Investigating the Flavour Anomalies

Gabriela Sousa, Teresa Escária

Supervisors: Alessio Boletti, Nuno Leonardo

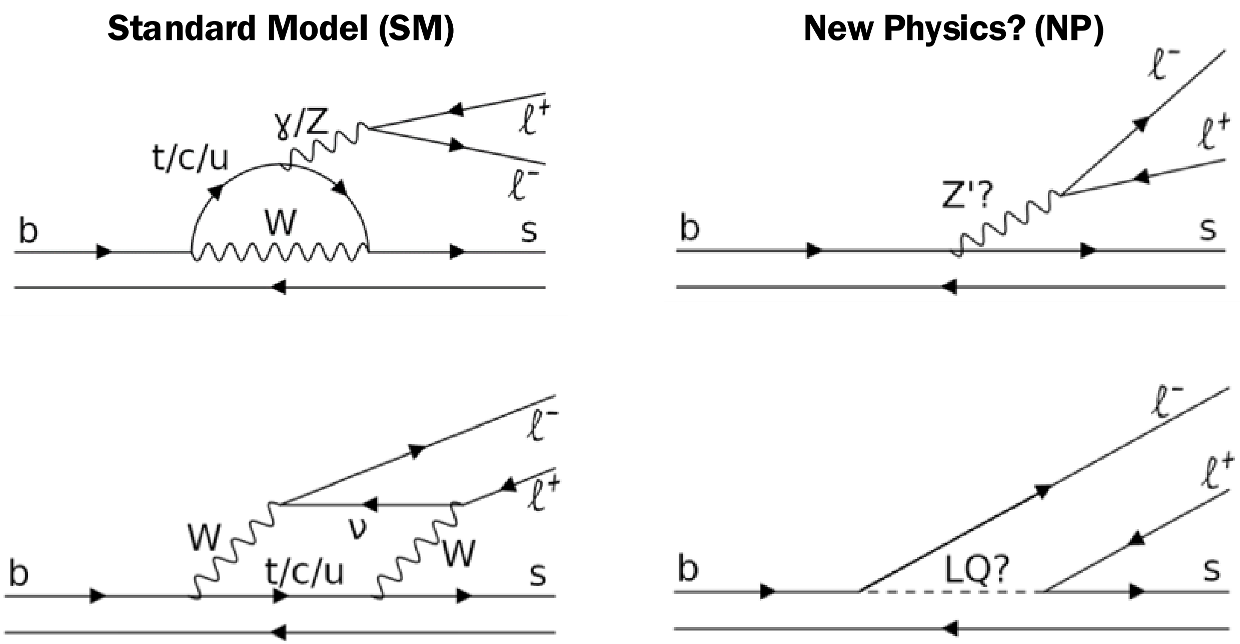
Flavour Anomalies

Lepton Flavour Universality (LFU)

- SM predicts different charged leptons have identical electroweak interaction strengths
- Violation of LFU may result in physics BSM

$b \rightarrow s l^+ l^-$ transitions

- Flavour-changing neutral currents (FCNCs) decays
- Forbidden at tree level in the SM

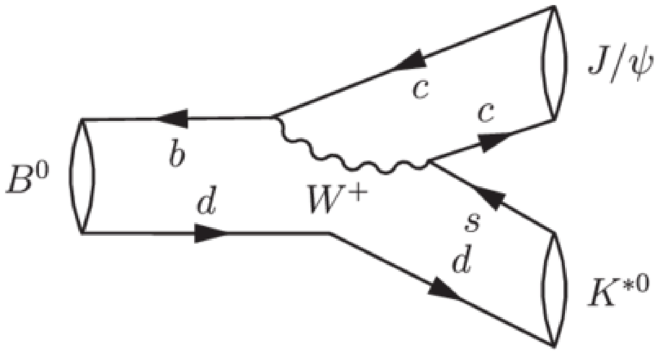


The $B^0 \rightarrow K^{*0}\mu^+\mu^-$ decay

Resonant Channel

$B^0 \rightarrow K^{*0}J/\Psi \rightarrow K^+\pi^-\mu^+\mu^-$

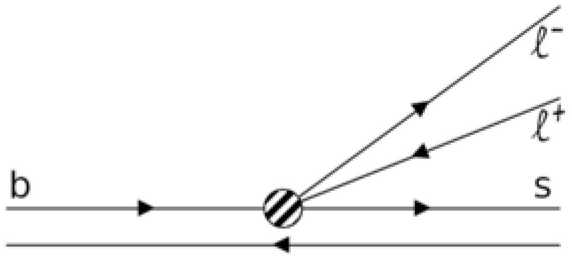
- Well understood within the SM
- Used as a control channel



Non-resonant Channel

$B^0 \rightarrow K^{*0}\mu^+\mu^- \rightarrow K^+\pi^-\mu^+\mu^-$

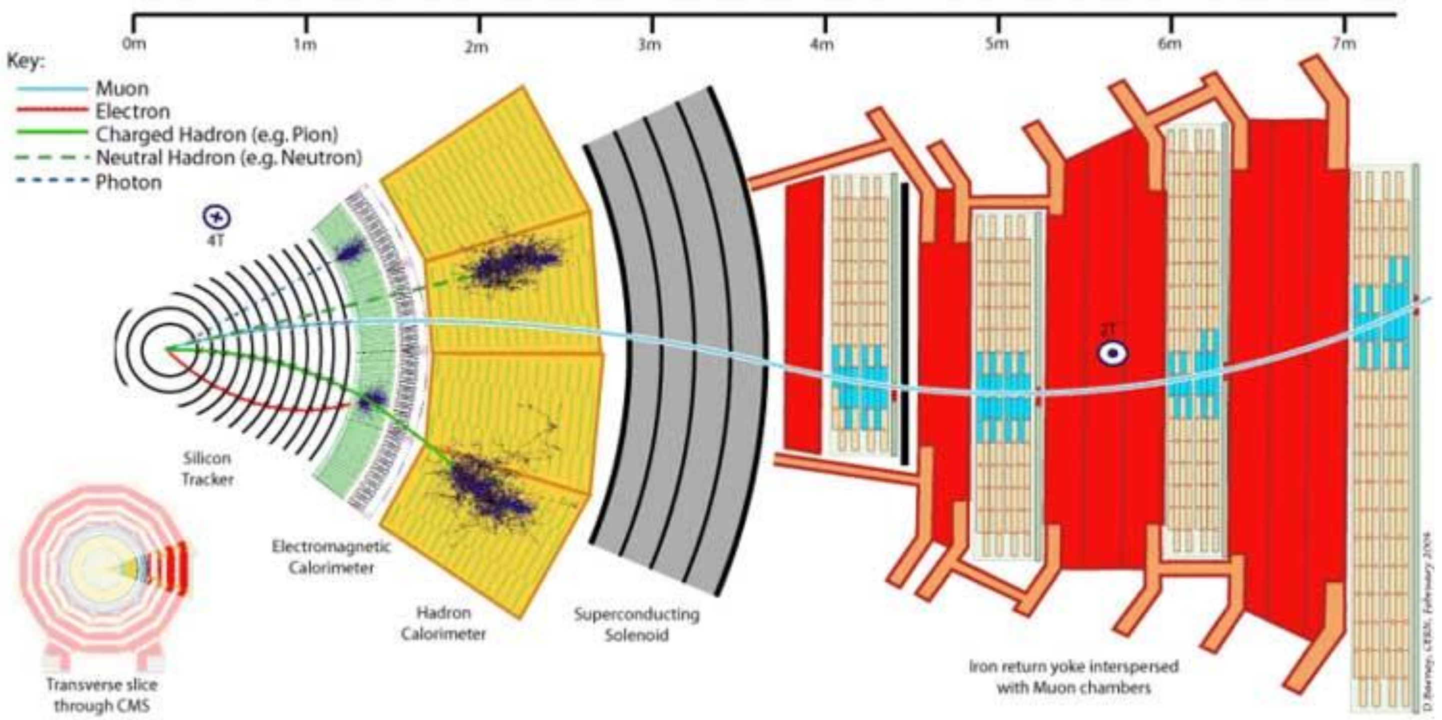
- Possibility of NP



Compact Muon Solenoid
CMS

CMS

General purpose detector at LHC



Two Key Subsystems:

Silicon Tracker

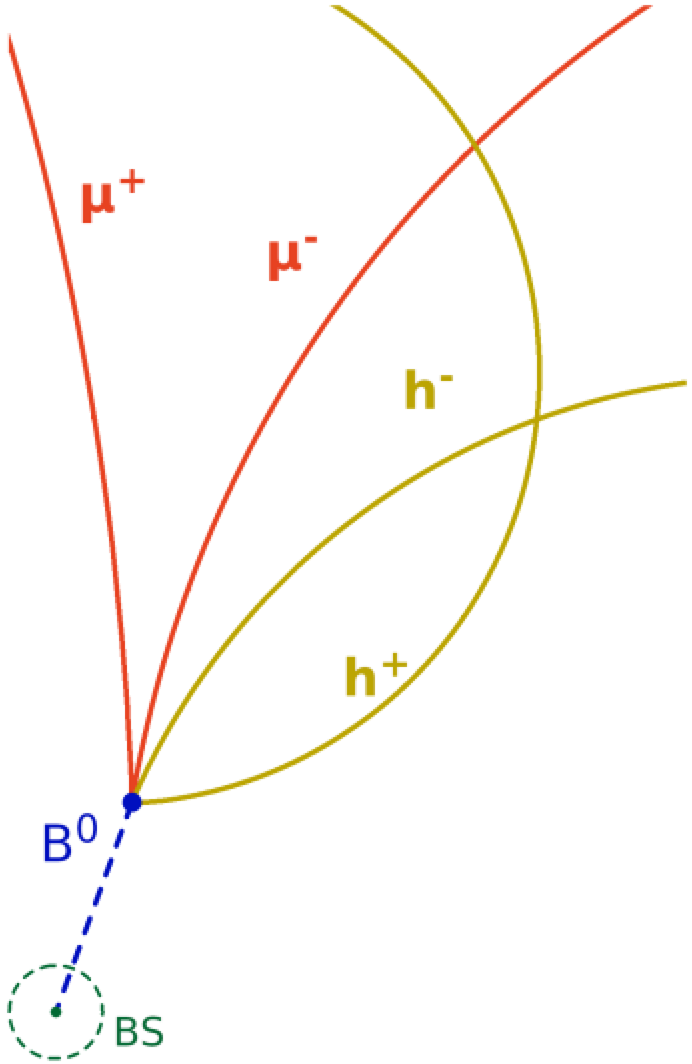
- Reconstructs charged particle trajectories, by collecting discrete hits
- Can fit precise tracks, determine momentum and reconstruct vertices for our desired decay

Muon Chambers

- Detect muons exiting the calorimeters

Candidate Reconstruction

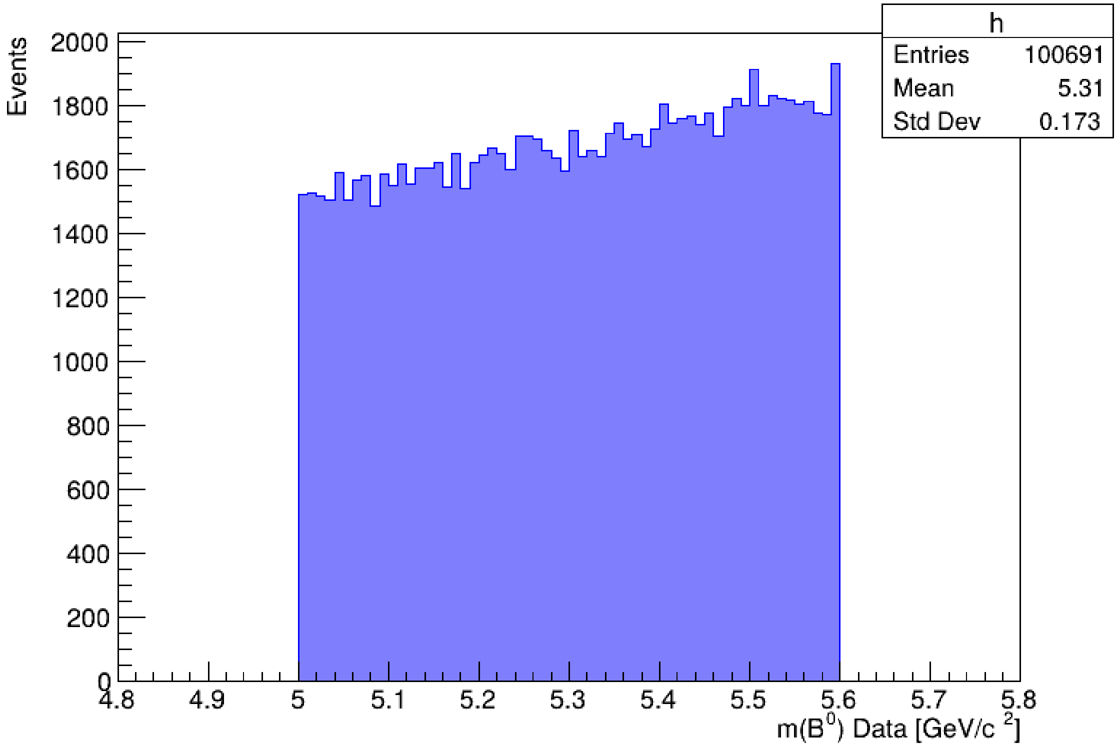
- Both $B^0 \rightarrow \boldsymbol{K}^+\boldsymbol{\pi}^-\mu^+\mu^-$ and $\bar{B}^0 \rightarrow \boldsymbol{K}^-\boldsymbol{\pi}^+\mu^+\mu^-$ decays occur
- CMS has no hadron PID:
  - the K/π combination closest to $m_{PDG}(K^{*0})$ is associated to the event (a fraction of events can be mis-tagged)
- B^0 has a long lifetime, leading to a secondary vertex displaced from the primary
- **Combinatorial Background:**
  - Tracker and muon system may reconstruct uncorrelated tracks that fake B^0 candidates
  - Contributes to a smooth, non-peaking background in the B^0 mass distribution
- **Physics Background:**
  - Other real B decays with similar final states can mimic the signal



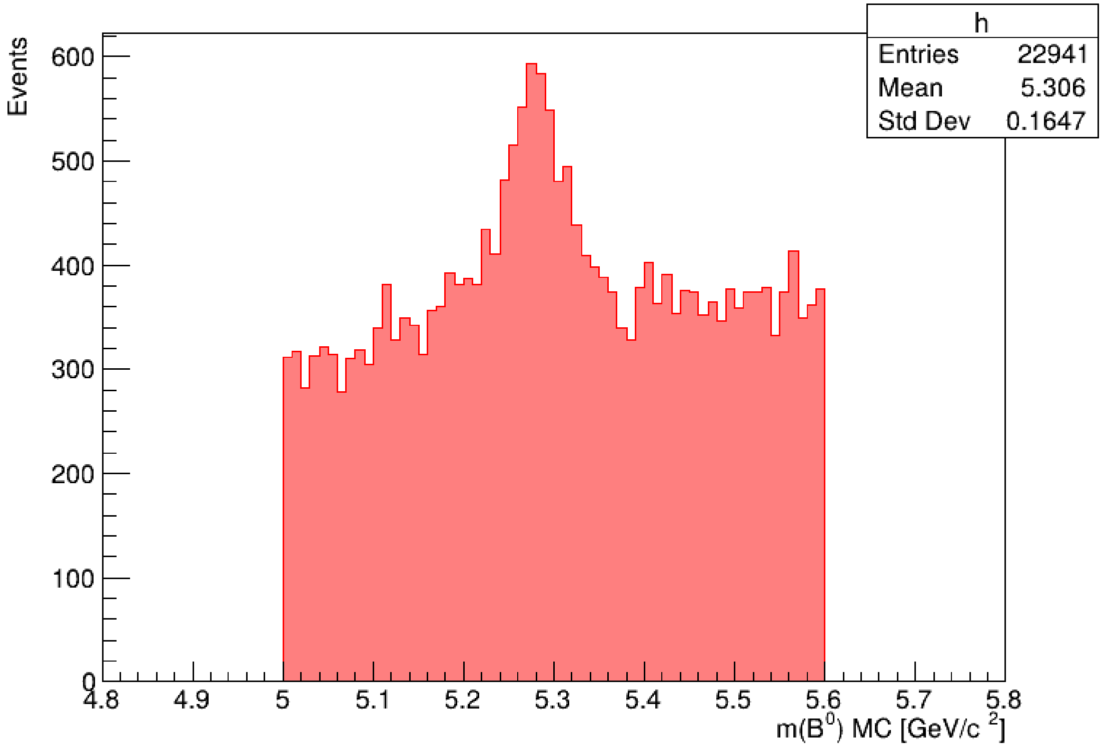
B^0 Mass Spectrum

$m_{PDG}(B^0) = 5279.63 \pm 0.20\ MeV$

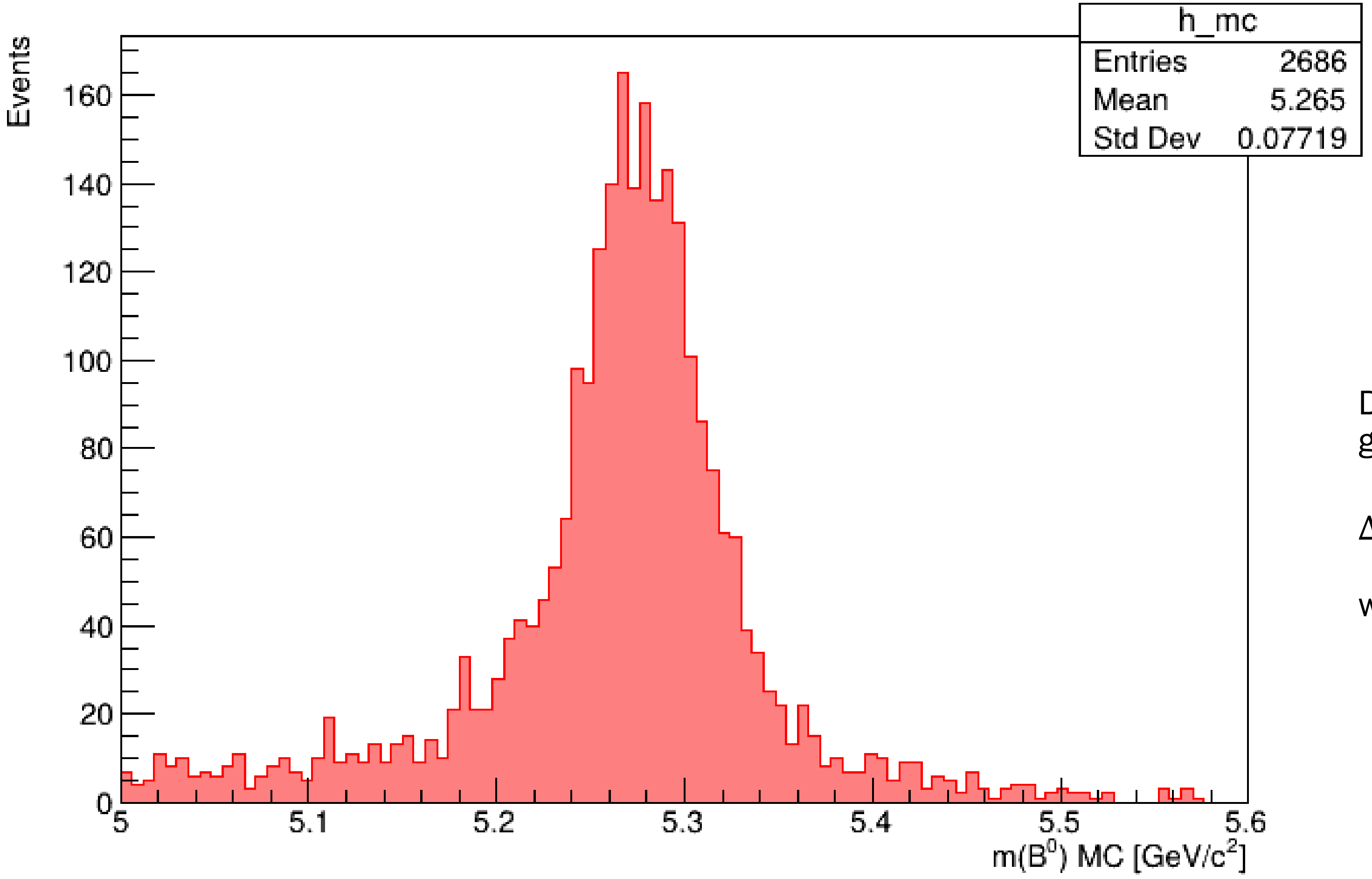
Data



MC (unmatched)



Truth Matching - MC



Done by calculating the angular distance between the generated and reconstructed variables:

$$\Delta R = \sqrt{\left(\eta_{reco} - \eta_{gen}\right)^2 + \left(\phi_{reco} - \phi_{gen}\right)^2},$$

where smaller ΔR means a closer match

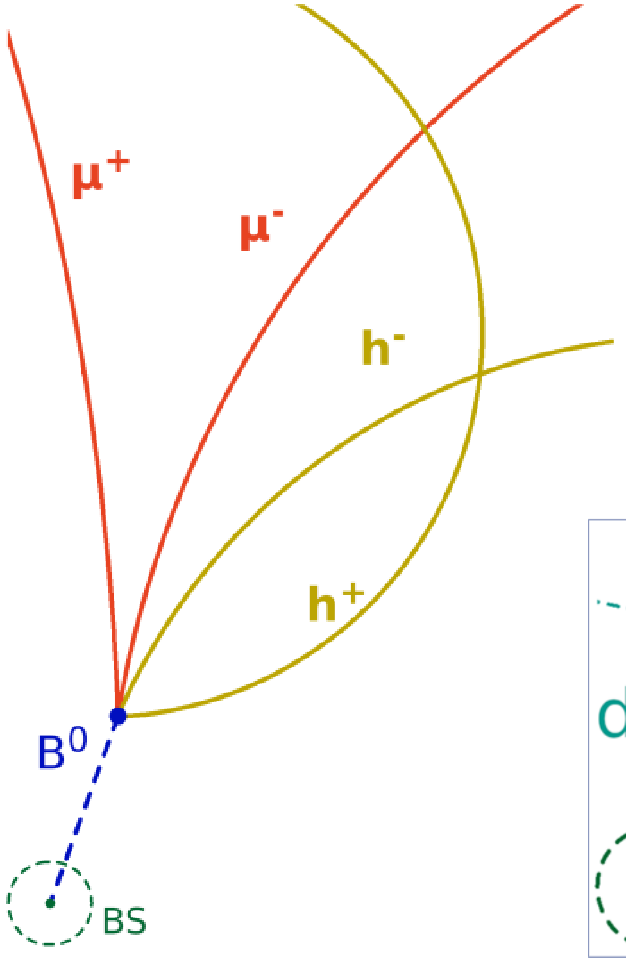
Motivation for Optimised Selection

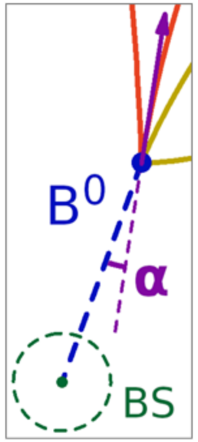
- Have a clear signal shape in simulation
- Want to achieve the same in real data, which is dominated by background
- Need to devise a strategy to **enhance the signal-to-background ratio**

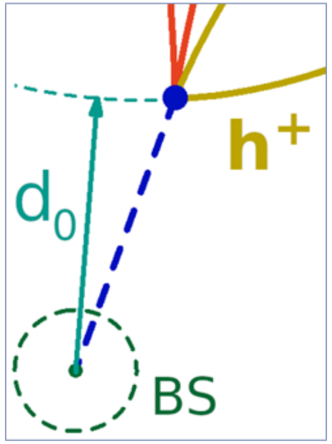
↓

Optimise Selection: apply cuts on variables that help distinguish real B^0 decays from background

Discriminating Variables





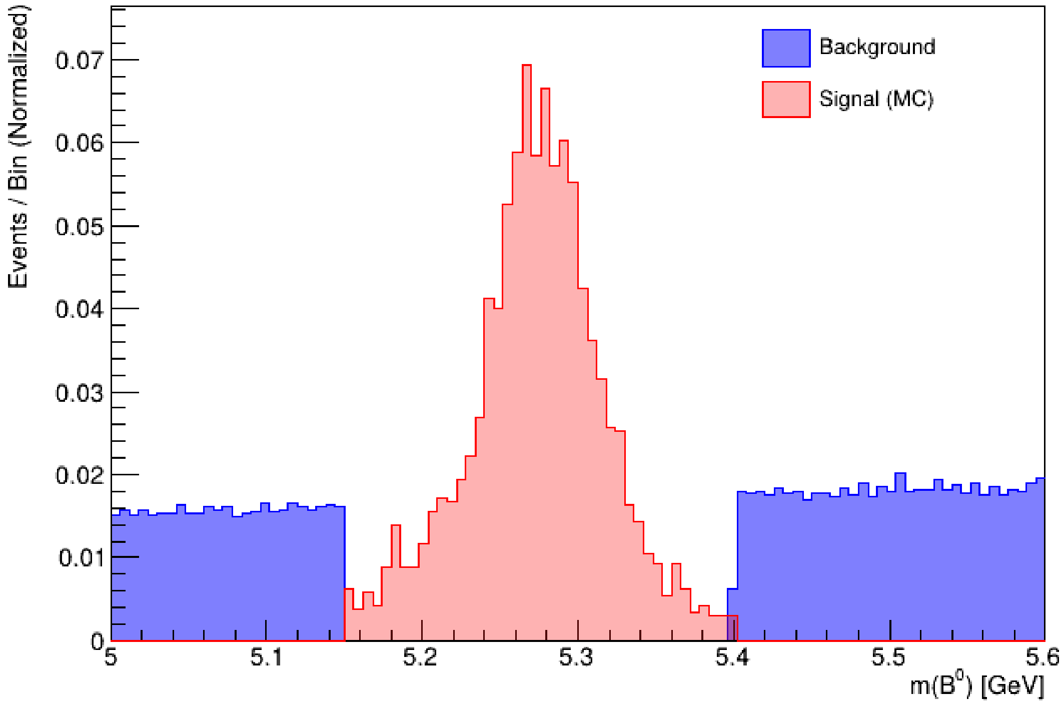


There are several kinematic and geometric variables that can help:

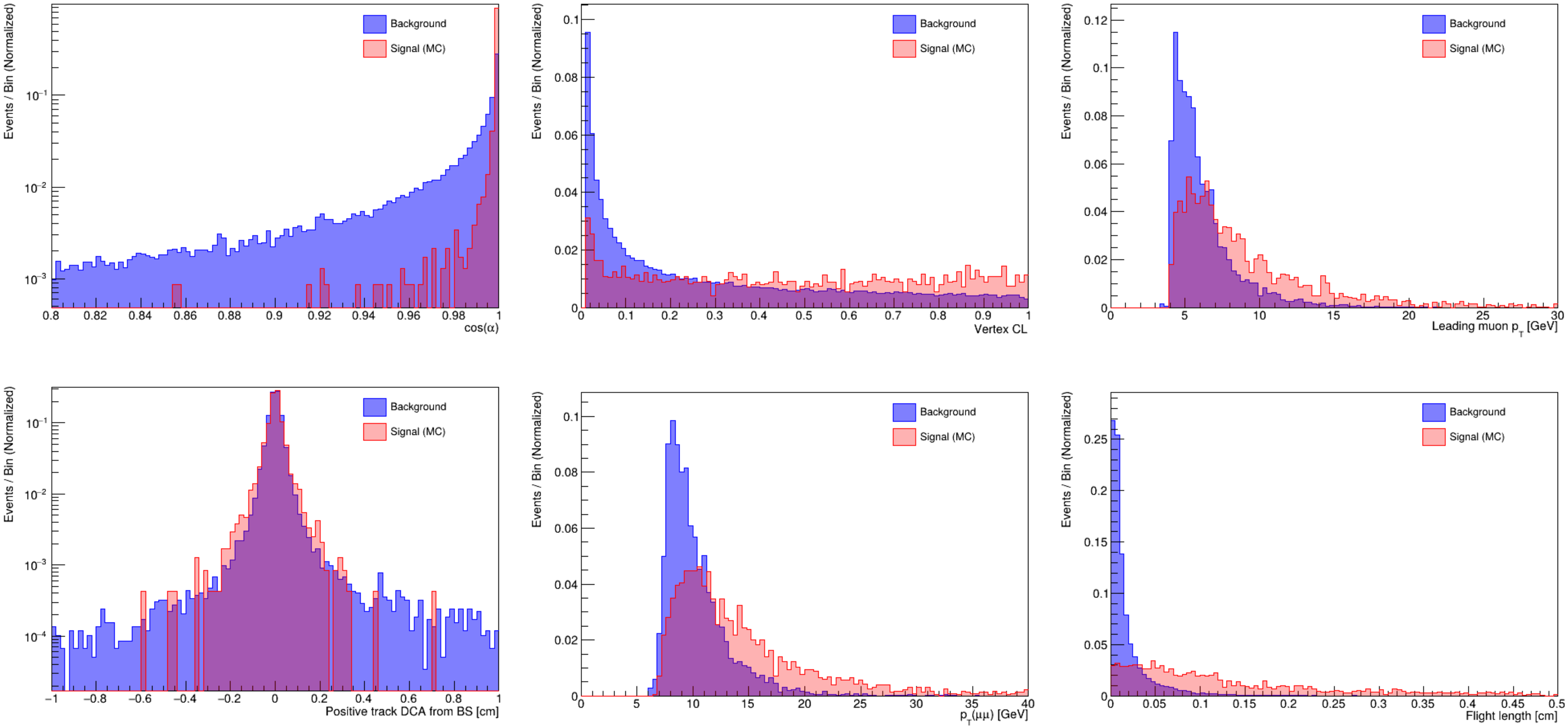
- **Flight Length:** distance between the primary vertex and the secondary vertex (B^0 decay vertex)
- **Cos(α):** cosine of the angle between the reconstructed B^0 meson momentum and its flight direction
- **Vertex CL:** measure of how likely the four tracks come from the same decay point
- **DCA:** how close a track comes to the beamline
- **p_T, η, ϕ:** transverse momenta, pseudorapidities and azimuthal angle of various variables

Background Sample from Data Sidebands

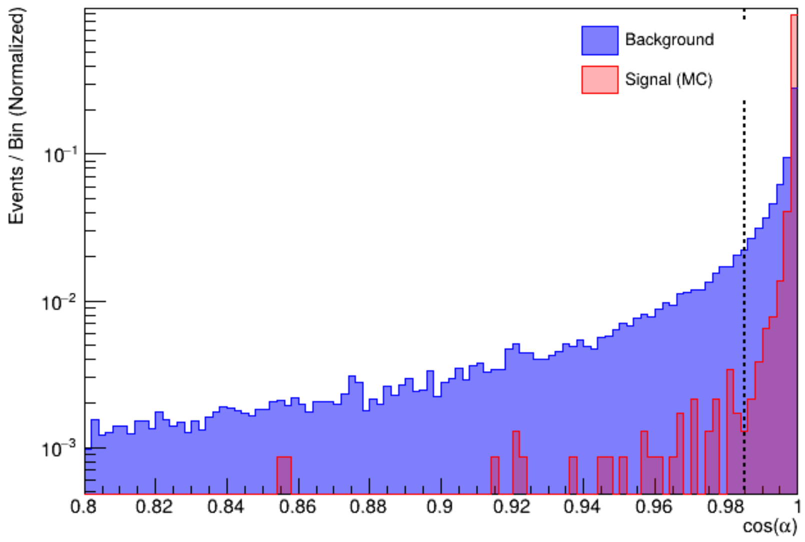
- **Signal sample:** from Monte Carlo
- **Background sample:** from data sidebands
    - ~3.5 σ away from mean of the signal peak
- Sidebands assumed to be dominated by background processes
- Events with extreme B^0 masses (< 5 GeV or > 5.6 GeV) excluded
- Compared MC signal vs. data background across 32 kinematic and geometric variables
- Each variable plotted as a normalized histogram with signal and background overlaid

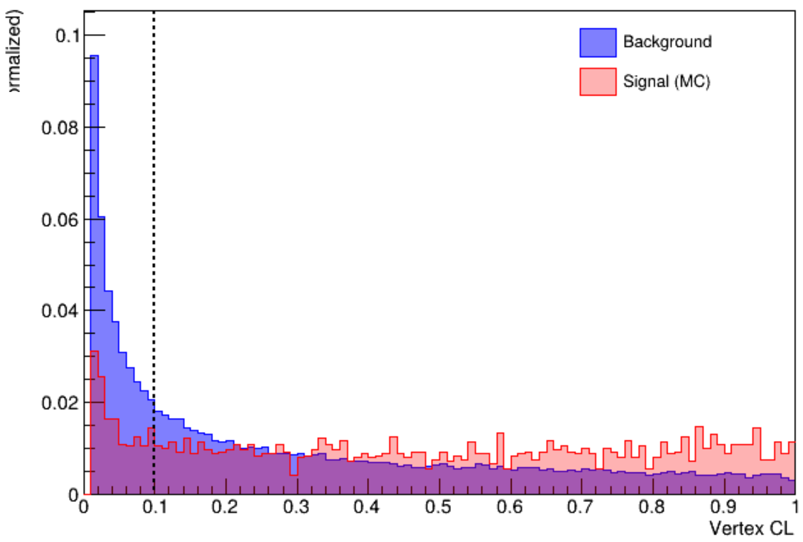


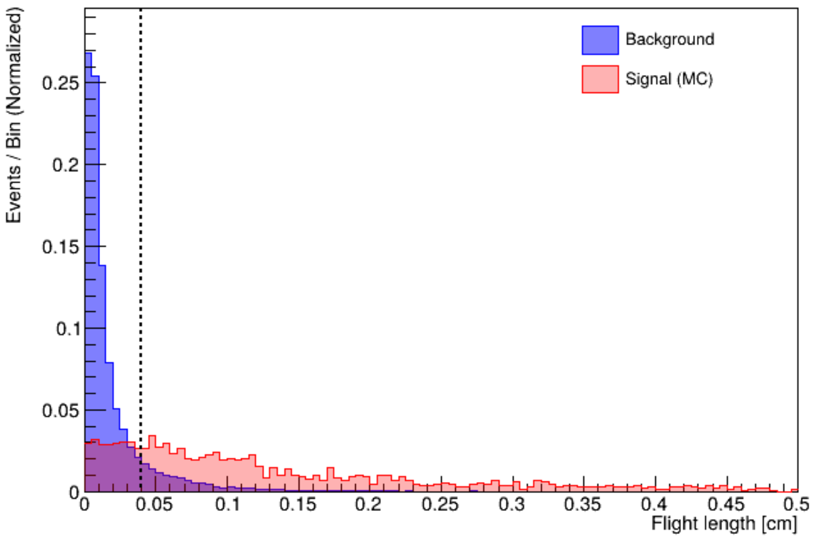
Signal vs Background

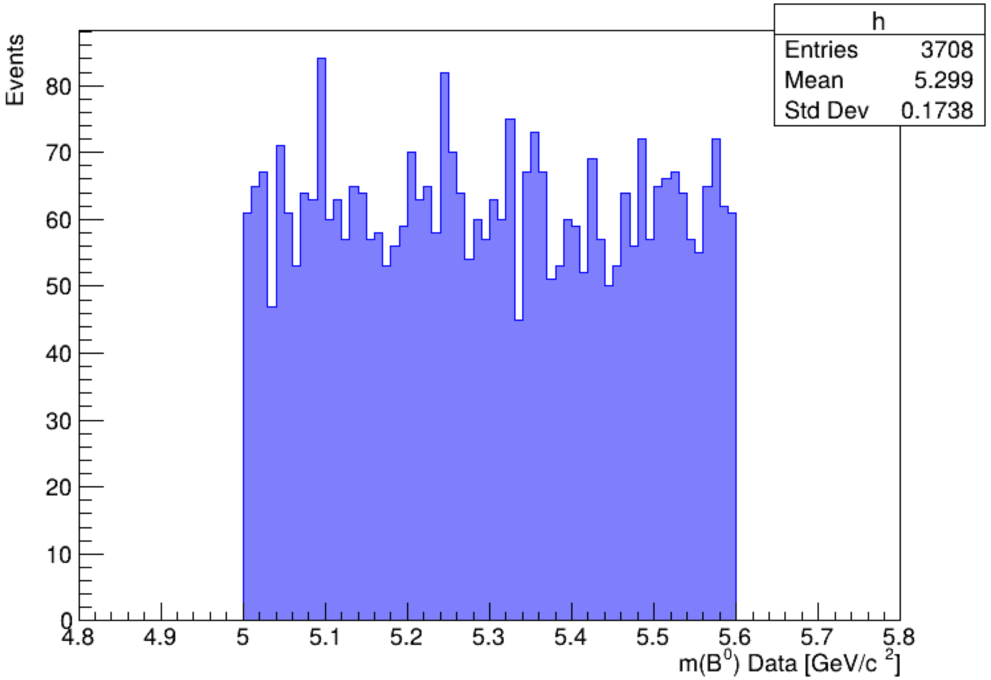


Applying Cuts



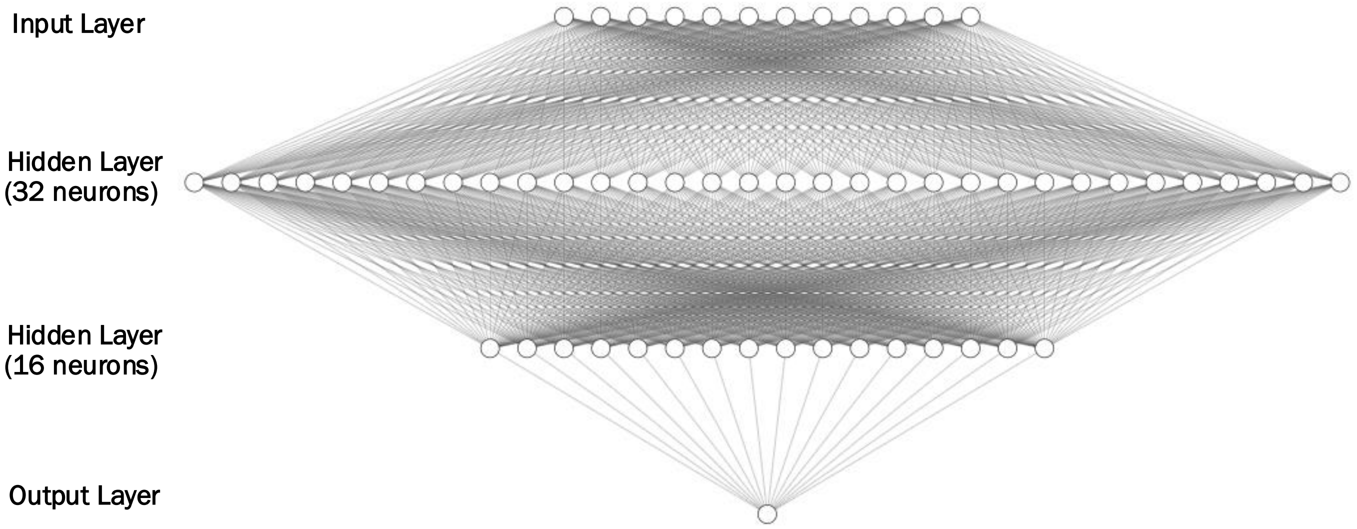


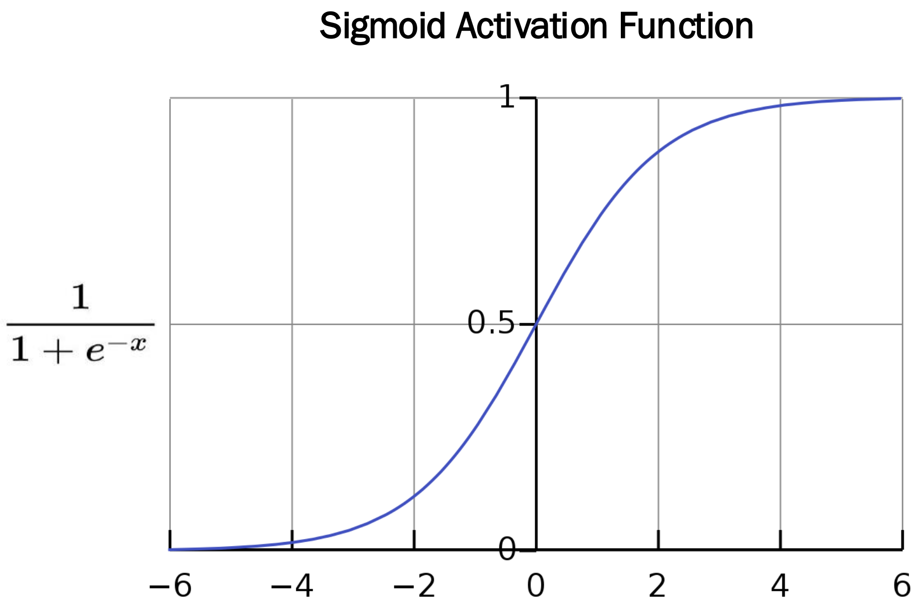




Neural Network Classifier

- Architecture:
  - Two fully connected hidden layers (32 and 16 neurons, respectively) with ReLU activation
  - Dropout layers (20%) to prevent overfitting
  - Sigmoid output for binary classification (signal vs background)





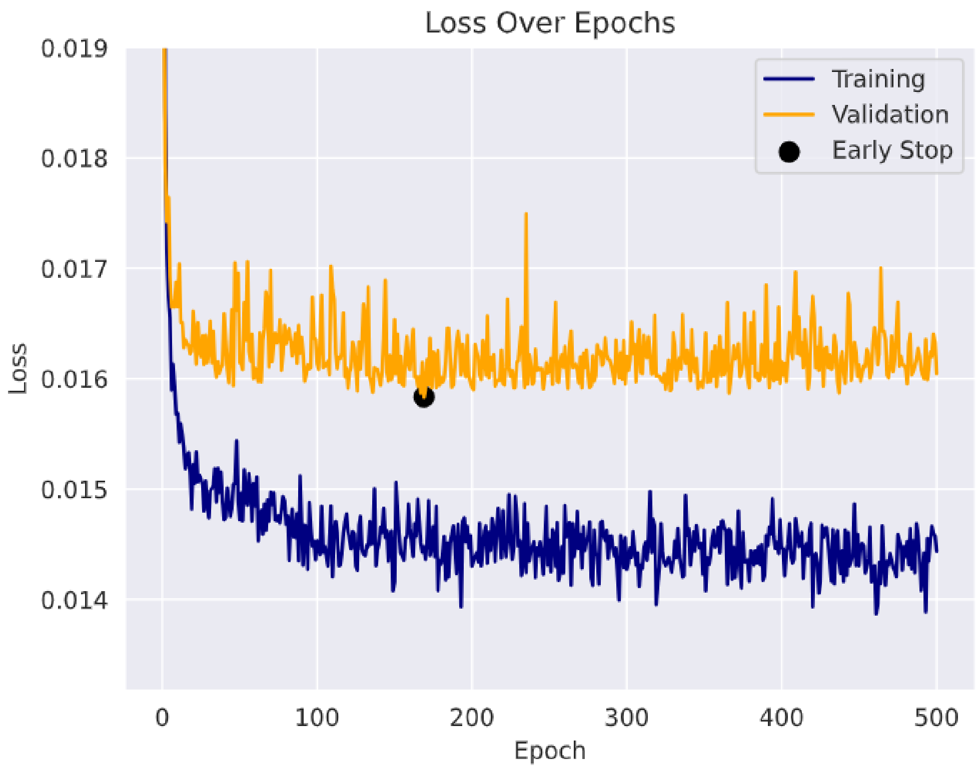
Loss Function and Class Imbalance

- **Binary Cross-Entropy (BCE):**
  - Compares predicted probabilities (0–1) to true labels (0 or 1)
  - Penalizes confident wrong predictions more strongly
  - For one example:

$$L_{BCE} = -[y \log(\hat{y}) + (1-y)\log(1-\hat{y})]$$

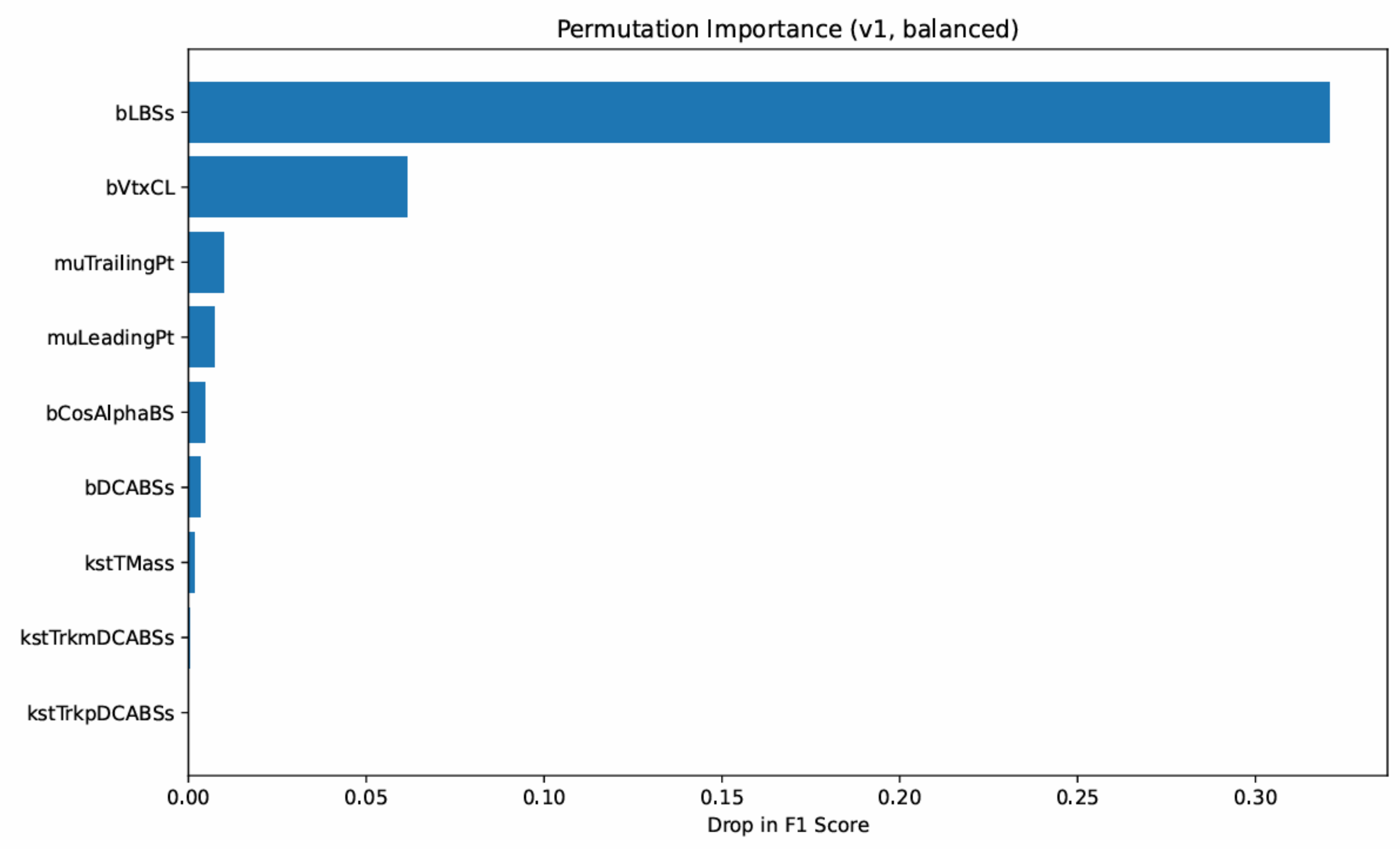
- **BalancedLoss (Weighted BCE):**
  - Extends BCE by introducing class weights via alpha
  - Helps address class imbalance by up-weighting underrepresented class
  - Ensures the model doesn't become biased toward predicting background only

$$L_{Balanced} = -[\alpha_1 y \log(\hat{y}) + \alpha_0(1-y)\log(1-\hat{y})]$$



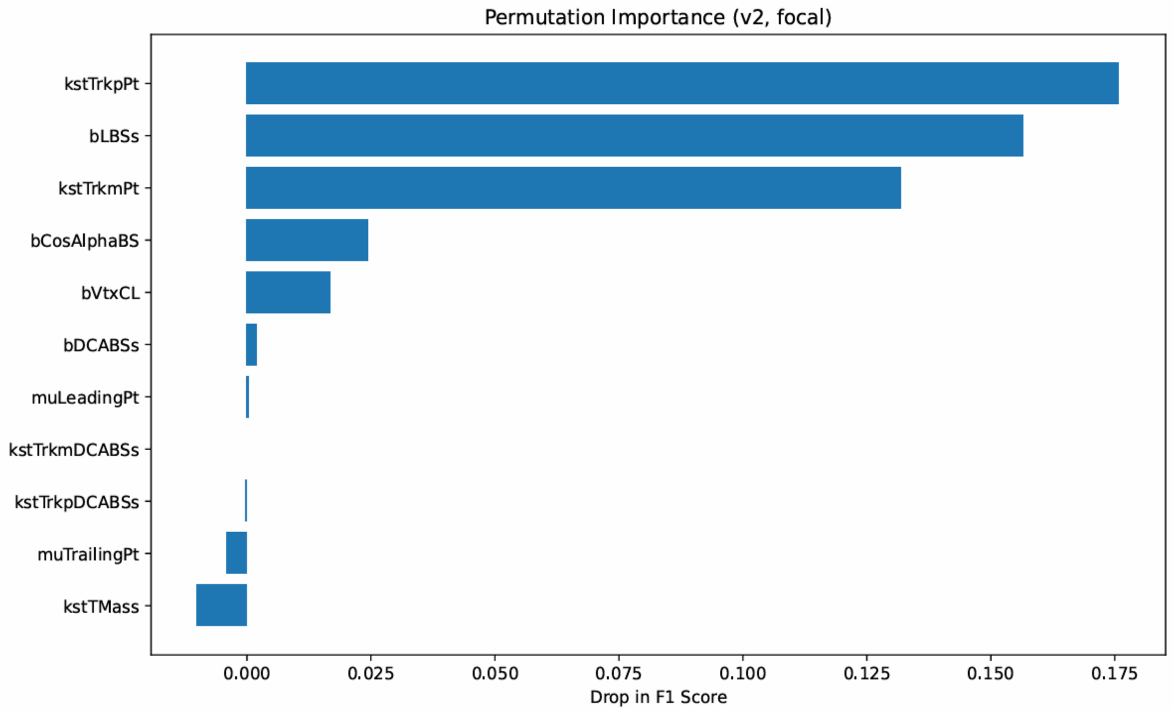
Feature Importance

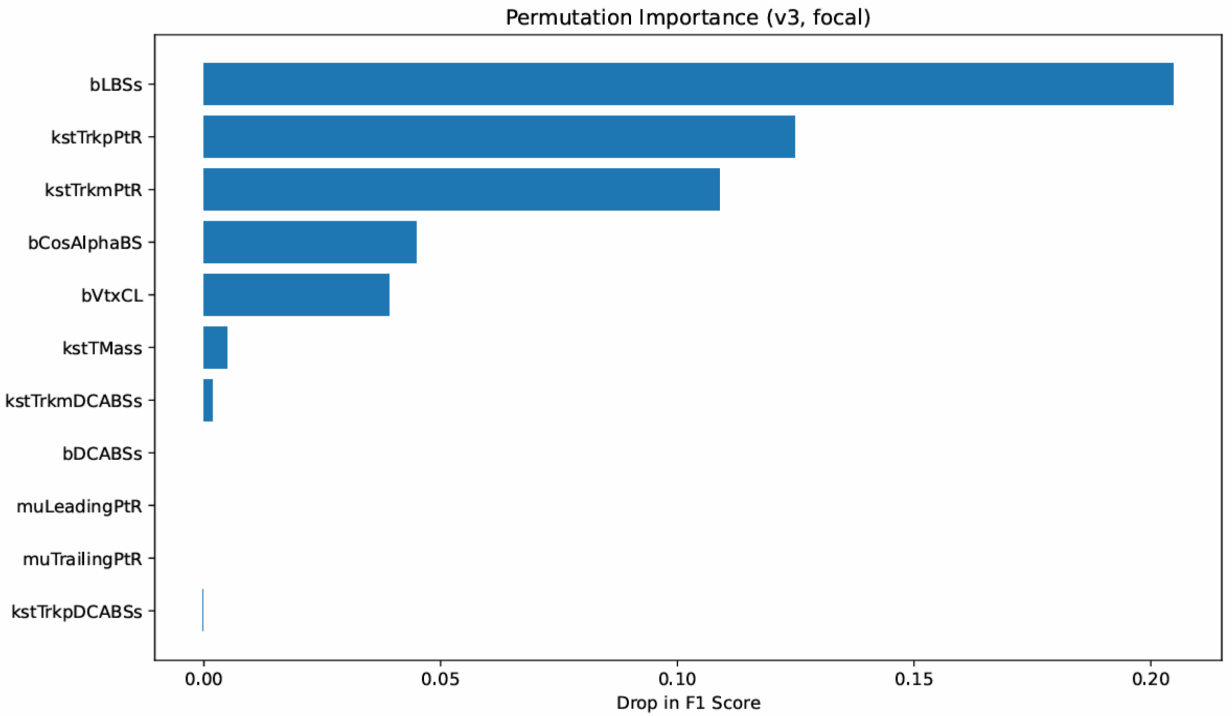
- We apply feature importance to identify which input variables most influence the model's predictions.
- This supports model interpretation, feature selection, and debugging.
- We use **Permutation Importance**, which measures how much the performance drops when individual features are randomly shuffled.



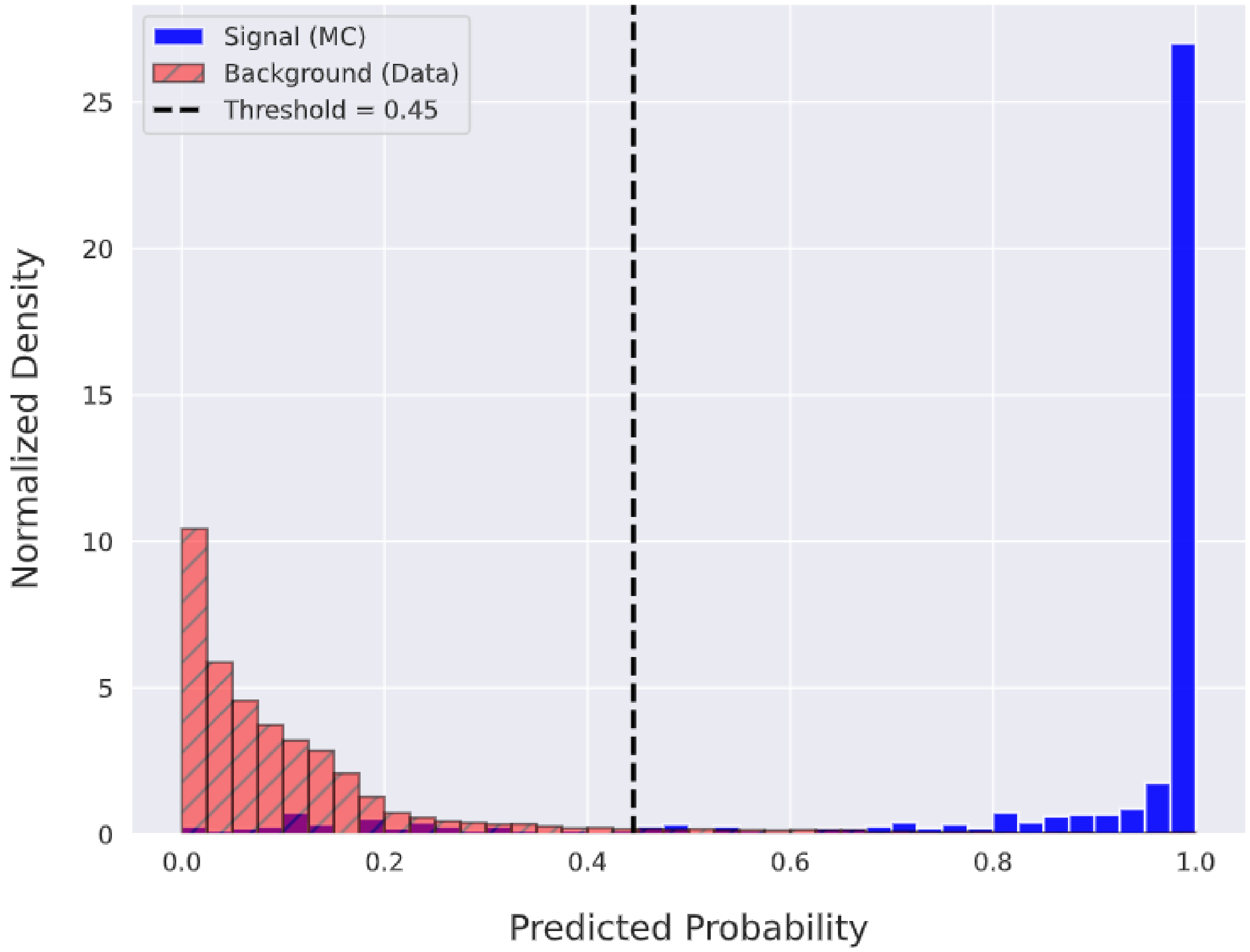
Feature importance

For different versions of the dataset and a different loss type.

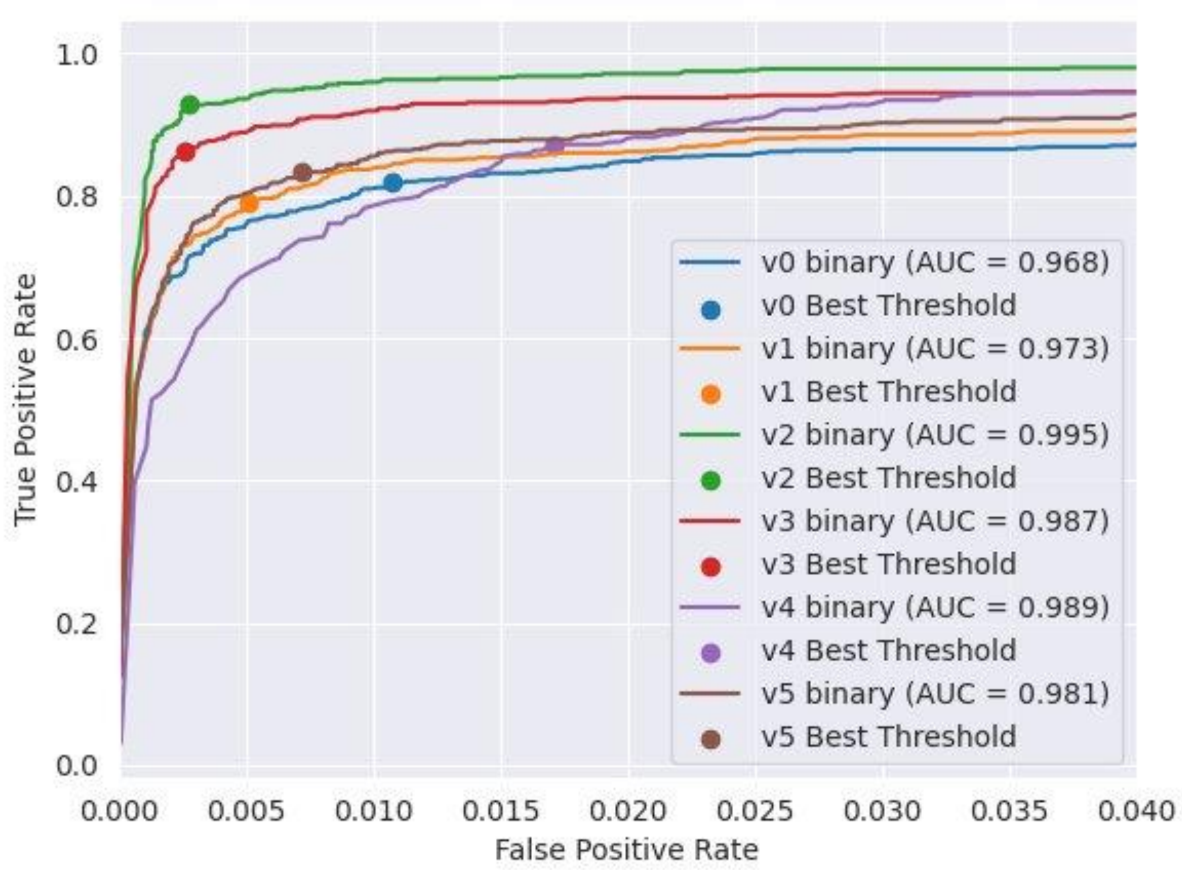




Neural Network Output



Model Evaluation



- Used **Receiver Operating Characteristic (ROC)** curves to evaluate classifier performance
- Plots True Positive Rate (TPR) vs. False Positive Rate (FPR) across all thresholds
  - **TPR** = signal efficiency;
  - **FPR** = background misclassification rate
- **Area Under the Curve (AUC)** gives a single performance metric
  - AUC = 1 perfect classification
  - AUC = 0.5 random performance

Threshold Optimisation

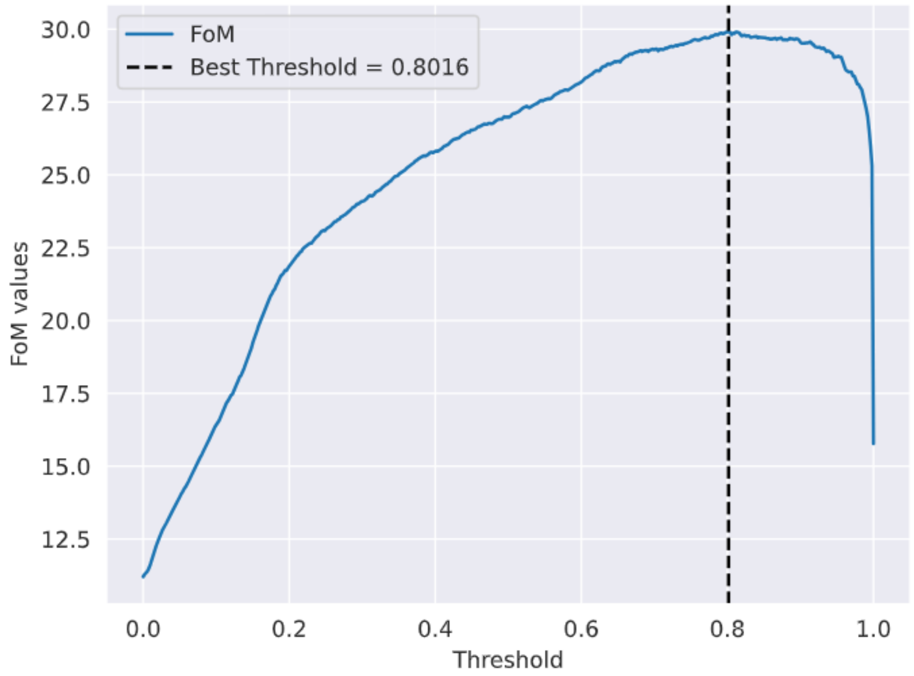
- Signal (MC) and background (data) have different statistics:
  - Raw event counts not directly comparable
- Used **Youden's J statistic** to guide initial threshold choice:

$$J = TPR - FPR$$

- Maximising J gave initial optimal threshold
- Applied model and initial threshold to full dataset to derive a scaling factor (estimate of background fraction under the peak of B0 mass spectrum)
- Scaling applied in a **Figure of Merit:**

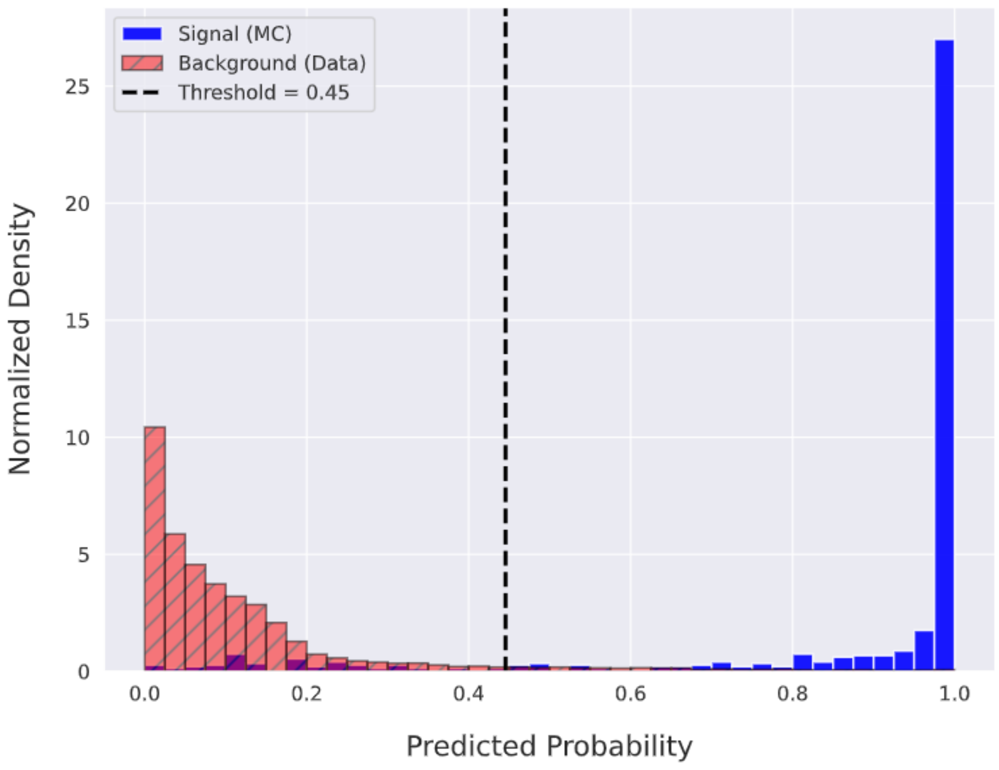
$$FOM = \frac{S}{\sqrt{S+B}}$$

- S/B: signal and background event counts above the threshold after applying estimated scaling

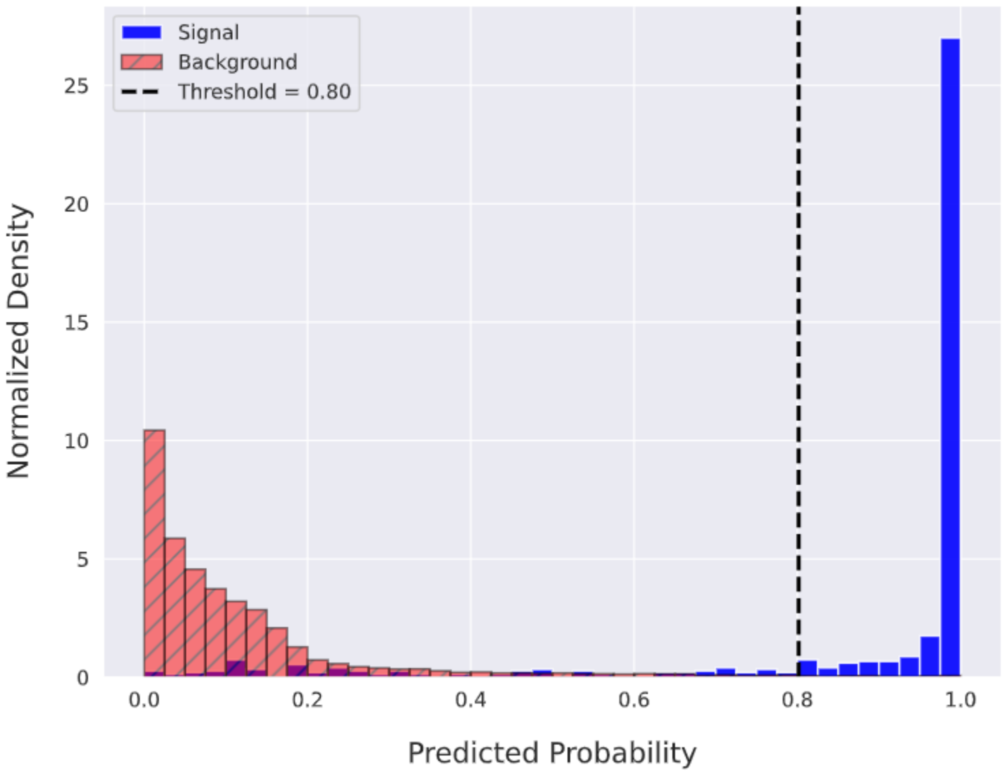


Neural Network Output

Youden's J statistic threshold



FoM optimal threshold



Next Steps:

- Look into model optimisation and explore different hyperparameters;
- Use other methods to calculate feature importance;
- Analyse the variable correlation matrix to detect redundancy;
- Apply trained model and optimal threshold to full dataset;
- Investigate presence of signal in real data after selection;
- Fit post-selection B^0 mass spectrum;
- Extract signal yield and compare to Standard Model expectations.

Backup Slides