

Anomalous Jet Identification via Variational Recurrent Neural Network

Authors: Alan Kahn, Julia Gonski, Inês Ochoa, Daniel Williams, Gustaaf Brooijmans
Nevis Laboratories, Columbia University, Address

0.1 Method

Our method employs a Variational Recurrent Neural Network (VRNN) to model jets as a sequence of constituents. A VRNN is a sequence-modeling architecture which replaces the standard encoder-decoder architecture of a Recurrent Neural Network with a Variational Autoencoder. This allows the VRNN to perform both sequence modeling as well as *Variational Inference*, which has been shown to be a very powerful tool in the task of Anomaly Detection [Ref]. Figure 1 shows a diagram of one VRNN cell.

Inputs to the VRNN consist of sequences of jet 4-vector constituents. Before training, we apply a pre-processing method which boosts each jet to the same reference mass, energy, and orientation, such that all jets are superficially the same with the only differences being their substructure. In addition, our pre-processing method includes a choice of *sequence ordering*, in which the constituent sequence input into the model is sorted by k_t -distance instead of by constituent p_T . In more detail, the n^{th} constituent in the list is determined to be the constituent with the highest k_t -distance relative to the previous constituent, with the first constituent in the list being the highest p_T constituent.

$$c_n = \max(p_{Tn} \Delta R_{n,n-1})$$

The loss function used is very similar to that of an ordinary Variational Autoencoder. It consists of a mean-squared-error loss between input constituents and generated output constituents as a reconstruction loss, as well as a weighted KL-Divergence from the latent space prior to the learned approximate posterior distribution. Since softer constituents contribute less to the overall classification of jet substructure, each KL-Divergence term, computed constituent-wise, is weighted by the constituent's p_T -fraction, averaged over all jets in the dataset to avoid correlations with constituent multiplicity. An additional weight coefficient is enforced as a hyperparameter, and has been optimized to be 0.1 in our studies.

$$\mathcal{L}(t) = MSE + 0.1 \overline{p_T}(t) D_{KL}$$

The architecture is built with 16 dimensional hidden layers, including the hidden state, with a two-dimensional latent space. All hyperparameters used result from a hyperparameter optimization scan.

Our model is trained on the leading and sub-leading jets of each event, where our input datasets consist of the entirety Background and Black Box events. After training, we determine an overall *Event Score* by choosing the most anomalous score between the leading and sub-leading jets. To ensure consistency between training scenarios, Event Scores are subject

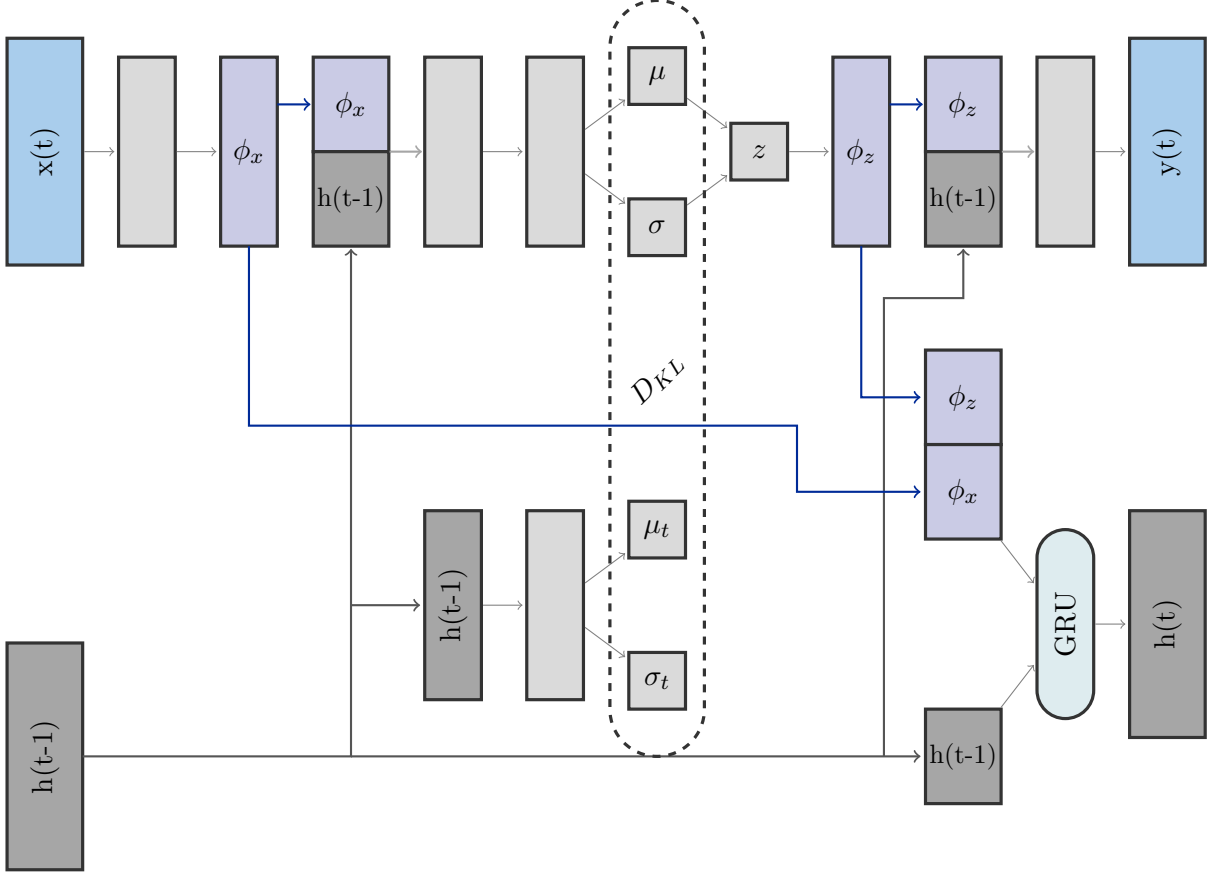


Figure 1. A Variational Recurrent Neural Network cell

to a transformation in which the mean of the resulting distribution is set to a value of 0.5, and Event Scores closer to 1 correspond to more anomalous events. Our results are then produced by applying a cut on the Event Score at a value of 0.65, as well as restricting the pseudorapidity of the leading and sub-leading jets to less than 0.75 to ensure that central, high momentum transfer events are being considered.

0.2 Results on LHC Olympics

We first present results with the R&D dataset. With a contaminated dataset consisting of 895113 background events and 4498 signal events corresponding to a contamination level of 0.5%, we aim to reconstruct the Z' mass peak using only a cut on the Event Score as a discriminator. Figure 2 shows the two-dimensional histogram of dijet combination mass vs Event Score, where we see the Z' mass at 3500 GeV clearly visible at higher values of the Event Score. Figure 3 shows the one-dimensional dijet mass distributions before and after a cut on the Event Score, along with the local significance of the signal in each bin. We see that a cut on the Anomaly Score dramatically increases the significance of the excess from

1σ to 6σ at a signal contamination of 0.5% while still retaining the smoothly falling behavior of the background

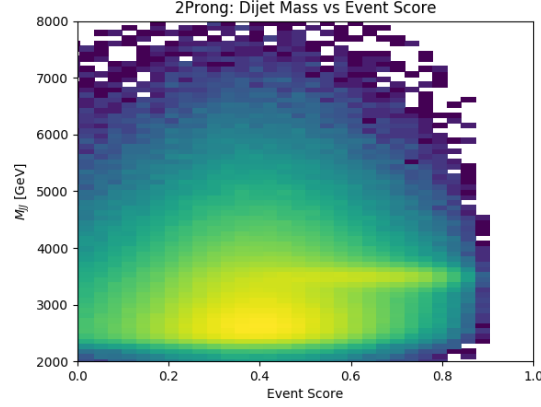


Figure 2. Description of the figure. Reproduced from Ref. [?].

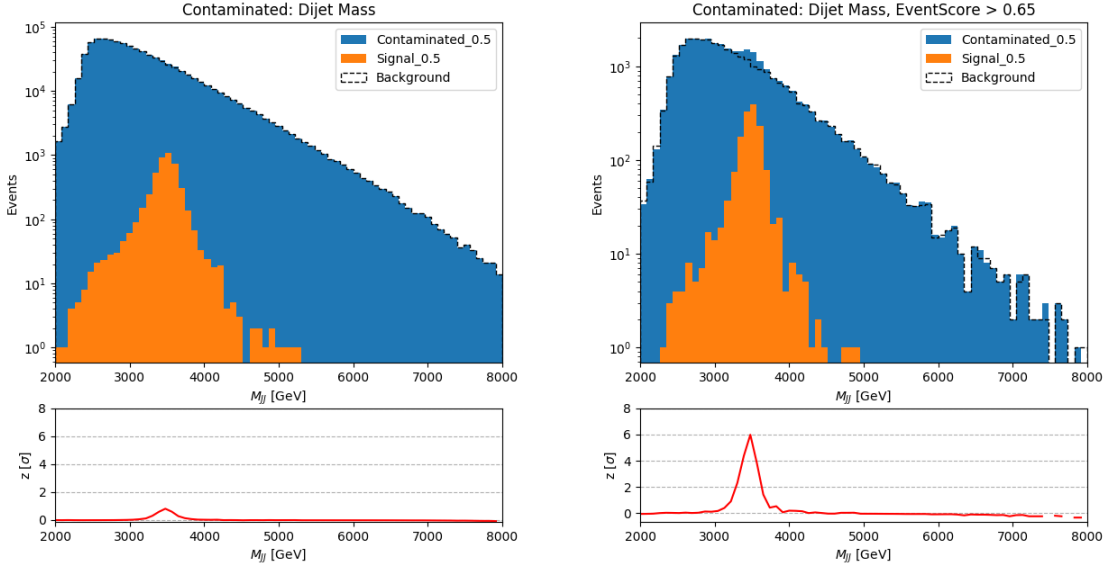


Figure 3. 2-Prong dijet mass distributions before (left) and after (right) a cut on the Event Score, at a signal contamination of 0.5%

Applying this method to Black Box 1, we arrive at the mass distributions of Figure 4. Plotted here is the mass of the dijet combination for both the Black Box 1 dataset as well as the Background dataset, displayed as a dashed line. The important feature in this result is

the shoulder present at around 4TeV, which corresponds to the present Z' signal at a mass of 3.8TeV.

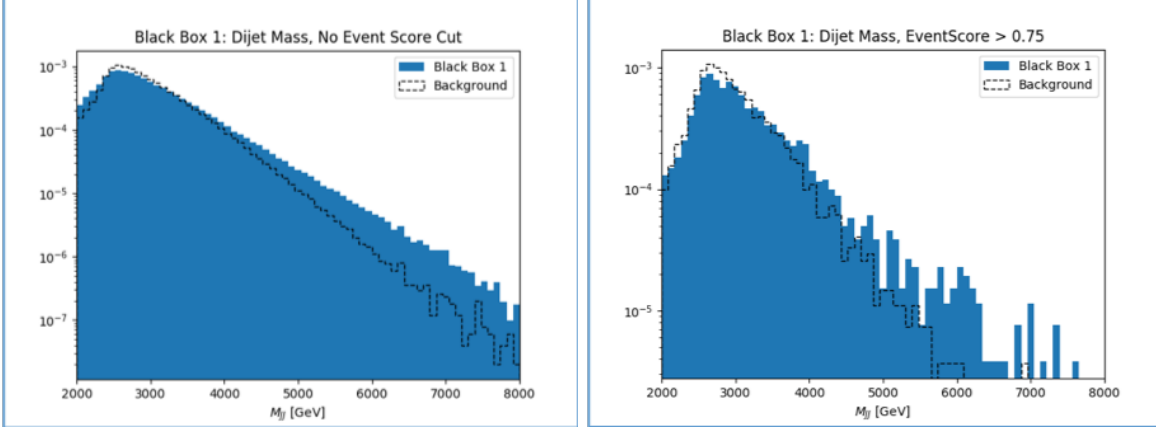


Figure 4. 2-Prong dijet mass distributions before (left) and after (right) a cut on the Event Score, at a signal contamination of 0.5%

In Black Box 2, our model shows no significant excesses. Furthermore, the behavior of the Event Score consistently affects the distributions of both Black Box 2 and the Background dataset. It is important to note that the model was trained independently on each dataset, and the resulting Event Scores are from entirely unique sets of network weights.

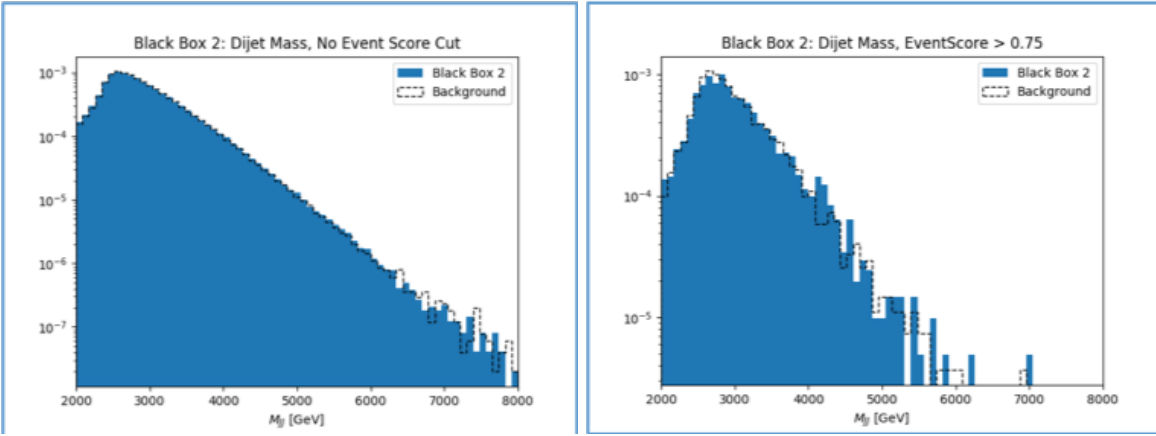


Figure 5. 2-Prong dijet mass distributions before (left) and after (right) a cut on the Event Score, at a signal contamination of 0.5%

Figure 6 shows our results for Black Box 3. Our model is specifically sensitive to boosted final states, and as a result, we are insensitive to the signal present in this Black Box.

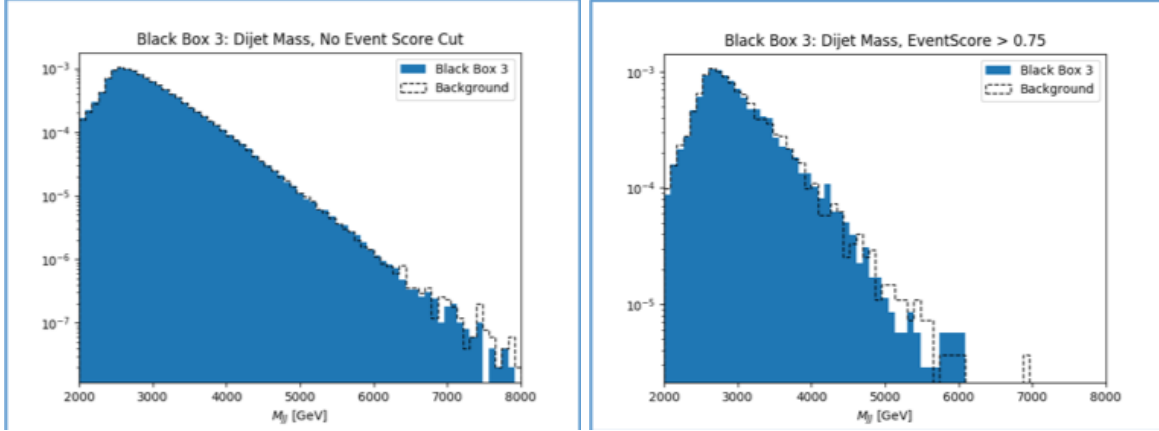


Figure 6. 2-Prong dijet mass distributions before (left) and after (right) a cut on the Event Score, at a signal contamination of 0.5%

0.3 Lessons Learned

This challenge presented a highly accessible avenue for development of our model. We are particularly surprised by the effect of our pre-processing method on the performance of the model, specifically by our choice of k_t -ordered sequencing, and find it to be a very interesting result which may have a number of applications in a wide range of analysis contexts. Overall, we view the VRNN as a model capable of identifying objects within a contaminated dataset as long as they can be characterized by sequential data. While we limited our scope in this study to be entirely unsupervised with absolutely no external information helping the model understand the elements of data that it is modeling, the simplicity of the VRNN’s architecture, being both a RNN and a VAE, opens the possibility of accommodating more supervised training scenarios. We also restricted some of our development tracks in the interest of simplicity, but feel that a number of adjustments to the architecture, such as a dedicated adversarial mass de-correlation network, or an additional input layer representing high-level features, are worthwhile avenues of exploration.

Acknowledgments

This work was supported by the U.S. Department of Energy, Office of Science under contract DE-AC02-05CH11231.

For the references, please use names from Ref. [?] . If your paper is not there or is not updated, please submit a MR!