When Jet Quenching Meets Machine Learning
Lihan Liu1, Marta Verweij2
1 Vanderbilt University, Nashville TN, USA
2 Utrecht University, Utrecht, Netherlands

Identification of Quenched Jets

In this poster I show how machine learning techniques can help to evaluate how quenched a jet is. To achieve this, we used the Long Short-term Memory (LSTM) model. Simulations were done with Monte Carlo event generators such as Pythia [1] and JEWEL [2] which simulates jets in vacuum and in medium respectively.

Feature Selection

Jets are recursively de-clustered into two subjects with Cambridge-Aachen (C/A) algorithm. The primary path, marked in red in Fig. 2, follows the harder branch at each branching. Each node on the primary path contains a set of substructure variables which are used to train a classifier based on LSTM model.

Input vectors to the LSTM network \([x_0, x_1, \ldots, x_n]\) are taken from jet binary tree which come from recursive de-clustering. At step \(t\), the feature vector \(x_t\) is a combination of substructure variables.

\[
z = \min(p_{T1}p_{T2}), \quad \Delta = \sqrt{(\eta_1 - \eta_2)^2 + (\varphi_1 - \varphi_2)^2}, \quad k_1 = p_{T1}^2 + \Delta, \quad m = \text{inv_mass}(j_1, j_2)
\]

Simulation & Training

For training, a classifier was trained using the Long Short-term Memory (LSTM) model. Calculations that happen in a LSTM cell can be expressed as follows:

1. Forget gate: \(f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)\)
2. Input gate: \(i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)\)
3. Update gate: \(C_t = f_t \cdot C_{t-1} + i_t \cdot \tanh(C_t)\)
4. Output gate: \(o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)\)

\(W, b\): learnable weights and biases.

Substructure Variables

Jets with different predicted values populate the Lund radiation plane in different ways (Fig. 7). Substructure variables such as \(z, \Delta\) and jet masses are compared (Fig. 8). Classifier no. 4 is used in making predictions.

Jewel jets with smaller predicted values (Istm<0.8) show similarities to Pythia jets which can be observed from distributions of substructure variables (see Fig. 8).

Table: Simulation and training details.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Input Features</th>
<th>Input Size</th>
<th>Hidden Size</th>
<th>No. Hidden Layers</th>
<th>No. of Parms</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[lnz, lnK]</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>248</td>
</tr>
<tr>
<td>2</td>
<td>[lnz, lnK, lnK_1]</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>248</td>
</tr>
<tr>
<td>3</td>
<td>[lnz, lnK, lnK_1, lnm]</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>256</td>
</tr>
</tbody>
</table>

This study has shown that the LSTM network can take jet substructure variables as input and develop its identification ability thus making it a practical approach to identify quenched jets.

It also shows that different feature selections may result in classifiers with different distinguishing abilities, as shown in Fig. 4.

References:

Footnotes:
1. It is configurable in Pythia framework that the data loader can run with multi-threads.
2. CPU: i7-8700K @ 3.70GHz: 4 out of 8 threads are used.
3. The total number of parameters scale with input size, output size and the number of hidden layers.