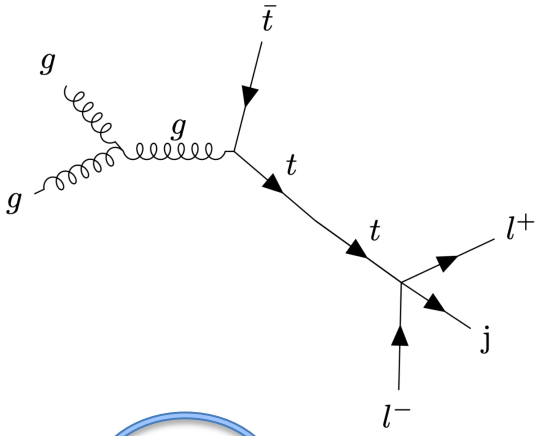
Overview of the Top FC Analysis

Meisam Ghasemi Bostanabad

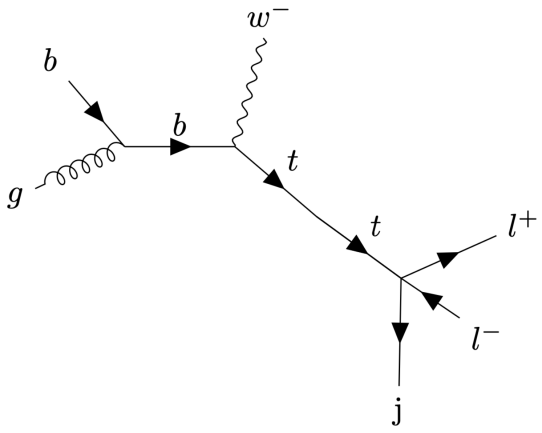
Analysis meeting
2023-4-2

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

- 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
- Presence of several charged leptons allows an efficient lepton trigger
- 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 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:

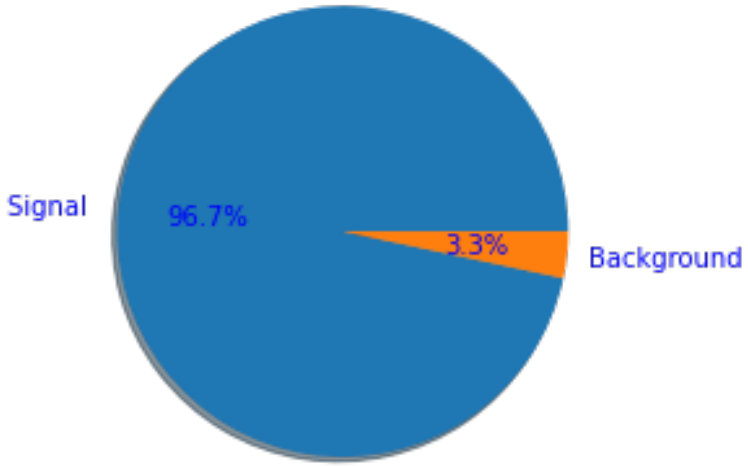
  weights = {'ttbarZ': 0.00431, 'tZ': 0.00375, 'tttt': 2.79520e-05, 'ZZ': 0.67125, 'ttbar': 0.9485, 'ttbarW': 0.00015, 'WZ': 0.13575,'signal_charm': 0.01376, 'signal_up': 0.01376}

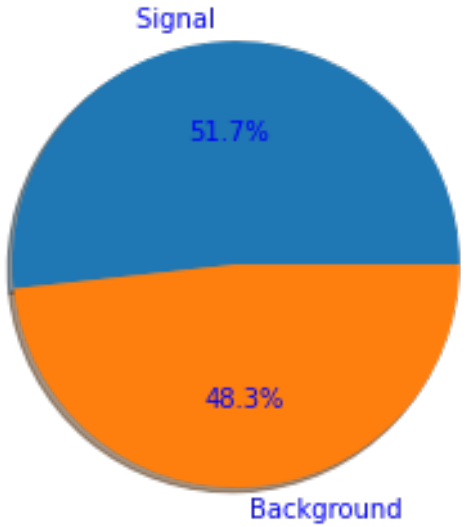
- The preselections applied:
  1. exactly 3 leptons (for now just electrons) with one pair of OS
  2. at least 2-jets with one b-tagged jet
  3. minimum P_T cut and η cut to pass di-lepton trigger

Signal-Background ratio

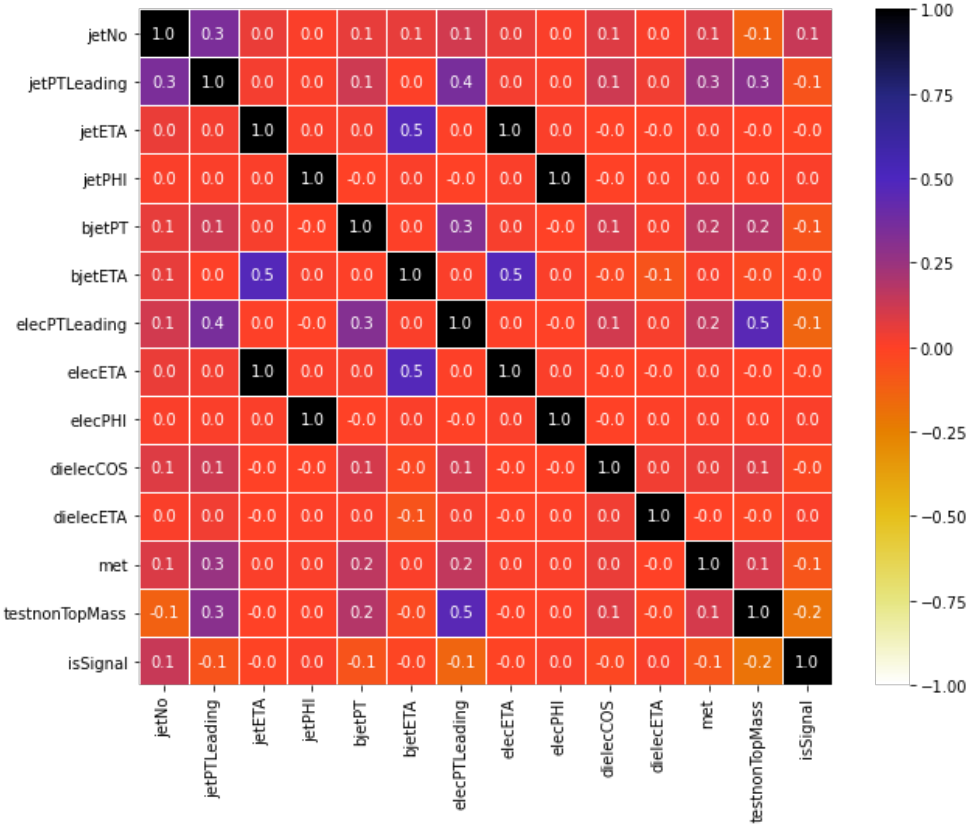
- After applying preselections, there are too many more signal than backgrounds! This make the dataset skewed and ML performance would be too weak. In each batch the model literally sees signals and no backgrounds!

- Still having contact with ML experts to deal with this kind of skewed data. In the meantime, 3.5k signal events are chosen randomly to have 51%-48% dataset.



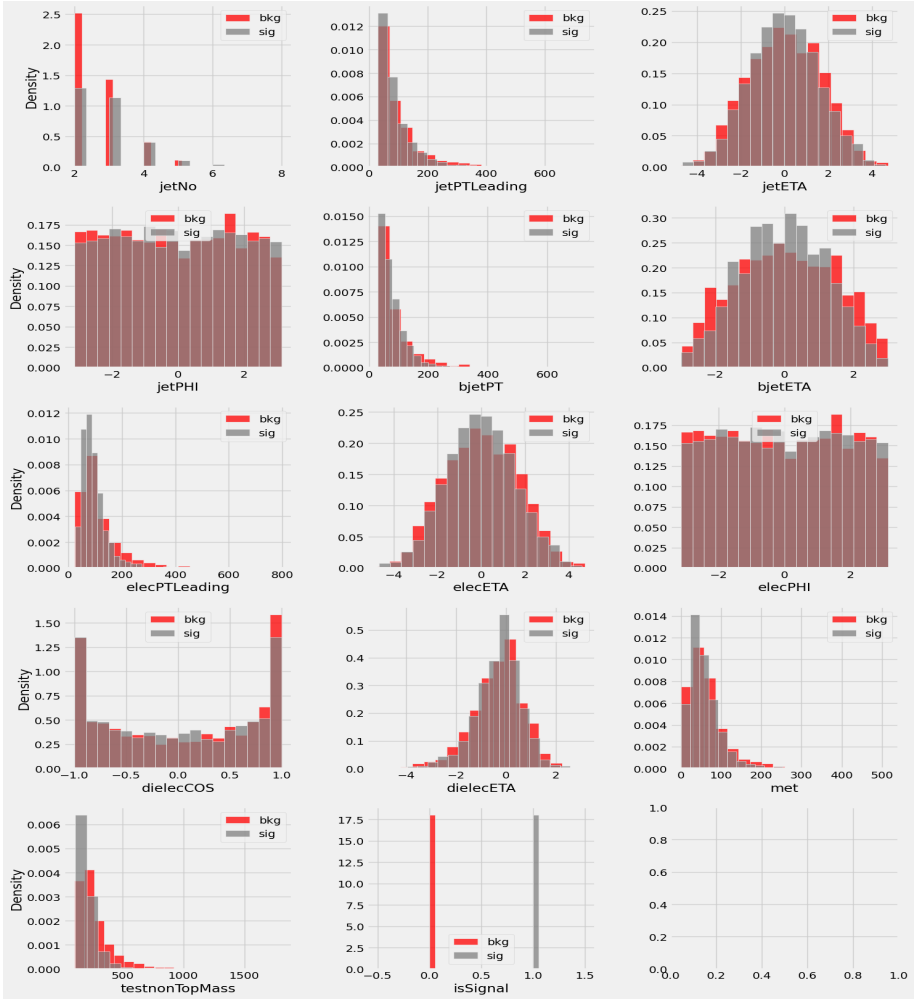


Correlations on the heatmap

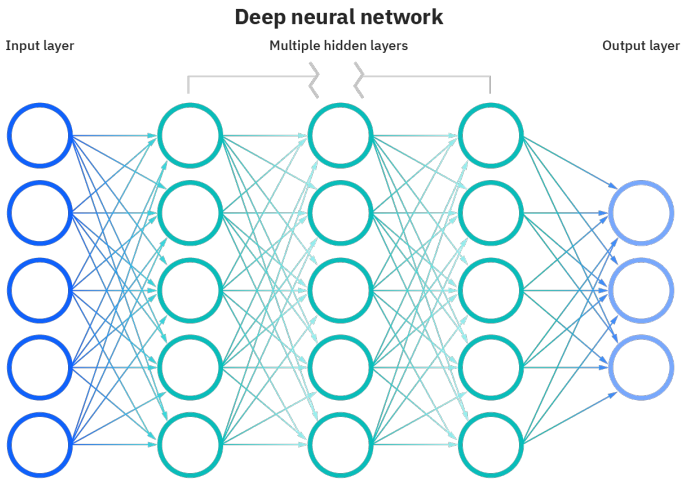


IsSignal is mostly (negatively) correlated to non-SM top mass!

Signal and background distributions



Neural Network

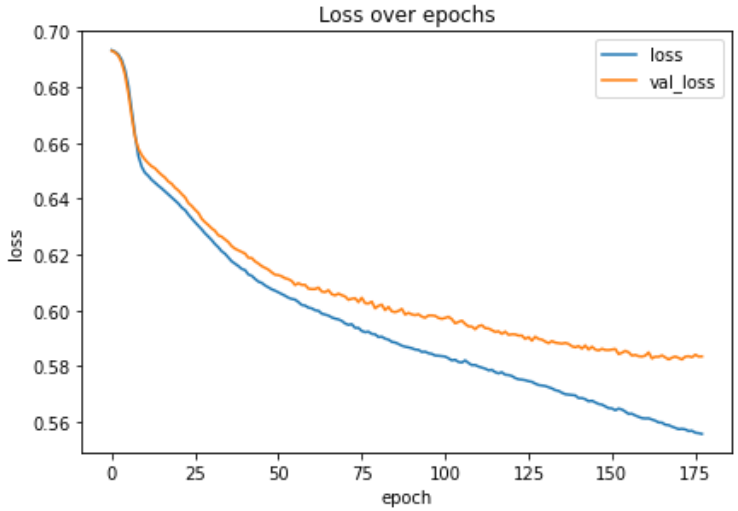


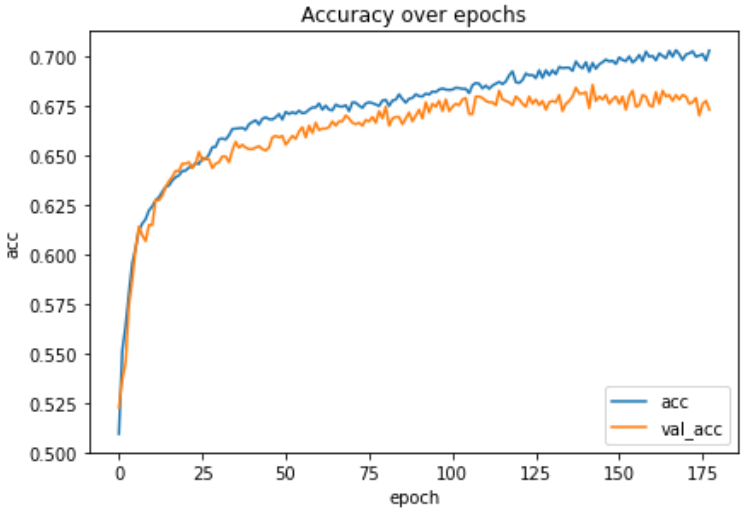
- This neural network model has 5 input nodes, 3 hidden layers with 5 nodes respectively, and 3 output nodes. Each node in the hidden layers uses an activation function to transform the weighted sum of inputs from the previous layer. The weights are learned during training using backpropagation, a method for updating the weights based on the error between the predicted output and the actual output. The output nodes produce the final prediction of the model. Neural networks can be used for a wide range of tasks, including classification, regression, and generative modeling.

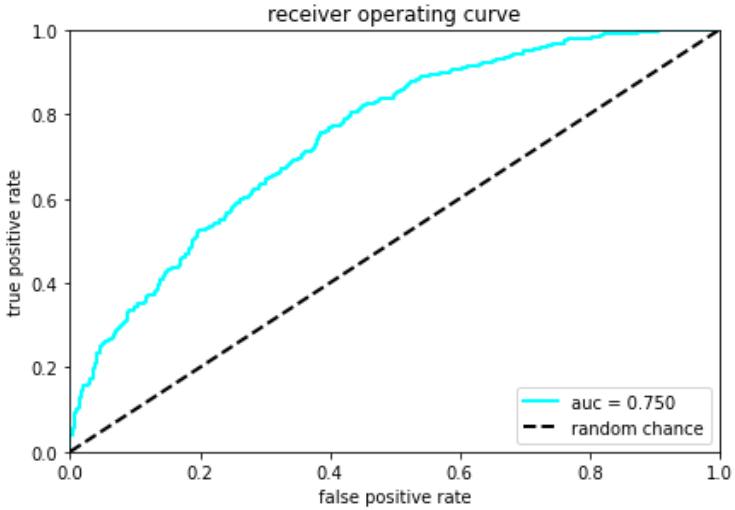
First try, simple NN

```
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
```

NN performance

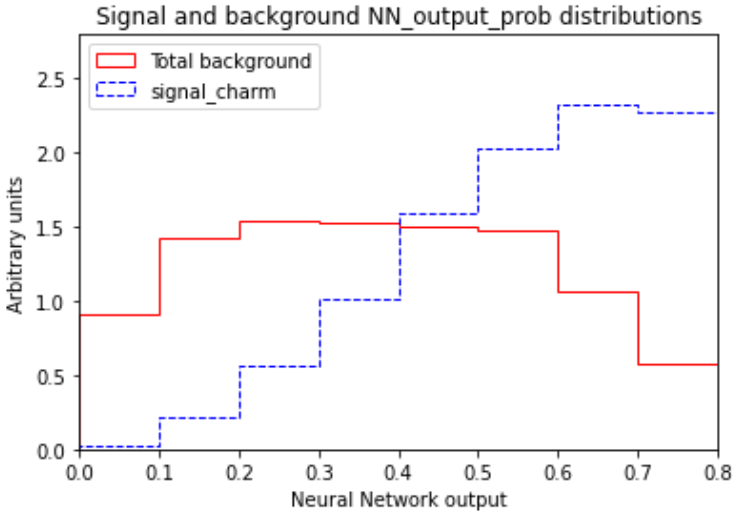


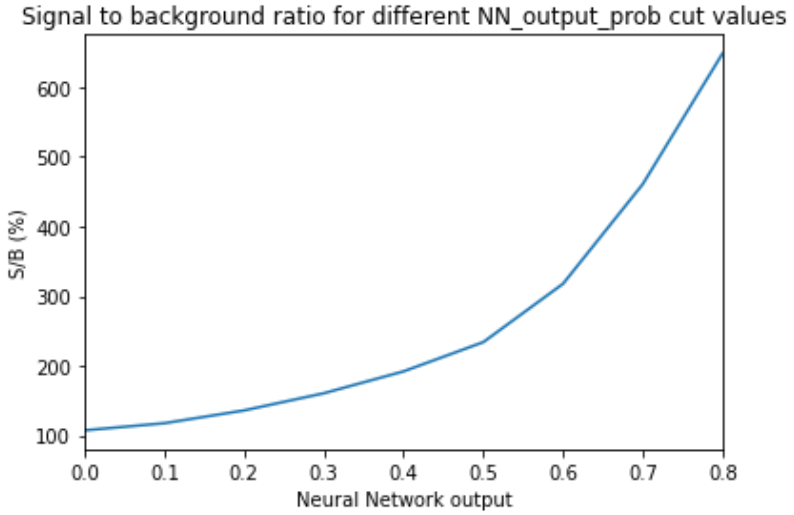




NN distributions and Significance

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

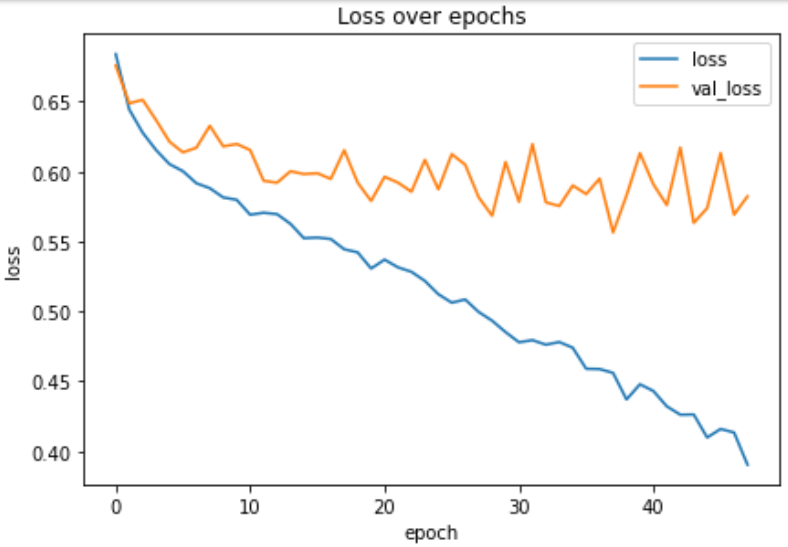


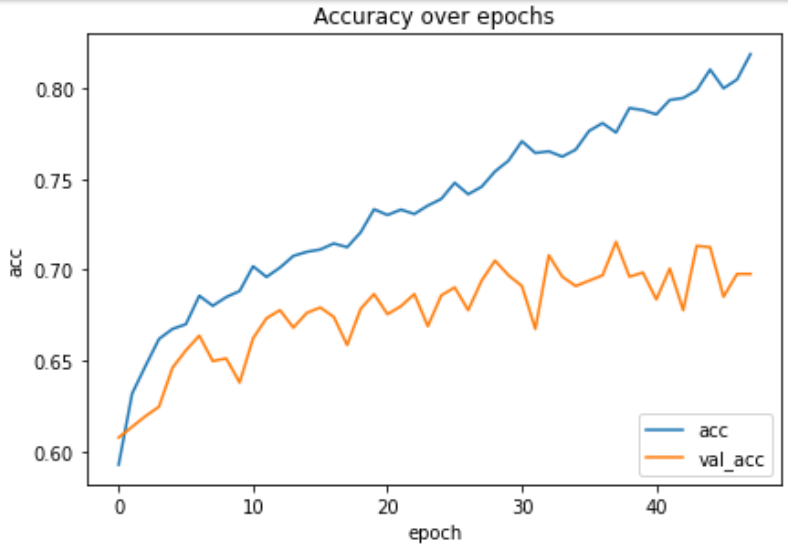


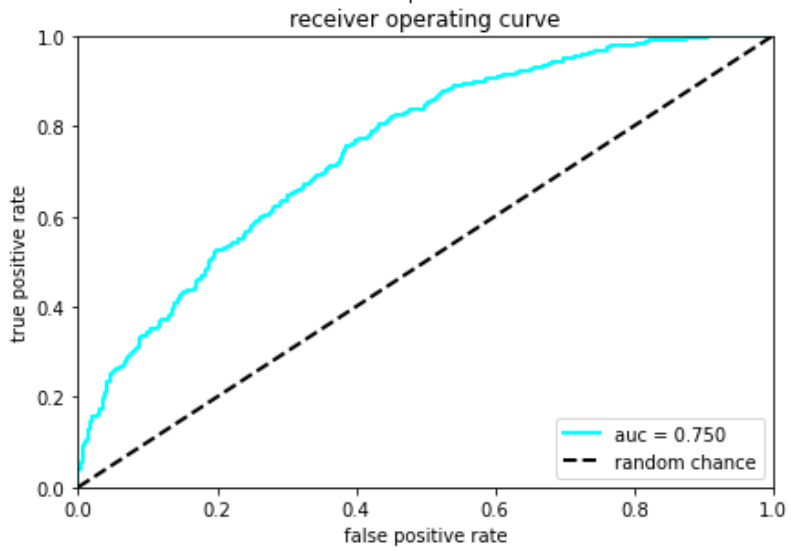
Keras-tuner to tune Hyperparameters

```
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')
```

Optimized NN performance

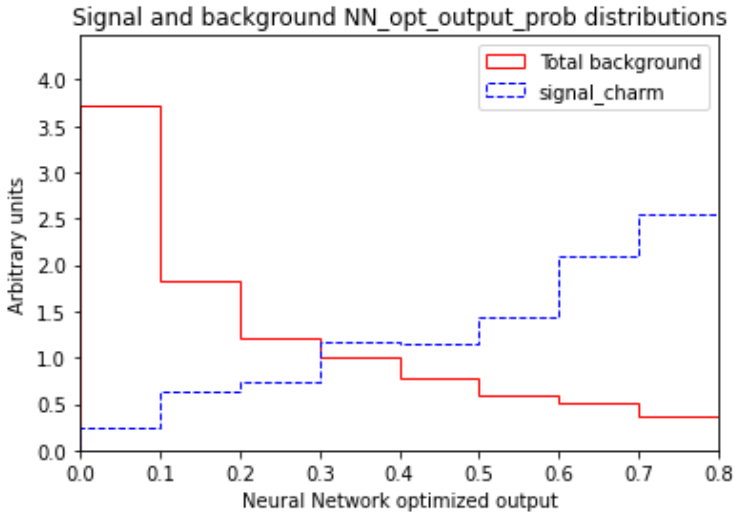


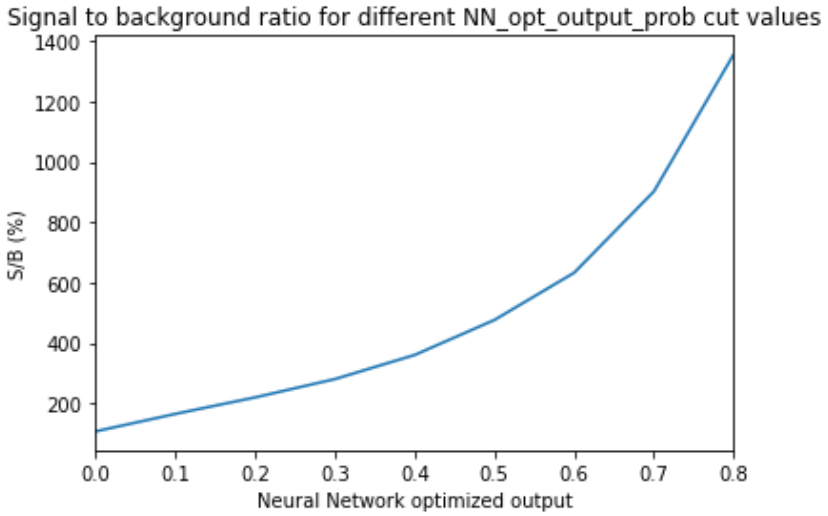




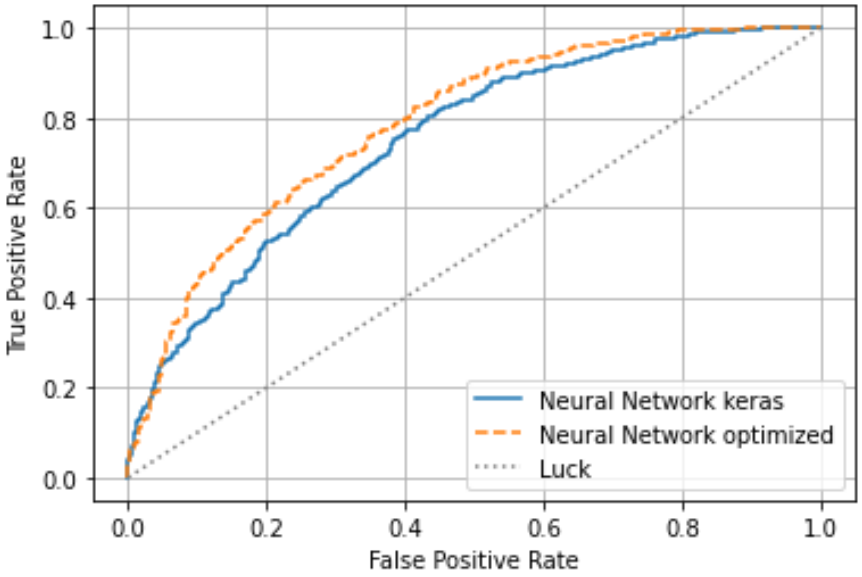
NN distributions and Significance

- Signal and background distributions based on the NN probabilities (left) is optimized and less backgrounds are classified as signal. Higher Signal/background ratio in different NN output (right) compared to simple NN.





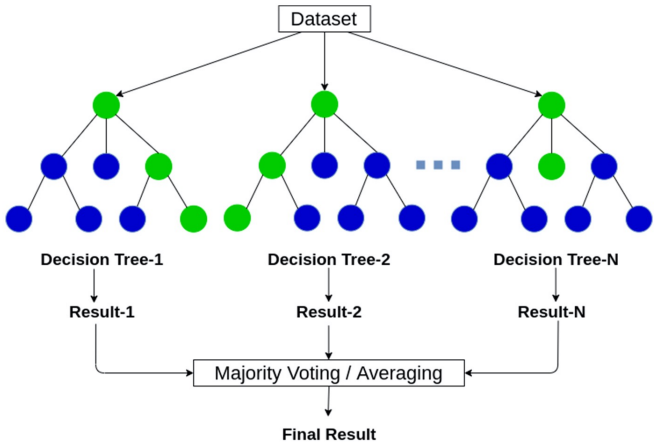
ROC curve



- 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.

Random Forest

Random Forest

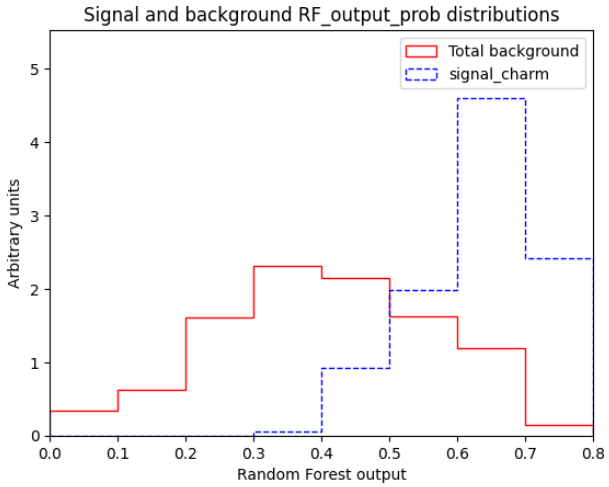


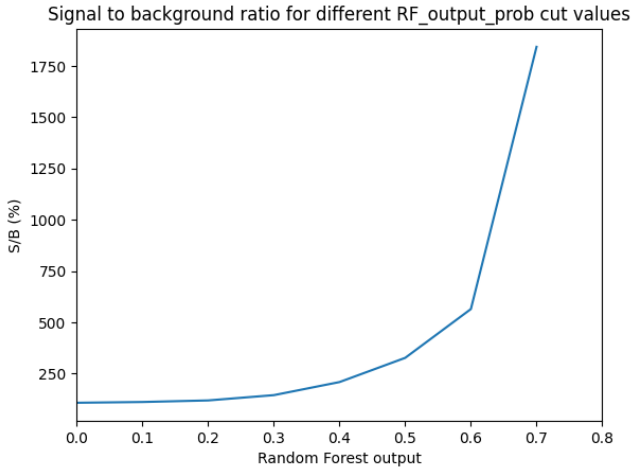
Random forests or **random decision forests** is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.

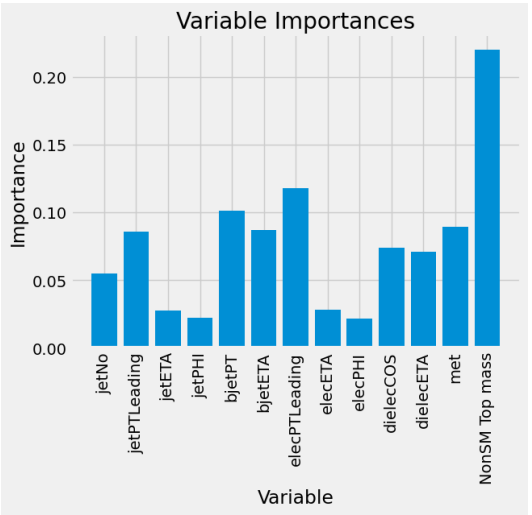
RF performance and importance

- 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

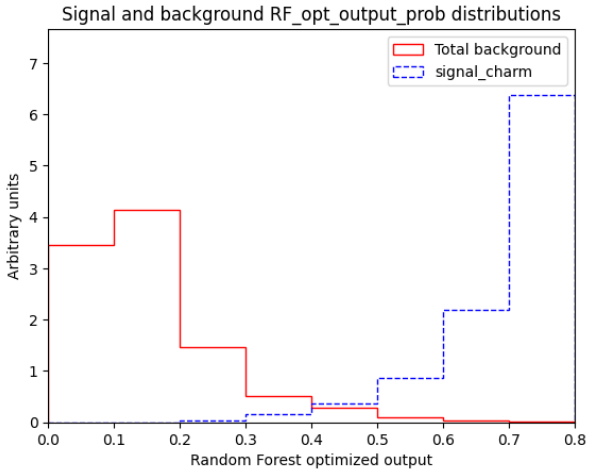
1. n_estimators = number of trees in the foreset
2. max_features = max number of features considered for splitting a node
3. max_depth = max number of levels in each decision tree
4. min_samples_split = min number of data points placed in a node before the node is split
5. min_samples_leaf = min number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

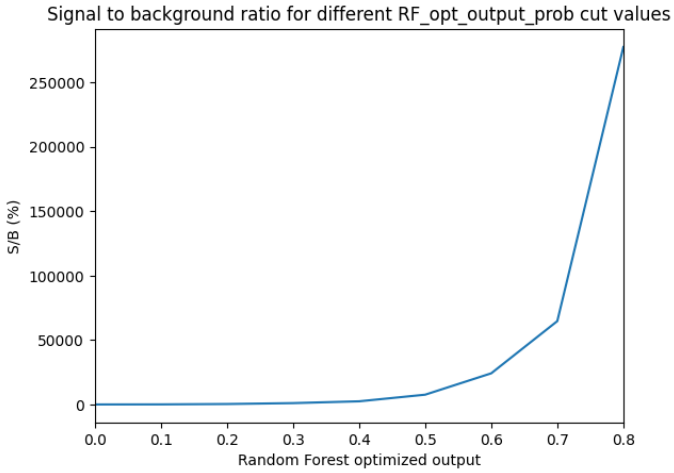
```
{'bootstrap': False,
 'ccp_alpha': 0.0,
 'criterion': 'mse',
 'max_depth': 50,
 'max_features': 'sqrt',
 'max_leaf_nodes': None,
 'max_samples': None,
 'min_impurity_decrease': 0.0,
 'min_impurity_split': None,
 'min_samples_leaf': 4,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators':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

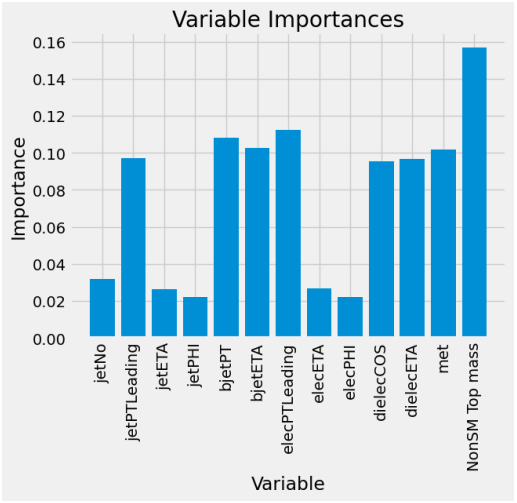
Optimized RF performance and importance

- 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. Huge

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

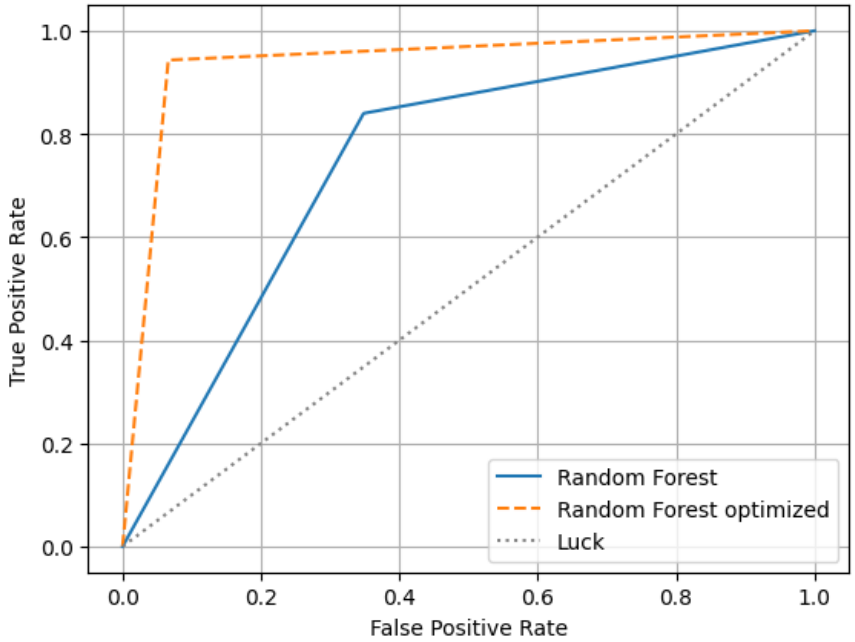




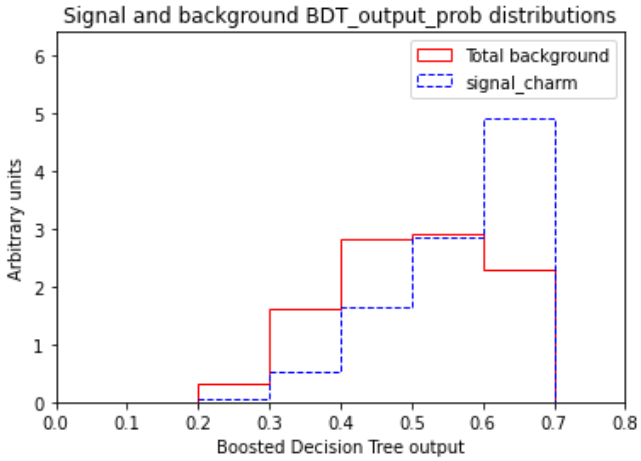


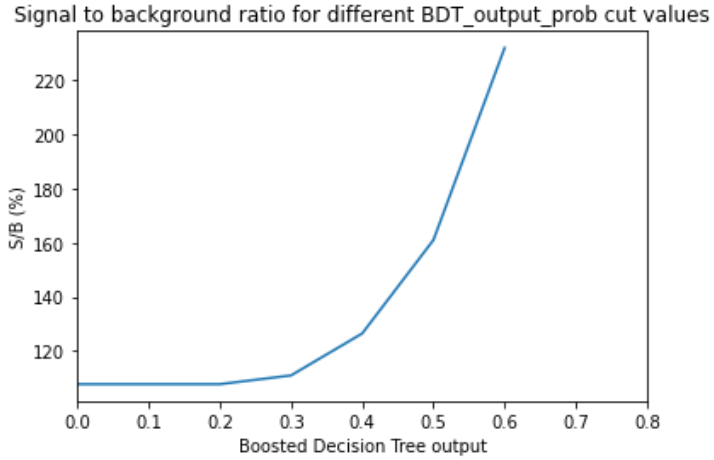
ROC curve for RF

- After RF optimization, derived higher TPR and lower FPR. Random search CV gives huge optimization in modelling and could be the best option for the rest of the analysis.



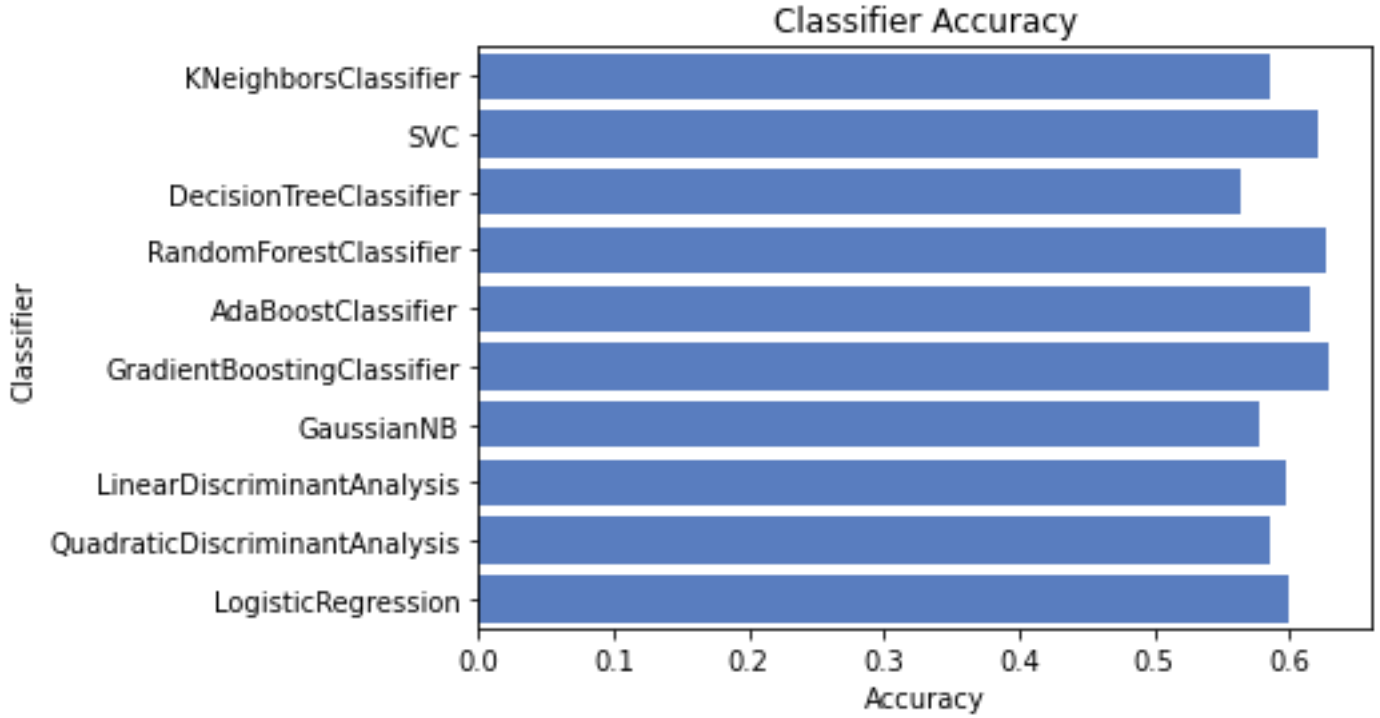
BDT performance





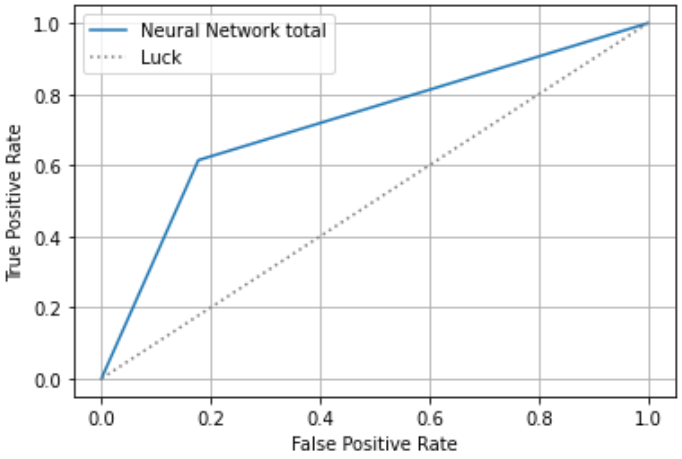

```
Time taken to fit BDT: 0.0s
AdaBoostClassifier(algorithm='SAMME',
                   base_estimator=DecisionTreeClassifier(max_depth=2),
                   learning_rate=0.5, n_estimators=12)
0.6183074265975821
Classification report for the test set
              precision    recall  f1-score   support

  background       0.67      0.49      0.57       293
      signal       0.59      0.74      0.66       286

    accuracy                           0.62       579
   macro avg       0.63      0.62      0.61       579
weighted avg       0.63      0.62      0.61       579

Classification report for the total set
              precision    recall  f1-score   support

  background       0.67      0.48      0.56      1393
      signal       0.62      0.78      0.69      1500

    accuracy                           0.63      2893
   macro avg       0.64      0.63      0.62      2893
weighted avg       0.64      0.63      0.63      2893
```

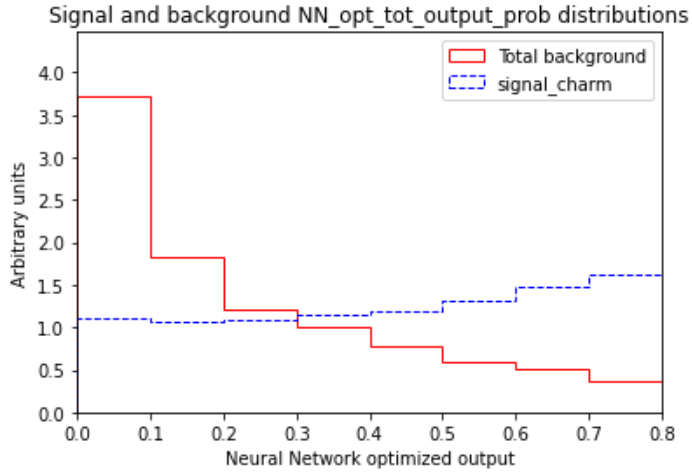
Other Classifiers

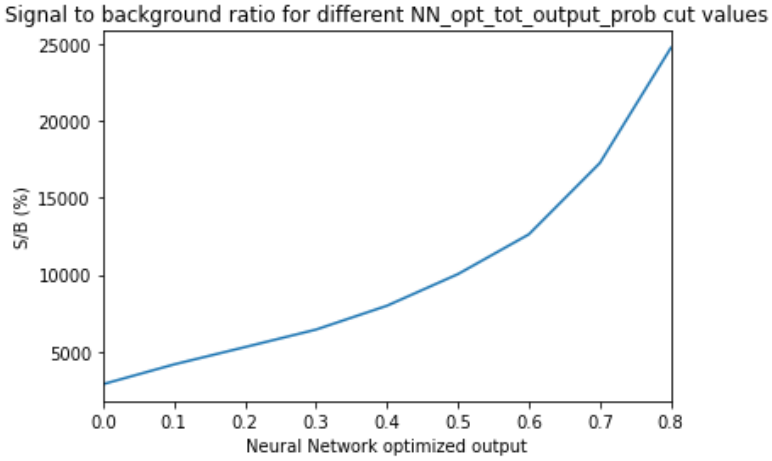


NN-opt with total data

```
Total accuracy 0.6215426727131593
Classification report for the total set
              precision    recall  f1-score   support

  background       0.07      0.82      0.13      3265
      signal       0.99      0.61      0.76     94643

    accuracy                           0.62     97908
   macro avg       0.53      0.72      0.44     97908
weighted avg       0.96      0.62      0.74     97908

Confusion matrix:
 [[ 2686   579]
 [36475 58168]]
```

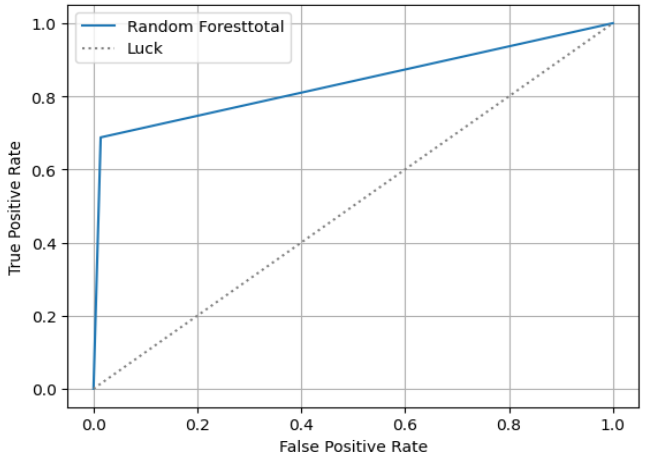


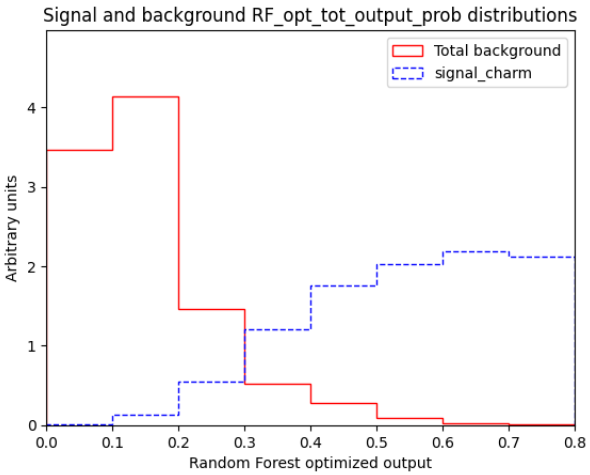


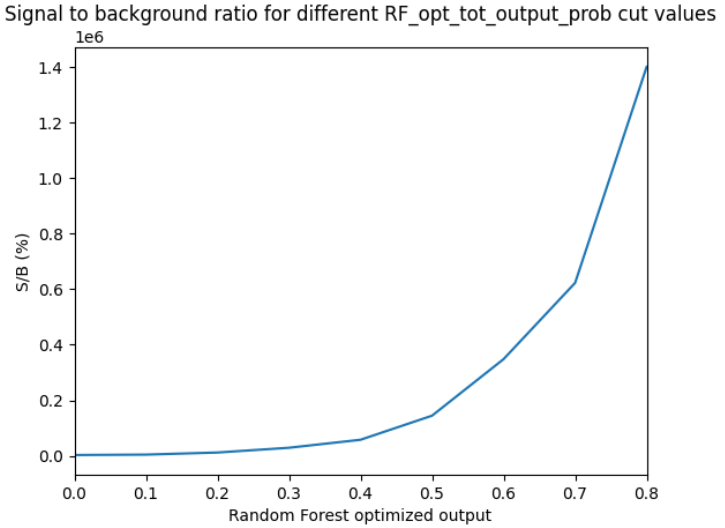


RF-opt with total data

```
Total accuracy 0.6976345140335826
Classification report for the total set
              precision    recall  f1-score   support

  background       0.10      0.99      0.18      3265
      signal       1.00      0.69      0.81     94643

    accuracy                           0.70     97908
   macro avg       0.55      0.84      0.50     97908
weighted avg       0.97      0.70      0.79     97908

Confusion matrix:
 [[ 3220    45]
 [29559 65084]]
```







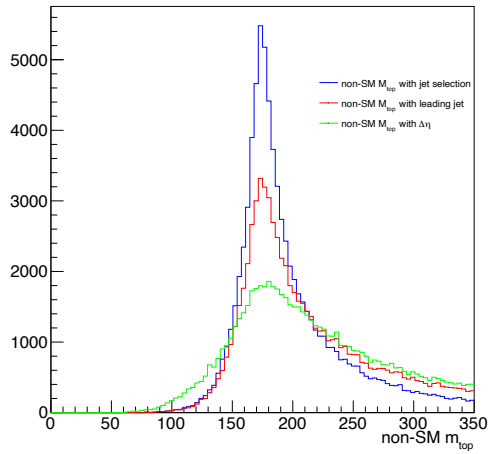
Summary & ongoing

- Several ML classifiers are trained using small subset of data and important analysis features.
- After model (NN and RF) optimization, both have good performance. RF gives higher accuracy score, TPR and lower FPR.
- RNN and BDT show weak performance (notebook)!
- Optimized RF and NN models are applied to the whole dataset but unfortunately TNR (real signals which are predicted as background) is high.
- Analysis tree production with important variables and plotting framework are done (tree framework, plotter framework).
- As the next step, need to make approximate median significance (AMS) plots. the primary goal is optimizing the discovery region for statistical significance.
- Your feedback is welcome and appreciated.

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