the appendix.

- The neural network formalism, unlike likelihood techniques, does not make assumptions about the underlying probability distributions (e.g., gaussian, gaussian mixture). Rather, it estimates the Bayesian probability directly, from examples of the two classes of events [7].
- Likelihood classifiers ignore correlations in the input variables; in the neural network formalism, there is no restriction on correlations in the input variables.
- Neural networks can learn to classify from examples; this is a particularly useful capability when there is no known algorithm for performing the classification. This is clearly the case for the $t\bar{t}$ multijet problem, since no full tree level calculation of the six–jet QCD background is available. A neural network can instead learn the characteristics of the background from real events in the training set.
- Neural network classifiers are more efficient than linear classifiers such as the Fisher Discriminant since they handle naturally those cases in which the decision boundary between classes is nonlinear.
- Finally, neural networks may have an advantage over *ad hoc* techniques since they do not rely upon projections of the data onto arbitrarily chosen axes which may not provide optimal separation.

The preparation of data for the neural network and training strategy, are discussed in section 3. The results of applying the network to the data are presented in section 4. Section 5 shows the advantages realized in combining the neural network method with the analysis in [1]. A preliminary look at the behaviour of data from the current run using our classifier (trained on 1988/89 data) is given in section 6. A discussion of systematic uncertainties appears in section 7. Directions for further work (i.e., **Part II**) are presented in section 8 and conclusions in section 9.

## 2 Event Selection

### 2.1 Background

The data used for the background was the 1988/89 miniDST sample for the SUMET_120 trigger (495,452 events). The details of the data selection are reported in [1]. Briefly, the cuts used were:

- only one primary vertex
- at least 6 jets (clustered with $R = 0.7$) with $p_T \geq 10\ GeV$ within $|\eta| < 2.4$
- event passes Main Ring splash veto (see [1])

A further cut requiring total summed $E_T$ of the event to be greater than 150 *GeV* was also applied, leaving a total 5910 events available for the neural network analysis. [4] The ensemble of cuts used to produce these events will be referred to as the *loose cuts.*

### 2.2 Top Signal

Samples of 6,000 *top* events with masses, $m_t = 100, 120, 130$ and $150\ GeV/c^2$ were generated using ISAJET 6.43, and HERWIG 5.3. QFL 3.4 was used to simulate the detector response, and TRGSIM was used to simulate the trigger. The cuts placed on the signal sample were identical to those described above for the background sample.

In the ISAJET generation, the W's were allowed to decay in all modes (*inclusive* generation) in order to model the situation in the real data, in which all decay modes will also be present. In fact, even after the topological cuts, non-hadronic decay modes still account for some 25% of our signal [1]. In the generation with HERWIG, which has a completely different treatment of fragmentation from that of ISAJET, the W decays were forced to be hadronic(*exclusive* generation). The use of *exclusive* HERWIG allows us both to examine the effect of using Monte Carlos with different fragmentation models and to construct a classifier which is explicitly sensitive only to hadronic decays. The performance of such an exclusive classifier is a lower limit to what should be achievable. These arguments will be returned to in the section 7.

## 3 Data Preparation and Training Strategy

Neural network training proceeds by choosing a network architecture (i.e., number of hidden units, see figure 2), preparing a 'training set' consisting of a mixture of *top* events and background events, and running the training program. The performance of the net must always be evaluated using a statistically independent 'test set' of events not used in the training in order to insure that the network has not simply fit to statistical fluctuations in

[4] In [1], an additional cut requiring six jets *after* reprocessing the data in version 7_0 was made. This cut was not applied in the present analysis; therefore our total initial number of events, 5910, differs slightly from that in [1]

the training set data. We give here a summary of the training procedure. Details are to be found in the appendices A and B.

The training forces the network output to be as close as possible to '0' for background events and to '1' for *top* events (however see appendix A.3.1). In general the input variables alone are not sufficient to uniquely assign all events to one class or another; rather, the network output after training represents the *probability* of an event to be *top* , given the relative fractions of the two classes in the training set. We shall designate this probability as $P_{top}(\alpha)$ where $\alpha$ is the ratio of background to *top* .

The architecture chosen was a feed forward net with 18 input units (i.e, the 18 input variables), a single hidden layer with 5 units, and a single unit in the output layer (figure 2) whose output is the probability. The training set consisted of 1700 real CDF QCD background events and 1700 *top* events of the 4 different masses. In creating the training set, the 4 *top* masses were combined so as to create a classifier which was not sensitive to the *top* mass. (The populations of the different top masses in the training set were not equal: the higher masses had larger populations. We shall return to this point in section 4.3.) A cut of total summed jet $E_t > 240 GeV$ was placed on all events in order to restrict the operation of the network to that region where the *top* and background classes overlap (see figure 3). The motivations for these choices are elaborated in appendix A. The training procedure used was standard gradient backpropagation ([10] and appendix B).

A number of standard 'tricks' were employed in order to speed up and improve the training and to aid in choosing the optimum network architecture. As these may not be immediately obvious to the average reader, their discussion has been relegated to appendices A and C. It is important however here to remark upon one consequence: the resulting network outputs, ranging from -.9 to .9, need to be modified before they can be interepreted as probabilities:

$$P_{top}(1) = \frac{out + .9}{1.8}$$

where *out* is the network output, and $\alpha$ here = 1 since the populations of *top* and background events are equal in the training set.

The choice of equal populations of *top* and background in the training set clearly does not reflect reality. This choice was made because the training procedure achieves maximal separation power with equal populations. The result for an *arbitrary* $S/B$ ratio can then be easily obtained from:

$$P_{top}(\alpha) = \frac{P_{top}(1)}{P_{top}(1) + \alpha(1 - P_{top}(1))}$$

(see Appendix C for a derivation) which represents the probability that a particular event is *top* , for a given ratio, $\alpha$, of background to signal. For simplicity in most discussions, cuts for enriching samples in *top* will simply be placed on the network output variable, *out*.

# 4 Results

## 4.1 Output Distributions for Neural Net and Fisher Discriminant

The distributions of network output for events passing the *loose cuts* are shown in figure 4 for training set and test set, for ISAJET and HERWIG. [5] The test set network output distribution for ISAJET is repeated in figure 5a for comparison with the distribution of the Fisher linear discriminant, calculated for the standard set of 18 input variables, shown in figure 5b.[6] The performance of the Fisher discriminant appears, from the figure, to be poorer, but this can be substantiated in a quantitative comparison.

The relative performance of two classifiers can be gauged in a plot of purity versus efficiency. Such a plot is used in figures 6a and b to compare the neural network and Fisher classifiers for HERWIG and ISAJET. These plots are constructed by moving a cut in output variable across the test set distributions in figures 5a and b, and recording the fraction of events above this cut which are signal (purity) versus the fraction of total signal events above the cut (efficiency). (Recall that in the test set, the populations of signal and background are equal.) For both Monte Carlos, the performance of the Fisher discriminant is substantially worse than that of the neural net.

A cut on the Fisher variable is equivalent to choosing a single plane in input variable space to separate signal from background, while a neural net can combine several such planes to better approximate the optimal surface for this separation. The superiority of the neural net indicates that, for the present choice of input variables, this surface is curved.

[5] Note that the signal and background distributions for the training sets are slightly better separated than those for the test sets. This is normal, and does not reflect overtraining, which shows a much larger deficit in performance on the test set. Unbiased measurements of network efficiency must be based upon data from the test set.

[6] To allow a direct comparison of the two classifiers, the Fisher coefficients were calculated using the events in the training set, while in the distribution shown in figure 5b, the Fisher variable is calculated using data from the test set.

## 4.2 Explanation of Tables

The efficiencies in figure 5 are relative; absolute efficiencies, separated into the different masses are presented in tables I (HERWIG) and II (ISAJET). The first line of the tables shows the number of background events and absolute efficiencies of the loose cuts alone. The rest of the first section shows the number of background events retained and absolute *top* efficiencies, broken down into the four top masses, for no cut on network output (i.e., total summed $E_t > 240GeV$ cut only), and for cuts on network output of 0.0, 0.1, 0.2, and 0.3. The middle section of the tables shows the expected number of top events for the $4pb^{-1}$ 1988/89 run (using the calculation of [8] for the *top* cross section) for the four masses, for network output cuts at 0.0, 0.1, 0.2, and 0.3. In the bottom section, the $S/B$ ratios are presented for the same cuts on network output.[7]

The efficiencies for ISAJET are about a factor of 1.6 higher than those of HERWIG, both for loose and tight cuts. There are two reasons for this. First, as mentioned earlier, there are well known differences between the ISAJET and HERWIG Monte Carlos; here, these apparently have the effect of making HERWIG resemble the background more than does ISAJET. The second reason has to do with the fact that the HERWIG $t\bar{t}$ pairs decay exclusively hadronically, while ISAJET $t\bar{t}$ pairs were allowed to decay into any final state. The leptonic ISAJET decays, perhaps surprisingly, in fact make up about 25% of the mulit–jet sample. The remaining 35% of the difference can therefore be attributed to differences between the Monte Carlos.

The tables show that the neural network provides considerable discriminating power. For example, for $m_{top} = 150\ GeV/c^2$ the $S/B$ for the loose cuts alone is about 1/600, while for the ISAJET neural net, it becomes 1/33 for $out \geq\ 0.3$, a gain of a factor of 20.

## 4.3 Comparison to 'Tight Cuts'

The loose cuts were designed essentially to clean the data and to provide a sample of topologically well defined events. The $S/B$ ratios obtained with these cuts alone are clearly too small to permit a physics analysis of $t\bar{t}$ in the multi–jet channel. Clearly, even restricting oneself to kinematical variables as input, a wide range of different approaches to improving the $S/B$ ratios is possible.

[7] In order to avoid a small bias which would result from performing the cuts on the *training set* distribution, the cut efficiencies were calculated based on distributions from data which had not been used for training (including data from the test set as well as additional (signal) data contained neither in test nor train set). The numbers in the tables are thus *unbiased* in this sense.

The neural network analysis described above was carried out in parallel with the analysis based upon standard kinematical cuts, presented in [1], which we shall refer to here as the 'tight cuts'. The cuts used in that analysis are:

- $\sum_i E_T^i \geq 180\ GeV$
- $\sum_i E_T^i \geq 60 + 15 \times N_{jet}^{0.7}$
- $\sum_i |\eta^i| \leq 6.0$
- $\sum_i p_T^i/|\eta^i| \geq\ 380\ GeV/c$
- number of jets ($p_T \geq\ 10\ GeV/c, |\eta| \leq 2.4$) *with 0.4 clustering* $\geq 6$ [8]

where $N_{jet}^{0.7}$ is the number of $R = 0.7$ clustered jets in the event. All $R = 0.7$ clustered jets (not just the 6 leading jets) with $p_T \geq 10\ GeV/c$ and $|\eta| \leq 2.4$ are included in the above sums, except for $\sum_i E_T^i$, which includes all ($R = 0.7$ clustered) jets.

Although both analyses use the same data set as a starting point (i.e., the loose cut data), and the same 18 topological variables, the tight cuts use additional variables not used in the neural network analysis, i.e., information on jets beyond the six leading jets and on jets clustered with a cone of $R = 0.4$. In spite of these differences, it is instructive to compare the two approaches. The last lines in each of the blocks of tables I and II present the efficiencies, expected number of top events, and $S/B$ ratios for the four masses as determined in the tight cut analysis.

The tables show that neural network approach is very competitive. For very similar efficiencies, the net gives $S/B$ ratios which are superior to those of the tight cuts. The neural network has its greatest advantage for $m_{top} = 150\ GeV$ where the $S/B$ ratio is more than a factor of 2 better than that of the tight cuts, for almost identical efficiency. We shall return to this point in the discussion of figure 9 in section 5.

The tables show that the neural network efficiency is higher for the larger *top* masses. This effect may be in part due to the higher population of the larger masses in the training set, as mentioned in section 3. It is also however possible that the lower mass events resemble more the background events. To determine which of these two effects is the dominant one, it will be necessary to repeat the analysis with equal populations of *top* masses in the training set. This is reserved for future work **Part II.**

[8] Note that this is not really a kinematic variable but concerns the shapes of individual jets.

It is important to understand whether the two classifiers are in fact selecting the *same events*. This point forms part of the discussion in next section.

## 5 Combining the Two Classifiers

Because the neural network analysis and the tight cuts use some different input information, it is interesting to enquire whether the two classifiers can be combined to advantage.

Figure 7 shows the network output, for HERWIG and ISAJET, for the tightly selected signal and background. It is reassuring to note that the effect of the tight cuts is to remove events with low *top* probability. For the signal events, this only skews the distribution slightly towards higher probabilities. For the background events, however, the effect is quite pronounced; in fact, the background distribution, which was previously peaked at low probability (figure 4) is now essentially flat. The fact that the background distribution is not completely depleted at low probability, however, implies that additional discriminating power can still be had by placing a cut on the network output variable. The effect of placing such cuts is summarized in tables III and IV, and in figure 9, described below.

The above arguments can be stated in another way which addresses the question of whether the two classifiers choose the same events. The neural net discards little of the tight cut data: a cut on network output of *out* $> 0$, discards only about 20% of the signal (while rejecting about 50% of the background). The tight cuts, however, have a harsher effect on signal events selected by the neural network, as can be seen in figures 8a and b, which show a superposition of ISAJET network output distributions for the loosely and tightly cut data, and their ratio, respectively. The tight cuts seem to discard events at all values of *out* with equal probability. The conclusion is that the neural network classifies most of the tight cut data as having high *top* probability; however, there are a significant number of other events with high *top* probability which the tight cuts reject.

Tables III (HERWIG net) and IV (ISAJET net) show the results of applying the neural network classifier to background and *top* events selected with the tight cuts. The format of these tables is identical to that of tables I and II.[9] The neural network has provided an

[9] The efficiencies quoted are based upon the *entire sample* of 344 events surviving the tight cuts, including those which appeared in the training set. The efficiencies are then 'biased' in the sense described in section 4.1. When the efficiencies in the table were recalculated with events which were not used in training, the differences, in cases where statistics were adequate, were found to be only a few percent of the figures quoted. In some cases, however,the statistical error introduced by using the unbiased number exceeded the expected sise of the bias itself. For this reason the 'biased' values were retained for use in the table. This has a negligable effect on the results of the analysis.

additional cutting variable which can be used to improve $S/B$ beyond what the tight cuts alone give, albeit with somewhat reduced efficiency. For a cut at *out* $> 0.3$ for instance, the net improves the $S/B$ by about a factor of 3 while reducing efficiency by only about 30%.

However, the relative performance of the tight cuts, neural network, and combined classifier can perhaps best be understood by examining figure 9, which shows, for the ISAJET classifier, the reciprocal of the $S/B$ ratio versus absolute *top* efficiency for the 4 *top* masses. In each plot, the upper right–most dark circle (network alone) or open square (combined classifier) represents *no* cut on network output; the 'envelopes' then evolve as the cut is moved across the network output distribution. This is similar to what was done in section 4.1 for the purity versus efficiency curves, except that here the efficiencies are absolute.

The near verticality of the right hand side of the combined classifier's envelope reflects the fact mentioned above that progressively tighter cuts on network output significantly improve the $S/B$ ratio with little loss in efficiency. It should be noted, however, that in those regions where the circle envelope is below that of the squares, superior performance can be had simply by cutting more tightly on net output *without applying the tight cuts.* At the lower values of efficiency, the envelope of the combined classifier intersects and dips below that of the neural network alone. In these regions, where the envelopes are nearly horizontal, the combined classifier give 20-30% better $S/B$ ratios than the neural network alone.

The fact that the combined classifier has better performance than either of the two taken separately can have two possible causes:

- One or both of the classifiers has not reached the Bayes limit for kinematic variables alone. If one or both had, combining the two could not have brought improvement.
- The improvement may be due to the fact that the tight cuts make use of information that the neural network did not use, i.e., information on the total number of jets and on jet shapes.

It is suspected that the latter reason is the more relevant, for reasons which will be elaborated in section 6.2.

## 6 The 1992/93 Data

### 6.1 Application of 1988/89 Networks

The data from run Ia was collected via the multi–jet *top* trigger, TOTAL_ET_CL_100_6JETS, at level 2, processed from Stream2 and selected in the *QCDX* split. Standard checks of data quality revealed no particular problems, and it was determined that the same cuts could be used to select the 1992/93 multi–jet sample as were used for the 1988/89 sample.

ISAJET and HERWIG network output for a 2.5 $pb^{-1}$ sample of new data selected by the loose cuts are shown in figures 10a and b. The shapes of the distributions are similar to those for the 1988/89 data.

Two points regarding the interpretation of the networks outputs are worth mentioning. First, since these data have never been seen by the network, any questions of test set versus training set 'bias' (as discussed in section 5) are not relevant. The networks trained on the 1988/89 data can therefore be directly applied to the new data. The second point, however has to do with the trigger. The *top* trigger for 1992/93 is different from that used in 1988/89 and thus in principle can produce relative probability distributions for signal and background which are different from those used in training. There are two alternatives for dealing with this: 1) Make a correction to the probabilities based on a measurement of the trigger efficiency for top and background; or 2) Retrain the network using data from the 1992/93 run. These questions will be dealt with in **Part II.**

### 6.2 Effect of Non-kinematic Cuts

The relative importance of the 'non–kinematical' tight cut, i.e., on the number of jets found with $R = 0.4$ clustering was assessed by examining a set of the 1992/93 data before and after application of this cut. The HERWIG network outputs for these two data samples are shown in figure 11. The figure shows that the application of this cut preferentially depletes events in the low *top* probability region, i.e., that it is a useful cut for *top* /background discrimination.

### 6.3 B–Tagging

One major advantage of the 1992/93 data is that they contain information from the SVX. As we shall discuss in the conclusions, the multi–jet *top* analysis (as for the other channels) will ultimately have to rely upon this information to unequivocally establish the existence of a top signal.

A B–tagging algorithm has been applied to a 1992/93 data sample corresponding to 6.8 $pb^{-1}$. Our algorithm [1] is based on the search for at least one secondary vertex using the tracks belonging to a jet and well reconstructed inside the SVX. Thirteen events are tagged. The HERWIG and ISAJET network output distributions for these events are shown in figures 12a and b. If we define samples with *out* > 0.3 as being 'top enriched', then any B–tagged event in that region will be interesting to evaluate in more detail. No candidates with *out* > 0.3 for *both* Monte Carlos are found in the current subset of data. However with a relaxed criterion, *out* > 0.0, two events are found: (run=43123, event=321735), (run=43368, event=68233). A lego plot of the second event is shown in figure 13.

# 7 Systematic Uncertainties

## 7.1 Inclusive versus Exclusive Generation

It is important to ask whether the neural network may have a different efficiency for leptonic $t\bar{t}$ than for hadronic $t\bar{t}$ , since the two types of events will clearly have some differences. To check this, the ISAJET sample was separated into leptonically and hadronically decaying events and these two samples were passed separately through both nets. Although a detailed study was not done, the network output distributions for these two classes are very similar, with perhaps a slight tendency towards worse identification for the leptonic events (see figure 14).

## 7.2 Intrinsic Uncertainty in Measuring $P_{top}(1)$

A neural network classifies events by partitioning input variable space with hyperplanes (one hyperplane per hidden unit), with the orientations of the hyperplanes being determined by the weight matrices. For a given neural network architecture, because of limited statistics, several different sets of hyperplanes (i.e., different sets of weights) may give essentially identical classification performance. However, the resulting classifiers may give somewhat different answers for $P_{top}(1)$ on an event by event basis, resulting in a systematic uncertainty in $P_{top}(1)$.

Figure 15 compares the network outputs for classifiers which were trained with different random initial sets of weights, but which had nearly identical *purity versus efficiency* curves at the end of training. Figure 15a shows the difference of network output for two different

ISAJET networks, while figure 15b shows a scatter plot of one of these networks versus the other. Figures 15c and d repeat these plots for two HERWIG networks. For both Monte Carlos, the widths of the difference plots are about .22, corresponding to $\Delta_{NET}P_{top}(1) \sim 8.5\%$. [10] A measure of the uncertainty due to model dependence can be had by comparing the outputs of networks trained with the two Monte Carlos. In order not to bias the measurement towards ISAJET or HERWIG, this measurement was made on a sample of $m_{top} = 130\ GeV/c^2$ data generated by a third Monte Carlo, PYTHIA. The $t\bar{t}$ events in this sample were allowed to decay into any allowed final state (i.e., inclusive generation). Figures 16a and b are the outputs of the HERWIG and ISAJET trained networks for the PYTHIA data. The two distributions look qualitatively very similar. Figure 16c plots the difference between the outputs of the two networks on this data sample, and figure 16d is a scatter plot of HERWIG versus ISAJET. The curve in figure 16c has a mean near zero, indicating no overall shift between ISAJET and HERWIG, and an RMS of .34, which implies $\Delta_{TOT}P_{top}(1) \sim 13\%$. This contains contributions from $\Delta_{NET}P_{top}(1)$ as well as from the model dependence.

As a further check of this result, figure 17a shows the difference between the HERWIG and ISAJET network output for the 1992/93 data. The mean here is about .04, which is consistent with the slight difference in means of the *out* distributions for HERWIG and ISAJET. This overall offset is small compared to the RMS of the curve, .36, and can be ignored. The width corresponds to $\Delta_{TOT}P_{top}(1) \sim 14\%$, in good agreement with the measurement made on the PYTHIA data. Figure 17b is a scatter plot of HERWIG output versus ISAJET output for this data set.

Plots like those in figures 16 and 17 were also made for HERWIG versus ISAJET networks operating on ISAJET and HERWIG data. Similar results for $\Delta_{TOT}P_{top}(1)$ were obtained.

Assuming that the errors from $\Delta_{NET}P_{top}(1)$ and from model dependence add in quadrature, we obtain

$$\Delta^2_{MC}P_{top}(1) = \Delta^2_{TOT}P_{top}(1) - 2\Delta^2_{NET}P_{top}(1)$$

$$\Delta_{MC}P_{top}(1) = \sqrt{(.14)^2 - 2(.085)^2} = 7\%$$

[10] There is a scale factor of 1.8 between *out* and $P_{top}(1)$, and we may divide by $\sqrt{2}$ since *both* Monte Carlos contribute to the width.

as the residual uncertainty due to the choice of Monte Carlo. The more relevant quantity, however, is the uncertainty associated with using $P_{top}(1)$ from a *particular* Monte Carlo, $\Delta_{HERWIG}P_{top}(1) \sim \Delta_{ISAJET}P_{top}(1) = \sqrt{(.14)^2 - (.085)^2} = 11\%$.

## 8 Directions for Future Work

The addition of B-tagging from the SVX will surely be one of the major improvements that can be made to the present analysis. An improvement of a factor of 20 in $S/B$ is estimated in [1]; this would provide overall $S/B$ ratios of order unity. However there are also other avenues which should be explored.

When the tight cuts were combined with the neural network, in certain regions of $S/B$ –efficiency space, performance was improved. One explanation for this, as was mentioned earlier, is that the tight cut analysis used additional information on the jets beyond the leading 6 and on the number of jets clustered with a $R = 0.4$ cone size. This latter is not a true kinematical variable, but rather contains information about the shapes of the individual jets.

Quark jets are thought to be more collimated and have lower multiplicity than gluon jets. Jet shape information should be important for $t\bar{t}$ analysis since these events will contain 6 quark jets, whereas QCD background multi–jet events will contain predominantly gluon jets. Additional hints that such jet shape information may be useful come from [1] and [6]. In [6], a factor of 100 improvement in $S/B$ for multi–jet *top* events is had by making cuts on the number of calorimeter towers above an $E_t$ threshold and on the number of charged tracks above a $P_t$ threshold (considered for the event as a whole). These cuts will have a tendency to select events with jets that have fewer, high $P_t$ tracks. Although that analysis used Monte Carlo background rather than real data, and a different set of input variables, it should nonetheless be interesting to see what effect such cuts may have for the CDF analysis.

Evidence for differences in shape between quark and gluon jets in CDF data has already been presented [14]. In that analysis, a feed forward neural network was trained on Monte Carlo quark and gluon jets to produce a quark probability variable. The mean value of this variable for samples of CDF dijet data was found to increase with jet $E_t$, consistent with a higher fraction of quark jets at higher $E_t$. A project is currently in progress to apply this network to jets in the $t\bar{t}$ Monte Carlo and CDF multi–jet samples in order to try to find a variable which can be used to help distinguish *top* from background. Plots of the variables used in [6] will also be made to determine if they provide additional separation power.

Another approach will be to create a new network with a new set of input variables which includes information on jets beyond the first 6, the quark probability variable, and the SVX information. Such a network will be able to correlate the B-tagging information and quark probability with specific jets, which is potentially a very powerful piece of information.

In summary, then, the directions for **Part II** will include:

- Continue to validate data from the new run and continue, via B-tagging to search for multi–jet *top* candidates.
- Understand the effect of the 1992/93 trigger on the *top* probability distribution. Possibly retrain the network with new data.
- Investigate using quark probability for the jets to improve *S/B* .
- Look at distributions of the variables used in [6].
- Try retraining a new network which includes more information: jets beyond the first 6; quark probability; and SVX information.
- Resolve the questions about the ratios of the populations of the different masses in the training set raised in section 3.
- Try to recover some of the events lost in the cut on total summed jet $E_t$ of 240 $GeV$.

## 9 Conclusion

We have shown that a neural network classifier, trained on real CDF multi–jet background and HERWIG and ISAJET *top* events, provides an output probability variable which can effectively be used, with high *top* efficiency, to provide data samples enriched in *top* . When the classifier was used in conjunction with another classifier employing additional information, results are improved even further. *S/B* ratios of order 1/10 are possible, for absolute *top* efficiencies of 5-10% (depending on the *top* mass). This performance by itself is probably not adequate for a physics analysis of *top* decays at the collider; however, additional means, including the SVX, may be used to produce *S/B* ratios of order unity or better. The multi–jet channel in that case will be very important for the *top* search at CDF.

## 10 Acknowledgements

D. Bisello, G. Busetto, T. Dorigo, M. Loreti, and L. Pescara have contributed to the analysis of the multi–jet events which were the input to this analysis. We also acknowledge useful conversations with J. Benlloch, J. Conway, M. Dickson, F. Fogelman-Soulié, R. Hamilton, G. Pauletta, and N. Wainer. Contributing to the neural network training software were S. Bianchin, N. Chriss, M. DallAgata, M. De Nardi, E.S. Lessner, L. Santi.

APPENDICES

# A Data Preparation and Training Strategy

The parameters of a neural network classifier, rather than being specified by an explicit algorithm, are 'learned' in a procedure in which the classifier is presented with a 'training set' containing examples of the two classes of events to be separated. This procedure will be described in more detail below, but first a couple of points regarding the choice of the examples in the training set must be elaborated.

## A.1 A Mixture of Top Masses

Any technique to separate *top* events from generic QCD background will depend upon the assumed mass for the *top* quark. One may then either construct a set of classifiers with different assumptions for the *top* mass, or construct a single classifier which is largely independent of the *top* mass. For a neural network classifier, one way to implement the latter strategy would be to prepare a training set containing equal populations of events with several different *top* masses. This was the spirit of the approach used in the current analysis; however, due to the strong mass dependence of the *top* efficiency of the loose cuts, the actual ratio of of masses 150:130:120:100 $GeV/c^2$ turned out to be about 3:2:2:1. Due to the training procedure used, this implies that the network may be more efficient at selecting the higher masses. In retrospect, in order to avoid this additional complication, it might have been wiser to generate enough *top* signal to equally populate the different masses.

## A.2 Removing 'Trivial' Correlations

One of the main advantages of of a neural network classifier is its ability to discover multivariate correlations in the input variables, correlations which cannot easily be visualized in one and two dimensional projections. In fact there is little to be gained from training a neural network to make classifications based on cuts which could as easily have been determined from a cursory examination of the input variables. One can bring to bear the full power of the neural network technique by making *a priori* cuts on the input data which remove those cases which can be classified 'trivially'.

In our case it was found that the distributions of the summed $E_t$ of the six leading jets for the two classes have rather different shapes (figure 2). In fact it is only in the region above about 240 *GeV* that classification becomes ambiguous. We have chosen to concentrate on this region. A total of 1253 events survive the cut. Two points must be stressed about the cut at 240 *GeV*:

- The training set for a neural network classifier must be large enough to well represent the statistical diversity of each class, and the relative populations of the two classes should be equal. [11] In addition, it is necessary to set aside a statistically independent 'test set' of events not used for training in order to monitor the performance of the net and to avoid overfitting to the training data (as discussed below). Four hundred each of background and signal events were reserved for this purpose. This however left only 850 background events, which experience dictates is marginal for training on a problem of this complexity. In order to augment the training set size, an additional 850 events were artificially produced by 'flipping' $\eta$ for $-\eta$ for each jet in the 850 remaining background events and including these also in the training set. Taking advantage of known symmetries in order to generate new training patterns is a standard technique in neural network training.

  Such 'flipped' events are clearly valid from a physical standpoint. They do not substantially enrich the statistical diversity of the training set, but help the training by populating regions of input variable space which the network would otherwise not have sampled. The test set thus consisted of these 1700 real data background events and an equal number of *top* events with the ratio of the populations of the masses as described above.

- The cut in fact eliminates a non-negligable fraction of the signal events, and this must be taken into account when calculating efficiencies. Presumably a more sophisticated network could attempt to recover some of the signal events lost, but we have not attempted to do so in the present work. This will be attempted in **Part II.**

[11] The populations should actually reflect the true relative populations of the two classes, in order that the network output be interpretable as a probability. In our case, the ratio of *top* to background is exceedingly small, which leads to problems in training. Instead, we artificially scale up the fraction of signal to 50% and correct the network output probabilities after the fact. See also section 3.

## A.3 Training the Neural Network

### A.3.1 Training Set

Events of different top mass, and the background data, existed in separate files. In making the training file, the input file for each consecutive event was chosen at random. Thus, the signal events of different masses and the background data are mixed at random throughout the training file. If the files instead had entered the training set consecutively, the network would tend to 'forget' the mass it has just 'learned' as it passes to the next, and so on.

Before training, the input variables in the training set were shifted and scaled so as to have zero mean and unit variance. This is done purely as a means of speeding up the training procedure [11] and does not affect the performance of the classifier. It amounts simply to a linear distortion of input variable space which does not affect the topological relationship of the two classes.

Each element in the training set consists of the 18 input variables and a 'target' value. During training, the network tries to make its output come as close as possible to this target value for each event. The target is set to -.9 for background events and +.9 for signal events. The choice of $|target| = .9$ decreases training time since it forces the training to operate in a range of finite slope of the neuron transfer function. The incremental weight change per iteration is proportional to this slope (see [11]). The choice of target value determines the range of the network output. With a choice of $|target| = .9$, the network output can be converted to a probability (see appendix C) through the equation:

$$P_{top}(1) = \frac{out + .9}{1.8}$$

where $P_{top}(1)$ is the probability that an event with a network output of *out* is a *top* event, for equal *top* and background populations in the training set. As mentioned earlier, the training set contained 1700 real data background events and 1700 Monte Carlo *top* events of all masses.

## A.4 Choice of Network Architecture

The architecture chosen was a standard feedforward neural network with an input layer of 18 neurons, corresponding to the 18 input variables, a single hidden layer, and one neuron in the output layer (figure 3). Training was done using standard gradient backpropagation ([10] and appendix B) with the training set described above.

Generally speaking, the discrimination power of a neural network increases with the number of units in the hidden layer. The Fisher Discriminant is equivalent to a feedforward neural network without hidden units. In the case of a linear decision boundary between two classes, the Fisher Discriminant is the optimal classifier, and addition of hidden units cannot improve on its performance. Additional hidden units are required in the case of a non–linear decision boundary, and should be added until the network is able to approximate well this boundary.

The appropriate number of hidden units also depends upon the amount of available training data. If the number of hidden units, and thus the number of parameters needed to describe the network, is too large, the training procedure will cause overfitting of the data, i.e., the network starts to fit to statistical fluctuations in the training data. This is analogous to fitting a curve to a set of points which lie essentially on a line but with some scatter. A linear function will produce a good fit with reasonable residuals. A fit to a function with many parameters will eventually pass exactly through all the points, with zero residuals; however, such a function may have a quite complicated shape in order to do so. Overfitting in neural network training is usually called 'overtraining'. Thus in some cases, one may be obliged to use a less powerful classifier than the problem warrants, simply because not enough data exists to accurately determine the parameters of the optimal classifier.

The standard procedure is first to try the Fisher Discriminant (zero hidden units), and then progressively more complex architectures until the addition of more hidden units no longer improves performance. For each new architecture, care must be taken to avoid overtraining. This can be easily monitored in the following way. As a network is trained, initially the performance of the network improves both on the training set and on the statistically independent test set. As overtraining begins to occur, the performance on the training set continues to improve but the performance on the test set begins to deteriorate (just as the many parameter fit to the line data will have poor residuals for new data points). The optimal weights are those existing before this deterioration sets in.

In the present analysis, a study was made of architectures with 2, 5, and 10 hidden units. Training continued for 8196 'epochs', or passes through the entire training set. In order to avoid the possibility that the backpropagation training become stuck in a local 'energy' minimum (see the next section for a discussion of backpropagation training), several sets of random starting weights were tried for each architecture, and the set giving the best final discrimination was chosen. Periodically during training the network, performance on the test set was evaluated, in order to be sensitive to the onset of overtraining. When optimally trained, the 5 unit net had better performance than the 2 unit net, but the 10 unit net showed no improvement over the 5 unit net. The five hidden unit net was then chosen as the standard architecture for this analysis. The relative performances of the nets with different architectures and starting configurations were assessed using the type of purity versus efficiency plots described in section 4.1.

## B Backpropagation Training

The neuron activation, $out_k$, of a neuron $k$ is multiplied by a weight, $w_{jk}$ and appears as $w_{jk}out_k$ at the input of neuron $j$. (The input 'neurons' are not function units, but just fanouts for the input variables.) Neuron $j$ forms at its input the sum,

$$x_k = \sum_k w_{jk} out_k$$

and produces at its output the nonlinear transfer function (see figure 2),

$$out_k = \frac{e^{x_k} - e^{-x_k}}{e^{x_k} + e^{-x_k}}.$$

We define an error or 'energy' function, $E$ by

$$E = \sum_l E^l = \sum_l \sum_j (out_j^l - t_j^l)^2$$

where $out_j^l$ is the network output for output neuron $j$ (in the general case there can be many output units) on event $l$ (using the current set of weights) and $t_j^l$ is the desired or 'target' output for this event. We wish to minimize the total squared error committed by the network by adjusting the set of weights $\{w\}$. We can do this by performing gradient descent on $E$ with respect to the weights:

$$\frac{1}{2}\frac{\partial E^l}{\partial w_{ji}} = \frac{1}{2}\frac{\partial}{\partial w_{ji}}(out_j^l - t_j^l)^2$$

$$= (out_j^l - t_j^l)\frac{\partial out_j^l}{\partial w_{ji}}$$

$$= (out_j^l - t_j^l) out_j^{l\prime} out_i^l \quad \text{for output units}$$

$$= [\sum_k (out^l_k - t^l_k) out^{l\prime}_k w_{kj}] out^{l\prime}_j out^l_i \quad \text{(for hidden units).}$$

where $out^{l\prime}$ refers to the derivative of $out$ with respect to its argument. The backpropagation expression for the change in the weight $w_{ji}$ due to event $l$, in iteration $N$, is

$$\Delta^l_N w_{ji} = -\epsilon \frac{\partial E^l}{\partial w_{ji}} + \beta \Delta^l_{N-1} w_{ji}$$

where $\epsilon$, called the 'learning coefficient', tells the distance along the energy gradient to move in this iteration, and the term containing the constant, $\beta$, called the 'momentum' coefficient, is a smoothing term. The total weight change in iteration $N$ is the sum over all events $l$, however in practice, weights are often updated much more frequently than once per epoch, or pass through the training set. For the training in the present work, $\epsilon = .001$ and $\beta = .25$ were used, and updating was done every 10 events. Typically 1000-4000 passes through the training set (1700 signal + 1700 background) were required to achieve optimal weights.

## C Equivalence of Neural Network Output and Bayesian Probability

We use the expression above for $E$, for one output unit, and take $t_i$ to be -.9 for background and +.9 for *top* . This can also be expressed as,

$$E = \int [n_{top}(\vec{x})(out(\vec{x}) - .9)^2 + n_{QCD}(\vec{x})(out(\vec{x}) + .9)^2] d^n\vec{x}$$

where $\vec{x}$ is the $n$–dimensional vector of input parameters, and $n_{top}(\vec{x})$ and $n_{QCD}(\vec{x})$ are the numbers of *top* and *QCD* events in the region $d^n\vec{x}$.

The purpose of the backpropagation training is to find a set of weights, i.e., a functional form of $out(\vec{x})$, which gives the global minimum of this expression. Under the conditions that the network have sufficient complexity (i.e., hidden units) to achieve the desired mapping, that the training set represents the full statistical diversity of the mapping, and that several sets of initial conditions have been tried in order to ensure that the minimum found is indeed global and not local, we may assume that the backpropagation training has found the optimal form for $out(\vec{x})$. We may then explore the consequences of this extremum condition on the functional form of $out(\vec{x})$. We can interpret the integral above as a sum over many small regions of space. Since each term is positive definite, we must chose *out* to minimize each term separately. Within each small region, *out* is just a parameter, which allows us to differentiate with respect to *out* and set the result equal to zero. We obtain,

$$0 = n_{top}(\vec{x})(out(\vec{x}) - .9) + n_{QCD}(\vec{x})(out(\vec{x}) + .9)$$

so that

$$out(\vec{x}) = \frac{.9[n_{top}(\vec{x}) - n_{QCD}(\vec{x})]}{[n_{top}(\vec{x}) + n_{QCD}(\vec{x})]}.$$

Making use of the fact that

$$P_{top}(1) = \frac{n_{top}(\vec{x})}{[n_{top}(\vec{x}) + n_{QCD}(\vec{x})]}$$

where we now suppose that the total populations of *top* and $QCD$ events are equal and $P_{top}(1)$ is the probability that an event with input vector $\vec{x}$ is *top* for equal populations. This gives

$$out(\vec{x}) = .9[2P_{top}(1) - 1]$$

or

$$P_{top}(1) = \frac{out(\vec{x}) + .9}{1.8}.$$

For an arbitrary ratio of $QCD$ events to *top* events, $\alpha$, we need simply multiply the number of $QCD$ events in any small region $d^n\vec{x}$ by $\alpha$, so that

$$P_{top}(\alpha) = \frac{n_{top}(\vec{x})}{[n_{top}(\vec{x}) + \alpha n_{QCD}(\vec{x})]}.$$

It is then straightforward to show that

$$P_{top}(\alpha) = \frac{P_{top}(1)}{P_{top}(1) + \alpha(1 - P_{top}(1))}$$

## D The Effect of Possible *top* Events in the 'Background' Sample: A Heuristic Argument

The question of the effect of possible *top* events in the 'background' training sample has often been raised. We consider here the case where the 'contamination' from *top* is very small, which will be the case for the $t\bar{t}$ to multi–jets case since the $S/B$ ratios for the data after the loose cuts are of order 1/600.

The problem can be treated by considering a two dimensional model (figure 18). We let the *top* events (crosses) and background events (squares) be described by only two variables and suppose that they are distributed in the space of these variables as shown in the figure. Two events in the background sample are actually *top* events (square with the letter 't' inside). In a) we show a plausible classification boundary as might have been created by a neural network with 4 hidden units (4 sided polygon). As is typical in many classification problems, some of the events in each class are in fact on the wrong side of the boundary. These will contribute to the error committed by the network (appendix A). The two *top* events in the background sample are in fact in the correct region, since they are *top* , but as far as the network is concerned, they are misclassified, and these, too, will contribute to the error.

If we continue to train the 4 hidden unit net, the polygon may move around slightly, but, if there are not too many mislabelled *top* events, will never be able to dramatically reduce the error committed by the net. If on the other hand we now introduce many more hidden units, the network will be able to build a much more complicated classification boundary, ultimately arriving at a situation as in b), in which perfect classification is achieved. Note that from the standpoint of the network, the two *top* events in the background sample are treated in exactly the same was as signal or background 'outliers'.

The situation we have just described, however, is just overtraining, as described in Appendix A, i.e., the network with too many internal parameters begins to fit too well to the training set. Clearly the irregular boundary formed will not be as efficient for a statistically independent set of events, and in fact might even perform rather poorly.

The key to dealing with possible mislabelled *top* events in the background sample, then, is simply to ensure that the network does not overtrain. Straightforward procedures for achieving this exist and have been described in Appendix A. As long as the 'contamination' from *top* in the background training set is small, the mislabelled *top* events in the data set will simply cause an irreducible contribution to the error function and will not interfere

**with the performance of the net. In fact, when the network is applied to a larger sample of multijet events which *does* contain some top, it will correctly classify them.**

## References

[1] D. Bisello et al., CDF/ANAL/HEAVYFLAVOR/CDFR/1871

[2] J. Bensinger, P. Kesten, S. Tarem, CDF/DOC/HEAVYFLAVOR/CDFR/1036

[3] C. Hyde, D. Amidei, S.B. Kim, CDF/ANAL/HEAVYFLAVOR/CDFR/1565

[4] J. Benlloch and N. Wainer, CDF note in preparation.

[5] L. Stanco and E. Velasquez, CDF/ANAL/HEAVYFLAVOR/CDFR/1800

[6] A. Cherubini and R. Odorico, *Z. Phys.* **C47** (1990) 547.

[7] M. Richard and R. Lippmann, *Neural Computation* **3** (1991) 461-483.

[8] P. Nason, S. Dawson, R.K. Ellis, *Nucl. Phys.* **B303** (1988) 607.

[9] L. Garrido and V. Gaitans, 'Use of Neural Networks to Measure the $\tau$ Polarization and its Bayesian Interpretation', UAB-LFAF-91-04, Universitat Autonoma de Barcelona, Spain, April, 1991.

[10] D.E. Rumelhart, G.E. Hinton, J.L. McLelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1, MIT Press, Cambridge, MA, 1986.

[11] F. Fogelman-Soulié, *Artificial Neural Networks*, Proceedings of ICANN 1991, T. Kohonen, Ed., Elsevier Sci. Pub., B.V. North Holland, 1991.

[12] B. Denby, 'Tutorial on Neural Network Applications in HEP: a 1992 Perspective', in *New Computing Techniques in Physics II*, Proceedings of the 2nd AIHEP Workshop, World Scientific, 1992, also Fermilab-Conf-92/121-E, and CDF/PUB/PUBLIC/1737.

[13] K.H. Becks et al., 'B-Quark Tagging Using Neural Networks and Multivariate Statistical Methods, a Comparision of Both Techniques', Wuppertal University WU-B-93-4, Wuppertal, Germany.

[14] B.Denby, M. Dickson, G. Pauletta, N. Wainer, CDF/DOC/JET/CDFR/1706.

S. Bianchin et al., CDF/DOC/JET/CDFR/1707 and Udine Report 92/04/GP, 26 February 1992.

W. Badgett et al., CDF/PUB/PUBLIC/1851 and Fermilab-Conf-92/269-E.

Table I: The 4 $pb^{-1}$ of the 88/89 run have been considered and, at the level of the loose cuts of the analysis, efficiencies, number of events, and $S/B$ ratios of the NET performance are reported for a few values of the *output* variable of the NET for *exclusive* HERWIG. These are compared with those of the *tight* kinematical analysis. The $S/B$ ratios are computed from the selected experimental events.

| | | *NET Efficiencies* for HERWIG | | | |
|---|---|---|---|---|---|
| | data | *Top* 100 | *Top* 120 | *Top* 130 | *Top* 150 |
| *loose cuts* | 5910 | 3.9% | 9.5% | 11.1% | 14.4% |
| $SUM\_ET \geq 240$ | 1253 | 1.8% | 5.6% | 7.5% | 12.3% |
| *and* $OUT \geq 0.0$ | 330 | 1.0% | 3.8% | 5.5% | 9.7% |
| *and* $OUT \geq 0.1$ | 267 | 0.9% | 3.5% | 5.1% | 9.3% |
| *and* $OUT \geq 0.2$ | 219 | 0.8% | 3.2% | 4.5% | 8.7% |
| *and* $OUT \geq 0.3$ | 169 | 0.7% | 2.8% | 4.1% | 8.2% |
| *tight cuts* | 344 | 1.0% | 4.0% | 4.9% | 7.7% |
| Expected *Top Events* in 88/89 run | | | | | |
| *loose cuts* | | 13.6 | 11.7 | 9.1 | 5.9 |
| $SUM\_ET \geq 240$ | | 6.3 | 6.9 | 6.2 | 5.0 |
| *and* $OUT \geq 0.0$ | | 3.5 | 4.7 | 4.5 | 4.0 |
| *and* $OUT \geq 0.1$ | | 3.1 | 4.3 | 4.2 | 3.8 |
| *and* $OUT \geq 0.2$ | | 2.8 | 3.9 | 3.7 | 3.6 |
| *and* $OUT \geq 0.3$ | | 2.4 | 3.4 | 3.4 | 3.4 |
| *tight cuts* | | 3.5 | 4.9 | 4.0 | 3.2 |
| *Signal* over *Background* Ratio | | | | | |
| *loose cuts* | | 1/434 | 1/505 | 1/649 | 1/1002 |
| $SUM\_ET \geq 240$ | | 1/200 | 1/182 | 1/204 | 1/248 |
| *and* $OUT \geq 0.0$ | | 1/94 | 1/70 | 1/73 | 1/83 |
| *and* $OUT \geq 0.1$ | | 1/86 | 1/62 | 1/63 | 1/70 |
| *and* $OUT \geq 0.2$ | | 1/78 | 1/56 | 1/59 | 1/61 |
| *and* $OUT \geq 0.3$ | | 1/70 | 1/50 | 1/50 | 1/50 |
| *tight cuts* | | 1/98 | 1/70 | 1/86 | 1/108 |

**Table II: The 4 $pb^{-1}$ of the 88/89 run have been considered and, at the level of the loose cuts of the analysis, efficiencies, number of events, and $S/B$ ratios of the NET performance are reported for a few values of the *output* variable of the NET for *inclusive* ISAJET. These are compared with those of the *tight* kinematical analysis. The $S/B$ ratios are computed from the selected experimental events.**

|  |  | *NET Efficiencies* for ISAJET |  |  |  |
|---|---|---|---|---|---|
|  | data | *Top* 100 | *Top* 120 | *Top* 130 | *Top* 150 |
| *loose cuts* | 5910 | 7.85% | 15.6% | 17.9% | 23.8% |
| *SUM_ET* ≥ 240 | 1253 | 4.2% | 10.3% | 13.0% | 20.3% |
| *and OUT* ≥ 0.0 | 345 | 3.1% | 7.5% | 9.9% | 16.3% |
| *and OUT* ≥ 0.1 | 298 | 2.8% | 6.9% | 9.3% | 15.2% |
| *and OUT* ≥ 0.2 | 229 | 2.4% | 6.5% | 8.8% | 14.3% |
| *and OUT* ≥ 0.3 | 175 | 2.1% | 5.5% | 7.8% | 13.0% |
| *tight cuts* | 344 | 3.0% | 7.0% | 9.3% | 14.1% |
|  |  | Expected *Top Events* in 88/89 run |  |  |  |
| *loose cuts* |  | 27.4 | 19.2 | 14.7 | 9.8 |
| *SUM_ET* ≥ 240 |  | 14.7 | 12.6 | 10.7 | 8.3 |
| *and OUT*≥ 0.0 |  | 10.8 | 9.2 | 8.1 | 6.7 |
| *and OUT*≥ 0.1 |  | 9.8 | 8.5 | 7.6 | 6.2 |
| *and OUT*≥ 0.2 |  | 8.4 | 8.0 | 7.2 | 5.9 |
| *and OUT*≥ 0.3 |  | 7.3 | 6.8 | 6.4 | 5.3 |
| *tight cuts* |  | 10.5 | 8.6 | 7.6 | 5.8 |
|  |  | *Signal* over *Background* Ratio |  |  |  |
| *loose cuts* |  | 1/216 | 1/308 | 1/402 | 1/603 |
| *SUM_ET* ≥ 240 |  | 1/85 | 1/99 | 1/117 | 1/151 |
| *and OUT*≥ 0.0 |  | 1/32 | 1/38 | 1/43 | 1/52 |
| *and OUT*≥ 0.1 |  | 1/30 | 1/35 | 1/39 | 1/48 |
| *and OUT*≥ 0.2 |  | 1/27 | 1/29 | 1/32 | 1/39 |
| *and OUT*≥ 0.3 |  | 1/24 | 1/26 | 1/27 | 1/33 |
| *tight cuts* |  | 1/33 | 1/40 | 1/45 | 1/59 |

**Table III: The events selected by the *tight* kinematical analysis of the 4 $pb^{-1}$ of the 88/89 run have been considered. Efficiencies, number of events, and $S/B$ ratios of the NET performance are reported and compared with those of the *tight* kinematical analysis. The efficiencies of the NET are computed for a few values of $OUT$ for *exclusive* HERWIG and the $S/B$ ratios are computed from the number of selected events.**

| | | *NET Efficiencies* for HERWIG | | | |
|---|---|---|---|---|---|
| | data | *Top* 100 | *Top* 120 | *Top* 130 | *Top* 150 |
| *tight cuts* | 344 | 1.0% | 4.0% | 4.9% | 7.7% |
| $SUM\_ET \geq 240$ | 282 | 0.8% | 3.5% | 4.4% | 7.4% |
| *and* $OUT \geq 0.0$ | 128 | 0.6% | 2.7% | 3.7% | 6.3% |
| *and* $OUT \geq 0.1$ | 110 | 0.5% | 2.6% | 3.6% | 6.1% |
| *and* $OUT \geq 0.2$ | 88 | 0.5% | 2.4% | 3.3% | 5.8% |
| *and* $OUT \geq 0.3$ | 77 | 0.4% | 2.2% | 3.0% | 5.3% |
| Expected *Top Events* in 88/89 run | | | | | |
| *tight cuts* | | 3.5 | 4.9 | 4.0 | 3.2 |
| $SUM\_ET \geq 240$ | | 2.8 | 4.3 | 3.6 | 3.0 |
| *and* $OUT \geq 0.0$ | | 2.1 | 3.3 | 3.0 | 2.6 |
| *and* $OUT \geq 0.1$ | | 1.8 | 3.2 | 2.9 | 2.5 |
| *and* $OUT \geq 0.2$ | | 1.7 | 2.9 | 2.7 | 2.4 |
| *and* $OUT \geq 0.3$ | | 1.4 | 2.7 | 2.4 | 2.2 |
| *Signal* over *Background* Ratio | | | | | |
| *tight cuts* | | 1/98 | 1/70 | 1/86 | 1/108 |
| $SUM\_ET \geq 240$ | | 1/101 | 1/66 | 1/78 | 1/93 |
| *and* $OUT \geq 0.0$ | | 1/61 | 1/39 | 1/43 | 1/49 |
| *and* $OUT \geq 0.1$ | | 1/61 | 1/34 | 1/38 | 1/44 |
| *and* $OUT \geq 0.2$ | | 1/52 | 1/30 | 1/33 | 1/37 |
| *and* $OUT \geq 0.3$ | | 1/55 | 1/29 | 1/32 | 1/35 |

Table IV: The events selected by the *tight* kinematical analysis of the 4 $pb^{-1}$ of the 88/89 run have been considered. Efficiencies, number of events, and $S/B$ ratios of the NET performance are reported and compared with those of the *tight* kinematical analysis. The efficiencies of the NET are computed for a few values of $OUT$ for *inclusive* ISAJET and the $S/B$ ratios are computed from the number of selected events.

| | | *NET Efficiencies* for ISAJET | | | |
|---|---|---|---|---|---|
| | data | *Top* 100 | *Top* 120 | *Top* 130 | *Top* 150 |
| *tight cuts* | 344 | 3.0% | 7.0% | 9.3% | 14.1% |
| $SUM\_ET \geq 240$ | 282 | 2.7% | 6.2% | 8.5% | 13.5% |
| *and* $OUT \geq 0.0$ | 129 | 2.2% | 5.3% | 7.3% | 11.9% |
| *and* $OUT \geq 0.1$ | 109 | 2.0% | 5.0% | 7.0% | 11.6% |
| *and* $OUT \geq 0.2$ | 91 | 1.7% | 4.7% | 6.5% | 11.0% |
| *and* $OUT \geq 0.3$ | 72 | 1.6% | 4.1% | 6.0% | 10.1% |
| Expected *Top Events* in 88/89 run | | | | | |
| *tight cuts* | | 10.5 | 8.6 | 7.6 | 5.8 |
| $SUM\_ET \geq 240$ | | 9.4 | 7.6 | 7.0 | 5.5 |
| *and* $OUT \geq$ 0.0 | | 7.7 | 6.5 | 6.0 | 4.9 |
| *and* $OUT \geq$ 0.1 | | 7.0 | 6.2 | 5.7 | 4.8 |
| *and* $OUT \geq$ 0.2 | | 5.9 | 5.8 | 5.3 | 4.5 |
| *and* $OUT \geq$ 0.3 | | 5.6 | 5.0 | 4.9 | 4.1 |
| *Signal* over *Background* Ratio | | | | | |
| *tight cuts* | | 1/33 | 1/40 | 1/45 | 1/59 |
| $SUM\_ET \geq 240$ | | 1/30 | 1/37 | 1/40 | 1/51 |
| *and* $OUT \geq$ 0.0 | | 1/17 | 1/20 | 1/22 | 1/26 |
| *and* $OUT \geq$ 0.1 | | 1/16 | 1/18 | 1/19 | 1/23 |
| *and* $OUT \geq$ 0.2 | | 1/15 | 1/16 | 1/17 | 1/20 |
| *and* $OUT \geq$ 0.3 | | 1/13 | 1/14 | 1/15 | 1/17 |

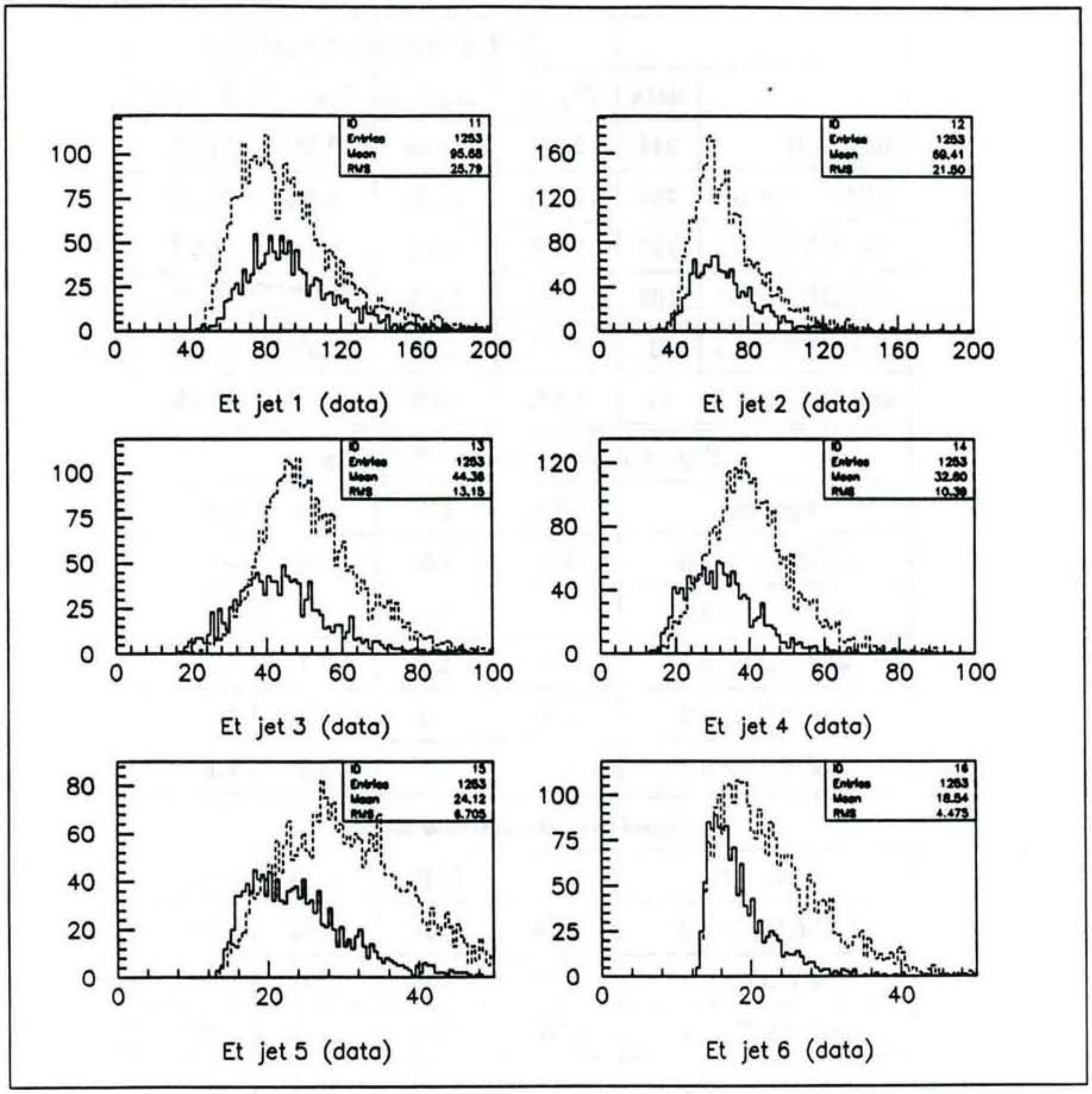


Fig.1a Distributions of $E_T$ of the leading six jets of the 88/89 multi-jet data (solid) and of the ISAJET sample of $t\bar{t}$ events (dashed) after full detector simulation. The normalization of the two curves is arbitrary.

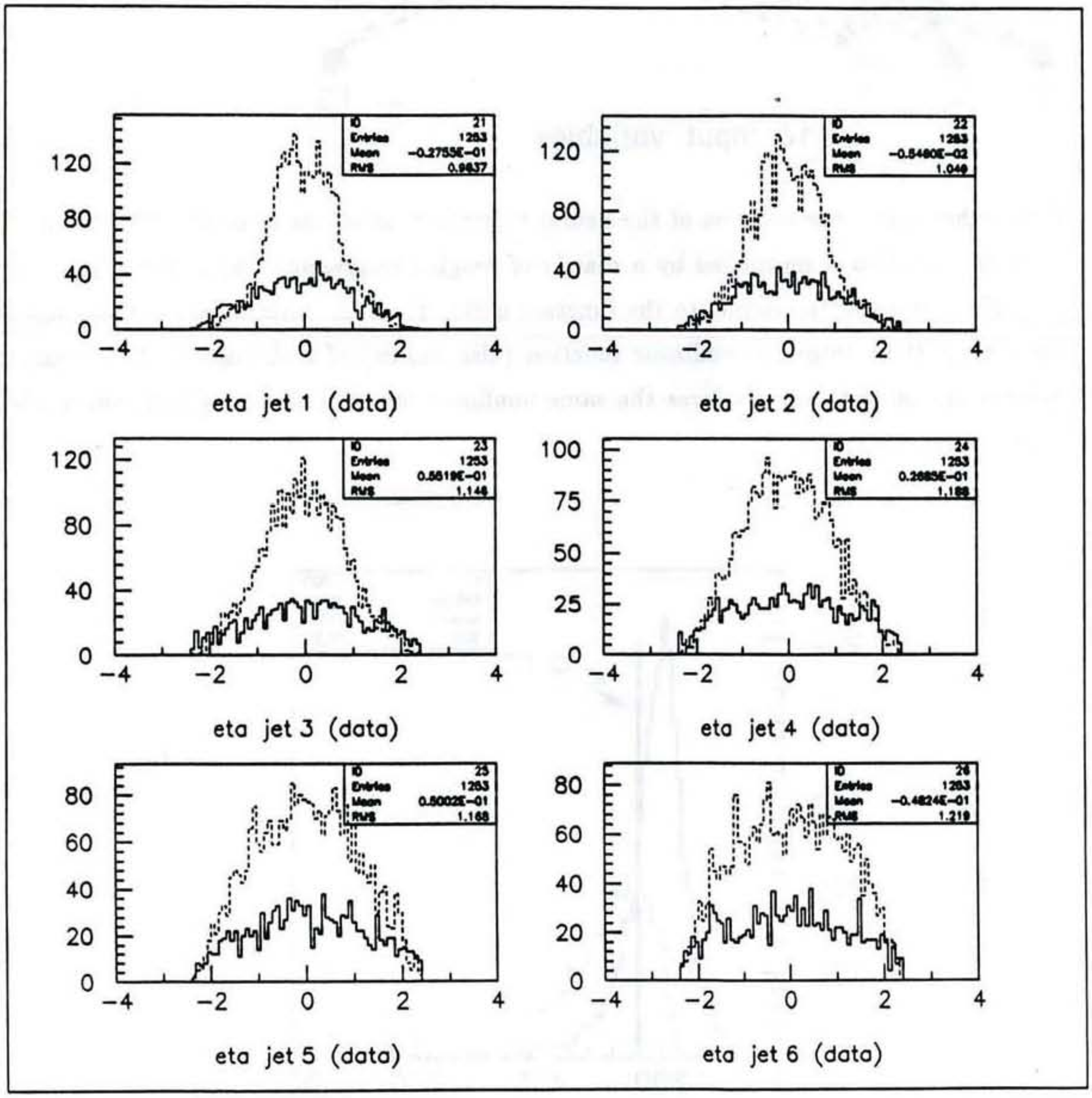


Fig.1b Distributions in $\eta$ of the leading six jets of the 88/89 multi-jet data (solid) and the ISAJET sample of $t\bar{t}$ events (dashed) after full detector simulation. The normalization of the two curves is arbitrary.

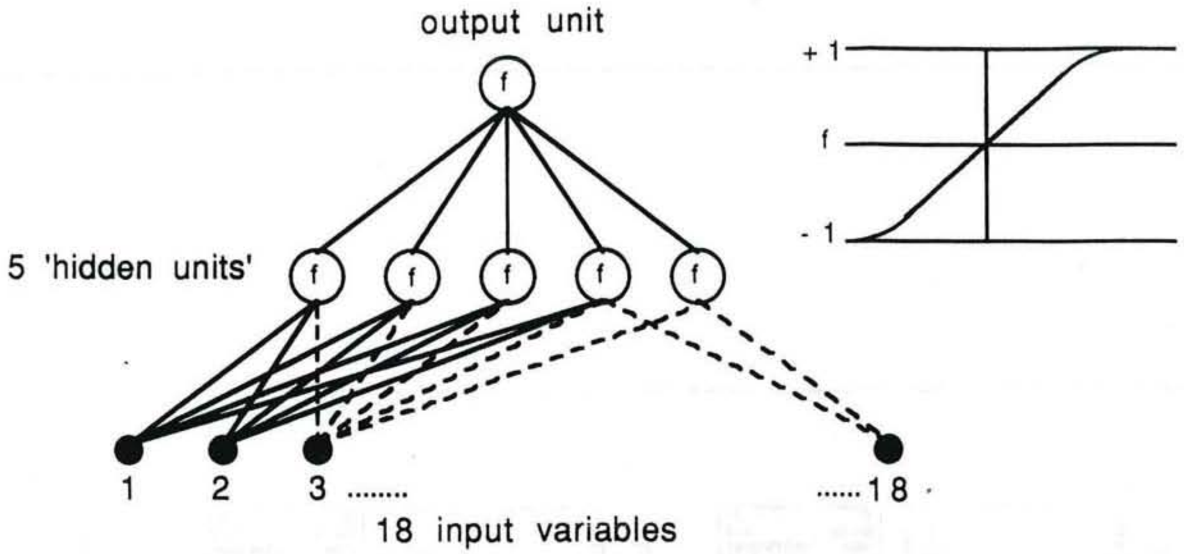


Fig.2 Schematic representation of the neural network used in the analysis. The vector of 18 input variables is multiplied by a matrix of weights (represented by lines) to produce 5 weighted sums at the inputs to the function units, 'f', in the hidden layer. These units produce at their output a nonlinear function (also shown) of their inputs. In a similar manner the output unit produces the same nonlinear function of a weighted sum of the hidden layer outputs.

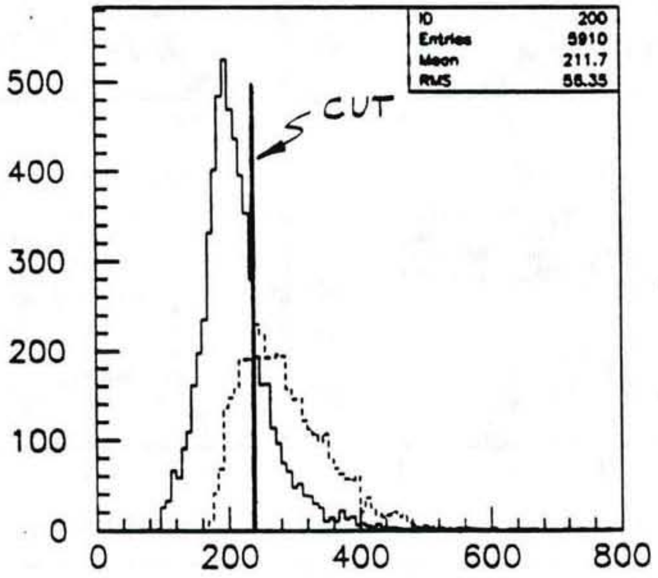


Fig.3 The $sumE_T$ of the leading six jets of the data and the ISAJET sample. The vertical line at 240 $GeV$ corresponds to the cut placed to restrict the operation of the network to a region where the *top* and background overlapped (see pg.6 and appendix A).

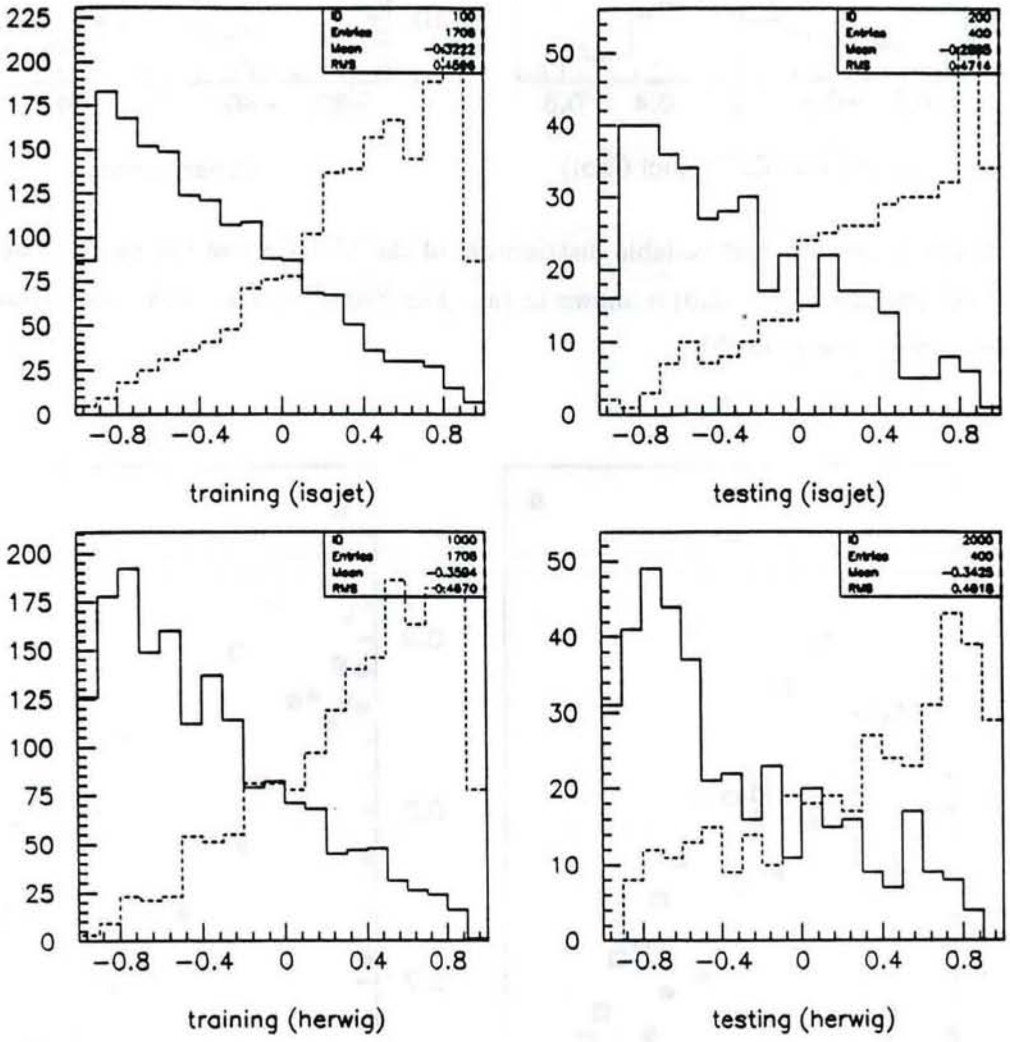


Fig.4 The distributions of the network *output* for the *TRAINING* and *TEST* sets of background events (solid) and $t\bar{t}$ events (dashed). The upper distributions are for the network trained with ISAJET, the lower ones for HERWIG.

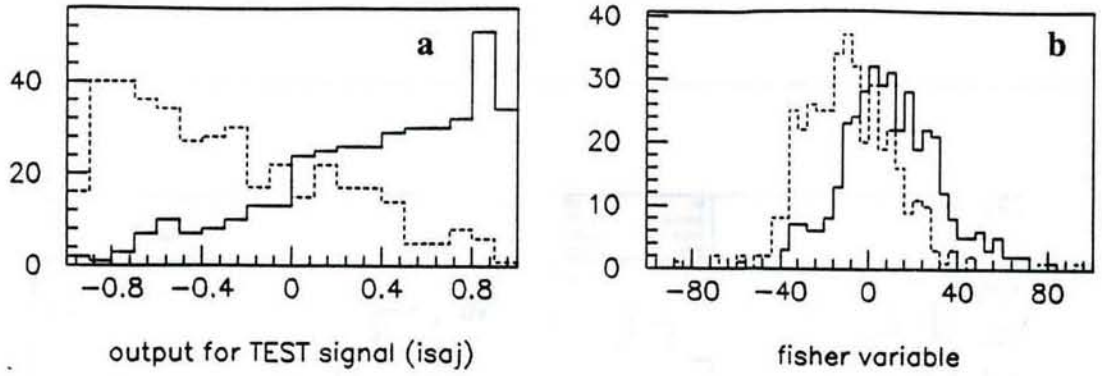


Fig.5 The network *output* variable distribution of the *TEST* set of the background events (dashed) and $t\bar{t}$ events (solid) is shown in (a). The Fisher discriminant distribution of the same events is shown in (b).

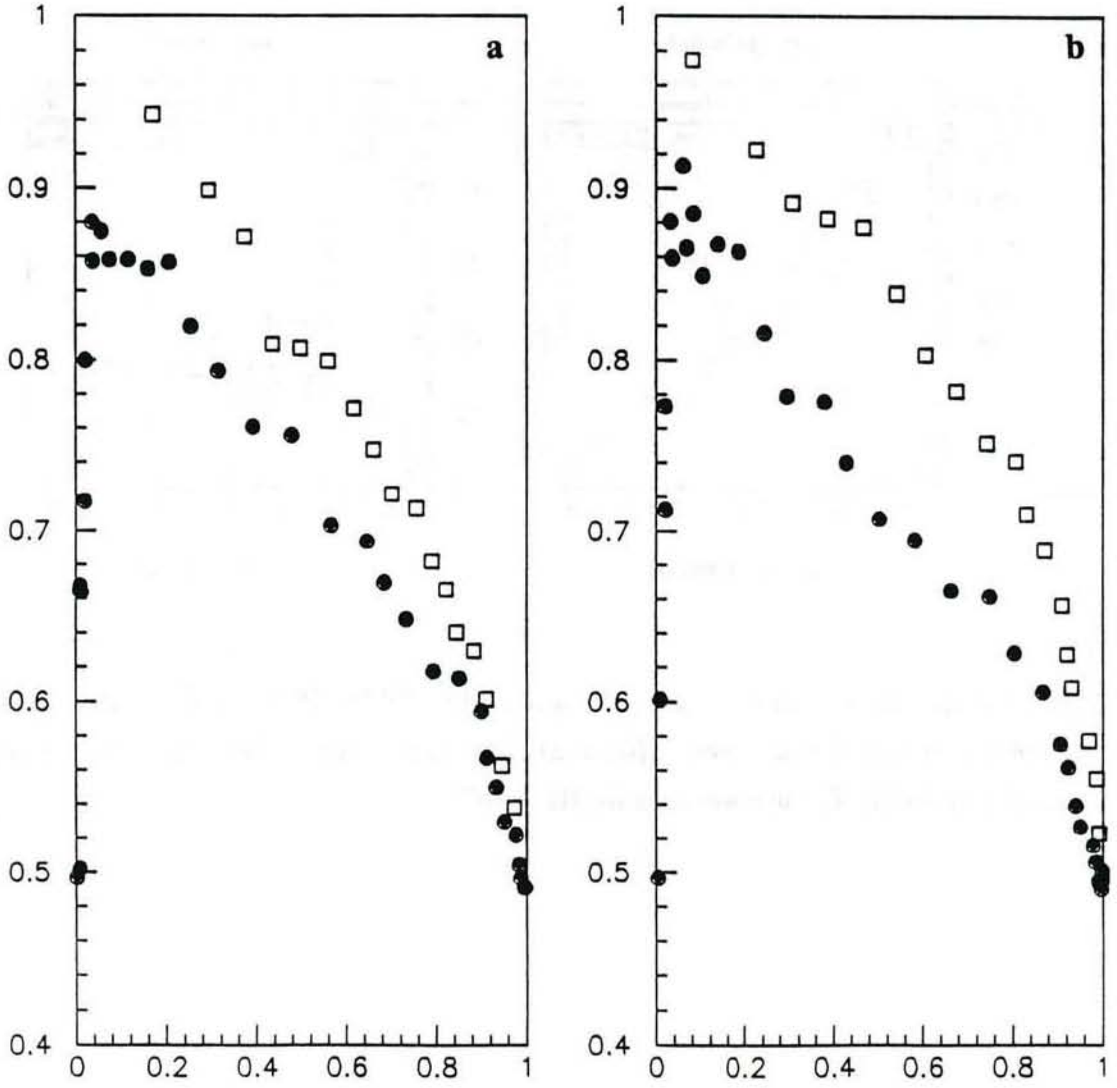


Fig.6 A direct comparison of the performance of the neural network and Fisher discriminant on the *TEST* data. The Purity *VS* Efficiency (see text) for the network *output* variable (white squares), and the Fisher discriminant (black dots), for HERWIG (a) and ISAJET (b). The performance of the network is better.

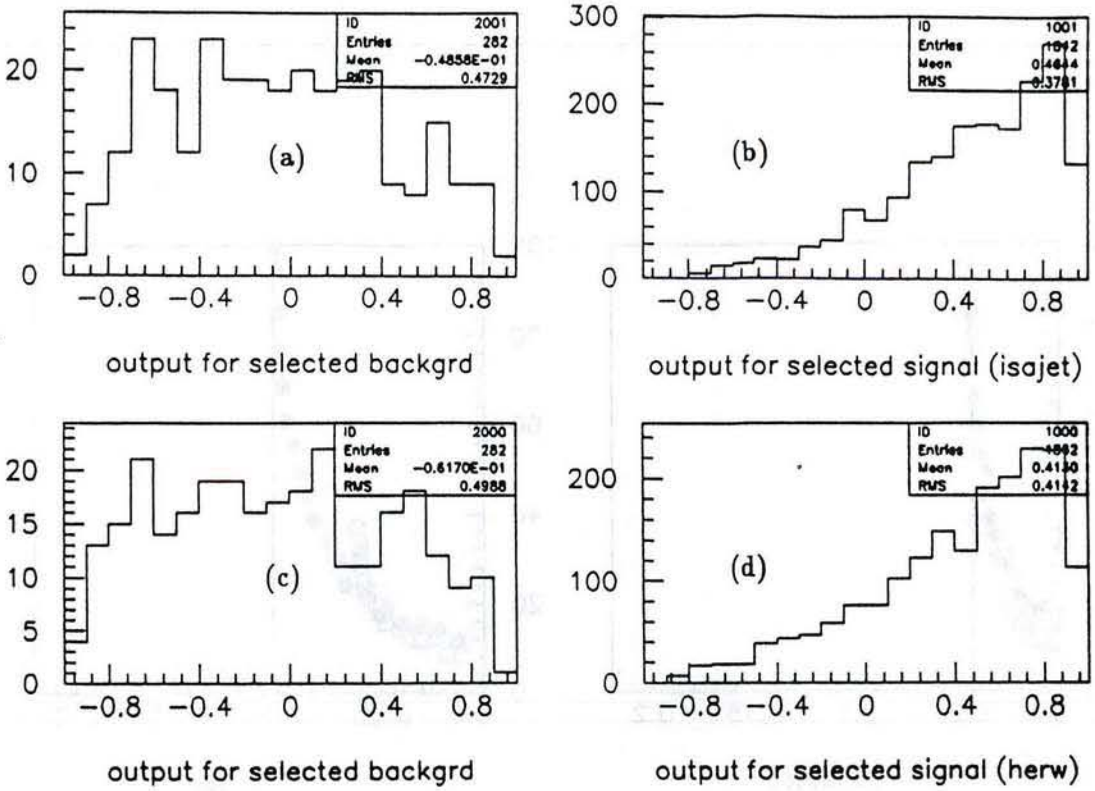


Fig.7 Neural nets applied to events passing the *tight cuts*; i.e. the combined classifier. In (a) the network *output* distribution of the selected backgorund events and in (b) the distribution of the selected ISAJET events, for the network trained with ISAJET. The performance of the network trained with HERWIG on *tightly* selected events are shown in (c) and (d).

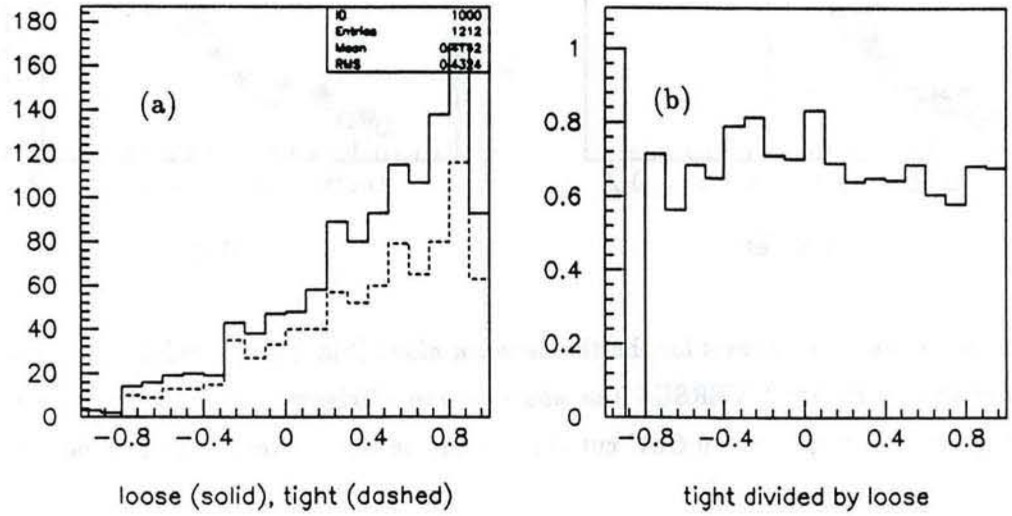


Fig.8 a) Distribution of *out* for ISAJET network for ISAJET $m_{top} = 150\ GeV/c^2$, for the loose cuts (solid curve) and tight cuts (dashed curve). b) Ratio of the tight distribution to the loose distribution. The tight cuts appear to discard events at all *out* values with roughly equal probability.

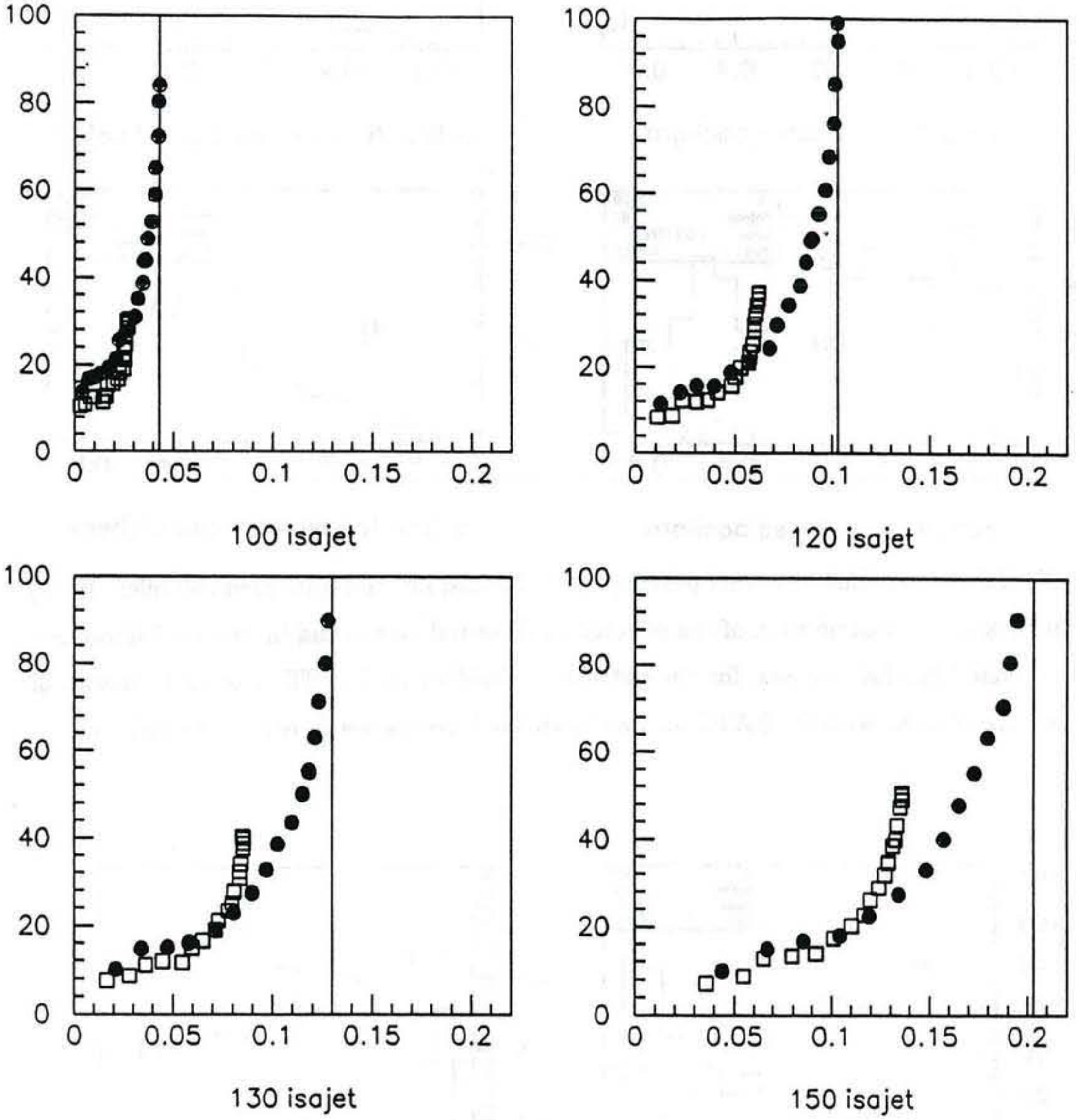
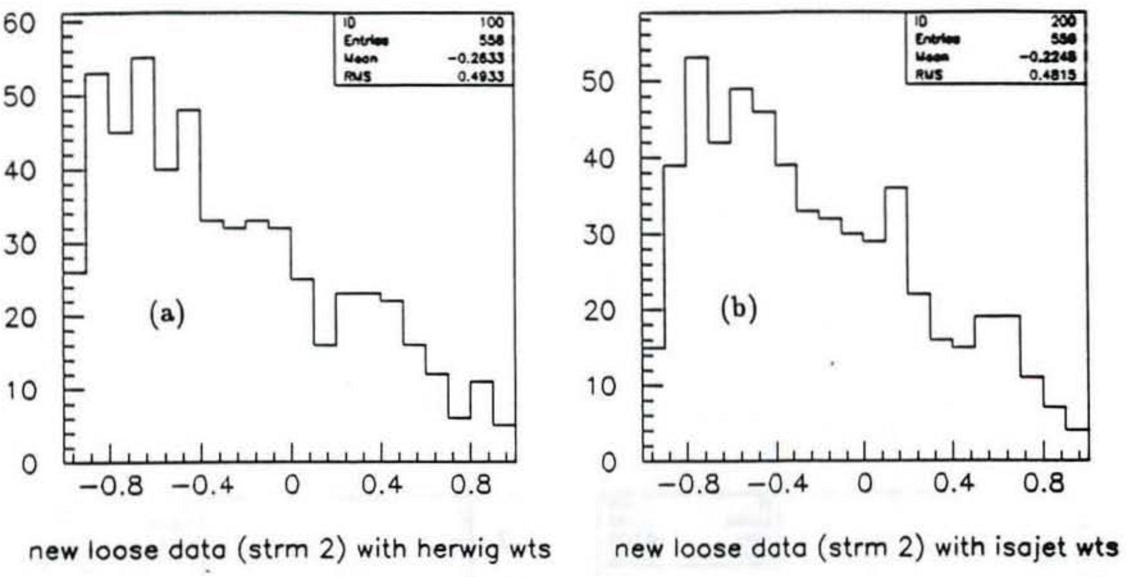


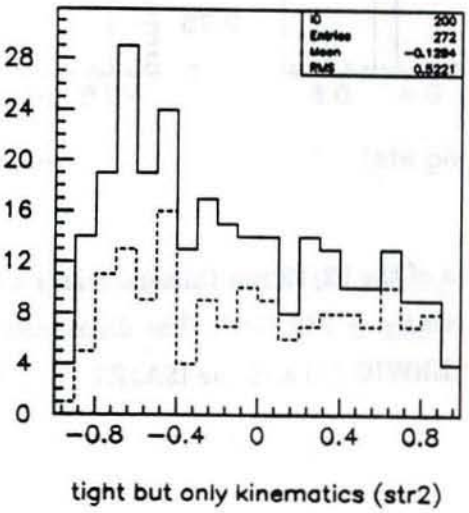
Fig.9 B/S for the 4 *top* masses for the the network *alone* (black dots) and for the *combined classifier* (white squares) VERSUS the absolute *top* efficiency. The vertical line is the efficiency for the $sumE_T \geq 240\ GeV$ cut (i.e. no cut on the network *output* variable).



new loose data (strm 2) with herwig wts

new loose data (strm 2) with isajet wts

**Fig.10 The *output* distributions of the first 2.5 $pb^{-1}$ data of the 92/93 run for the HERWIG and ISAJET networks are shown in (a) and (b) respectively.**



tight but only kinematics (str2)

Fig.11 The network *output* variable distribution of selected events from the first 2.5 $pb^{-1}$ data of the 92/93 run (from *Stream 2*). The solid line distribution corresponds to events selected only with the purely kinematic *tight* cuts. The dashed distribution is of those events that in addition pass the $R = 0.4$ *clustering cut.* The network used was that trained with HERWIG.

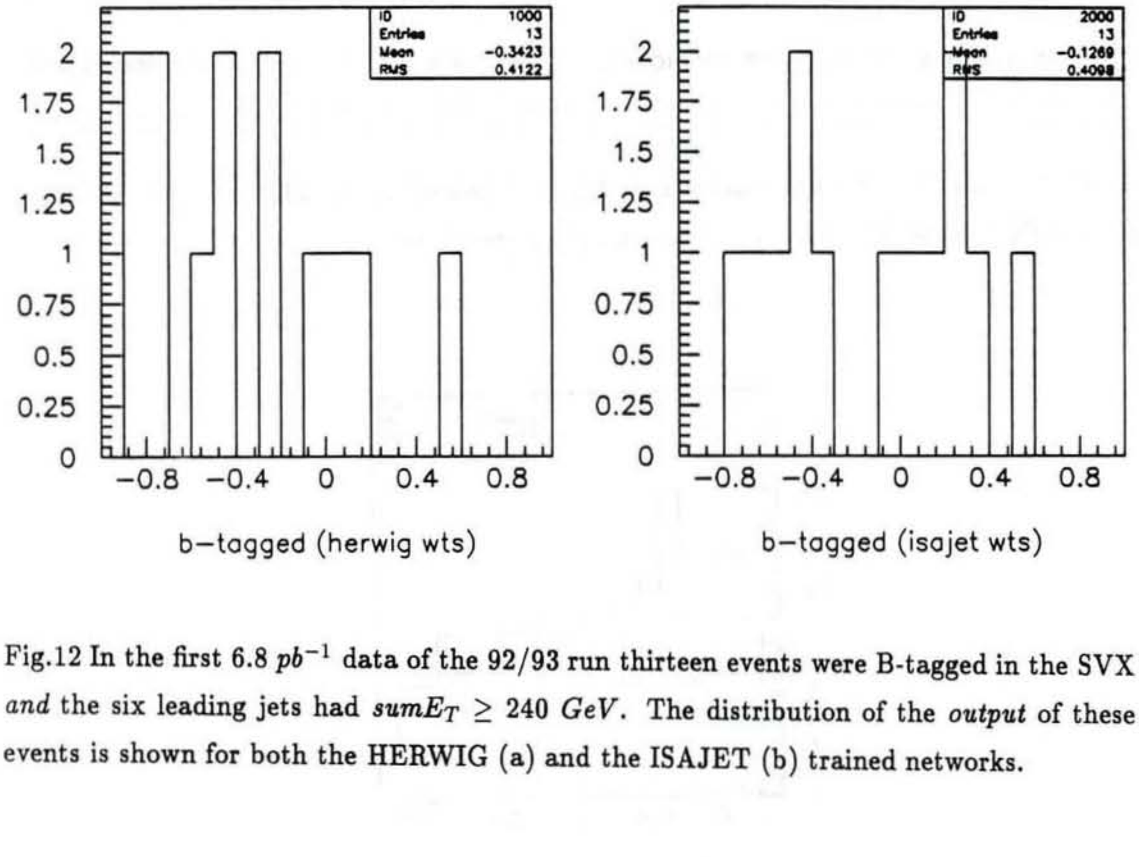
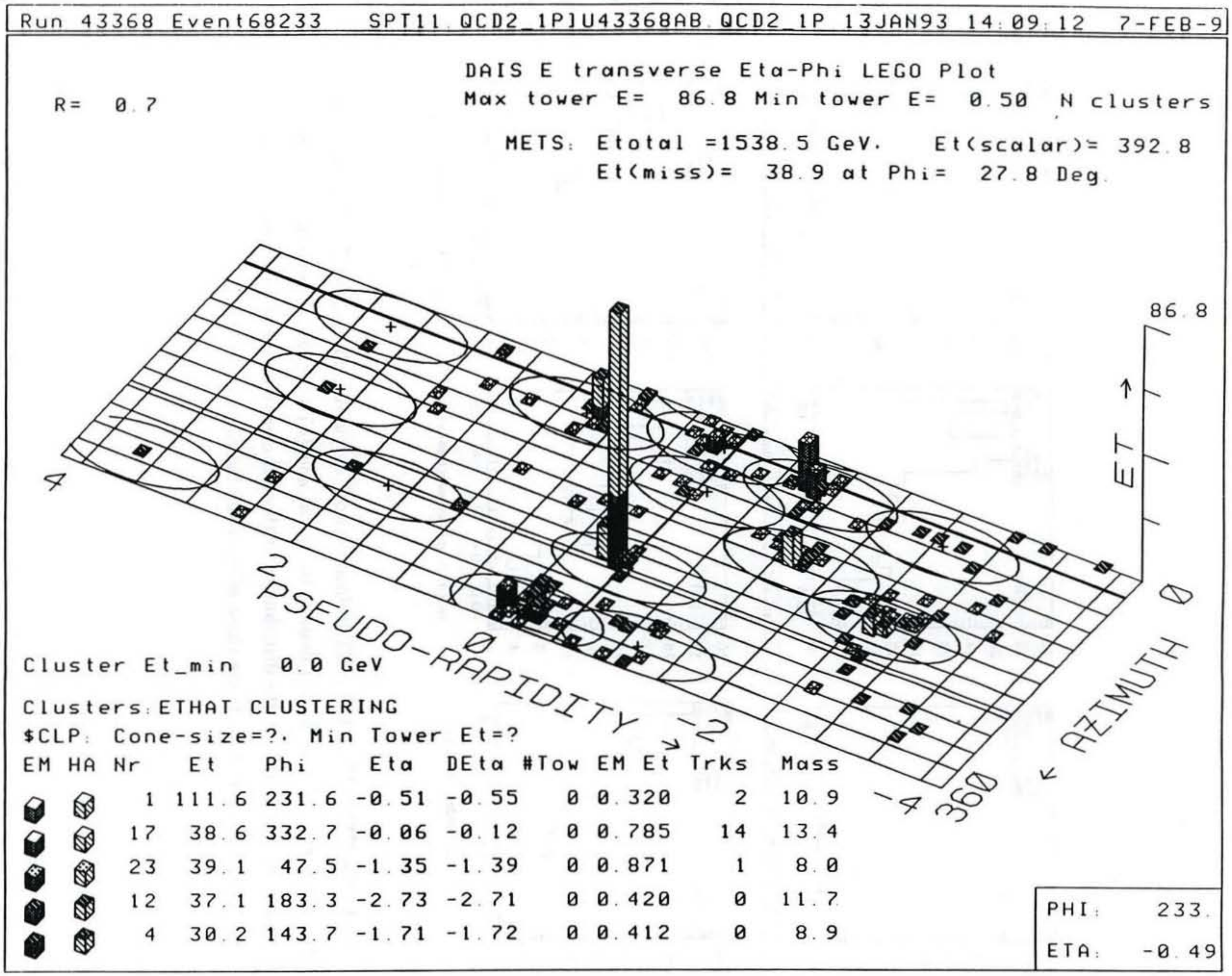


Fig.12 In the first 6.8 $pb^{-1}$ data of the 92/93 run thirteen events were B-tagged in the SVX *and* the six leading jets had $sumE_T \geq 240\ GeV$. The distribution of the *output* of these events is shown for both the HERWIG (a) and the ISAJET (b) trained networks.



```
Run 43368 Event68233   SPT11.QCD2_1P]U43368AB.QCD2_1P 13JAN93 14:09:12  7-FEB-9

                            DAIS E transverse Eta-Phi LEGO Plot
   R=   0.7                 Max tower E=  86.8 Min tower E=  0.50  N clusters

                              METS: Etotal =1538.5 GeV.   Et(scalar)= 392.8
                                    Et(miss)=  38.9 at Phi=  27.8 Deg.
```

```
Cluster Et_min   0.0 GeV

Clusters:ETHAT CLUSTERING
$CLP: Cone-size=?. Min Tower Et=?
EM HA Nr    Et    Phi    Eta   DEta #Tow EM Et Trks   Mass
●  ▨    1 111.6 231.6 -0.51 -0.55   0 0.320    2   10.9
●  ▨   17  38.6 332.7 -0.06 -0.12   0 0.785   14   13.4
●  ▨   23  39.1  47.5 -1.35 -1.39   0 0.871    1    8.0
●  ▨   12  37.1 183.3 -2.73 -2.71   0 0.420    0   11.7
●  ▨    4  30.2 143.7 -1.71 -1.72   0 0.412    0    8.9

                                                        PHI:     233.
                                                        ETA:    -0.49
```

Fig.13 Lego plot of one of the two 'candidate' events which survive the tight cuts, have *out* > 0. for both HERWIG and ISAJET networks, and are b–tagged.

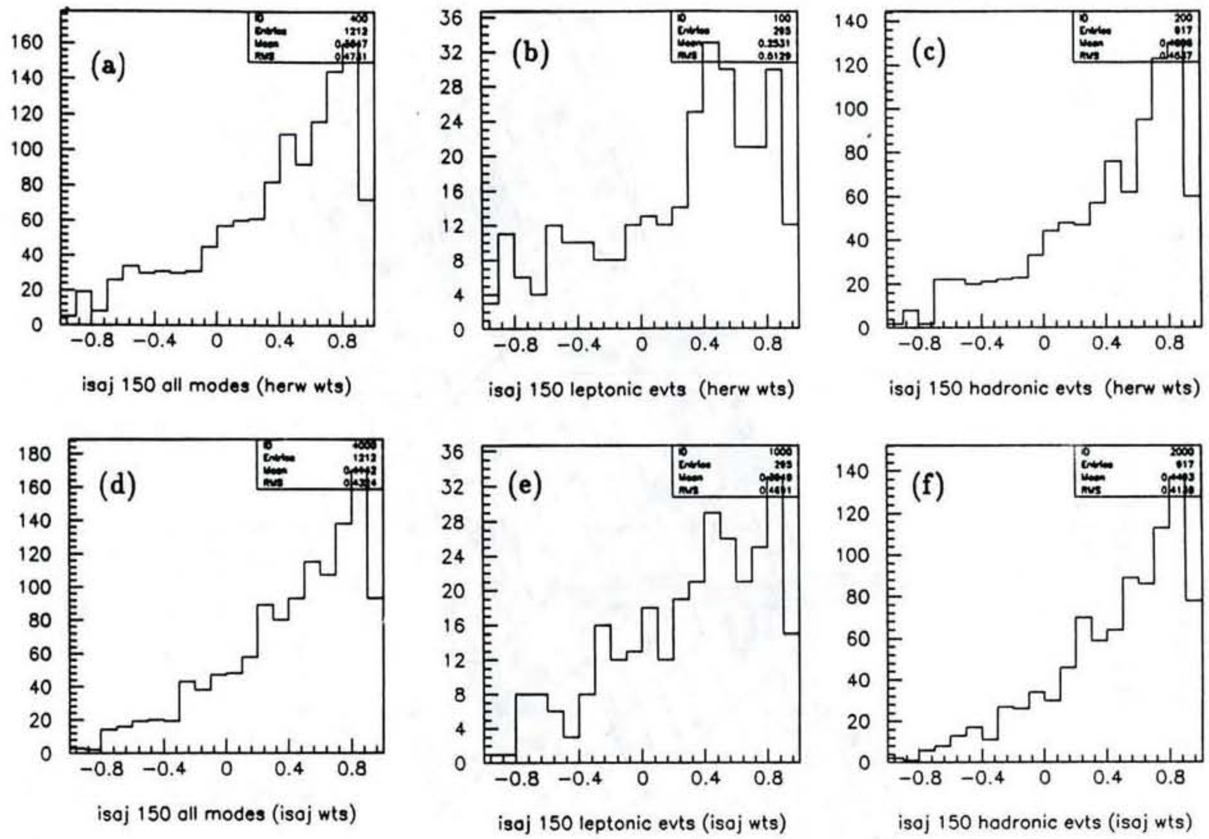


Fig.14 The HERWIG and ISAJET trained network *output* distribution for ISAJET $m_t$=150 $GeV/c^2$ events selected by the *loose* cuts. In (a) and (d) are the distribution of all the $t\bar{t}$ events. In (b) and (e) are the distribution of those $t\bar{t}$ events where at least one $W$ decayed leptonically, and (c) and (f) are those of the truly hadronic events.

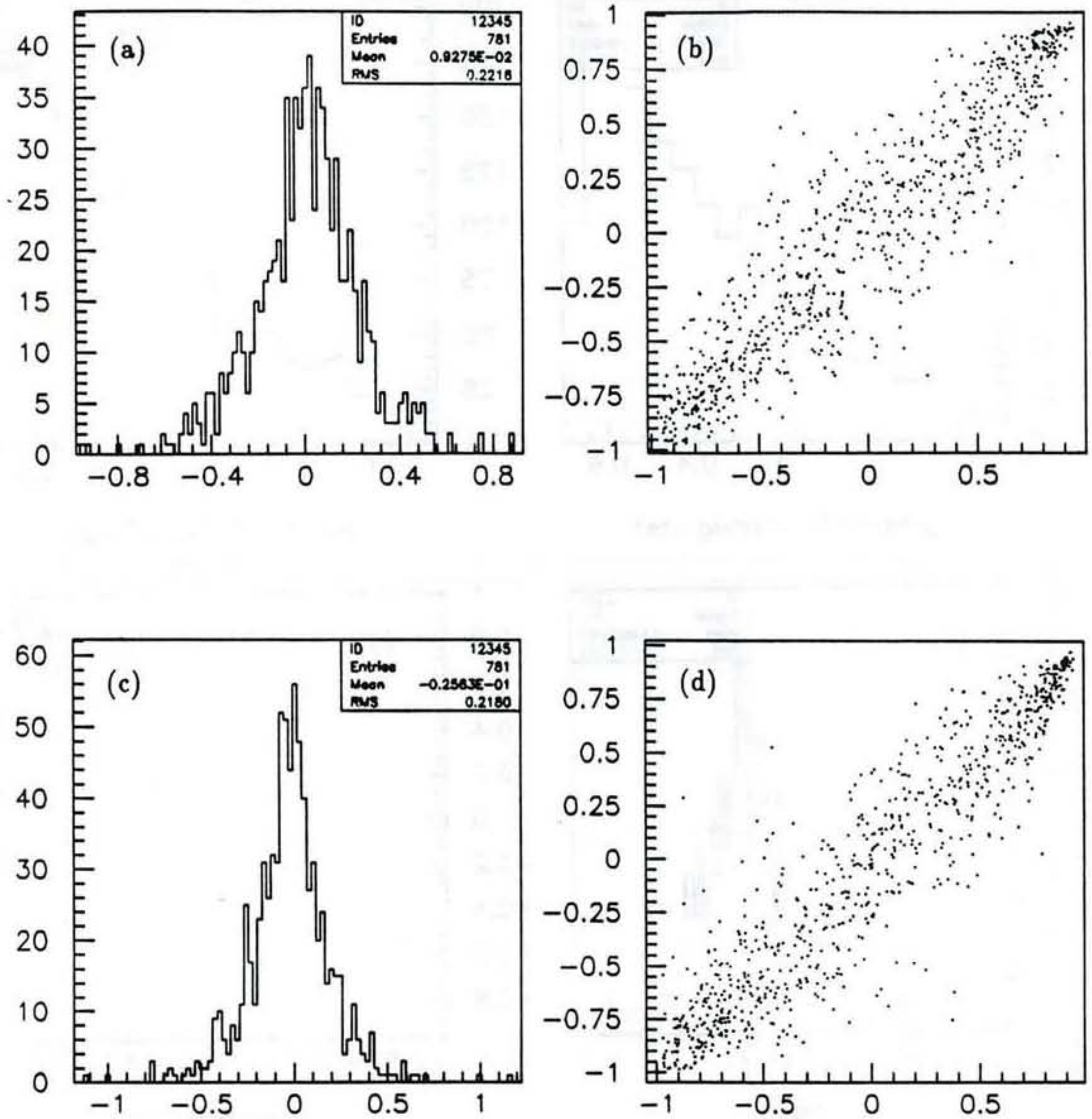


Fig.15 In (a), the difference between the network outputs for ISAJET nets with two different sets of initial weights; b) is a scatter plot of the one net versus the other. The width of the distribution in (a) represents an intrinsic uncertainty in $P_{top}(1)$ due to finite training statistics. The plots are repeated in (b) and (c) for two different HERWIG networks.

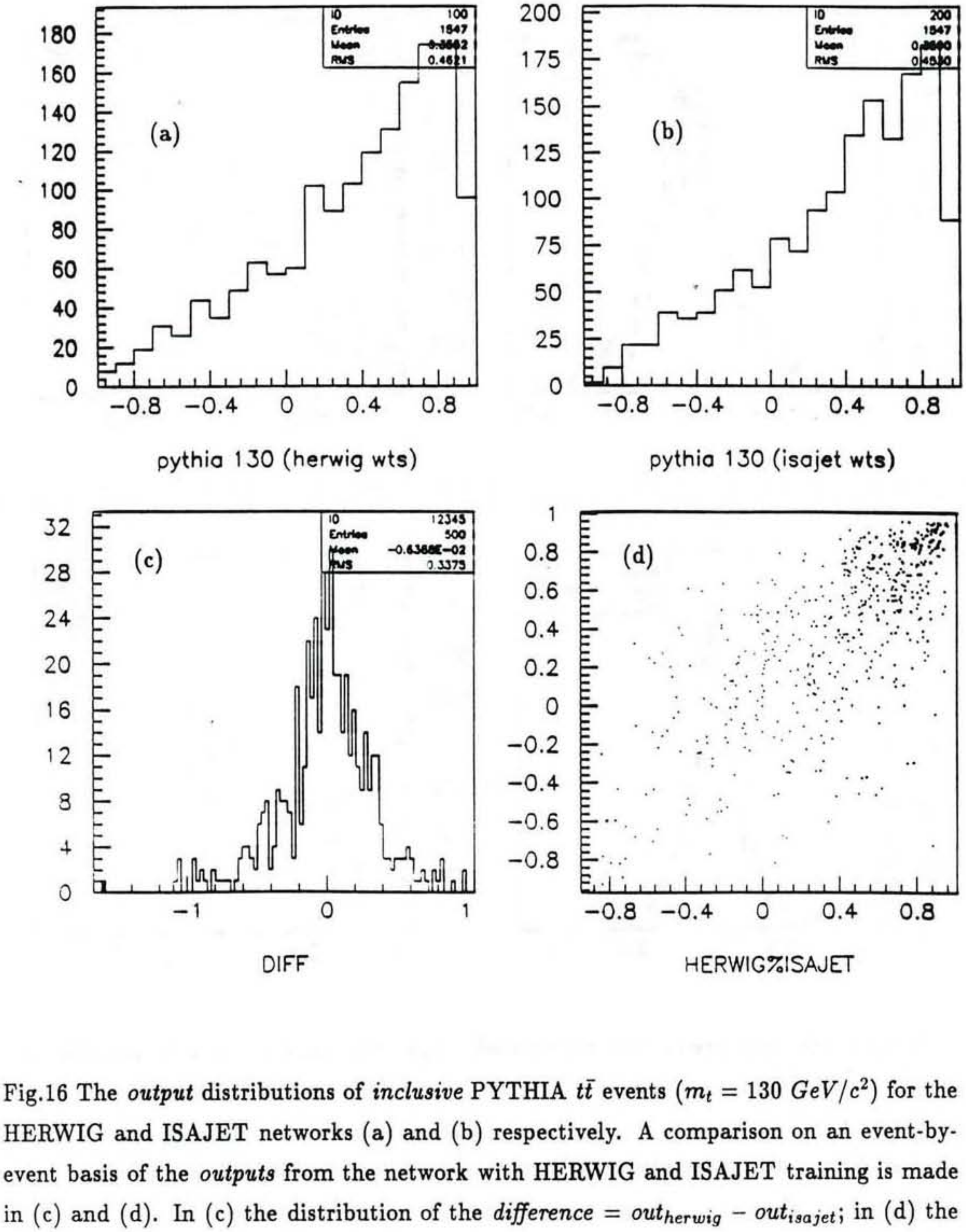


Fig.16 The *output* distributions of *inclusive* PYTHIA $t\bar{t}$ events ($m_t = 130\ GeV/c^2$) for the HERWIG and ISAJET networks (a) and (b) respectively. A comparison on an event-by-event basis of the *outputs* from the network with HERWIG and ISAJET training is made in (c) and (d). In (c) the distribution of the *difference* $= out_{herwig} - out_{isajet}$; in (d) the scatter plot of $out_{herwig}$ $VS$ $out_{isajet}$.

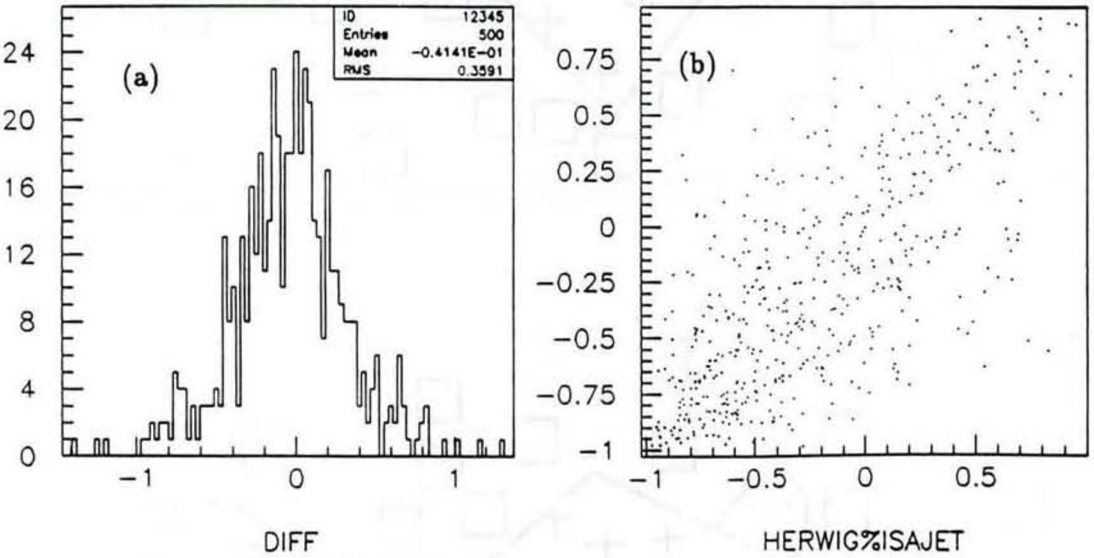


**Fig.17 A comparison of the *outputs* from the network with HERWIG and ISAJET training for the first 2.5 $pb^{-1}$ of the 92/93 data. In (a) the distribution of the *difference* = $out_{herwig} - out_{isajet}$; in (b) the scatter plot of $out_{herwig}$ *VS* $out_{isajet}$.**

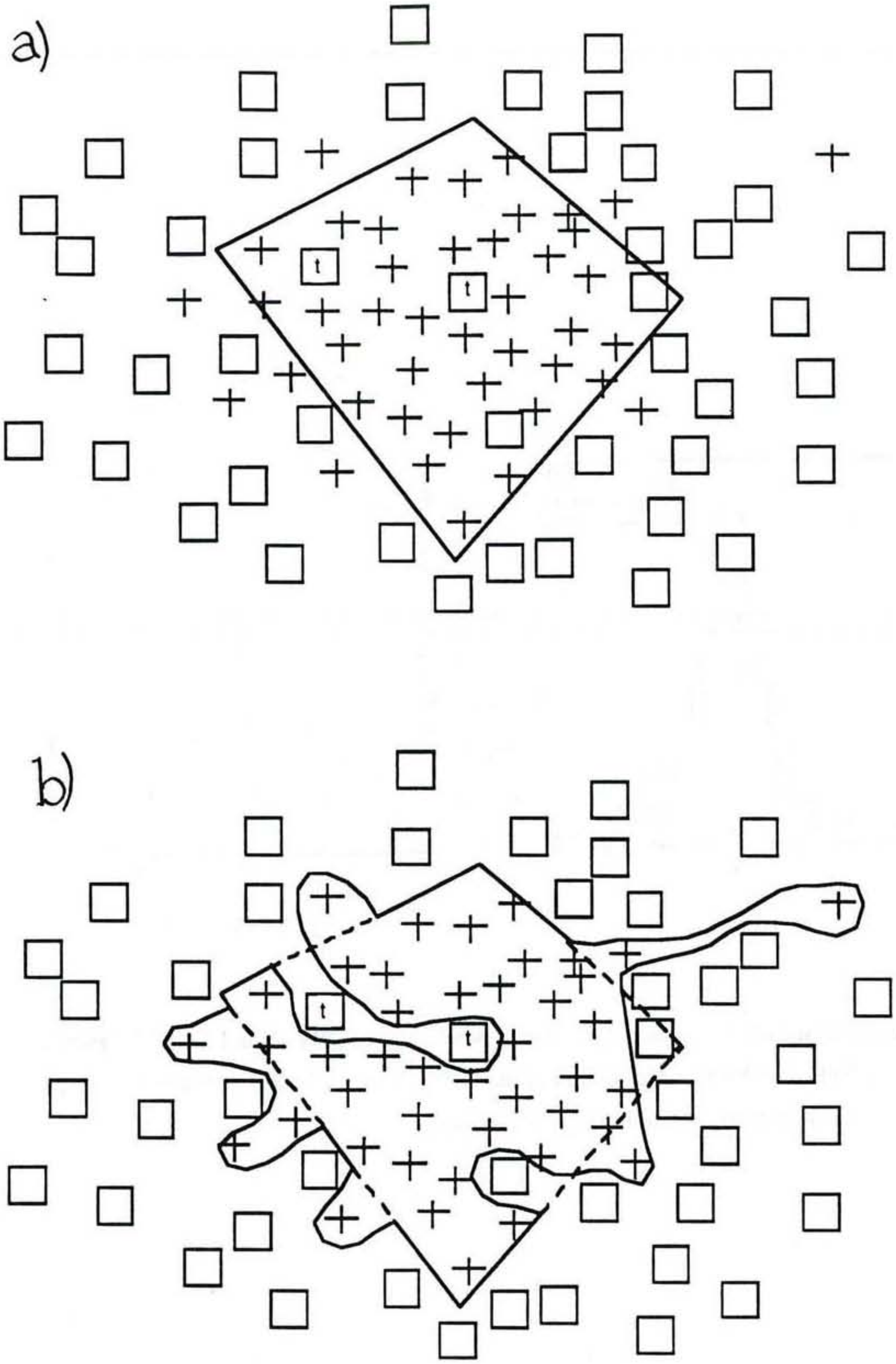


Fig.18 Two dimensional model of the $t\bar{t}$ to multijets problem. Crosses represent *top* events, squares, background. The two events with a 't' inside the square are *top* events which have been mislabelled as background. In a) is shown plausible classification boundary created by a neural network with 4 hidden units. In b), the much more complicated boundary created by an overtrained network with many hidden units, which achieves perfect classification.