

Model interpretability methods for high energy physics analysis

A. Zaborenko¹, L. Dudko¹, P. Volkov¹, G. Vorotnikov¹, E. Abasov¹

¹SINP MSU, Moscow

Outline

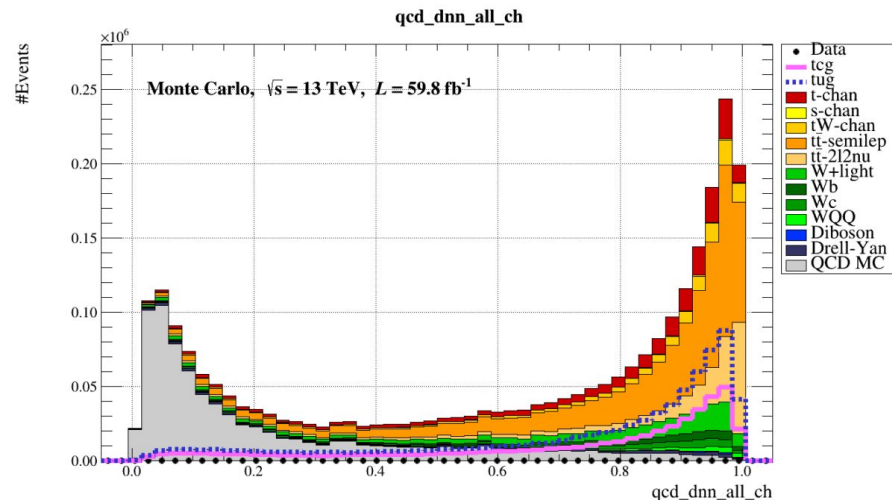
1. The physics task:
 - 1.1. The data
 - 1.2. The model
2. SHAP Explainer:
 - 2.1. Mathematical overview
 - 2.2. Global and local explanations
 - 2.3. Visualization
3. Glass-box models: EBM
 - 3.1. Generalized additive models
 - 3.2. Results

Physics

In this talk we will focus on explaining the output of the model used for QCD multijet background suppression.

It's a binary classification task with kinematic variables as input features.

A multilayer perceptron neural network is used to separate background and signal events.



An example of the discriminator of the QCD suppression network. Background events (grey) are grouped near 0.0, while all other events are grouped near 1.0.

Model interpretability: why do we need it?

Model interpretability methods are used to answer the question: why did a machine learning model make a specific “decision”?

It might be useful for many reasons:

- Providing insights into the data
- Making the model more trustworthy
- Debugging the model
- Detecting bias for more fair predictions

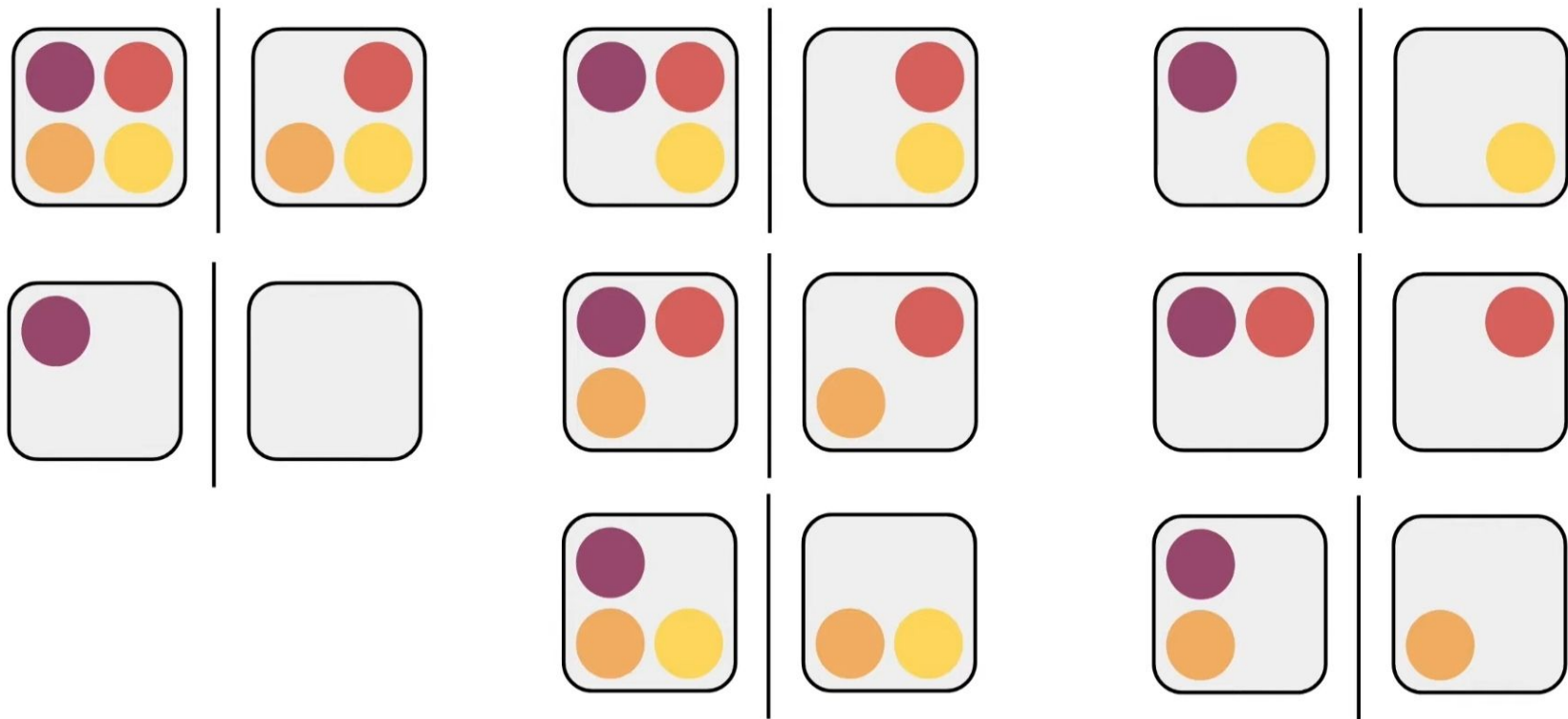
SHAP as a model interpretability example

Shapely Additive Explanation technique uses methods originating from game theory called Shapley values. These methods allow one to calculate each member's contribution in a coalition.

The general outline of the algorithm:

1. Get all subsets that don't contain the feature i
2. Find the marginal contribution of the feature i to each of these subsets
3. Aggregate all marginal contributions to compute the contributions of the feature i

Generating subsets and measuring contributions

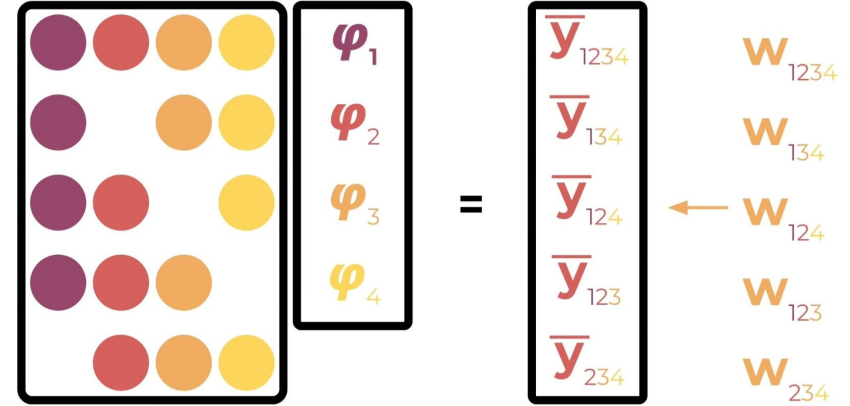


Shapley kernel

The number of coalitions to sample grows exponentially with number of features, so a process of approximating the Shapley values was created.

First, permutations of original example are created. The model output is then recorded for each permutation.

Then, by solving a linear regression with specific weights, the true Shapley values for each feature are obtained. These values indicate feature contribution for this specific example.



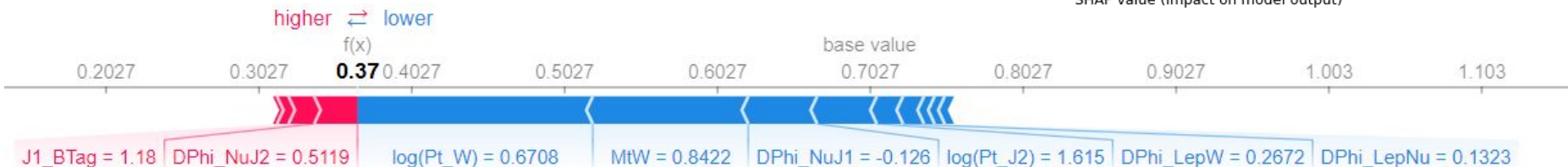
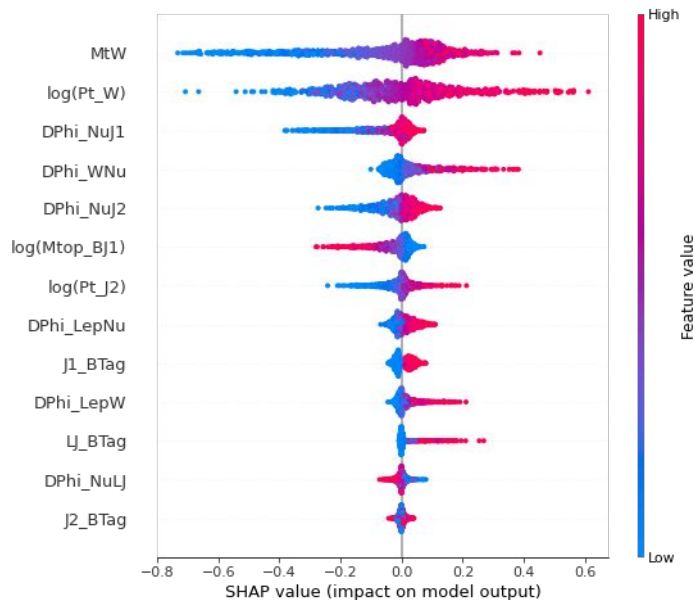
$$w_C = \frac{\# \text{ total features} - 1}{\# \text{ coalitions of size } |C| * \# \text{ included features in } C * \# \text{ excluded features in } C}$$

SHAP: local and global explanations

By aggregating local explanations we can understand how a certain feature influences the decision of the model in general.

Global explanations (Right)

Local explanation (Bottom)




Explainable Boosting Machine (EBM)

The EBM is a generalized additive model (GAM) which was designed to be interpretable (glass-box).

EBM uses gradient boosting to find the function f_j for every input feature. Then these functions are combined to create the final prediction.

This way to understand the effect a certain variable has on model output, we just need to plot this f_j .

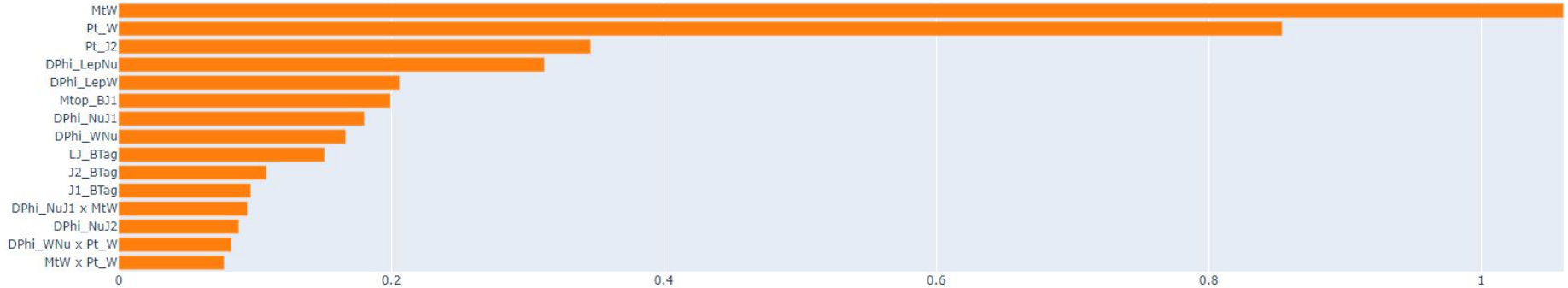
$$g(E[y]) = \beta_0 + \sum f_j(x_j)$$


Learned through gradient boosting

$$g(E[y]) = \beta_0 + \sum f_i(x_i) + \sum f_{i,j}(x_i, x_j)$$

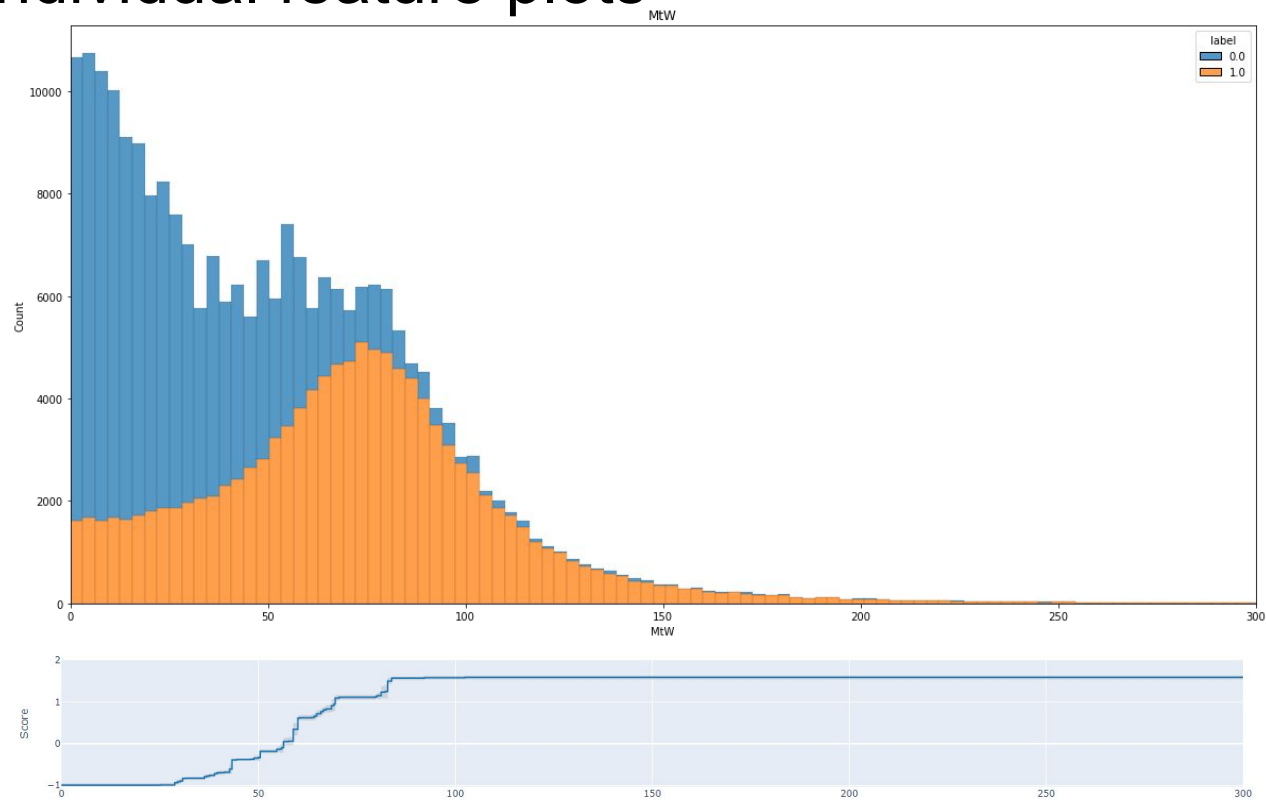
Pairwise interactions can also be modelled

EBM: feature importance plot

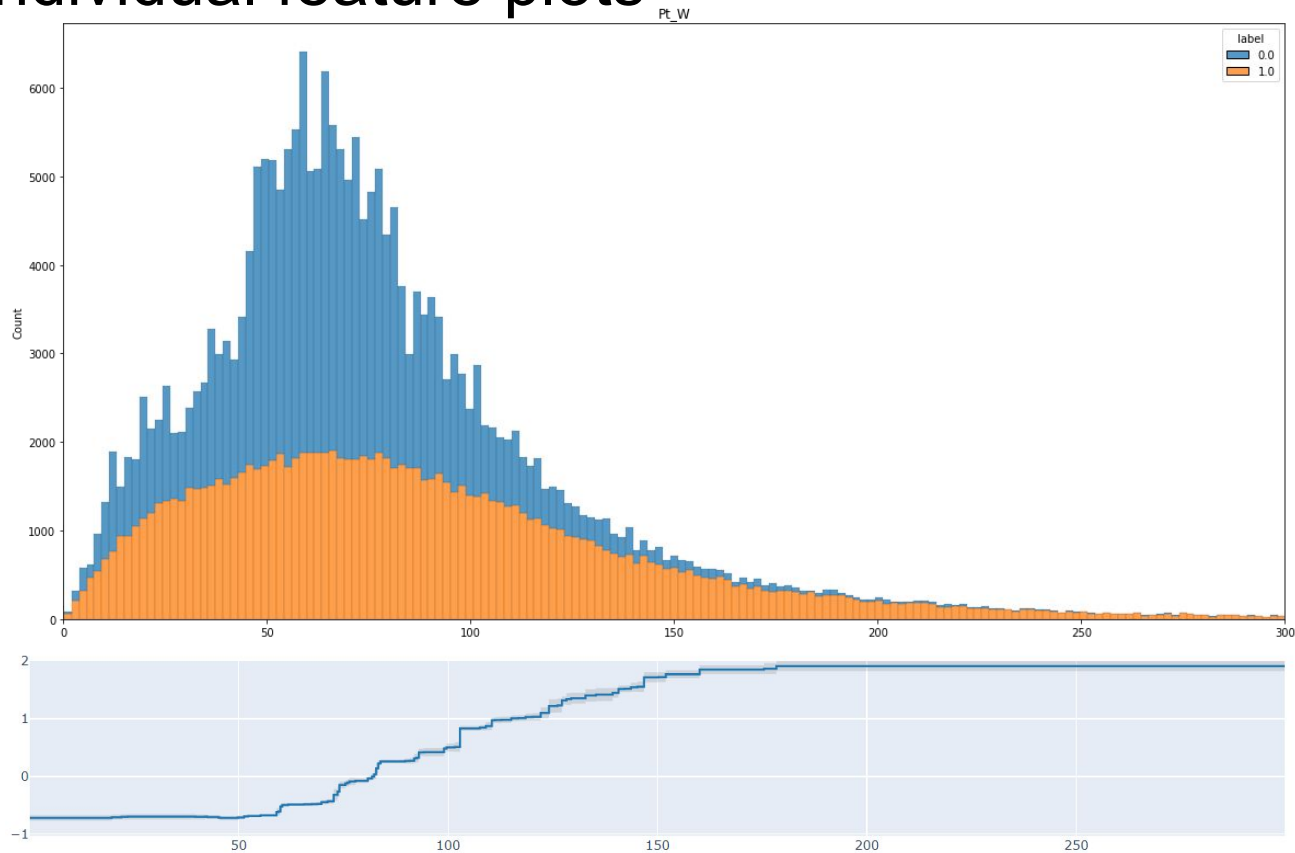


A standard feature importance plot to show how much each learned feature function affects the prediction.

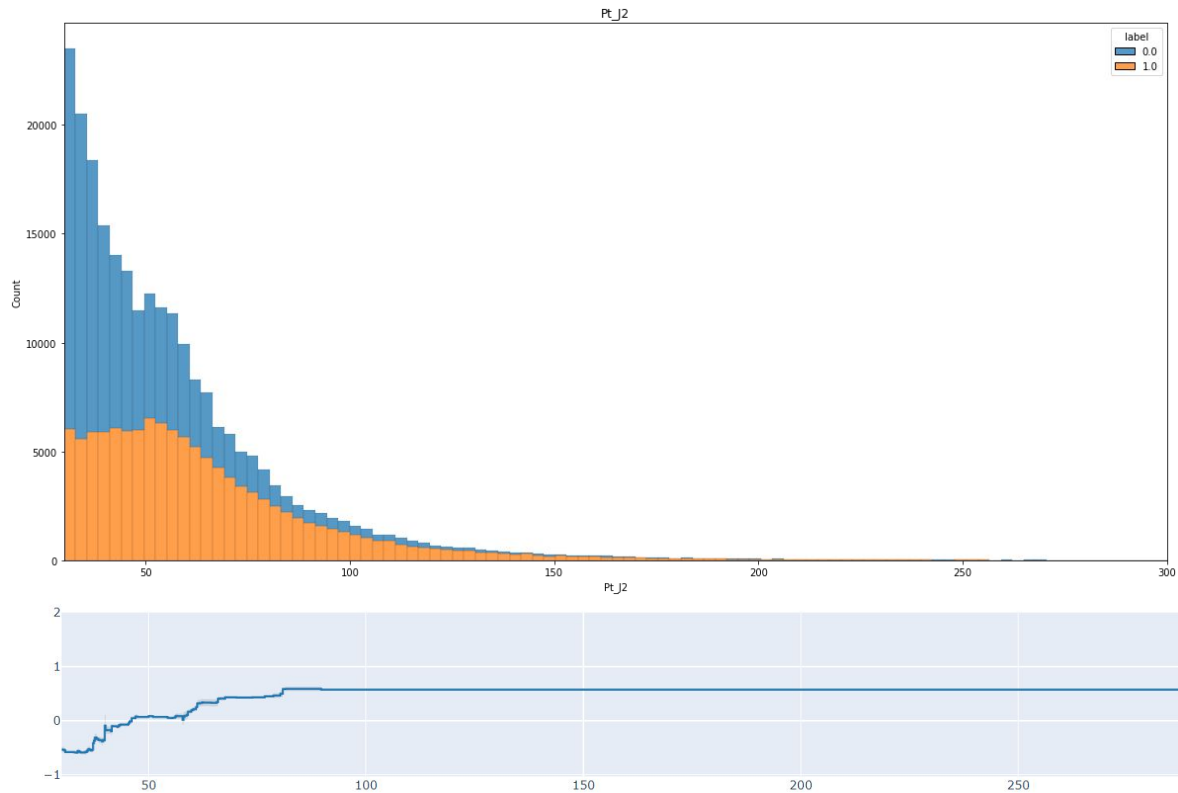
EBM: individual feature plots



EBM: individual feature plots

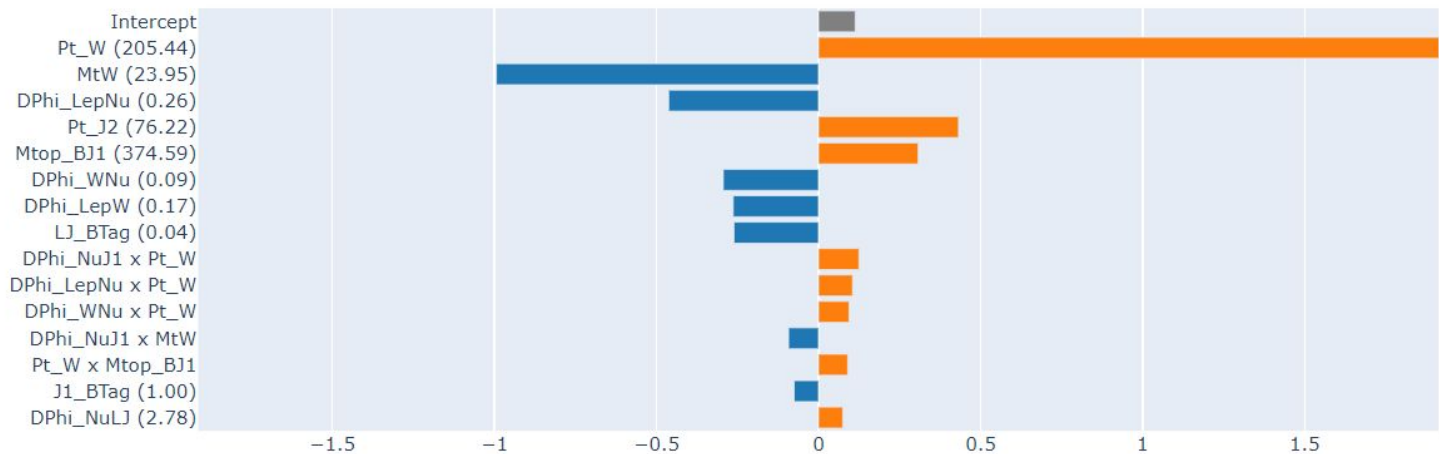


EBM: individual feature plots



EBM: local explanations

Predicted (1.0): 0.690 | Actual (1.0): 0.690



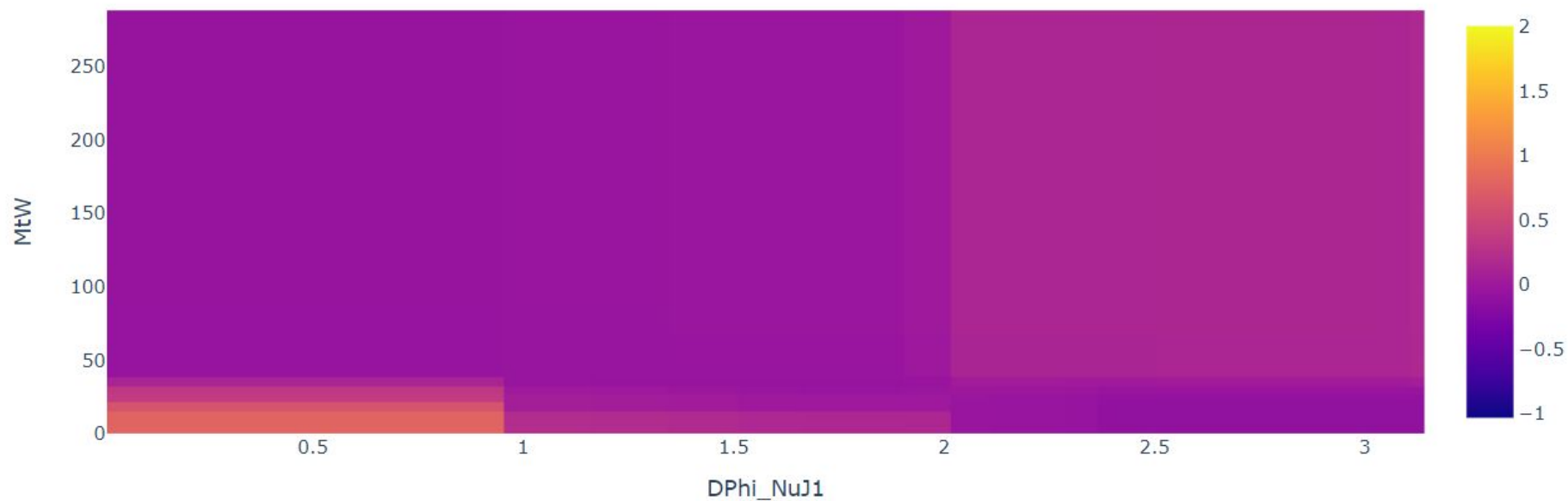
To obtain local explanations one only needs to plug feature values in already learned functions.

Thank you for your attention!

Backup

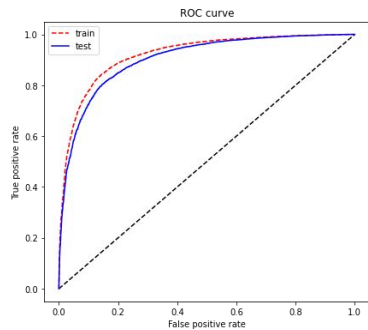
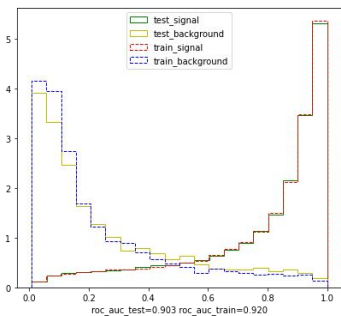
Pairwise interactions in EBM

DPhi_NuJ1 x MtW



EBM vs DNN performance comparison

EBMClassifier



EBM with default hyperparameters

DNN with one hidden layer