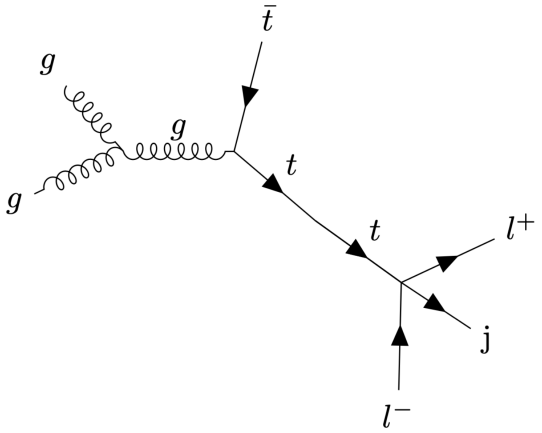
Overview of the Top FC Analysis

Meisam Ghasemi Bostanabad

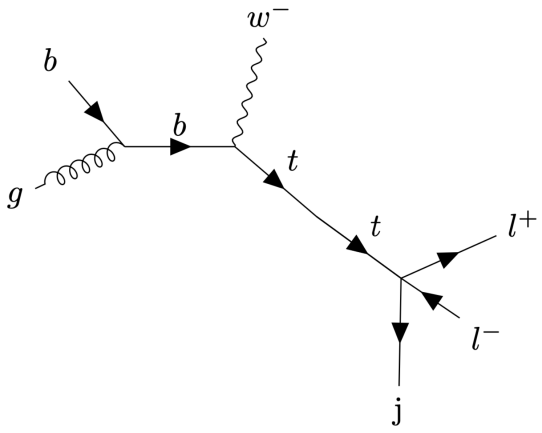
Analysis meeting
2023-4-22

Flavor Changing in Top sector

In this analysis we looking for FC ($t \to u$ or $t \to c$) in top sector as the heaviest quark which may be an indicator of new flavor physics.



ttbar



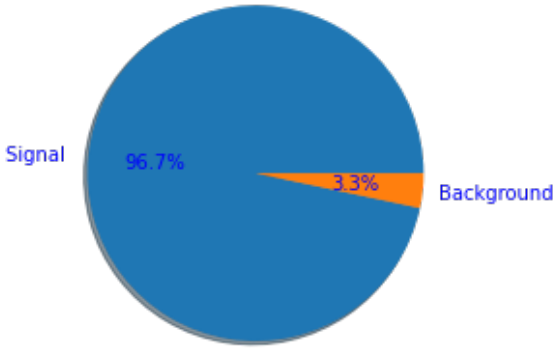
tW (3M events being generated)

- Starting with **ttbar**, targeting **final states** with three leptons (a pair of OP) and a b-tagged jet (one of the tops decays leptonically via w $\to l\, \nu_l$)
- There are **at least** two jets – other jets might come from showering
- The leading potential backgrounds are tZ, $t\bar{t}W$, $t\bar{t}Z$, $t\bar{t}t\bar{t}$, WZ, ZZ, $t\bar{t}$

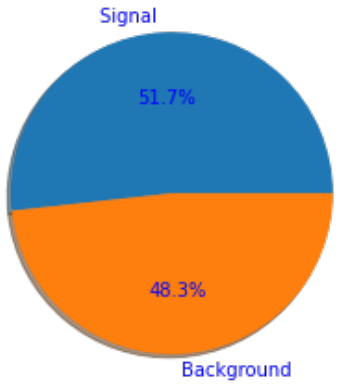
Signal-Background ratio

- After applying preselections, there are too many more signal than backgrounds! This make the dataset skewed and ML performance would be too weak.
- Still having contact with ML experts to deal with this kind of skewed data. In the first try, **3k signal events** are chosen randomly to have 51%-48% dataset.
- To make it even better, more backgrounds are generated to be able to include more signals (30k) in the train-test batches.

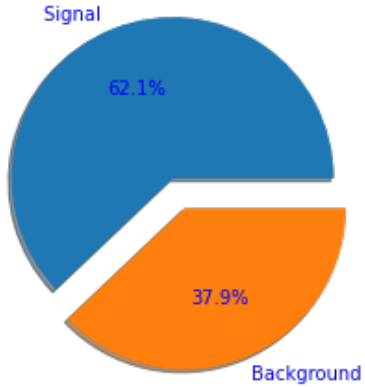
unbalanced data



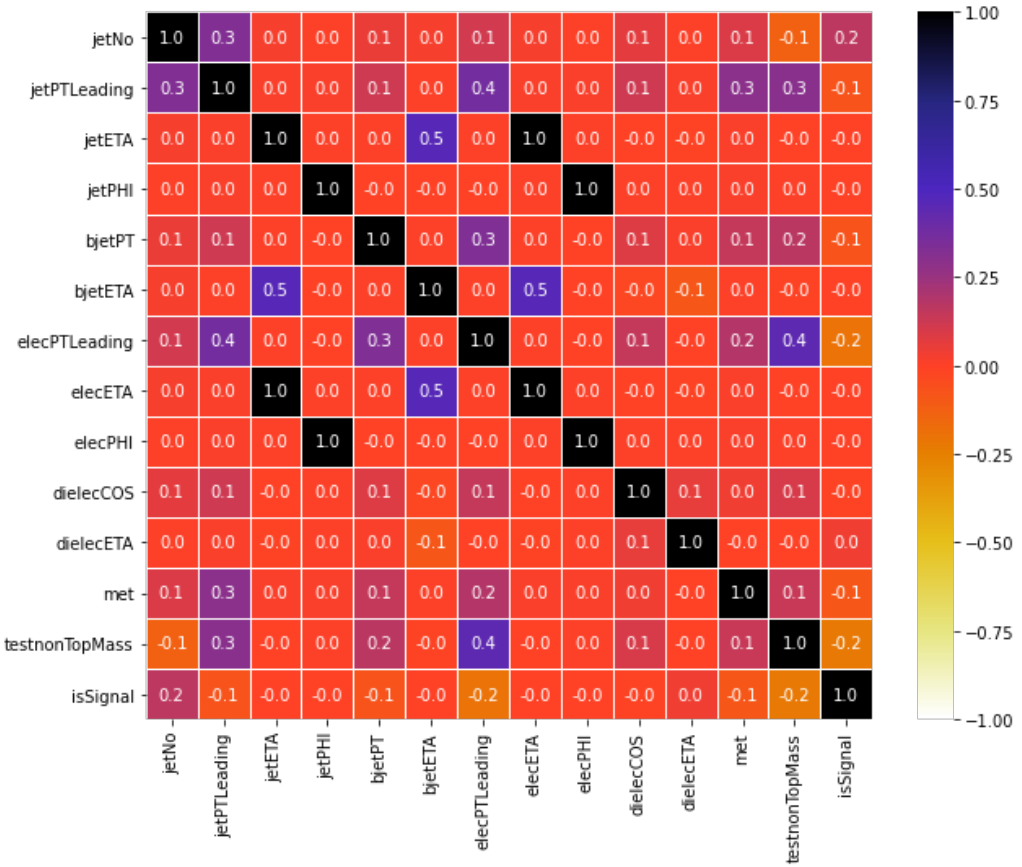
balanced data with 3k signal



balanced data with 30k signal



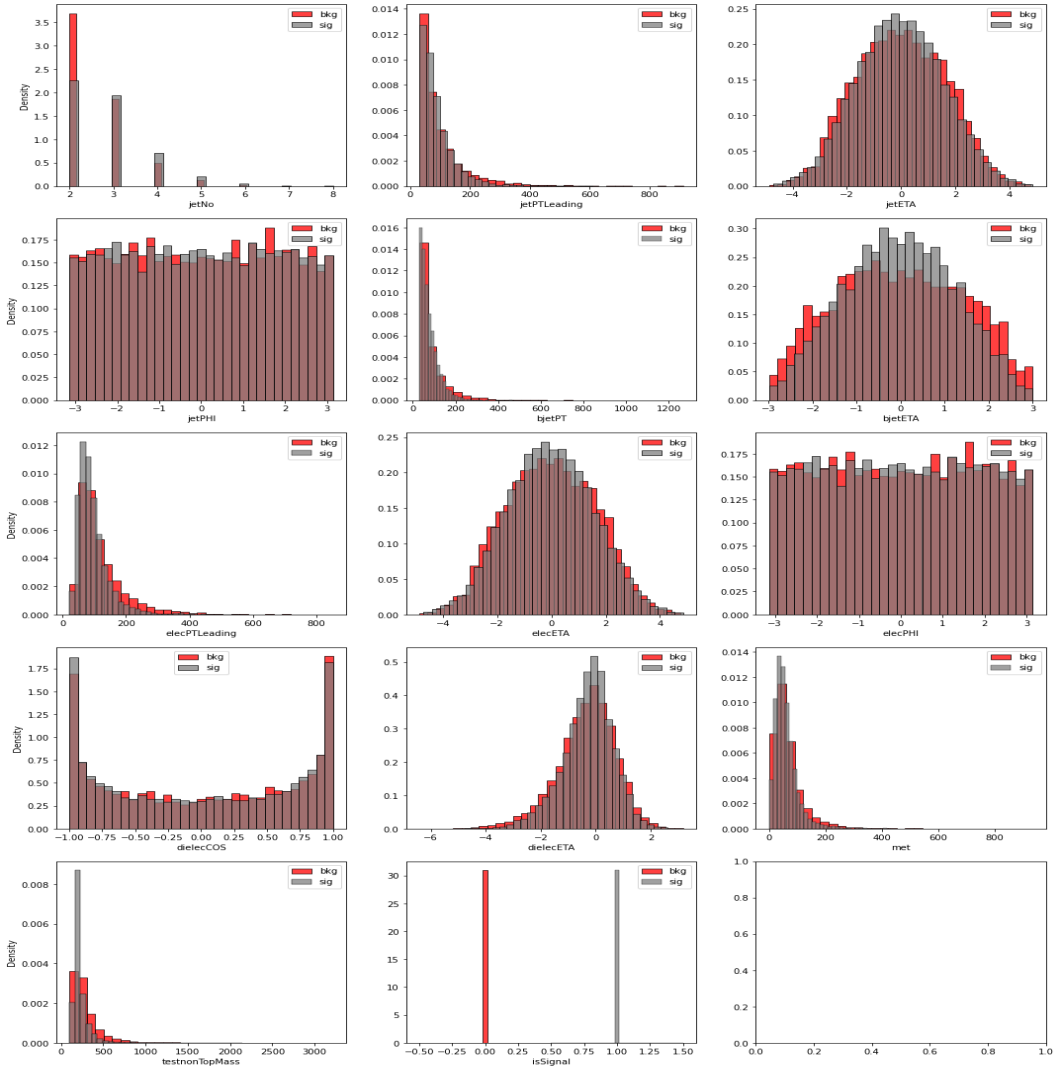
Correlations on the heatmap



IsSignal is mostly (negatively) correlated to non-SM top mass

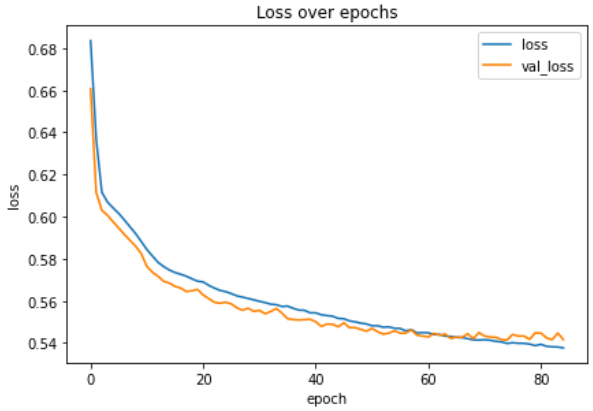
JetNo is (positively) correlated – means signal prones to more jets

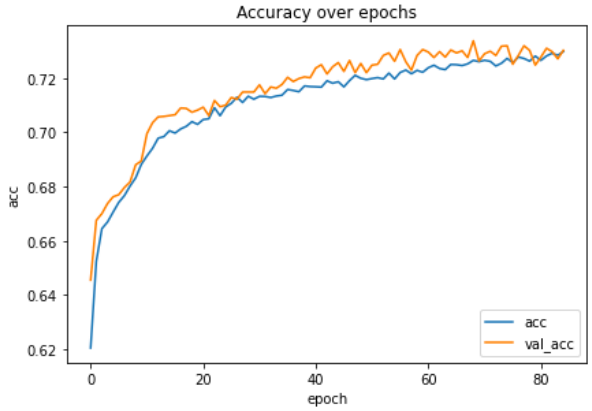
Signal and background distributions

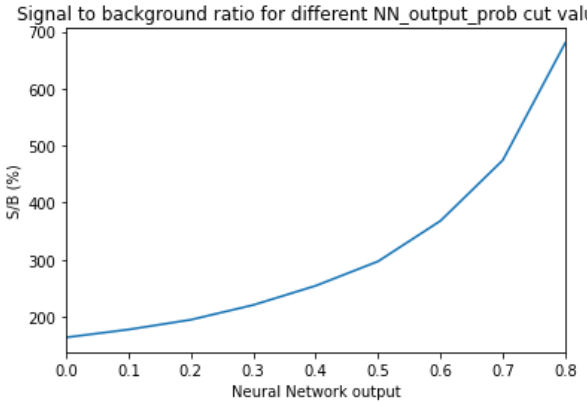


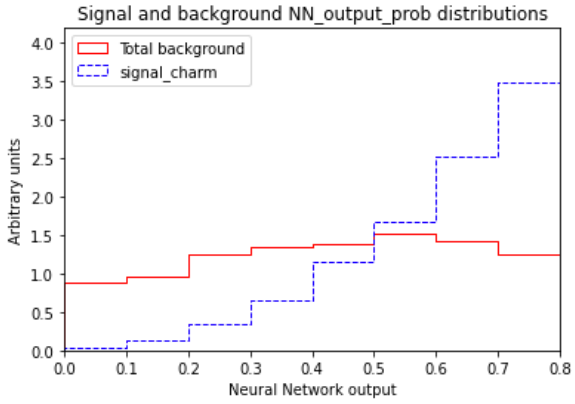
Simple NN performance

Structure details in the backup



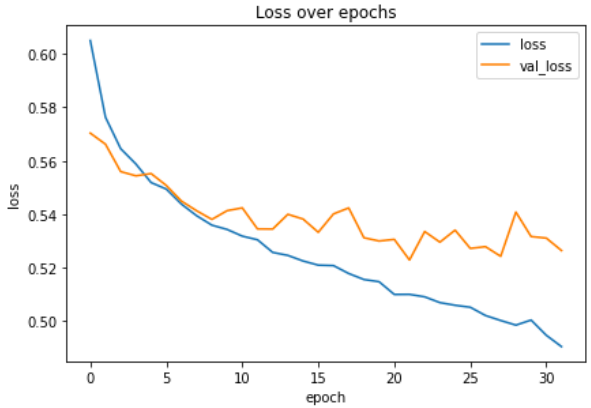


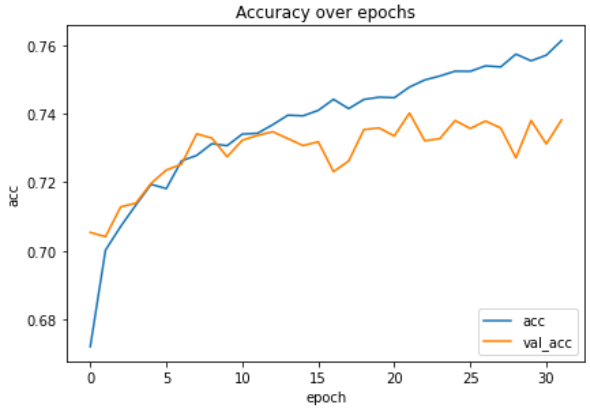


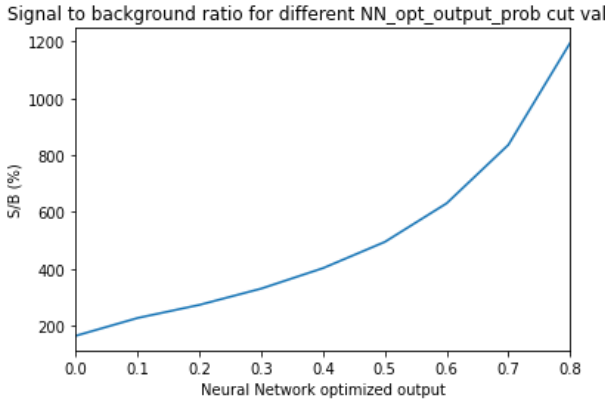


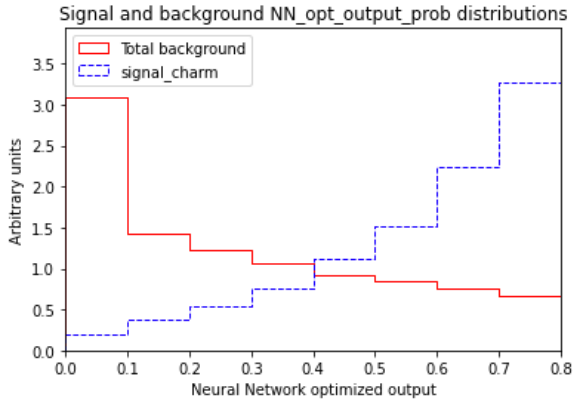
Optimized NN performance

Structure details in the backup

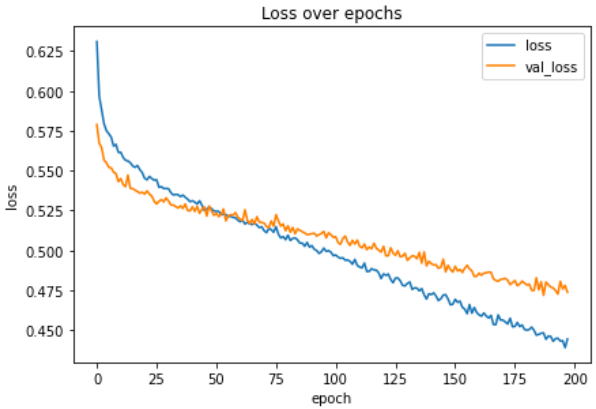


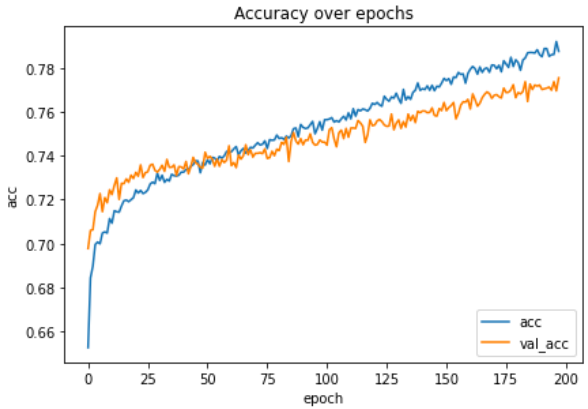


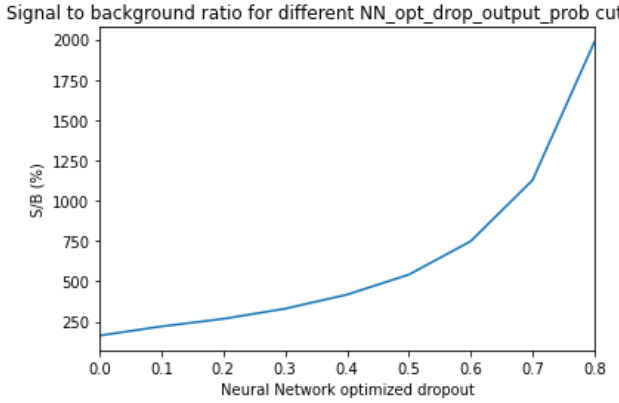


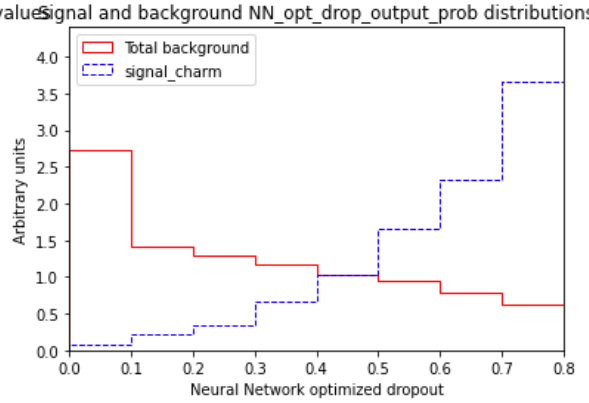


Optimized NN (dropout layer) performance

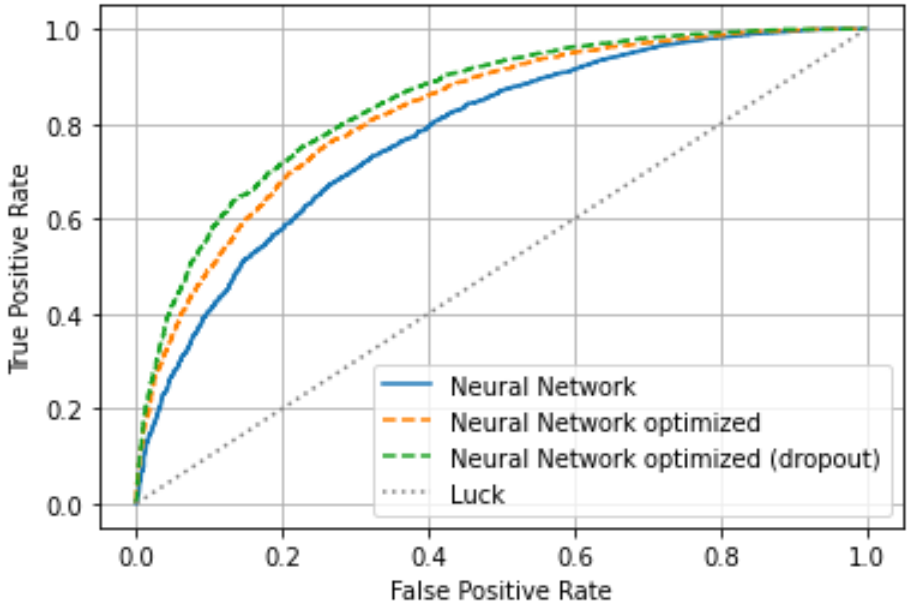






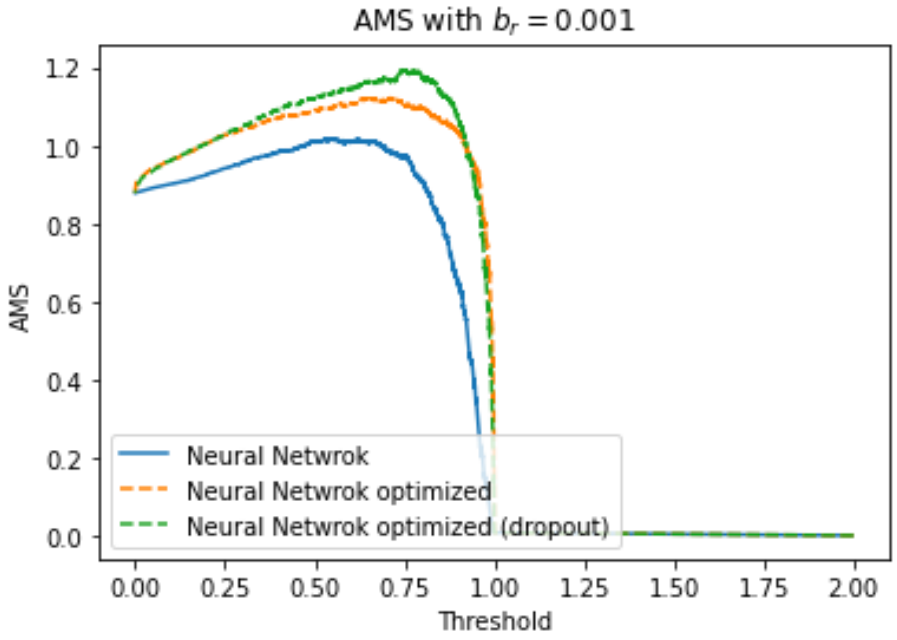


ROC curve



- Receiver Operating Characteristic curve is a graphical representation of the performance of a binary classification model. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at different classification thresholds. A perfect classifier would have a ROC curve that passes through the top-left corner, indicating a high TPR and low FPR.

AMS curve

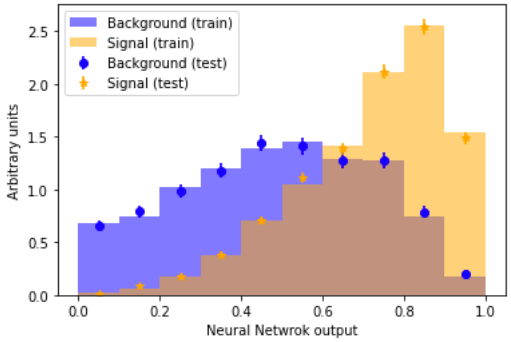


$$\text{AMS} = \sqrt{2\left((TPR + FPR + b_r)\ln\left(1 + \frac{TPR}{FPR + b_r}\right) - TPR\right)}$$

- In classifying signal or background events, the primary goal is optimizing the discovery region for statistical significance. As discussed here, this metric is the approximate median significance (AMS). This metric is used in Higgs Kaggle competition.

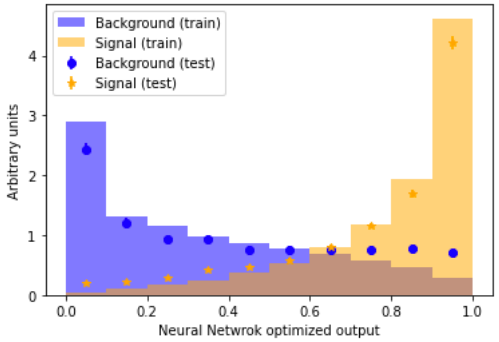
Overfit checking

Comparing a machine learning model's output distribution for the training and testing set is a popular way in High Energy Physics to check for overfitting. The plots show the machine learning model's decision function for each class, as well as overlaying it with the decision function in the training set.



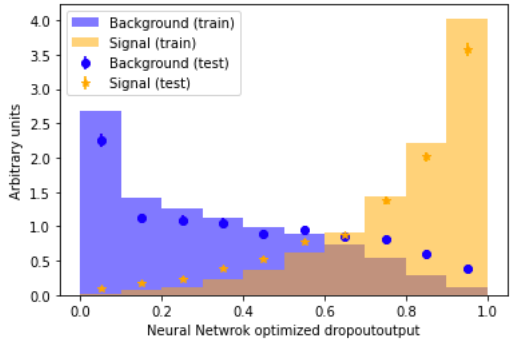
simple NN

```
Test acc score: 0.7314498602918348
Train acc score: 0.7290825830487426
```



optimized NN

```
Test acc score: 0.7638932008692952
Train acc score: 0.8361921763427507
```



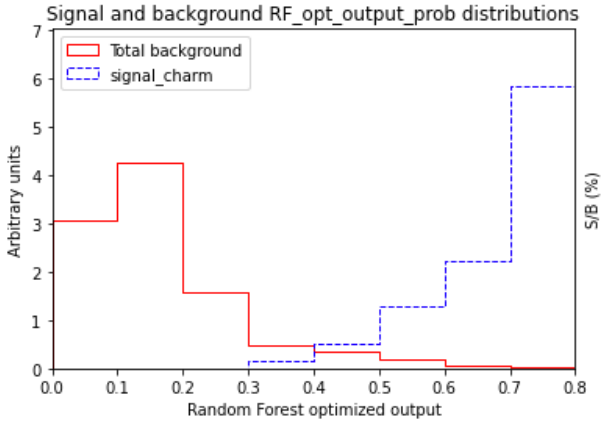
optimized NN with dropout

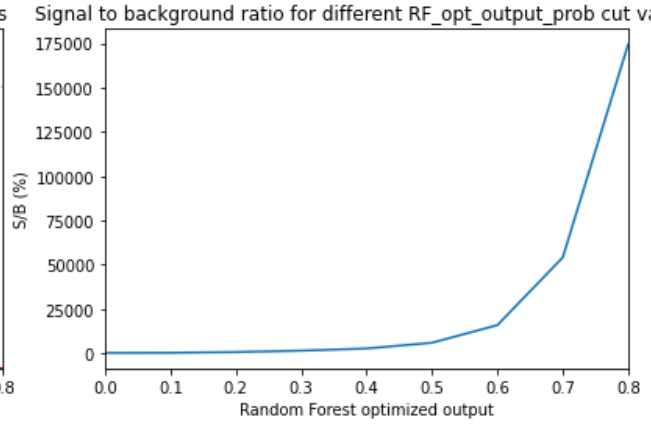
```
Test acc score: 0.7786401738590499
Train acc score: 0.8542378143433716
```

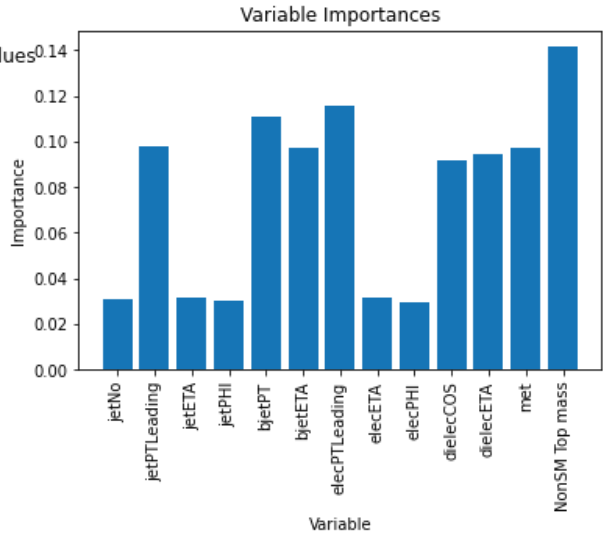
Optimized RF performance and importance

- Signal and background distributions based on the RF probabilities (left) and Signal/background ratio in different RF output bins (middle). RF threshold with the max S/B could be a good option to define signal region.

- Feature importance in model training is on the right. Non-SM top mass is the most important variable in training!

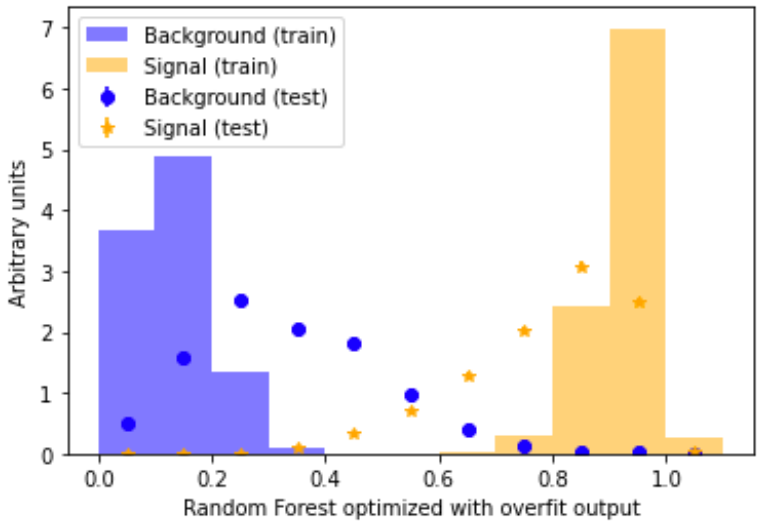






Overfit checking

Comparing a machine learning model's output distribution for the training and testing set is a popular way in High Energy Physics to check for overfitting. The plots show the machine learning model's decision function for each class, as well as overlaying it with the decision function in the training set.



optimized RF

Summary & ongoing

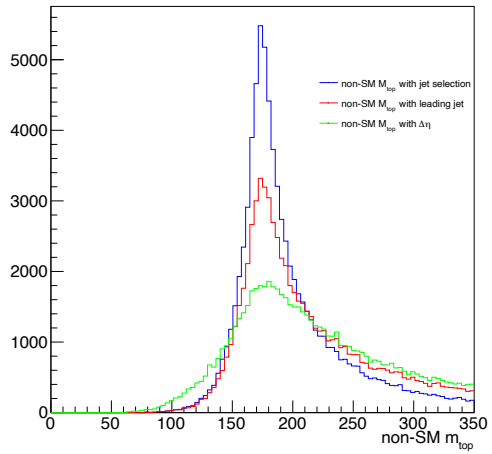
- Several ML classifiers are trained using subset of data and important analysis features.
- After model (NN and RF) optimization, both have good performance. NN gives higher accuracy score, TPR and lower FPR plus no overfit is observed. Overfit observed in RF.
- RNN and BDT show weak performance (notebook)!
- NN models (simple and optimized) are applied to the whole dataset and the NN weights are saved in a separate tree.
- Analysis tree production with important variables and plotting framework are done (tree framework, plotter framework, ML weights).
- As the next step, we can define signal regions based on significance and AMS plots and using pyhf for statistical analysis.
- IPM affiliation for publications.
- Your feedback is welcome and appreciated.

Backup

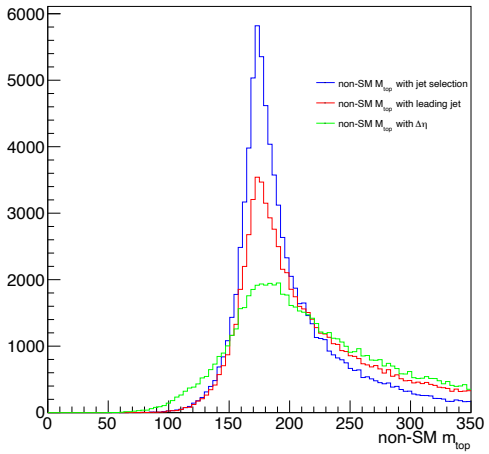
Non-SM top mass reconstruction

- Three algorithms used to reconstruct non-SM top mass:
  1. the min $\Delta\eta$ between electrons is used to select OP electrons and subsequently non-SM top mass reconstruction (green)
  2. the leading non-btagged jet and the 3 electrons are the inputs for $\min(|m_{llq} - m_{top}|)$ to choose the best selection for OS electrons (red)
  3. Loop over all the electrons and jets to get $\min(|m_{llq} - m_{top}|)$. The combination will be used to indicate OS leptons (blue)

Signal charm



Signal up



Signal and background generation

- Signal and background events are generated with MG5 (for ME) + PYTHIA (for PS and HAD) + Delphes (for HLLHC CMS card detection). almost 3M events for both charm and up signals and 2M events for each background.

- Weights look fine (<1) for all signal and background events. Extra 15M $t\bar{t}$ events are being generated to have better ML training (the third lepton in $t\bar{t}$ should be fake btw).

- Here is the weight summary for all analysis processes:

  weights = {'ttbarZ': 0.00431, 'tZ': 0.00375, 'tttt': 2.79520e-05, 'ZZ': 0.67125, 'ttbar': 0.9485, 'ttbarW': 0.00015, 'WZ': 0.13575,'signal_charm': 0.01376, 'signal_up': 0.01376}

- The preselections applied:
  1. exactly 3 leptons (for now just electrons) with one pair of OS
  2. at least 2-jets with one b-tagged jet
  3. minimum P_T cut and η cut to pass di-lepton trigger

First try, simple NN

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input (InputLayer)          [(None, 13)]              0

 hidden1 (Dense)             (None, 20)                280

 hidden2 (Dense)             (None, 20)                420

 output (Dense)              (None, 1)                 21

=================================================================
Total params: 721
Trainable params: 721
Non-trainable params: 0
```

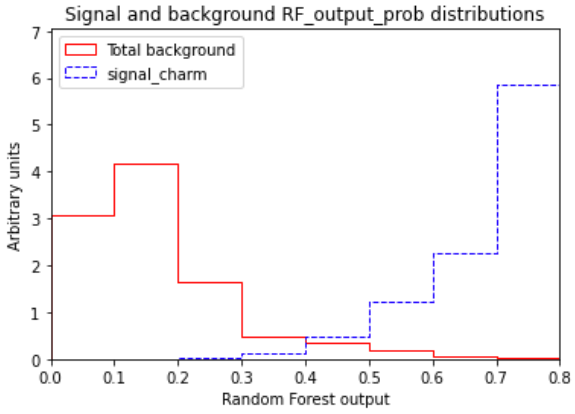
Keras-tuner to tune Hyperparameters

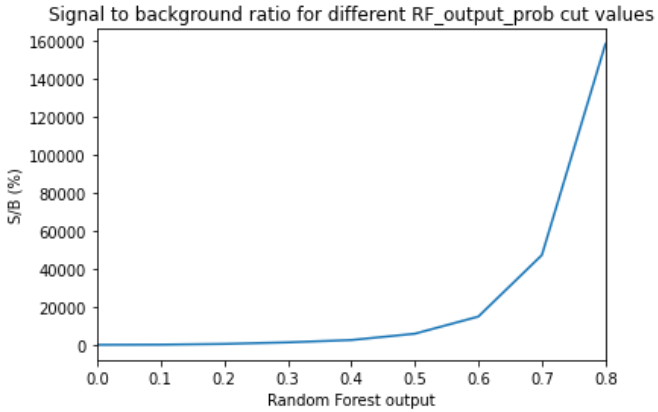
```
RandomizedSearchCV(cv=5,
                   estimator=Pipeline(steps=[('scaler', StandardScaler()),
                                             ('clf',
                                              <keras.wrappers.scikit_learn.KerasClassifier object at 0x7fe642490880>)]),
                   n_iter=5,
                   param_distributions={'clf__activation': ['selu', 'relu',
                                                            'tanh'],
                                        'clf__batch_size': [64, 128, 256, 512],
                                        'clf__dropout_rate': [0.1, 0.01],
                                        'clf__epochs': [5, 10, 15, 50, 100,
                                                        200],
                                        'clf__k_initializer': ['lecun_normal',
                                                               'normal'],
                                        'clf__network_layers': [(32, 32),
                                                                (64, 64),
                                                                (128, 128,
                                                                 128)],
                                        'clf__optimizer': ['Nadam', 'Adam',
                                                           'SGD'],
                                        'clf__verbose': [0]},
                   scoring='accuracy')
```

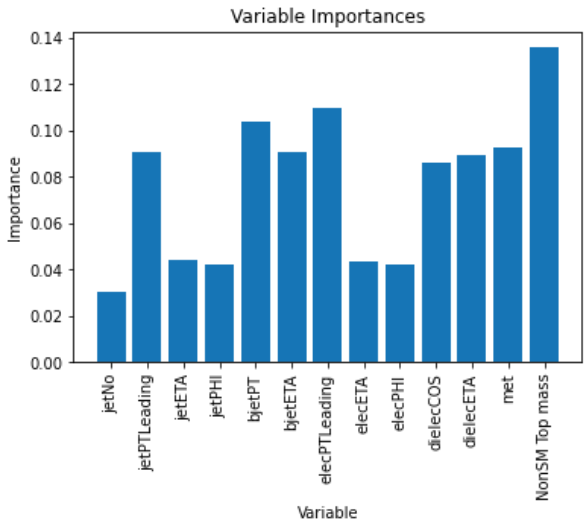
RF performance and importance

- Signal and background distributions based on the RF probabilities (left) and Signal/background ratio in different RF output bins (middle). RF threshold with the max S/B could be a good option to define signal region.

- Feature importance in model training is on the right. Non-SM top mass is the most important variable in training!







Random search cross validation tunning

1. n_estimators = number of trees in the foreset
2. max_features = max number of features considered for splitting a node
3. max_depth = max number of levels in each decision tree
4. min_samples_split = min number of data points placed in a node before the node is split
5. min_samples_leaf = min number of data points allowed in a leaf node
6. bootstrap = method for sampling data points (with or without replacement)

```
{'bootstrap': False,
 'ccp_alpha': 0.0,
 'criterion': 'mse',
 'max_depth': 50,
 'max_features': 'sqrt',
 'max_leaf_nodes': None,
 'max_samples': None,
 'min_impurity_decrease': 0.0,
 'min_impurity_split': None,
 'min_samples_leaf': 4,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators': 800,
 'n_jobs': None,
 'oob_score': False,
 'random_state': 42,
 'verbose': 0,
 'warm_start': False}
```