Non-parametric Peak Finder for General Searches in ATLAS data

Michael Erickson

Argonne National Laboratory, Physics Undergraduate

Student Research Participation Program

The College of New Jersey

Ewing, NJ

Argonne National Laboratory

Argonne, IL

August 12, 2011

Abstract

The ATLAS experiment, which is currently operating at the Large Hadron Collider (LHC) at CERN, constantly requires faster and more efficient methods for analyzing the data which is being produced and stored. The implementation of a general search which analyzes all stored data for new particles would be extremely useful as a method for preliminary analysis. The first goal of this project was to understand the applicability of various analytical functions to the description of background of invariant masses of two or more particles (jets). The second goal was to develop a technique for detecting statistically significant peaks without any a priori assumptions on the shape of the background distribution. To achieve the second goal, Python and ROOT were used to develop program NPPFinder to detect possible peaks in invariant mass distributions and then calculate each peak's statistical significance using numerical methods. These statistical significances are then graphed in order to observe any important values which should be investigated further. The program has been run successfully over 1.0 fb^{-1} of ATLAS data. Overall, this method shows great promise in providing a simple, unique, and effective approach for quick preliminary analysis of ATLAS data.

Introduction

The European Organization for Nuclear Research (CERN) in Geneva is home to the Large Hadron Collider (LHC), which performs experiments in the 7 TeV energy range with plans for even higher energies in the future. The LHC is used to accelerate both protons and heavy ions to over 99% the speed of light and then allow them to collide at certain points along the structure's beam line. At one of these points located in Meyrin, Switzerland is housed the ATLAS detector, which encircled the beam line in order to capture these events. For each of the almost 1 billion events produced per second, ATLAS detects information such as position, energy, and transverse momentum on every particle produced by the collision. Since every second of raw data recorded requires about 23 petabytes of storage space, a number of hardware and software 'triggers' are used to store only interesting and

important data. However, even after this selection process, about one petabyte of storage space per year is needed to record all of the data collected at the LHC. This information is made available for further analysis by researchers around the world.

Thanks to the work of a previous summer student here at Argonne, program InvMass has already been created in order to access this data and convert it into a more useful histogram format. Each histogram is classified according to the type of particles or structures produced in each event and contains information on the number of events produced for each value of the invariant mass of the collision. This classification results in hundreds of histograms, each of which contains potentially interesting information which must be explored.

For this reason, the ATLAS experiment is in constant need of faster and more efficient methods of analysis which can quickly alert researchers about important segments of the data which require further study. One such method is a general peak search algorithm, which runs over all data channels to identify possible peaks which are a sign of new physical phenomenon or particles. The general method for a peak search algorithm is to construct a background for the shape of each histogram and then compare the actual data to this background in search of significant deviations from the background. The goal of this project has been to develop such an algorithm and to test it on the latest data collected by the ATLAS detector. This report first discusses the background behind some of the attempted approaches and why they ultimately were unsuitable for the task at hand. It then goes on to present the settled upon method and discuss its merits and shortcomings.

Analytical Curve Fitting

One method for constructing a background for a given histogram uses the method of analytical curve fitting to fit a predefined, analytical function to the data using a fitting program. This was the initially chosen method used to develop the algorithm, due to it's widespread use in constructing backgrounds for individual histograms of ATLAS data. However, after numerous attempts at fitting

histograms in this way it became obvious that it is extremely difficult to find analytical function capable of fitting each histogram over its entire range (it is in fact doubtful whether or not such a function exists). Figure 1 is taken from a recent ATLAS report which shows a well fitted background. However, this background is clearly fit over a partial range of the data. Figure 2 is a set of plots taken from the output of the initial analytical peak search algorithm, which shows how a function may partially fit the data but deviate significantly outside of a certain range. For this reason it was decided that analytical curve fitting would not be acceptable for a general peak search algorithm, and further methods were investigated.

Non-Parametric Curve Smoothing

The next method investigated was that of non-parametric curve smoothing. When dealing with peak searches, it is most important to generate a good background for the given data. The original data can then be compared to the background in order to locate any possible peaks, and their statistical significance. Non-parametric curve smoothing seemed to provide a quick and easy method for smoothing out a given set of data points and therefore creating a usable background. The ROOT data analysis framework already provides a number of types of data smoothers including SmoothSuper and SmoothLowess, the latter of which was developed rather recently by Cleveland [4]. For the purposes of this project, SmoothLowess was chosen because of its ability to extrapolate data points to areas where data was not present before, which is extremely useful when generating a flat background for a peak. SmoothLowess was run over a number of different ATLAS histograms in an attempt to generate a reasonable background for the data. It should be noted that for testing purposes, a fake statistically significant peak was introduced into the histogram along with the original data. After numerous attempts it became clear that SmoothLowess was far too sensitive to peaks to be of any use for background generation, as shown in Figure 3. Using SmoothLowess, it proved impossible to generate a background that was flat enough compared to the peak, and the method was therefore abandoned.

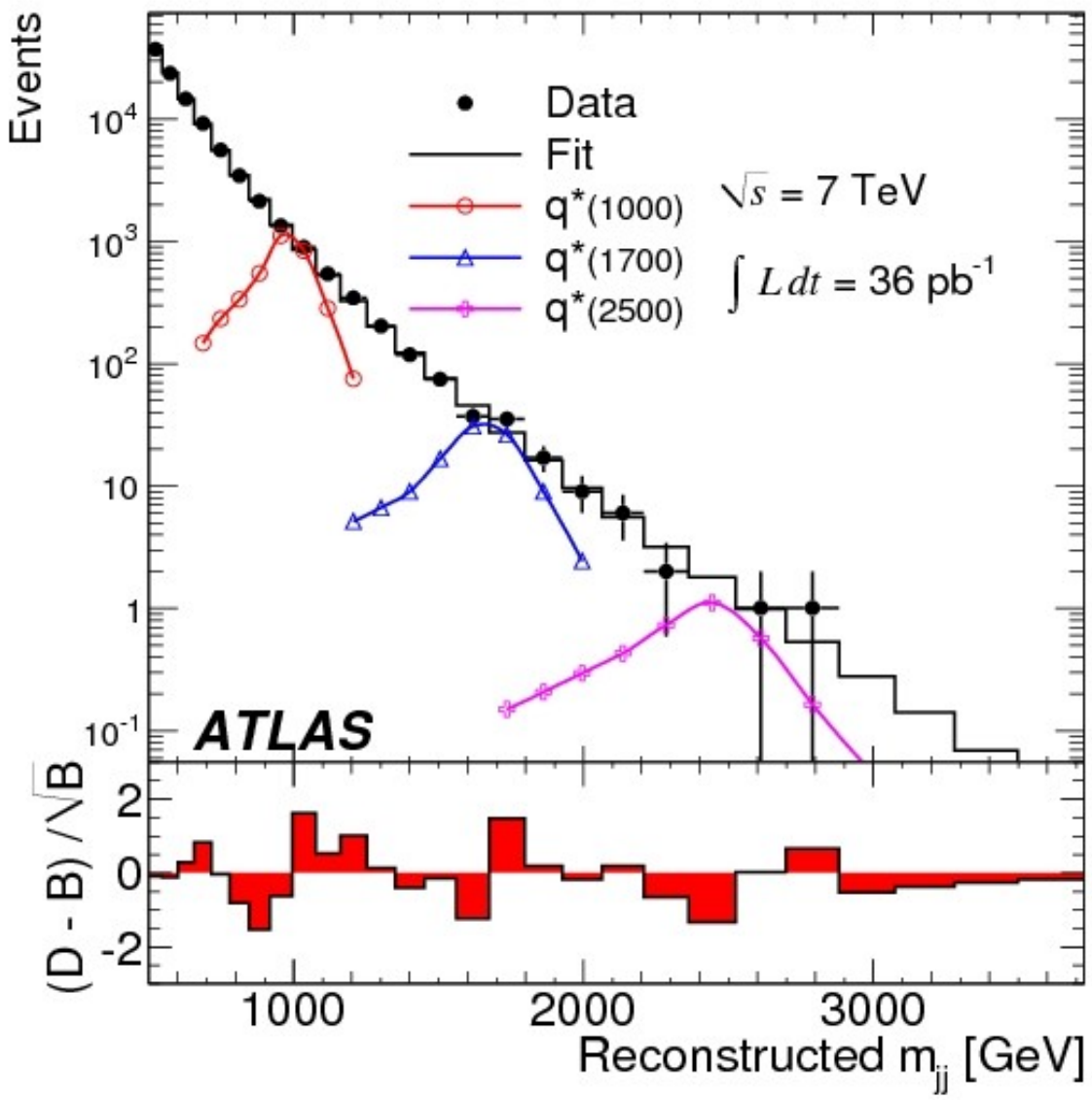
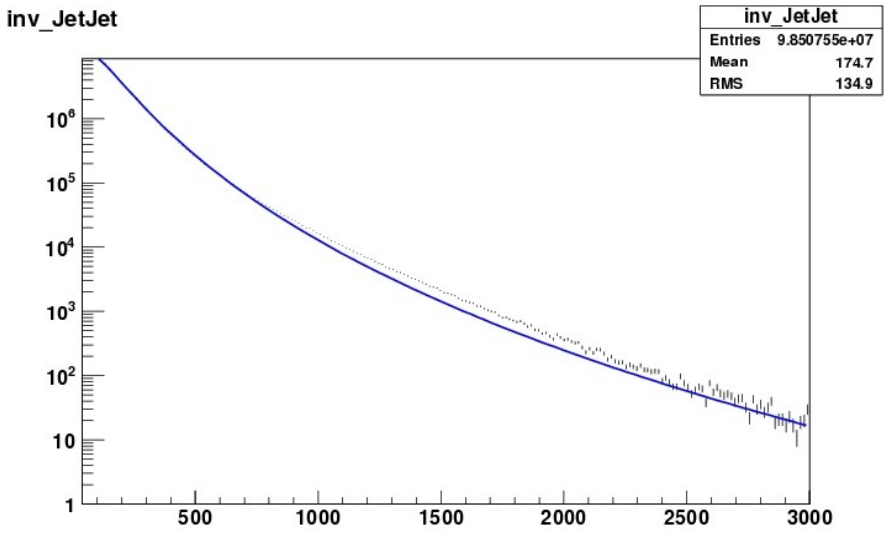
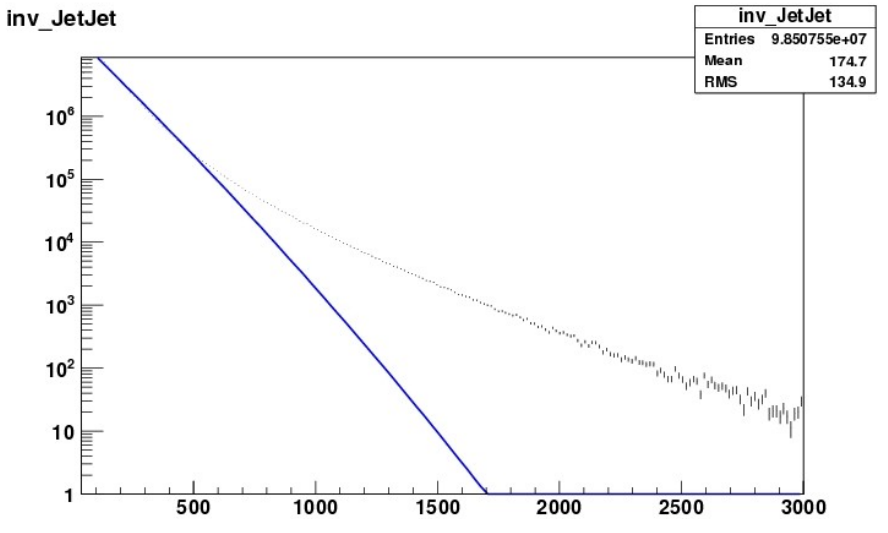


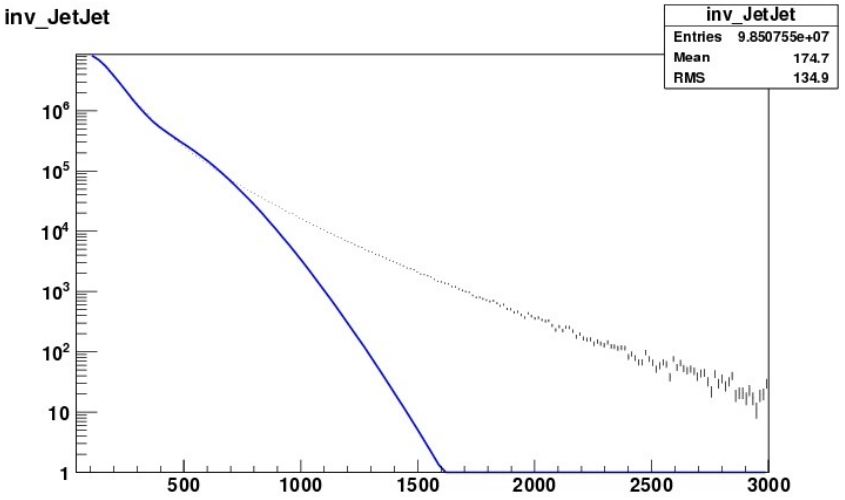
Figure 1: ATLAS fit for Dijet mass distribution [1]



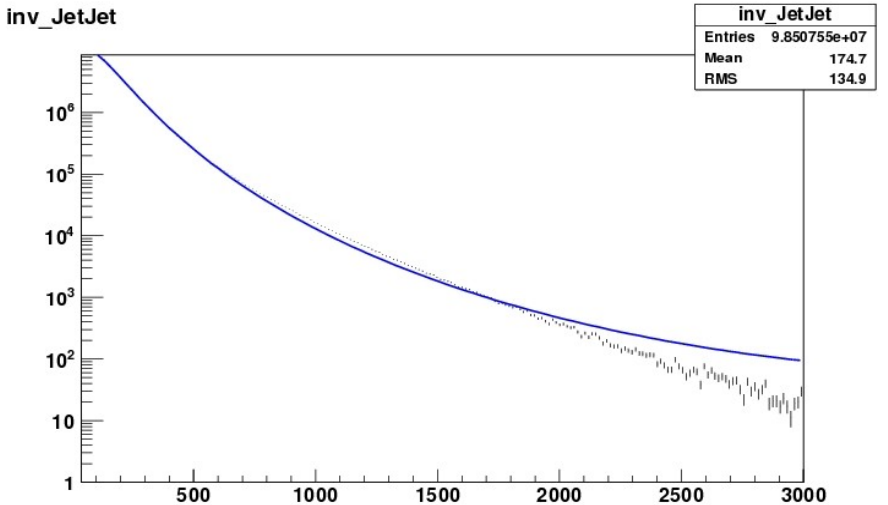
Dijet Function [3]



ATLAS Dijet Function [1]



ATLAS note function [2]



CMS function [5]

Figure 2: Plots for inv_JetJet histogram created using various fit functions

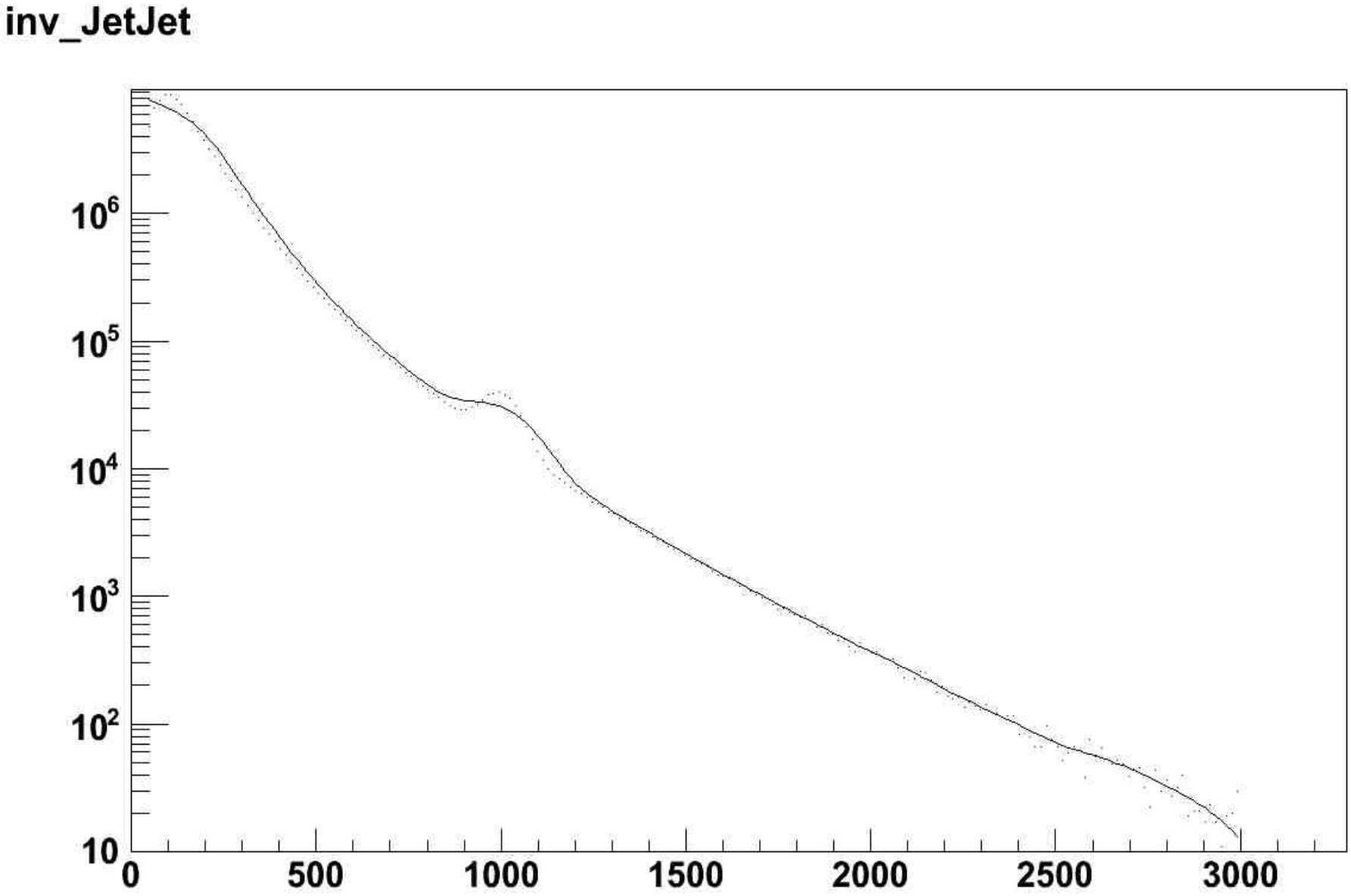


Figure 3: Plot of SmoothLowess applied to inv_JetJet histogram

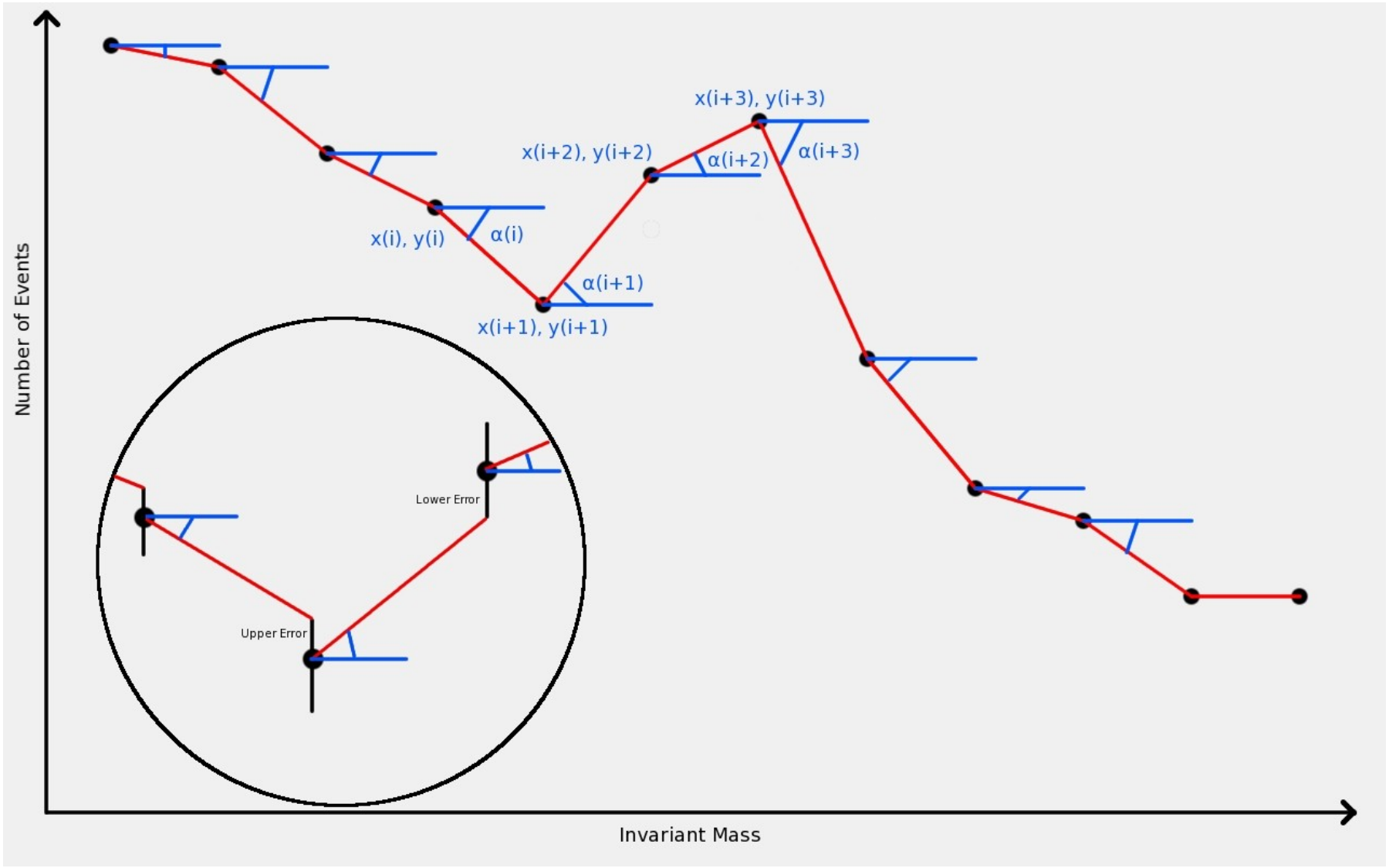


Figure 4: Illustration of NPPFinder algorithm

Non-parametric Analysis Method: NPPFinder

Due to the problems with the previous two approaches, it became clear that a novel approach was necessary to accurately detect possible peaks in the background spectrum. For this reason, the program Non Parametric Peak Finder (NPPFinder) was developed using a numerical, iterative approach and taking into account statistical uncertainties. In short, NPPFinder iterates through any given histogram and, using only data from the histogram and one input sensitivity parameter, determines the location and statistical significance of any possible peaks. Figure 4 is helpful for visualizing the approach, which is now outlined. For each point *i* in the histogram, the derivative α_i between points *i* and *i+1* is found taking into account statistical uncertainties. This is done by calculating the slope between two points with the error bars taken into account. If point i+1 is lower than point i, the upper error is used, while if point i+1 is higher than point i, the lower error bar is used. This process is shown in Figure 4 and can be summarized by the equation

$$\alpha_i = \frac{(y_{i+1} \pm dy_{i+1}) - y_i}{x_{i+1} - x_i}$$

where dy_{i+1} is the upper or lower uncertainty in y. The derivatives are then averaged up to some point N. This can be summarized by the equation

$$\bar{\alpha}_N = \frac{1}{N} \sum_{i=0}^{N} \alpha_i$$

where α_i is the value of the first derivative between points *i* and *i+1* and N is the total number of points so far. While this occurs, NPPFinder also checks whether $d\alpha_{i+1}$ and $d\alpha_{i+2}$, the changes in α_{i+1} and α_{i+2} respectively, are greater than $\bar{\alpha} + \sigma$, where σ is a user defined free sensitivity parameter. This can be summarized by the conditions

$$d\alpha_{(N+1)} > \bar{\alpha}_N + \sigma$$
$$d\alpha_{(N+2)} > \bar{\alpha}_N + \sigma$$

When this condition is true, NPPFinder registers a possible peak and begins classifying the following

points as part of the peak. This continues until $d\alpha_{i+1}$ and $d\alpha_{i+2}$ are both less than zero, which signifies the maximum of the peak has been reached. This can be summarized by the conditions

$$d\alpha_{(N+1)} < 0$$
$$d\alpha_{(N+2)} < 0$$

When this condition is met, NPPFinder exits the peak and adds an equal number of points as there were going up the peak to the end of the peak (since most peaks are symmetrical).

After all possible peaks have been detected, NPPFinder then iterates through the list of possible peaks in order to form a background for each peak. This is achieved by performing a linear regression of points between the first and last points in the peak. If $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ are the first and last points in a peak respectively, then the slope between them can be written as

$$m = \frac{(y_2 + dy_2) - (y_1 + dy_1)}{x_2 - x_1}$$

where dy_1 and dy_2 are the statistical uncertainties in y_1 and y_2. Here the statistical uncertainties are added instead of subtracted in order to always be on the conservative side for any given peak. The linear constant can be found by

$$b = (y_1 + dy_1) - m x_1$$

or

$$b = (y_2 + dy_2) - m x_2$$

Once m and b have been found for a specific peak, a linear regression line for the peak can be found by calculating new y values for each point *i* in the peak according to the linear equation

$$y_i = m x_i + b$$

NPPFinder then stores these new points as the background for the given peak.

Finally, NPPFinder uses the background points to calculate the statistical significance of each peak in a given histogram. This is done by summing the residuals of the original points in a peak with respect to the calculated background points, and then dividing this value by it's own square root. The

residuals, r_i, for each point *i* taking into account uncertainties are given by the equation

$$r_i = y_i(peak) - dy_i(peak) - y_i(background)$$

The sum of all values of r_i for a given peak can be found by

$$S = \sum_{n=1}^{N} r_i$$

where N is the number of points in the peak. The statistical significance of each peak is then calculated according to the equation

$$\sigma = \frac{S}{\sqrt{(S)}}$$

The result for each peak is then stored, and a graph showing the peak, it's background and it's statistical significance is printed. Figures 5, 6, and 7 show results from running NPPFinder over inv_JetJet. inv_GammaGamma, and inv_ElectronElectron histograms, with the detected peak points, linear regressions, and statistical significances displayed for each peak. For inv_JetJet and inv_GammaGamma, two false peaks were added at 1000 and 2500 for testing purposes. However, for inv_ElectronElectron no peaks were added and the detected peak represents the already known Z-boson particle.

Conclusion

Overall NPPFinder is a promising and novel approach for detecting peaks. The program can detect peaks automatically taking into account statistical uncertainties and has been tested using a variety of input conditions. The program has also been run successfully over 1 fb^{-1} of ATLAS data and the results show that no statistically significant peaks occur in the current set of data apart from peaks which are already known to be present. In addition to its applications to ATLAS data analysis, NPPFinder also has the potential to be useful in any area which requires peak detection and background estimation, and could in fact easily be extended to these areas of research. There are still a few possible

improvements to NPPFinder which are currently being addressed. These include but are not limited to the detection of broad peaks and the automatic determination of the sensitivity parameter. However, even in its current form, NPPFinder is a very useful tool for general peak detection and analysis.

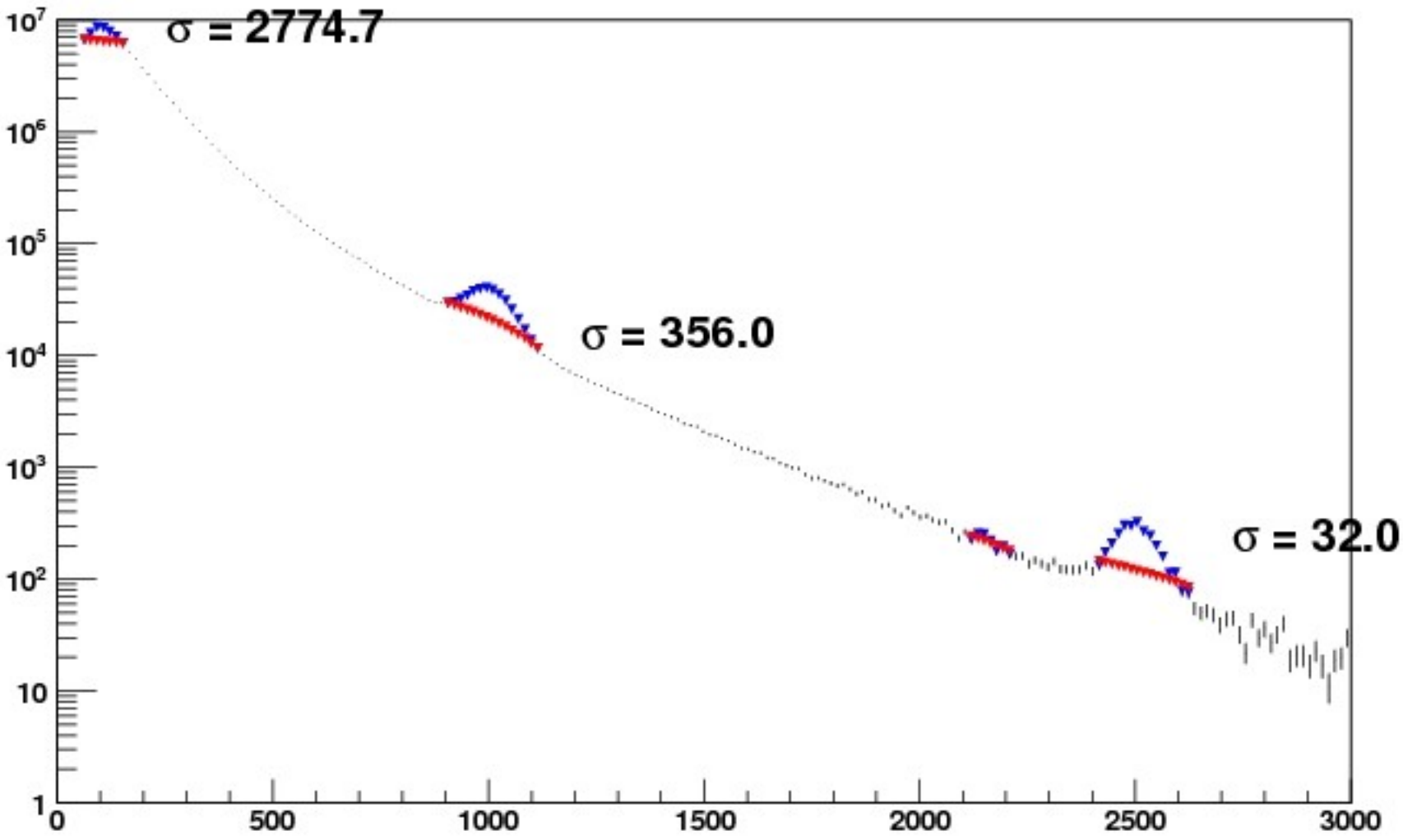


Figure 5: Output from NPPFinder for inv_JetJet

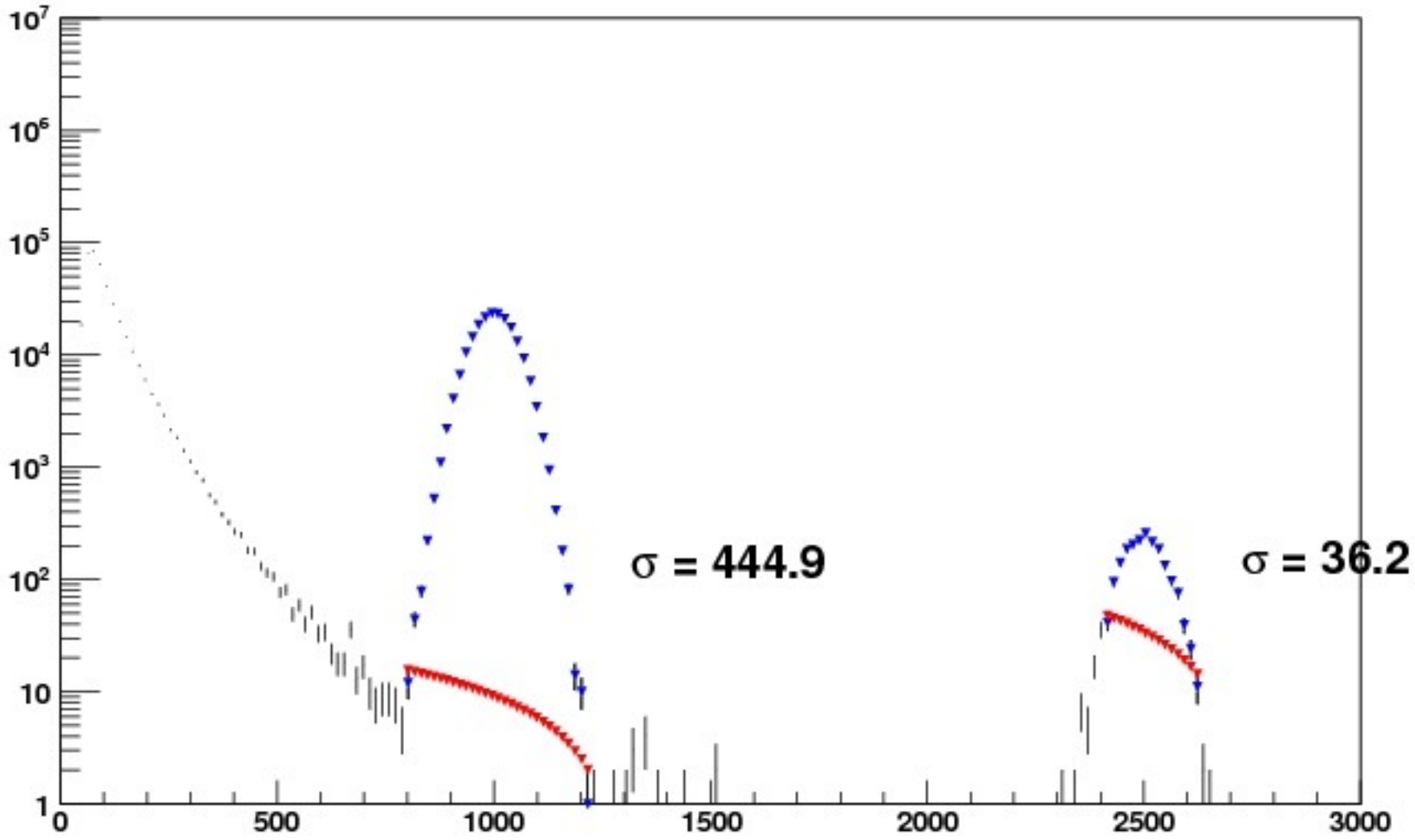


Figure 6: Output from NPPFinder for inv_GammaGamma

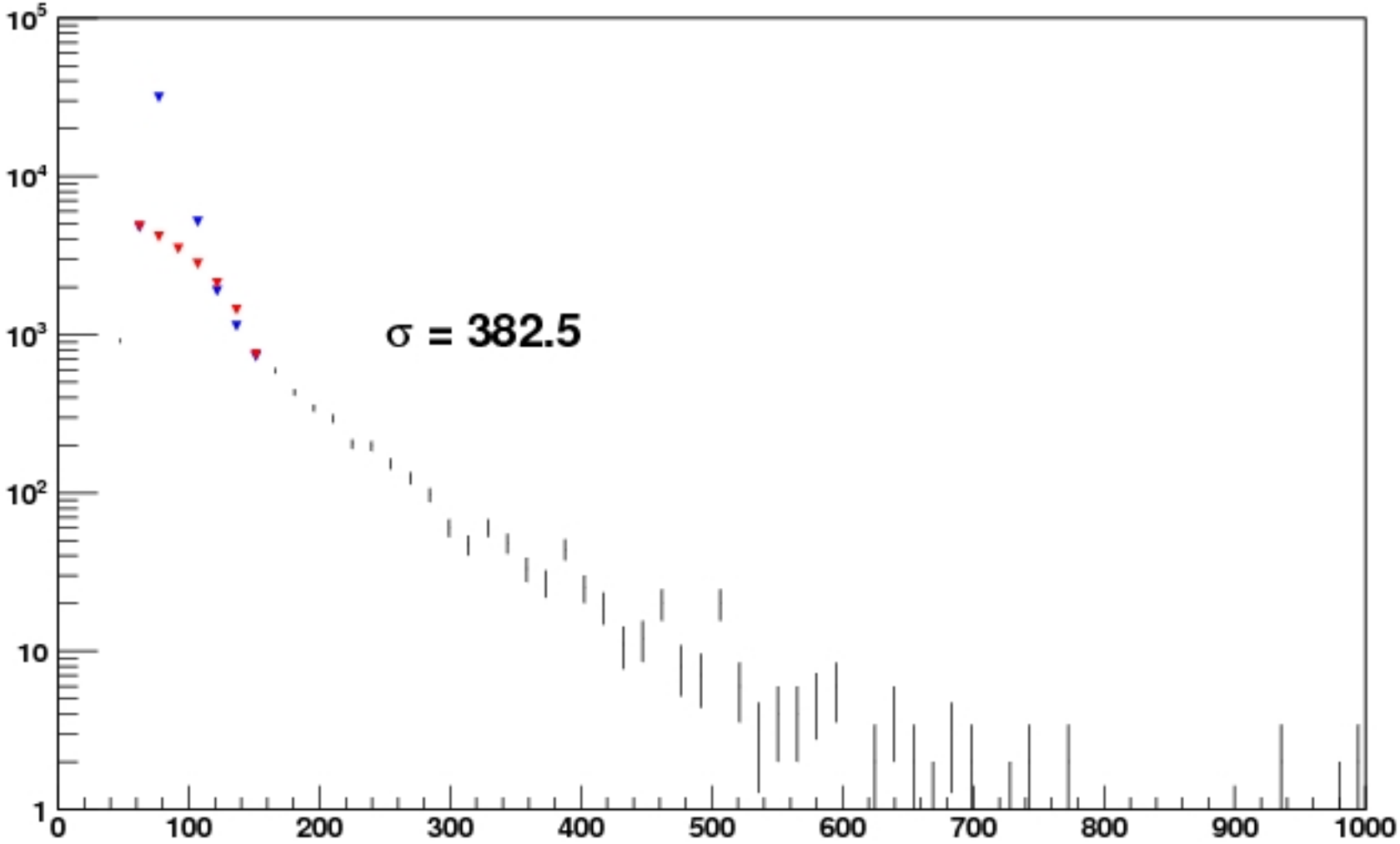


Figure 7: Output from NPPFinder for inv_ElectronElectron

References

[1] The ATLAS Collaboration, “Search for New Physics in Dijet Mass and Angular Distributions in pp collisions at $\sqrt{(s)}$ = 7 TeV Measured with the ATLAS Detector”, *New J. Phys.*, Mar 2011

[2] ATLAS Note

[3] A. Kupco, “Measurement and QCD analysis of inclusive dijet mass cross section in $p\bar{p}$ collisions at $\sqrt{(s)}$ = 1.96 TeV”, *Doctoral Thesis, Faculty of Mathematics and Physics at Charles University, Prague*, July 2003

[4] Cleveland, W.S. "Robust Locally Weighted Regression and Smoothing Scatterplots". *Journal of the American Statistical Association* *74* (368): 829–836.

[5] CMS