Interpretability of Interaction Network for Identifying $H \rightarrow b\bar{b}$ jets from QCD Background

Avik Roy, Mark Neubauer
University of Illinois at Urbana-Champaign

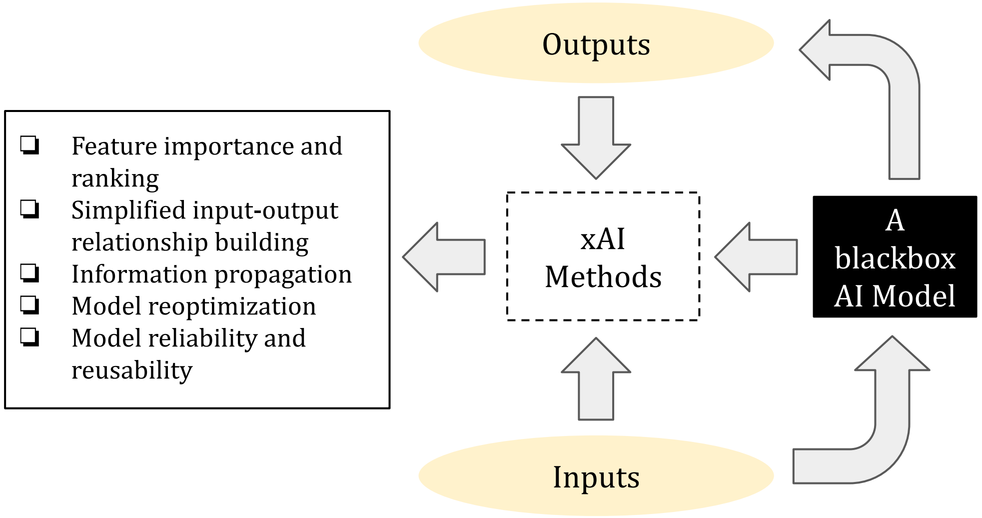
July 9, 2022

FAIR4HEP

ICHEP 2022

Interpretability of AI Models

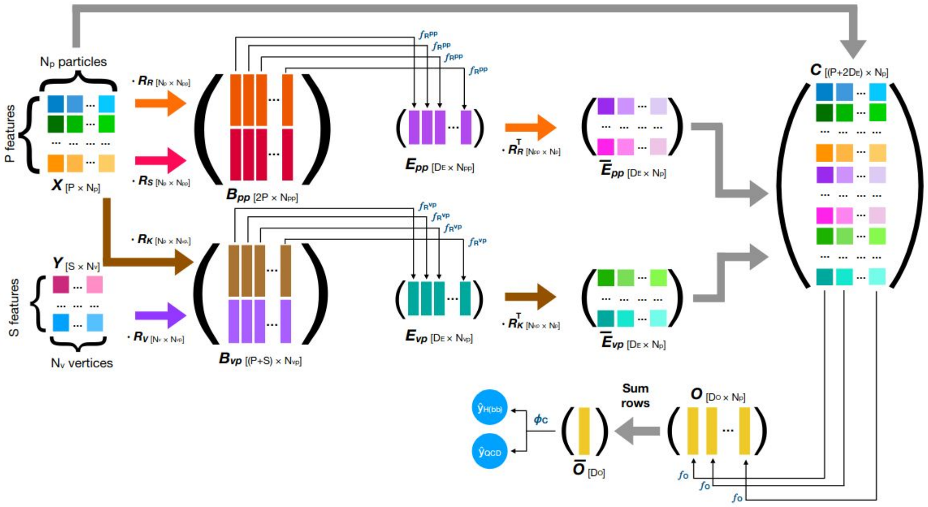
- AI and ML models are successful but largely *mysterious*
- Methods of Explainable AI (xAI) allow better understanding of why AI models work
- Still quite novel in HEP applications
- Prospect and scope has been discussed in the Snowmass white paper: [arxiv 2206.06632]



The Interaction Network Model

- State of the art Graph Neural Network (GNN) Model trained to distinguish $H \to bb$ jets from QCD background
- Trained with full CMS detector simulation, made available via CMS Open data
- Input to the model:
  - 60 particle tracks, 30 features per track
  - 5 secondary vertices, 14 features per vertex
  - Particle-particle and particle-vertex interaction matrices create an interaction network
  - Three MLP as transformation networks:
    - f_r : particle interaction
    - f_r^{pv} : particle-vertex interaction
    - f_o : pre-aggregator

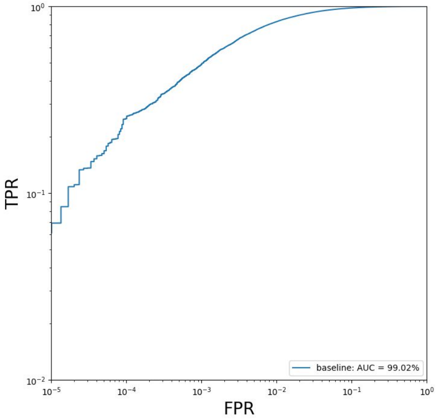
Image from: 10.1103/PhysRevD.102.012010



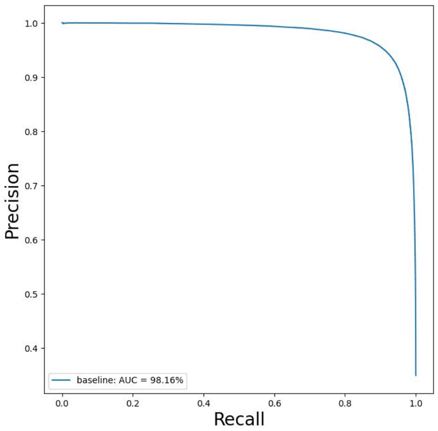
FAIR4HEP

Baseline Model architecture and Performance

- Each MLP has three hidden layers with 60 nodes per hidden layer
- Other hyperparameters:
  - D_e = dimension of particle-particle and particle-vertex interaction internal representations = 20
  - D_o = dimension of pre-aggregator network representation = 24
- Trained with dataset that roughly has a 2:1 distribution for QCD and Hbb jets
  - Validation accuracy of 95% (for a decision threshold of 0.5) with an ROC-AUC of 99.02%



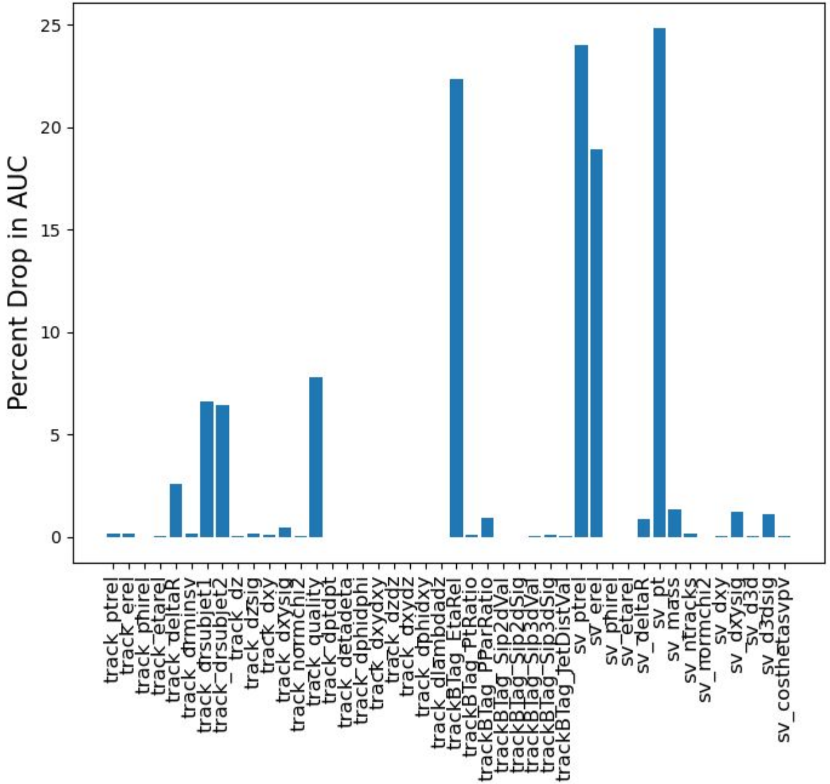
ROC curve



Precision Recall curve

Identifying feature importance

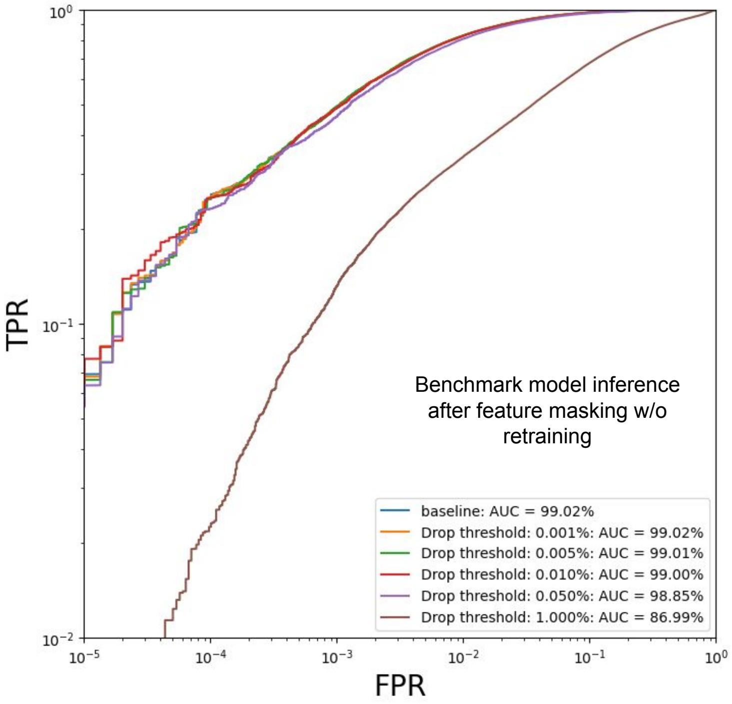
- Which particle and secondary vertex features play the most important roles?
- **Occlusion Test**: Identify by masking each feature across all nodes and evaluating the model's performance characteristic AUC for ROC and precision-recall curves
- Masking done by replacing entries by zero



Model Reevaluation with Multiple Features Masked

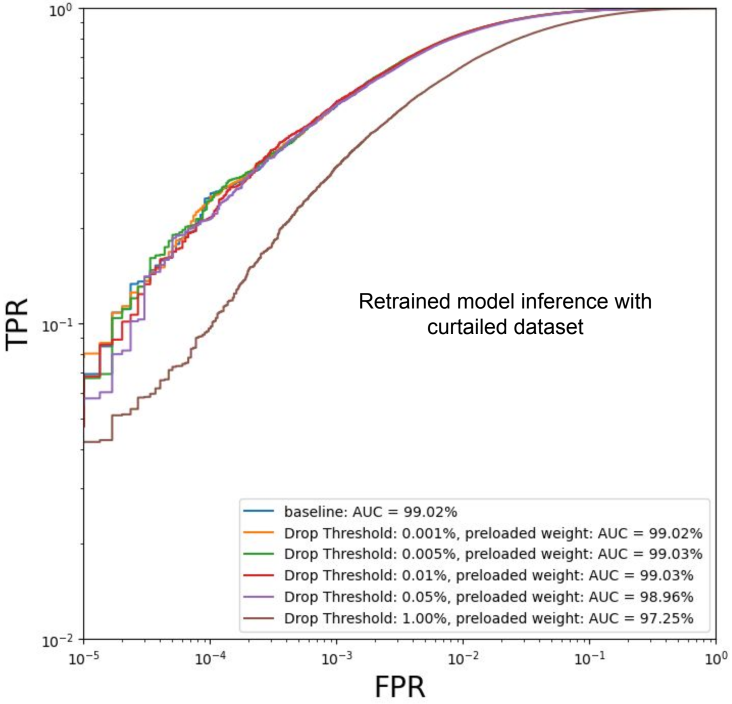
- Based on the ΔAUC measure list of *relatively unimportant* features can be found
- The model's performance was reevaluated by simultaneously dropping multiple features

ΔAUC threshold	# Particle features dropped	# Vertex features dropped
0.001%	8	2
0.005%	9	2
0.01%	11	3
0.05%	14	4
1.00%	25	8



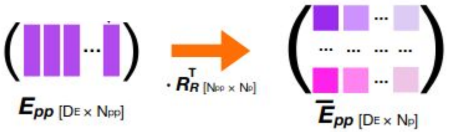
Model Retraining

- To compensate for performance loss, the model was retrained with reduced feature space
- To accelerate model training, relevant weights were preloaded from the baseline models
- Preloading allowed training to be 3x as fast
- Model performance was recovered for all cases, including the large drop case of 1% drop threshold

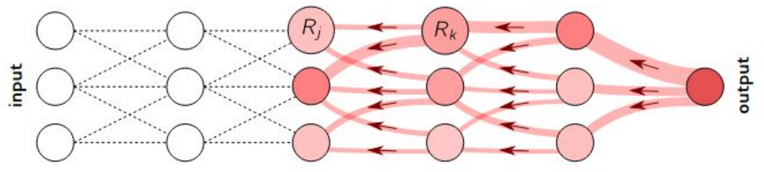


Layerwise Relevance Propagation (LRP)

- Propagates the output of a network backwards and distribute it among input features
- Across each layer of an MLP, relevance score from the next layer is back-propagated and linearly redistributed to the current nodes
- For the Interaction Network model, the relevance propagation formula needs to be adjusted
- For 

- For 

$$R_j = \sum_k \frac{z_{jk}}{\sum_j z_{jk}} R_k$$

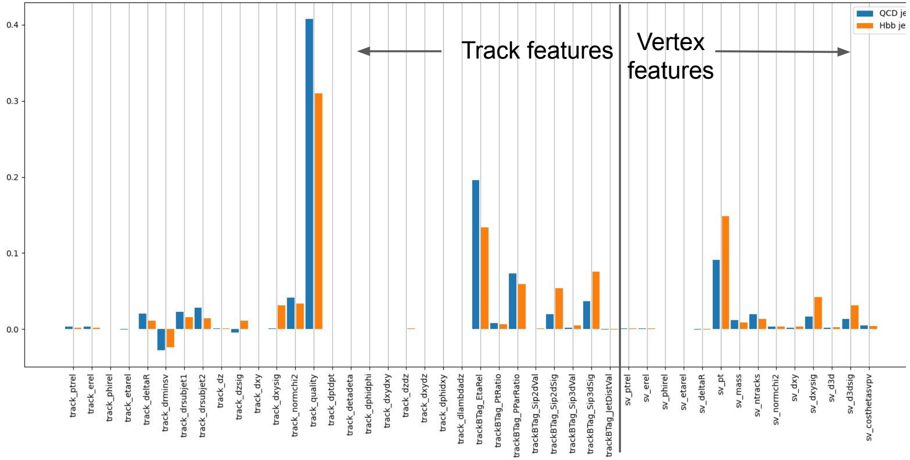


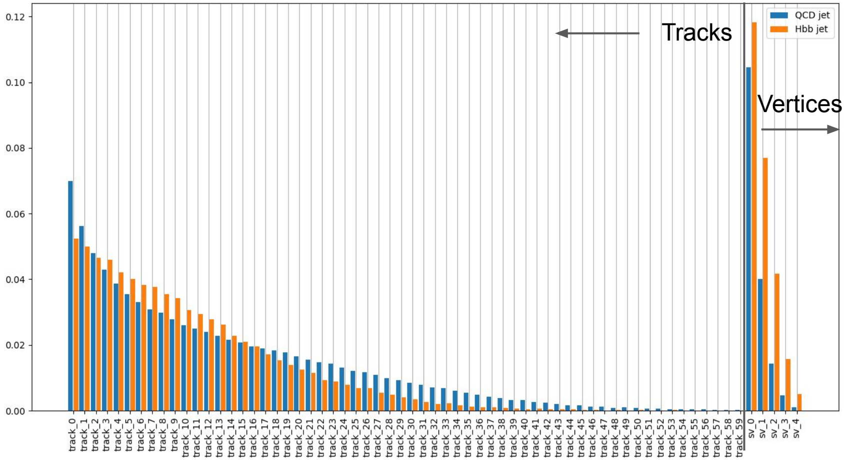
$$R_{kn} = \frac{o_{kn}}{\sum_n o_{kn}} \bar{R}_k$$

$$R_{km} = e_{km} \sum_n \frac{\bar{R}_{kn}}{\bar{e}_{kn}} R_{r_{nm}}$$

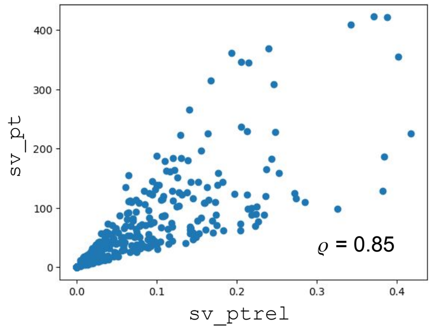
Results from LRP-γ

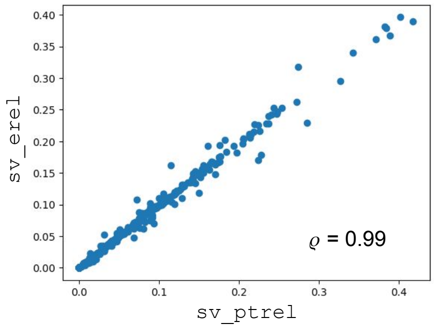
- Chose to use the LRP-γ algorithm with $\gamma = 2.0$
- Lossless LRP- total relevance preserved, redistributes relevance via positive weights and suppresses correlation among features

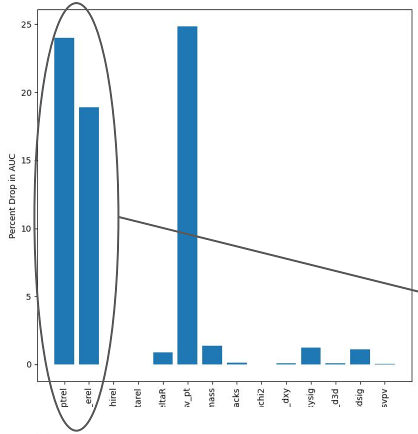


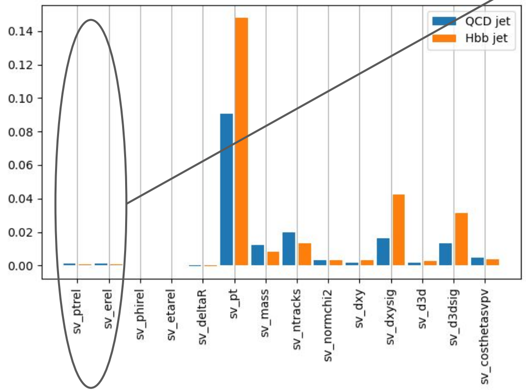


Taking a Closer Look I: The secondary vertex features





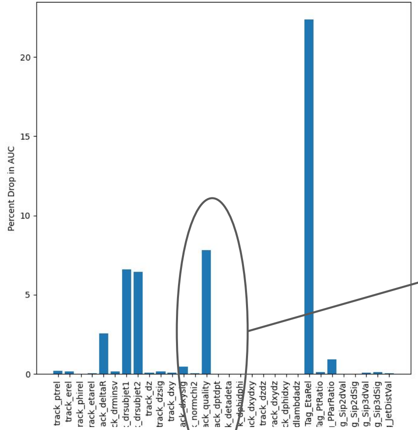


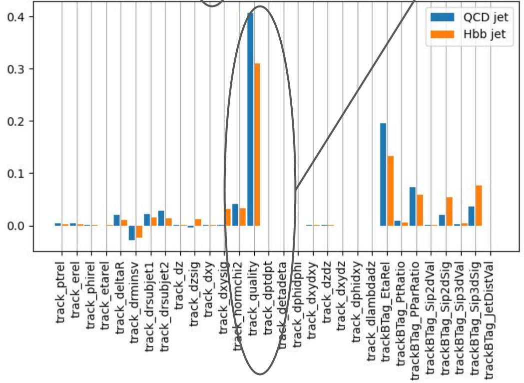


`sv_ptrel, sv_erel` rank very high in AUC score ranking but very low in LRP.

LRP-γ suppresses importance of correlated features by concentrating score to features with positive weights

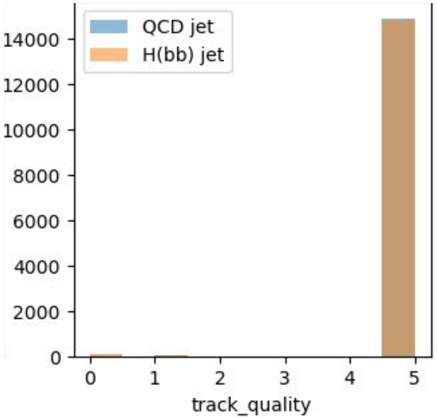
Taking a Closer Look II: The feature that actually doesn't matter





`track_quality` ranks very high in both feature ranking.

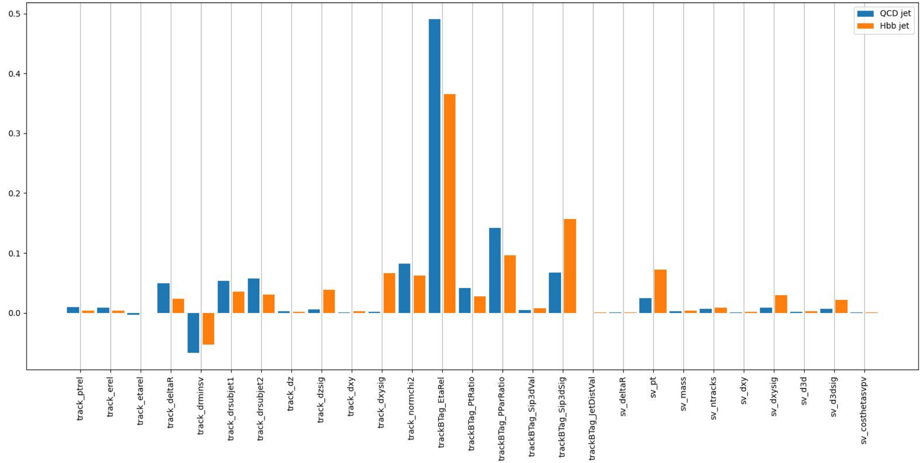
It is an arbitrarily chosen flag and distribution is almost identical for both jet categories

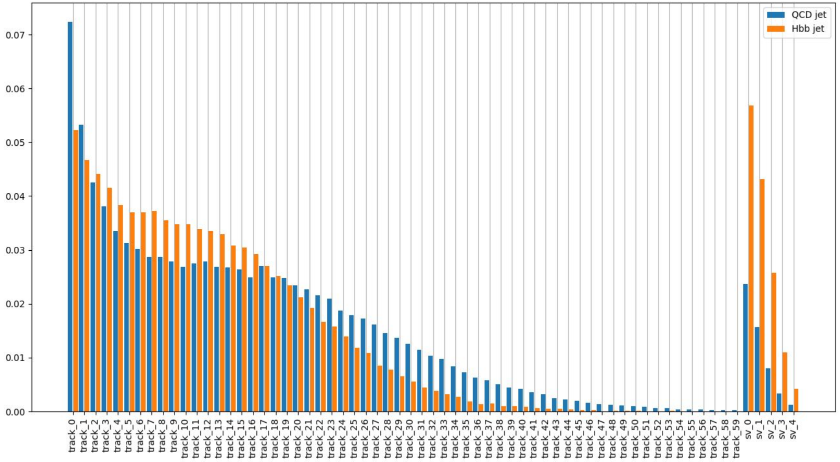


Almost all tracks have the same track_quality value, so it has no discriminating power at all. Just acts as a large additive contribution towards model output (like a bias)

Retraining model without these features

- Retrained model without these features
- Model performance is almost identical to benchmark (AUC = 99.00%)



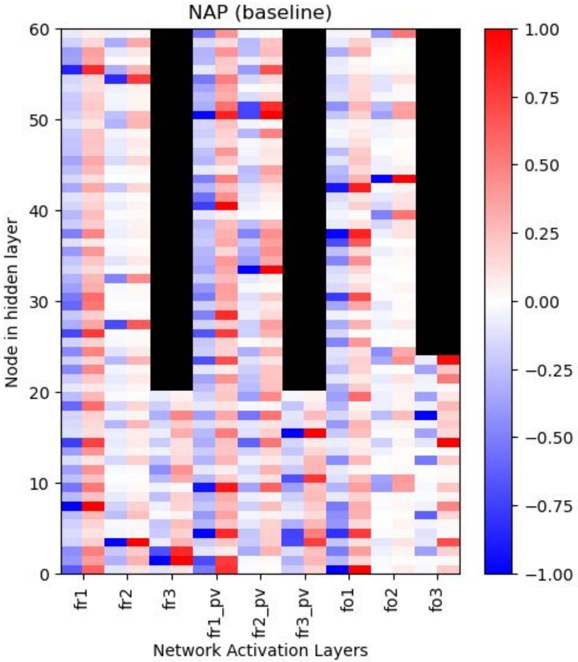


LRP score distribution for other features/tracks/vertices are similar

Neuron Activation Patterns (NAPs)

- Feature importance metrics don't reveal any information about the model's inner workings
- Understanding the model's inner workings help with hyperparameter reoptimization
- To see how the hidden layers respond to input data, we look at the Relative Neural Activity (RNA) score for different nodes within a layer

$$\mathrm{RNA}(i, j; \mathcal{S}) = \frac{\sum_{i=1}^{N} a_{j,k}(s_i)}{\max_j \sum_{i=1}^{N} a_{j,k}(s_i)}$$

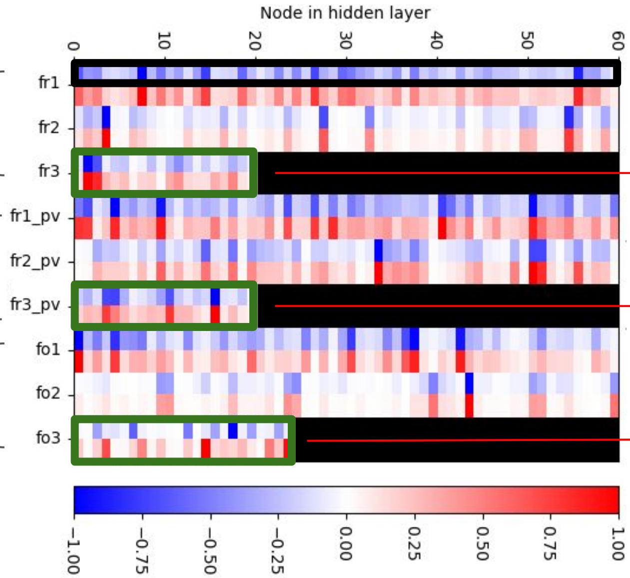


NAP: Taking a Closer Look

Hidden layers for particle interaction transformation network

Hidden layers for particle-vertex interaction transformation network

Hidden layers for pre-aggregator network



For each hidden layer, the distribution is normalized w.r.t largest aggregated neuron activation in the respective layer

Output layer of the particle interaction transformation network, dimension D_e

Output layer of the particle-vertex interaction transformation network, dimension D_e

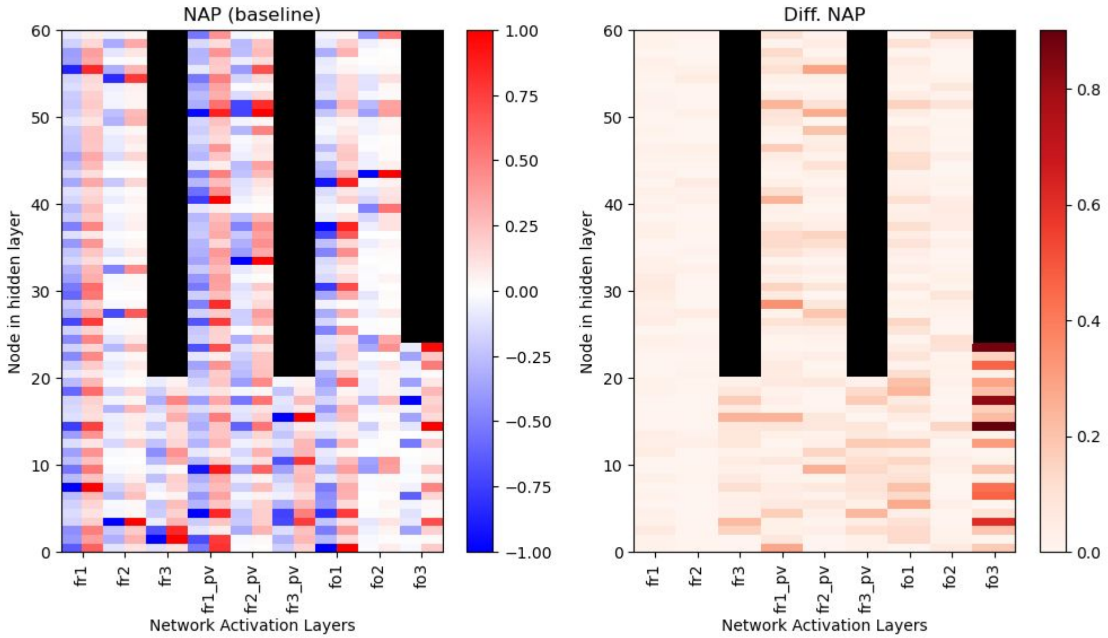
Output layer of the pre-aggregator network, dimension D_o

Activation for QCD jets

Activation for signal (H > bb jets)

What do NAPs Tell us?

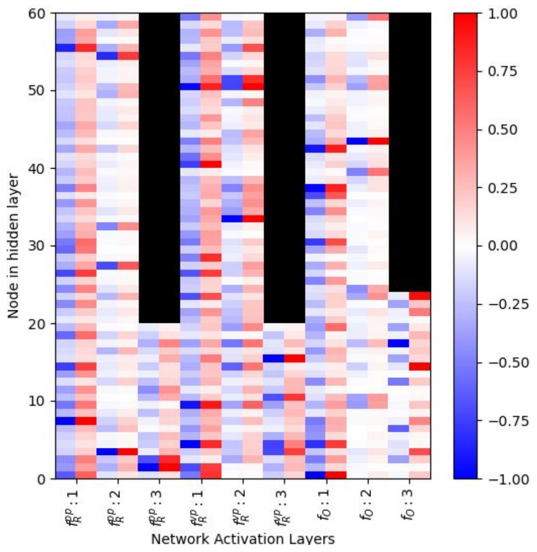
- Activation is rather sparse, largest activations at each layer is are shared by handful of nodes
  - There is scope for model simplification
- Until the very last layer (*fo3*), the activation patterns for signal and background are similar
  - Internal space distributions might not be effective classifiers



Hyperparameter Reoptimization

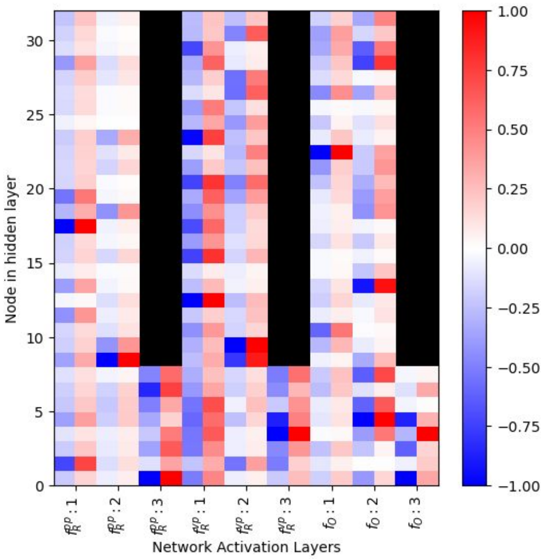
- NAPs reveal crucial sparsity in model's internal structure
- Can be measured by calculating the number of nodes with RNA score less than a given threshold, e.g. 0.2
- This can be further illustrated by comparing NAPs for a *squeezed* model:
  - Features associated with a ΔAUC < 0.01% along with `track_quality`, `sv_ptrel`, `sv_erel` are dropped
  - 32 nodes/layer, $D_e = D_o = 8$
  - Gives an ROC-AUC of 98.84%

Baseline model NAP



Sparsity: 0.44

Squeezed model NAP



Sparsity: 0.36

Summary

- These results (along with additional details) are compiled in the notebooks in this repository
- Our studies suggest:
  - Feature importance metric (via occlusion test or methods like LRP) can be useful in selecting important features
  - NAP diagrams are useful indicators of model sparsity and hence can allow better insight into model complexity reduction and hyperparameter reoptimization
  - Additional physics insight may be needed to determine reliability of xAI metrics
- Future direction of studies:
  - xAI for a wider range of collider physics problems
  - Developing physics inspired metrics for xAI in HEP
  - Interlink between model explainability and uncertainty quantification