Department of Physics and Astronomy
Heidelberg University

Bachelor Thesis in Physics
submitted by

Anselm de Jonge

born in Veldhoven (Netherlands)

2003

Study on a possible opposite side proton tagger for LHCb Upgrade II using a TORCH detector

This Bachelor Thesis has been carried out by Anselm de Jonge at the
Physikalisches Institut in Heidelberg
under the supervision of
Prof. Dr. Stephanie Hansmann-Menzemer

Abstract

Neutral B mesons can oscillate between their particle and antiparticle state. The measurement of these oscillations is an important topic at the LHCb experiment in order to probe the standard model of particle physics, our most complete theory describing fundamental particles and their interactions. To measure these oscillations, it is necessary to know the B flavour at the moment of its production and decay. In order to determine the production flavour, specific algorithms, usually referred to as flavour tagging algorithms, are used. Flavour tagging algorithms exploit the properties of specific particles to correlate their charges with the initial B flavour.

For the next major upgrade, Upgrade II, of the LHCb detector, a scenario with the addition of a new time of flight detector called TORCH is considered. This would significantly improve the distinction between protons and kaons. Thanks to the potential improvement of the detector, the possibility of a tagging algorithm which uses protons from the opposite side B decay, should be studied.

Using Monte Carlo simulated data, where the origin and associated particle of a track are known, this thesis confirms that the selection of protons improves significantly with a perfect proton identification at track momenta below 10 GeV.

Crucially, neural networks were trained to perform flavour tagging on these selections, however they were unable to perform well, despite the improved selection. This study reveals a new challenge for such a tagging algorithm, due to protons originating from the opposite side B decays having conflicting relations to the initial B flavour depending on what decay the protons originate from. This thesis demonstrates that distinguishing these decays based solely on the track information is not feasible, indicating the need for further investigation to develop a more effective approach.

Zusammenfassung

Neutrale B Mesonen können zwischen ihrem Materie und Antimaterie Zustand oszillieren. Die Messung dieser Oszillation ist ein wichtiges Thema am LHCb Experiment um das Standardmodel der Teilchenphysik, unsere vollständigste Theorie um fundamentale Teilchen und deren Wechselwirkungen zu beschreiben, zu erproben. Um diese Oszillationen zu messen, benötigt man den "flavour" Zustand des B Mesons zum Zeitpunkt der Entstehung und des Zerfalls. Um den Zustand zum Entstehungszeitpunkt zu bestimmen werden sogenannte flavour tagging Algorithmen verwendet. Diese nutzen die Eigenschaften bestimmter Teilchen aus, um deren Ladung mit dem Zustand des B Mesons zu korrelieren.
Für die nächste große Aufrüstung, Upgrade II, des LHCb-Detektors wird ein Szenario in Betracht gezogen, bei dem ein neuer Flugzeitdetektor namens TORCH eingebaut werden würde. Dieser würde die Unterscheidung zwischen Protonen und Kaonen erheblich verbessern. Dank dieser potenziellen Verbesserung des Detektors sollte die Möglichkeit eines tagging Algorithmus untersucht werden, der Protonen aus dem "opposite side" B-Zerfall verwendet.
Mithilfe von Monte-Carlo-simulierten Daten, bei denen der Ursprung und das zugehörige Teilchen einer Teilchenspur bekannt sind, bestätigt diese Arbeit, dass sich die Auswahl der Protonen mit einer perfekten Protonenidentifizierung bei Impulsen unter 10 GeV deutlich verbessert.
Zudem wurden neuronale Netze zum flavour tagging auf der verbesserten Auswahl trainiert, jedoch konnten diese trotz der verbesserten Auswahl keine guten Ergebnisse liefern. Diese Studie weist auf eine neue Herausforderung für einen solchen Algorithmus, da die "opposite side" Protonen, je nachdem, aus welchem Zerfall sie stammen, einen anderen Zusammenhang zum Zustand des B Mesons zum Entstehungszeitpunkt haben. Diese Arbeit zeigt, dass die Unterscheidung dieser Zerfälle allein unter Verwendung der Spurinformationen nicht möglich ist und weist auf die Notwendigkeit weiterer Studien für die Entwicklung effektiverer Ansätze hin.

Contents

Chapter 1

Introduction

Particle physics is a field of research which attempts to explain phenomena in nature at the most fundamental level possible by studying the interactions between the smallest building blocks of the universe. Our current knowledge is accumulated in the Standard Model (SM) of particle physics [1], which is a highly tested and studied theory. The SM is successful at describing the electromagnetic, strong, and weak interaction, however, it is not a complete theory as it fails to incorporate gravity or to explain the matter-antimatter asymmetry in the universe.
The study of the SM is a key topic at the European Organization for Nuclear Research (CERN), the world leading organisation for nuclear and particle physics research. It hosts the largest particle accelerator in the world: the Large Hadron Collider (LHC) [2], at which protons are collided with a center of mass energy of up to 14 TeV. These collisions generate a large number of particles, which may undergo interactions and decays, the products of which are then measured by four detectors located around the collision points, each with their respective experiment. CERN also hosts several smaller detectors and experiments.
This study uses data collected by the LHCb experiment [3], one of the four main experiments at the LHC. The LHCb experiment is dedicated to precision measurements of decays of particles containing c and b quarks (heavy flavour decays), which give insight into important parameters of the standard model and might reveal new physics. A key challenge in heavy flavour physics is measuring the oscillation of B^0 or B_s^0 mesons [4], where knowledge of the the state in which the meson was created (initial state/flavour) is crucial. The process of discerning the initial flavour of a B meson, which in the context of this study is referred to as the signal B meson, is called flavour tagging. The flavour tagging method used in this thesis is opposite side tagging [5], which takes advantage of the fact that in proton-proton collisions the b quarks are produced in pairs with excess energy, which causes the two quarks to separate and form new hadrons in a process called hadronisation. While one of the b quarks can form a neutral B meson which can oscillate, the partner quark might form another hadron (opposite side B hadron) and decay in ways that conserve its initial information. The opposite side tagger aims to identify this.
The next large upgrade (Upgrade II [6]) for the LHCb detector is planned for 2033 and measurements for Run 5 & Run 6 should continue in 2035 with support for an instantaneous luminosity of up to $1.5 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, which is 7.5 times larger than current design specifications.

Several different scenarios are proposed for Upgrade II, including the construction of a new time of flight detector TORCH[7]. This would allow better particle identification at low momenta, especially for protons, kaons and pions.
The OS proton tagger, which uses protons originating from the opposite side B decays, has not been considered viable so far, due to the smaller fractions of protons in an event and also the difficult distinction from kaons, especially at low momenta. With the introduction of TORCH, however, this might become viable and studying the potential of such a tagger is the goal of this thesis.

Chapter 2

Physical foundations and methods

2.1 The LHCb experiment

The LHCb experiment is one of the four major experiments at the LHC at CERN. One of its main goals is the measurement of heavy flavour decays, which are decays involving c and b quarks. In order to suit this, the detector is designed as a single arm forward facing detector and covers a pseudorapidity range of $\eta \in [2,5]$, to take advantage of the higher production cross section of heavy quark pairs at low angles θ in proton-proton collisions. The pseudorapidity is defined with the angle θ relative to the beam as: $\eta = -\log(\tan(\theta/2))$.
A schematic of the detector can be seen in figure 2.1.

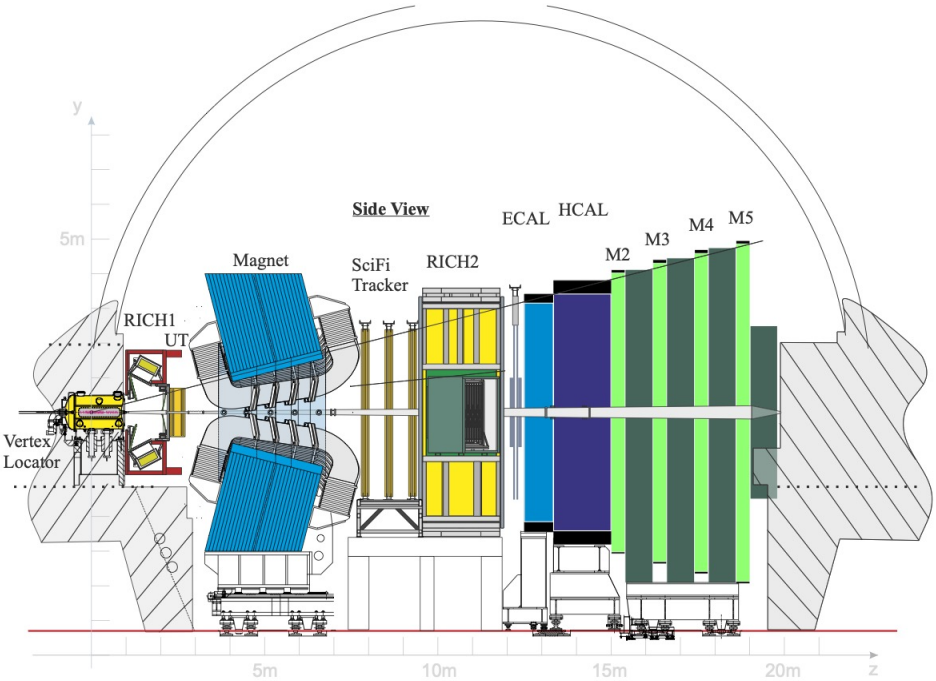

Figure 2.1: Schematic cross section of the Run 3 LHCb detector. [8]

2.1.1 Tracking system

The tracking system of the LHCb detector is responsible for reconstructing the trajectories of charged particles, called tracks, as well as associated information like the momentum, vertex positions and charge. The collection of signals and tracks produced in a single proton-proton collision is referred to as an event.
Starting at the proton-proton collision point, the vertex locator (VELO) is positioned around this point order to provide precise measurements for the reconstruction of the primary vertex (PV) and the determination of the impact parameter (IP) of particle tracks. The primary vertex is the position at which the collision occurs and the impact parameter is the closest distance of approach of an extrapolated track to the PV.
In order to determine a particle's charge and momentum, a large dipole magnet with an integrated field of $\approx 4\,\mathrm{T\,m}$ bends charged particles in their path through the detector. The silicon-strip upstream tracker (UT) is positioned before, or upstream of, the magnet, and the scintillating fiber (SciFi) tracker is positioned downstream of the magnet. By exploiting the stray magnetic field between the VELO and UT, a preliminary momentum estimate with 15% precision can be performed to speed up track matching with SciFi hits. The UT also provides tracking for particles decaying after the VELO. The SciFi tracker, consisting of three layers, then provides precise tracking for particles passing through the detector and accurately determines their momentum using the full extent of the magnetic field

2.1.2 Particle identification system

The particle identifications (PID) system serves to supply information to help identify what type of particle created a track.
The two ring imaging Cherenkov detectors (RICH1 & RICH2) can distinguish between different hadrons. By exploiting the dependence of the Cherenkov angle on the velocity of a particle and using the measured momentum, the particle's mass can be approximated. RICH1 is filled with C_4F_10 gas and is effective in a momentum range from 2.6 to 60 GeV[1]. It is situated between the VELO and UT. Figure 2.2 shows the relation between the Cherenkov angle and momentum of particles in RICH1. Due to their higher mass and thus lower velocity at the same momentum, kaons and protons require a higher momentum to exceed the emission threshold for Cherenkov radiation. The RICH2 detector contains CF_4 and is effective for a higher momentum range from 15 to 100 GeV. It is located downstream of the SciFi tracker.
Further downstream are the calorimeter systems, which initiate particle showers and measure the deposited energy. The electromagnetic calorimeter (ECAL) generates electromagnetic showers and measures the energy of photons and electrons. The hadronic calorimeter (HCAL) generates hadronic showers and measures the energy of any heavier particles or hadrons.

[1]This thesis uses natural units with the convention $c = 1$.

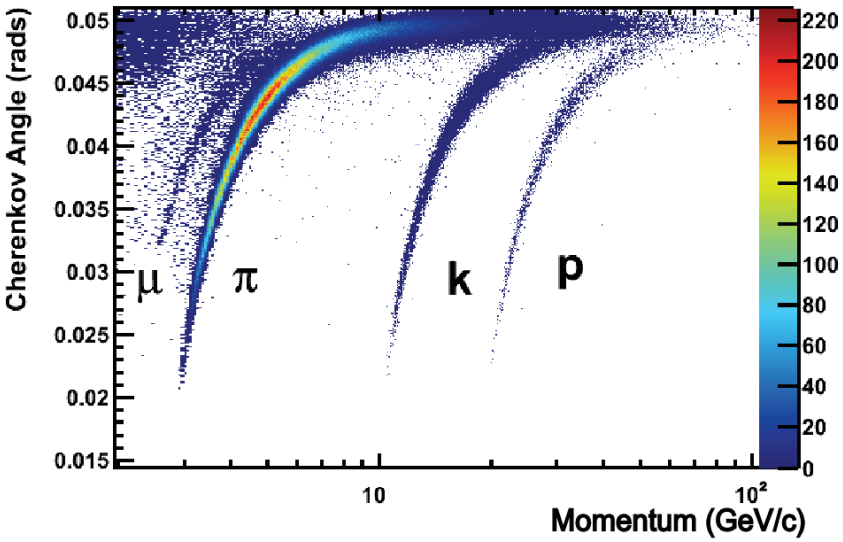

Figure 2.2: Reconstructed emission angle of Cerenkov radiation in relation to measured track momentum. The different particles (muon, pion, kaon, proton) follow different curves[9].

Muons are minimum ionizing particles, meaning they lose a minimal amount of energy when passing through material and are not stopped by the calorimeters. The muon chambers (M2-M5), located further downstream of the calorimeters identify these and provide further tracking.

2.1.3 Time of flight detector

Due to the expected increase in luminosity in in Run 5 and onwards, after Upgrade II taking place from 2033 to 2035, the VELO will be required to detect tracks with a timing precision of 20 ps or less to be able to discern tracks from different collisions. By installing a new detector with similar timing precision further downstream, the time of flight of tracks could be determined. For this purpose, the detector, Time Of internally Reflected CHerenkov light (TORCH)[10], is proposed. It would be installed either directly before or after RICH2 and aims to achieve a time resolution of 13 ps through the detection of Cherenkov photons emmited in quartz bars.
By precisely measuring the time of flight for particles to traverse from the VELO to TORCH, the particle's velocity can be determined. Together with the momentum measurement this would provide a mass measurement which can be used to identify the particles. The TORCH detector would be effective within a momentum range of 2 - 10 GeV, which is especially useful for kaons and protons, as they are difficult to distinguish in this range with the current PID systems.

2.1.4 Monte Carlo simulation

This study uses simulated data generated through Monte Carlo (MC) simulations. By using simulated data, information which would otherwise be unknown or unclear in the resulting data can be retained. This is for example information about the type of particle a track consists

of or what kind of decay a track originates from. This information can then be used for example to filter out backgrounds or to look at specific types of tracks or events, that would otherwise be indistinguishable from the rest.
At LHCb, MC simulations are generated using the Gauss framework [11], which combines several packages in order to create a complete simulation. First the Pythia [12] package randomly generates the proton-proton collisions. Then the EvtGen [13] package simulates the hadronisation and decay of particles generated in the collision and the Geant4 toolkit [14] simulates the interactions of the particles with the detector using a computer model of the LHCb detector. Finally the Boole framework [15] simulates the detector signals and electronic hardware, resulting in data in the same format as real data, which can then be processed by the same software.

2.2 Machine learning

This study employs supervised machine learning to analyse data. For supervised machine learning, a machine learning model is optimized (trained) on a dataset that is labelled according the function or pattern it is intended to learn. After training has completed, the model is expected to be able to analyse new data and predict the correct labels as accurately as possible.

2.2.1 Decision tree classifier

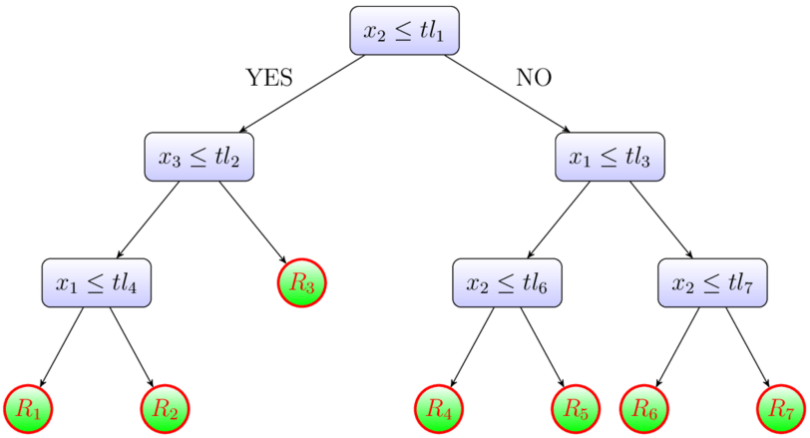

Figure 2.3: Schematic of a decision tree with variables x_i, threshold levels tl_i and end nodes R_i[16].

A simple machine learning model, which is used in this study, is the decision tree classifier [17]. A decision tree (DT) uses a tree like structure to classify data into different subsets. An example can be seen in figure 2.3. Starting with the full dataset, the data is split into two subsets based on whether the variable x_i is smaller than the threshold tl_j. This is then repeated iteratively with multiple variables and thresholds until a predefined depth, purity or subset size is reached. In this example the tree has a depth of 3 and 7 subsets. A split in the tree is also called a node and the final subsets are called end nodes. A sequence of splits in the tree from the start to an end node is called a path.
The goal of the DT training is to find the optimal thresholds to achieve the purest classification of the data, so the ratio of elements with a certain label within the subset has to be the highest. Training is performed by choosing the optimal variables and splits in each node so the average impurity of the subsets weighted by the number of elements within the subset is minimised. To measure the impurity of a subset, the commonly used Gini impurity measure is used in this study. For a given node m with the subset Q_m, where p_{mk} represents the probability of choosing an element with label k within the subset, the Gini impurity is defined by equation 2.2.1.

$$\text{Gini}(Q_m) = \sum_k p_{mk}(1 - p_{mk}) \tag{2.2.1}$$

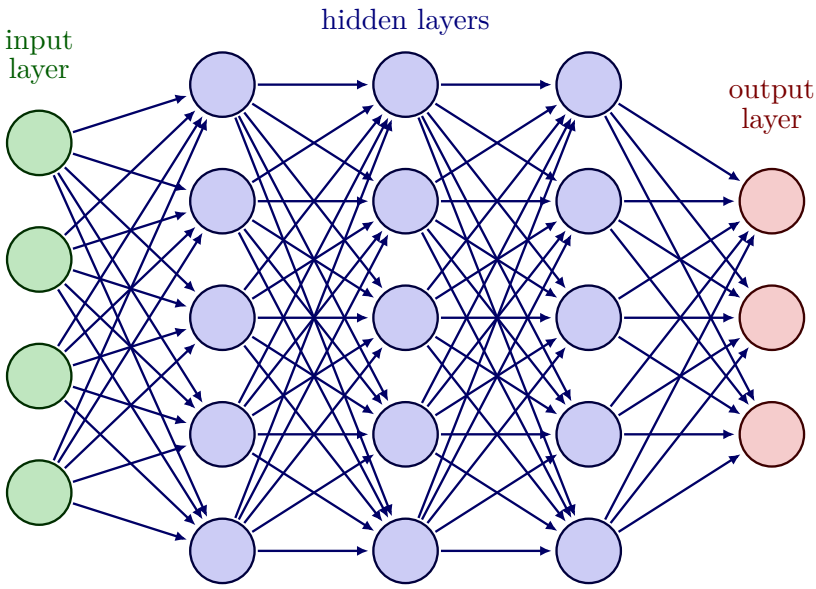

Figure 2.4: Schematic of a NN, circles are nodes and a vertical array of nodes is a layer [18].

2.2.2 Neural networks

A neural network (NN) [19] is a model inspired by biological neurons and can contain a large amount of parameters. For this study a feed forward neural network is used, which consists of so-called layers and nodes. The number of nodes in a layer is the size of the layer and the setup of layers and nodes is called the network architecture. An example schematic for a NN architecture can be seen in figure 2.4. Starting with the input layer (on the left), each node corresponds to an input variable from the data, giving an input vector x_0. This is then passed onto each node of the next layer, which transforms it to a value $x_{1,i}$ and together the entire layer generates the new output vector x_1. This is then repeated with the next layer and so on, until the output layer is reached, which returns the output of the NN. The layers in between the input and output layers are called hidden layers.
The transformation $x_{(n-1)} \rightarrow x_n$ of a layer at position n can be represented with a weights matrix $W_{(n)}$ and a bias vector $b_{(n)}$ with equation 2.2.2:

$$x_{(n)} = W_{(n)} x_{(n-1)} + b_{(n)} \tag{2.2.2}$$

The transformation in equation 2.2.2 is, however, a special type of transformation called an affine transformation. A property of affine transformations is that chaining these together, as in combining multiple layers, results in an affine transformation again, so adding more layers does not do anything, which limits the capabilities of the network to model complex functions or patterns. To circumvent this and improve the functionality of the NN, a nonlinearity has to be introduced to the model. This can be done by applying a nonlinear function $f(x)$, called activation function, to the output of a layer, given in equation 2.2.3.

$$x_{(n)} = f\left(W_{(n)} x_{(n-1)} + b_{(n)}\right) \tag{2.2.3}$$

In the case of this study, the commonly used exponential linear unit (ELU) function (applied element wise $z = x_i$) is used as defined in equation 2.2.4. The parameter α is set to the default value 1.

$$\mathrm{ELU}(z) = \begin{cases} z & \text{if } z > 0 \\ \alpha(e^z - 1) & \text{if } z \leq 0 \end{cases} \tag{2.2.4}$$

The sigmoid function is applied to the output layer to transform the output to a value between 0 and 1.

$$\mathrm{sigmoid}(z) = \frac{1}{1 + e^{-z}} \tag{2.2.5}$$

With the activation function applied, increasing the number of layers will enhance the neural network's ability to model deeper, increasing the level of abstraction. Additionally, different layers can have different widths and increasing the amount of nodes per layer enables the model to recognise more complex patterns and features within the dataset. However, the number of free parameters depends on the number of nodes, so increasing the size of the NN also means increasing the training time and the likelihood of overfitting, so a good balance has to be found.
To train the neural network a method called back-propagation [19] is used. First, the predictions and their performance are calculated. The performance is determined through a metric called loss, which compares the predictions with the expected result. Then, in a backwards pass through the network, the loss with respect to each weight from the weights matrices is calculated and the weights are then adjusted to reduce the loss. How strongly the weights are adjusted is determined by a parameter called learning rate.
Especially for large datasets, this becomes very computationally expensive, which is why conventionally the data is randomly split up into small samples called minibatches with a parameter batch size, and the loss averaged over the minibatches. These parameters, that help define the network are also called hyper parameters. A complete pass of the dataset during training is considered an epoch. The process is then repeated until the network doesn't improve anymore or stopping conditions are met.
The loss function used in this study is the binary cross entropy loss (BCELoss [20]) function, a commonly used loss function, which is specifically designed for problems with two choices (binary) and well suited for networks that output probabilities. It is given in equation 2.2.6 with the label or target y, the prediction x, and the batch size N.

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} \ell_n \qquad \ell_n(x,y) = -\left[y_n \log x_n + (1 - y_n) \log(1 - x_n)\right] \tag{2.2.6}$$

A common problem with neural networks is overfitting. Overfitting occurs when the model learns structures present in the dataset that do not, however, reflect the correct model. This occur, for example, due to spiking around statistical fluctuations and is usually observed when measuring the performance of the NN on an independent dataset. Such an independent dataset is usually obtained by splitting the original dataset (here with an 80:20 ratio) into a training sample and an independent validation sample.

The technique employed in this thesis to prevent overfitting is called early stopping. The network is trained on the training sample as before, however, the loss is also calculated for the validation set. Once the validation loss stops reducing, while the training loss continues improving for multiple epochs, overfitting is apparent. The training is then stopped and the network from the epoch with the lowest validation loss is taken as the final model.
For classification tasks, where neural networks have to decide between two choices (positive and negative), the output is usually structured to be a number between 0 and 1, and a threshold value to determine what choice is made, with 1 corresponding to positive and 0 corresponding to negative. A common way to graphically visualize the performance of such a NN is through receiver operating characteristic (ROC) curves, an example of which can be seen in figure 2.5. In ROC curves the true positive rate (the rate of positive decisions that are also correct) is plotted against the false positive rate (the rate of positive decisions that are incorrect) for each threshold value between 0 and 1. The amount of area beneath the curve amounts to the performance of the model and a flat diagonal line indicates the model has no statistical power (random guessing).

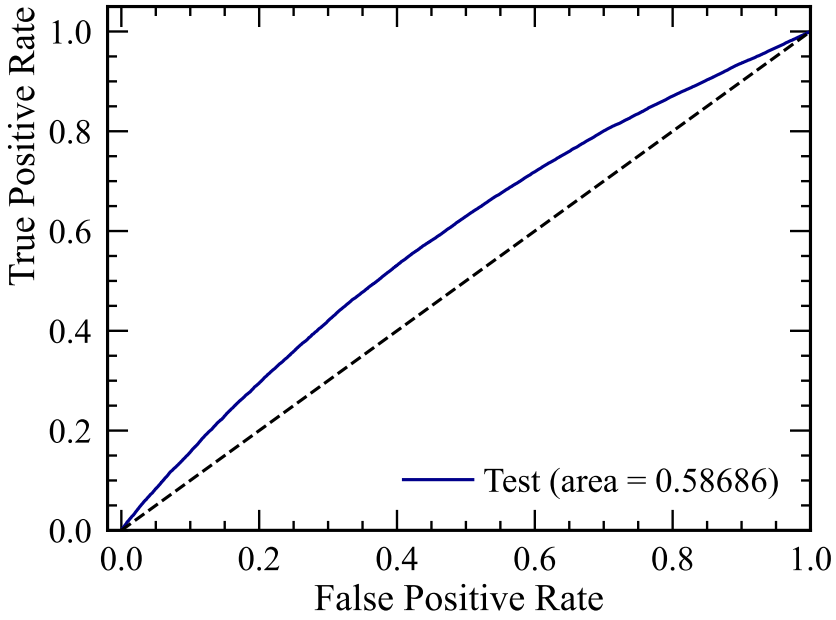

Figure 2.5: Example of a ROC curve in blue with the dotted line indicating a ROC curve with no statistical power.

2.3 Standard model

The Standard Model of particle physics [1] is a widely accepted model, able to describe three of the four fundamental physical interactions: electromagnetism, weak interaction and strong interaction.
An overview of the elementary particles in the Standard Model can be seen in figure 2.6.
The standard model defines 12 fundamental fermions with spin $1/2$, subdivided into 6 leptons and 6 quarks. These are split into 3 generations, each generation containing a charged lepton, its corresponding neutrino, an up type quark with charge $-1/3$ and a down type quark with charge $-2/3$. The term flavour, which is used throughout this thesis, generally refers to the type of quark or lepton. Additionally a corresponding antiparticle with opposite charge exists for every fundamental fermion.
Together these particles combine to form all known observable matter. Quarks cannot remain isolated due to the strong interaction, known as colour confinement, and quickly combine with other quarks in a process called hadronisation. The resulting particles are called hadrons and are typically either a meson, consisting of a quark and an antiquark pair, or a baryon consisting of 3 quarks or 3 antiquarks.
Interactions between these particles are mediated by the gauge bosons with spin 1. The photon, γ, is massless and neutrally charged and mediates the electromagnetic interaction between charged particles, described by quantum electrodynamics (QED). The quarks posses an additional quantum number called colour charge (red, green, and blue) which together with the massless gluon g are responsible for the strong interaction. The strong interaction is described by quantum chromodynamics (QCD). Lastly, the weak interaction is mediated by the charged $W^{\pm}$ bosons for the weak charged current and the Z boson for the weak neutral current. These bosons have a mass of around 80 GeV and 91 GeV [21] respectively. In addition to these gauge bosons, the Higgs boson with a spin of 0 provides the elementary particles with their mass through the Higgs mechanism[1].

2.3.1 Flavour physics

Flavour is a descriptor for quark types, however, it can be broadened in the sense that hadrons, particles made up of multiple quarks, also carry a flavour. Determining a hadron's flavour means determining its quark content.
The b quark is of special interest to the LHCb experiement because it plays a crucial role in probing the standard model and also allows searching for new physics. In the case of proton-proton collisions, like at the LHCb experiment, the $b\bar{b}$ quark pair is usually generated with excess energy, which drives the quarks apart. Due to colour confinement, the quarks are still bound by the strong force and increasing amounts of energy are stored in the field between the quarks as they move apart. Unlike electromagnetism, the force increases with distance until there is enough energy to spontaneously create new quark-antiquark pairs. The gluons are fragmented in this manner until the quarks involved have a low enough energy to combine into bound

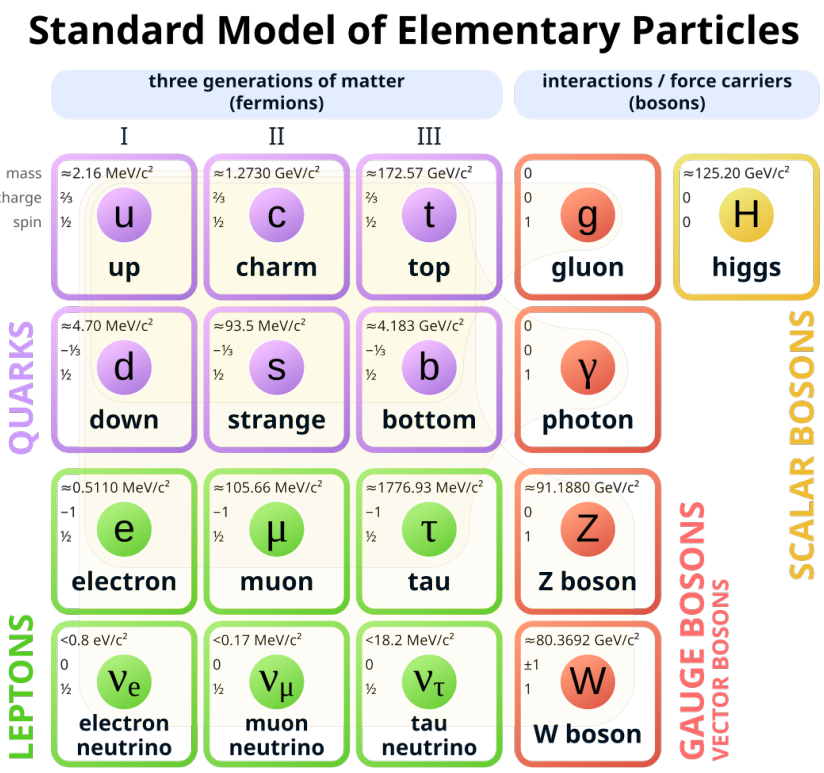

Figure 2.6: Elementary particles of the standard model with mass, charge and spin [22].

hadrons. This is a process known as fragmentation and hadronisation and is illustrated in figure 2.7.

2.3.2 Neutral meson oscillation

Flavour is conserved in the electromagnetic and strong interaction, however the weak interaction allows for quarks to change flavour (quark mixing), through the W boson. The complex coefficients with which this mixing can occur are summarized in the Cabibbo-Kobayashi-Maskawa (CKM) matrix, which is a unitary 3×3 matrix, where $|V_{ij}|^2$ of a matrix element is the probability for a down type quark of the ith generation to transition into an up type quark of the jth generation via the W boson. The complex phases of the coefficients explain CP violation within the weak interaction. CP symmetry is the combined symmetry of charge conjugation (C), so the inversion of charge, and parity transformation (P), the inversion of spacial coordinates. This symmetry, is observed to be broken in the weak interaction and could prove crucial for explaining the matter-antimatter imbalance in the universe.

Due to quark mixing, neutral mesons are able to transition between their particle and antiparticle state through so called box diagrams at lowest order, shown in figure 2.8. Through the exchange of two W bosons, the quarks within the meson are able to switch their respective states and the meson turns into an antimeson or vice versa. This process is referred to as neutral meson oscillation [4] and occurs with a certain frequency, which can be measured and gives

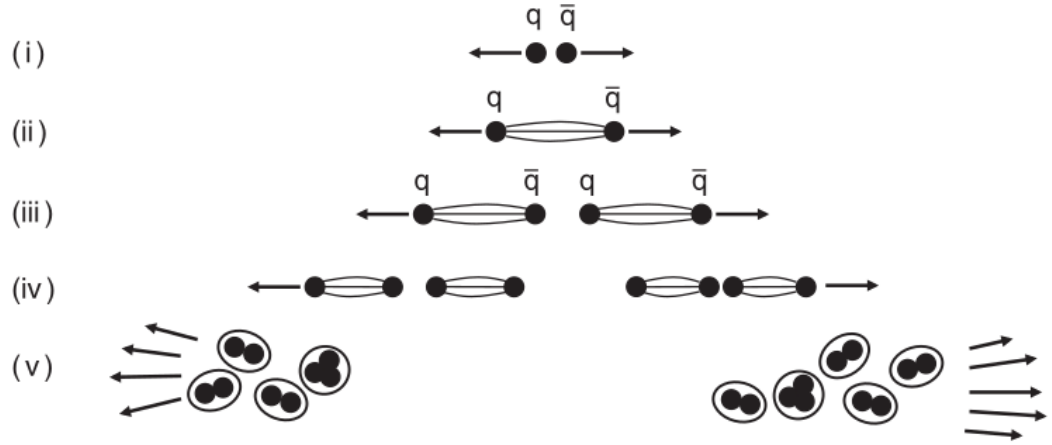

Figure 2.7: Illustrative example of the steps in the hadronisation process [1].

important insights into the CKM matrix. As an example, the current measurement of the B_s^0 meson oscillation performed by the LHCb collaboration is at $17.7656 \pm 0.0057\,\mathrm{ps}^{-1}$ [4].
This study focuses on the $B^0(\bar{b}d)$ and $B_s^0(\bar{b}s)$ meson. The reconstructed B meson and its decay products in an event are referred to as the signal in this study. The dominant Feynman diagrams of B^0 oscillation can be seen in figure 2.8.

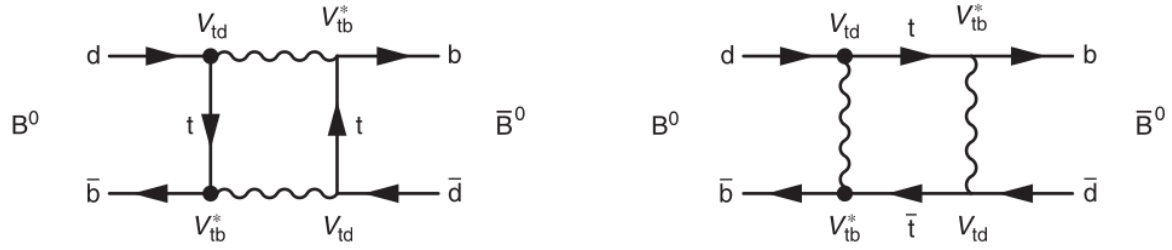

Figure 2.8: Leading order Feynman diagrams of B^0 meson oscillation, dominated by the t quark. B_s^0 oscillations are analogous with a s quark instead of a d quark [1].

2.4 Flavour tagging

Measuring neutral meson oscillations requires a precise reconstruction of the decay time of the signal particle, as well as the identification of its initial and final flavour states. In flavour specific decays the final state of the meson can be determined from the charges of the decay products, however, this does not work for the initial state. The initial state has to be determined by looking at other particles, which are created in the B meson's hadronisation process. The process of obtaining initial state information is called flavour tagging.
For every event, a flavour tagging algorithm will give a predicted flavour, the tagging decision d and a predicted mistag η, which is an estimate for the probability of the tagging decision being incorrect. The tagging decision is defined as: $d = +1$: The signal has a $\bar{b}$; $d = -1$: The signal has

a b and $d = 0$ if no decision can be made. The predicted mistag is defined as: $\eta \in [0,0.5]$ with $\eta = 0$ corresponding to an exact prediction and $\eta = 0.5$ a random prediction.

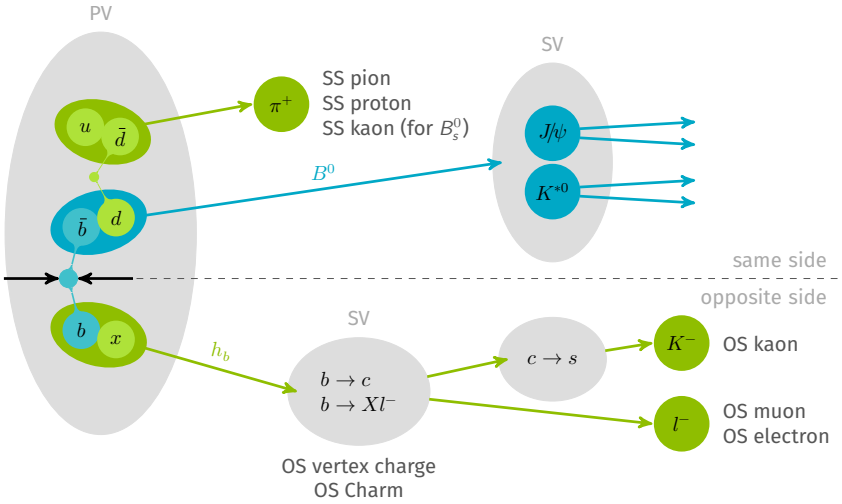

Figure 2.9: Schematic overview of the B meson hadronisation process and the different tagging algorithms. [23]

At the LHCb several algorithms called taggers are developed, each targeting a different part of the hadronisation process and the decay products. The algorithms are split into two categories: same side (SS) and opposite side (OS) decays. A schematic overview of the hadronisation process and the different algorithms can be seen in figure 2.9.
Same side decays originate from the signal B meson fragmentation, the principle of which can be seen in the green parts of the top half of the figure. In order to create the d, u or s quark in the signal, a pair has to be created so an antiquark partner has to exist and hadronise into a hadron. The algorithm attempts to identify the particle that is most likely to originate from this partner quark and bases the tagging decision on its charge.
Due to the $b\bar{b}$ pair production, another B hadron can be created in the same event with the B signal. The opposite side taggers exploit this and look for the decay particles of the partner B hadrons. Since different decay processes are involved in creating different particles, the taggers are trained on individual tagging particle types and again the tagging decision is based on the charge of the best candidate. The principle of OS taggers can be seen in the bottom half of the figure 2.9.
Of special importance for this study is the OS kaon (OSK) tagger and the potential OS proton (OSP) tagger. The OS kaon tagger is characterised by the weak decay chain of $b \to c \to s$, while the other quark in the B meson stays the same. The kaon charge then corresponds with the charge of the OS B meson at the time of its decay, which, if does not oscillate, corresponds with the initial state of the signal B meson.
For the OS proton tagger, which is the main focus of this thesis, an assumption can be made that these protons are likely to originate from Λ_b^0 baryons via the decay $\Lambda_b^0(bud) \to \Lambda_c^+(cud) \to p(uud)$, due to the high branching fraction of $\Lambda_b^0 \to \Lambda_c^+$anything and $\lambda_c^+ \to p$anything [21]. In this case the proton charge would determine the flavour of the Λ_b^0. A schematic example for the opposite

side proton tagger can be seen in figure 2.10.

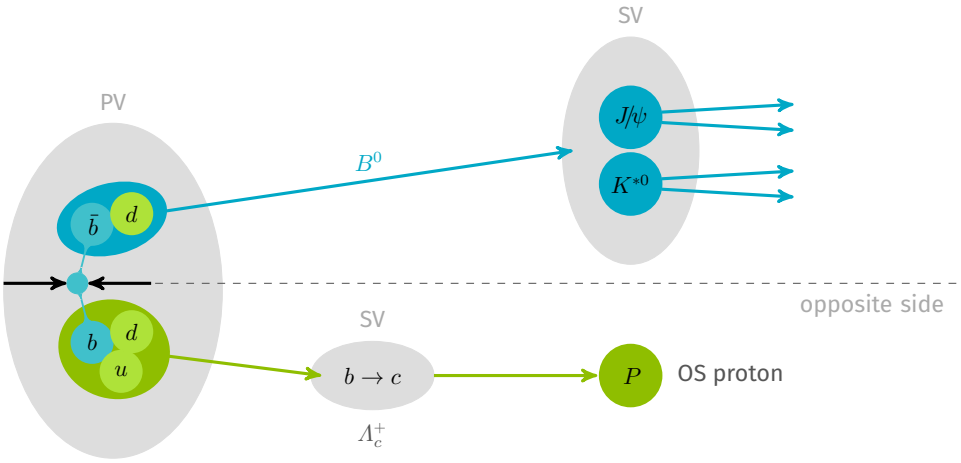

Figure 2.10: Schematic overview of the opposite side proton tagger with a Λ_b^0 decay [23]

.

To achieve the best results, the different taggers are all evaluated on an event and their individual results combined. For this the taggers need to be calibrated first, since it is unclear whether the algorithms return correctly scaled mistag predictions. This is done by fitting a linear transformation from the predicted mistag η to the measured mistag ω of the dataset as in equation 2.4.1. To determine ω, similar values of η are grouped together and the probability for a tagging decision within that group being incorrect is then ω.

$$\omega = \alpha \cdot \eta + c \tag{2.4.1}$$

The calibrated taggers can then be combined to form a combined mistag η_{comb} and combined tagging decision d_{comb} which are based on the probability P_b and $P_{\bar{b}}$ for the signal candidate to contain a b $\bar{b}$ respectively. For tagging candidates i with tagging decision d_i and mistag ω_i the combined decision and probability can be calculated via equations 2.4.2 and 2.4.3.

$$d_{\text{comb}} = \text{sign}(P_{\bar{b}} - P_b); \qquad \eta_{\text{comb}} = 1 - \max(P_b, P_{\bar{b}}) \tag{2.4.2}$$

$$p_b = \prod_i \left(\frac{1 + d_i}{2} - d_i[1 - \omega_i] \right); \qquad p_{\bar{b}} = \prod_i \left(\frac{1 - d_i}{2} + d_i[1 - \omega_i] \right) \tag{2.4.3}$$

Finally, two metrics to measure the performance of the tagging algorithm are defined in equation 2.4.4.

$$\varepsilon_{\text{tag}} = \frac{N_{\text{tag}}}{N_{\text{tot}}} \qquad \omega = \frac{N_{\text{false}}}{N_{\text{tag}}} \tag{2.4.4}$$

The tagging efficiency ε_{tag} is the ratio of the number of tagged events N_{tag} and the total number of events N_{tot}, the average mistag ω is the ratio of the number of falsely tagged events N_{false} and the number of tagged events. These can be combined to give the effective tagging efficiency or tagging power:

$$\varepsilon_{\text{eff}} = \varepsilon_{\text{tag}}(1 - 2\omega)^2 \tag{2.4.5}$$

A comparison of the tagging power of LHCb algorithms and several published analyses across different experiments can be seen in figure 2.11. The OS kaon tagger has a tagging power of 1.4%.

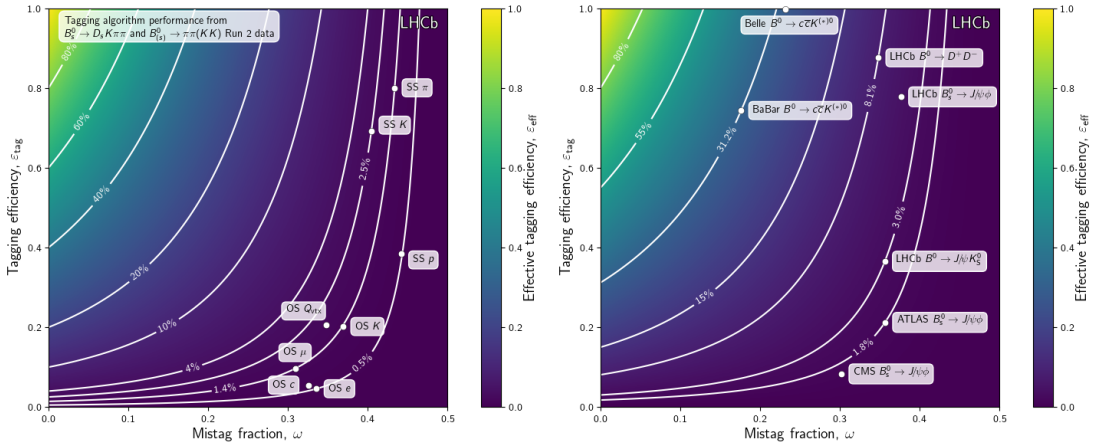

Figure 2.11: The tagging power ε_{eff} displayed in the ω-ε_{tag}-plane for different LHCb flavour tagging algorithms in Run II (left) and for multiple published analyses across different experiments (right) [24].

2.4.1 Tagging algorithm

This study uses a conventional approach at flavour tagging based on the preselection of candidate tracks within an event and analysis of these [5].
First, the preselection is performed via predefined cuts on different variables. Since it is performed on the entire dataset, a decision tree is chosen for its simplicity and quick execution time. The DT, as explained in chapter 2.2.1, classifies the data into categories relevant to the tagger (e.g. identifying OS kaons for an OS kaon tagger, etc.).
Then a neural network analyses each selected track and gives an output x from 0 to 1 in order to predict the mistag. The predicted mistag is the output of the NN $\eta = x$, however, if it is above 0.5, it is flipped via $\eta = 1 - x$ and the tagging decision of the track is also inverted. This ensures that each selected track has a tagging decision $d = \pm 1$ and a predicted mistag $\eta \in [0,0.5]$. Tracks that are not preselected are given a tagging decision $d = 0$ and are not further analysed by the NN.
In order to train the decision tree and the neural network, labelled data is required, which means that simulated data has to be used, which contains information about the tracks origin and true particle ID.

Chapter 3

Study of the opposite side proton tagger

3.1 Brief overview of the analysis

The aim of this study is testing the impact of the improved particle identification provided by TORCH on the potential development of an opposite side proton tagger. By including a functional OSP tagger, the current combined tagging algorithms would benefit from additional tagging efficiency, due to the inclusion of OS protons, and an improved tagging predicition for events selected by the OSP tagger.
In the current run an OSP tagger is not considered feasible, due to the low amount of opposite side protons and their bad separability from kaons. Since TORCH would improve PID, specifically selecting protons should become better, which could be sufficient for a new OSP tagger.
It is assumed that most OS protons originate from Λ_b^0 baryons, which have a high likelihood of decaying to protons. In this case the proton charge and tagging decision are the same. However, since the b quark is more likely to hadronise into a meson, rarer decay modes, like $B^+ \to pnX$ or $B^+ \to p\bar{p}X$ [21] could contribute a significant amount, which has to be investigated.

3.1.1 Truth information

In order to filter for opposite side protons a simulated dataset is required. The simulation can save information from steps in the simulation process that would otherwise be unclear or unknown. This truth information enables the dataset to be labelled for supervised machine learning.
Two truth information variables used in this study are *TRUEID* and *Origin_Flag*.
TRUEID employs the Monte Carlo numbering scheme [25] to encode what type of particle is responsible for creating the track. A proton corresponds to the number 2112.
The variable *Origin_Flag* stores information relevant to flavour tagging about the origin of a track. This can roughly be divided into 5 categories: Tracks originating from the signal B decay, tracks originating from the same side fragmentation, tracks originating from the opposite side B decay, tracks originating from the opposite side fragmentation and tracks from other sources

or background tracks. The latter category includes ghost tracks, which are falsely reconstructed tracks with no corresponding particles, and tracks not associated with the signal or OS B decay.

3.1.2 The dataset

This study uses an available dataset generated from Monte Carlo simulated events containing $B^+ \to J/\psi K^+$ decays under 2024 LHCb conditions. This means all events contain a reconstructed $B^+ \to K/\psi K^+$ (or charge conjugated) decay for the signal, while the OS B could be anything, which aims to be similar to real measurements where such decays are reconstructed.
Optimally simulations in Run 5 conditions should be used, however, these are not available, so the improved PID, which is the focus of this study, will be implemented with the help of the *TRUEID* variable. The signal decay type is not important, as long as it contains a B meson, since this study is on the opposite side proton tagger. This dataset is chosen due to it being the largest available with 1209157 events. The composition of the dataset is analysed using truth information and given in table 3.1.

Track type	Count	% total
Signal B decay	3646759	2.9%
SS fragmentation	3809378	3.0%
OS B decay	2919514	2.3%
OS fragmentation	2763866	2.2%
other / background	114302118	89.7%
total	127441635	100%

Particle	Count	% total	% in OS
Kaons	13541754	10.6%	
OS kaons	481207	0.4%	16.5%
Protons	9483435	7.4%	
OS protons	81027	0.1%	2.8%

Table 3.1: Composition of the simulated $B^+ \to J/\psi K^+$ dataset with respect to the origin categories (left) and kaon & proton fractions out of all tracks (right).

In total there are 127 million tracks, or on average around 105 tracks per event. Background tracks and tracks from other sources make up around 90% of the dataset with the other four categories having approximately equal shares. The tracks in the category OS fragmentation are from fragmentation of the opposite side B, however in the context of this thesis, OS only refers to OS B decay. There are around 481 thousand kaons originating from the OS B decay, accounting for approximately 0.4% of the entire dataset. In contrast there are around 81 thousand OS protons, which make up less than 0.1% of the dataset, roughly 6 times fewer than the kaons. As such, the potential OSP tagger has fewer tagging candidates and is expected to have a lower tagging efficiency. This also shows how, with a difficult distinction between kaons and protons, an OSP tagger algorithm could struggle to remove enough kaons from the dataset. Figure 3.1 shows the momentum distribution of protons in the dataset with and without a PIDP > 5 cut, where PIDP is the variable representative of the proton identification (see more in appendix B). A clear drop in efficiency is visible towards lower momenta.

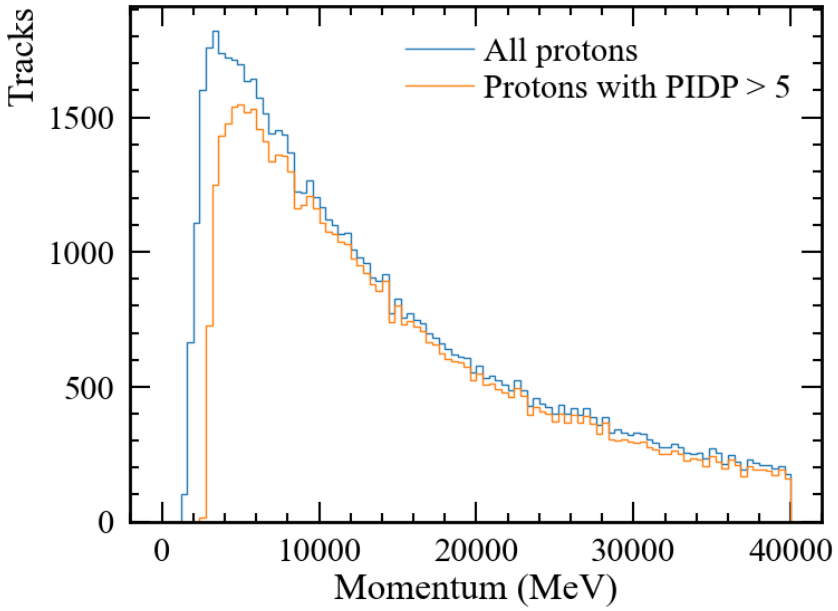

Figure 3.1: Momentum distribution of protons with and without PIDP > 5 cut.

3.2 Analysis of OSP tracks

As mentioned in section 3.1, the Λ_b^0 decays are expected to account for most OS protons, however, rare B meson decays cannot be ignored. In order to better understand the dataset and to verify the assumptions, an analysis is done to measure how large these contributions are.

3.2.1 Investigating the proton origin

With the *Origin_Flag* variable it is possible to ascertain whether an OSProton stems either from an unexcited B meson (B^0, B^+, B_s^0, B_c^+ and antiparticle partners) or from any other hadron containing a b quark, including Λ_b^0.
Figure 3.2 shows the correlation between opposite side protons from these two different origins and the charge of the signal $B^\pm$ meson. The Other B decays, which contain the $\Lambda_b^0 decays$ and only have a small opposite charge contribution, actually amount to less than one third of all opposite side protons, with the rest coming from the unexcited B meson decays. Additionally, both types of origins have opposing charge correlations, meaning they mostly cancel out (seen in the bars on the right), leading to a potential high mistag. The flavour tagging algorithm used in this study, only looks at tracks individually to assign the tagging decision and mistag estimate. Since it is difficult to determine the mother particle of a single track, this could become a challenge for the algorithm.

3.2.2 Removing multi OS proton events

Besides the opposite charge correlation being dominant for protons from unexcited B decays, there is also a large contribution for the same charge correlation. To reduce the height of both

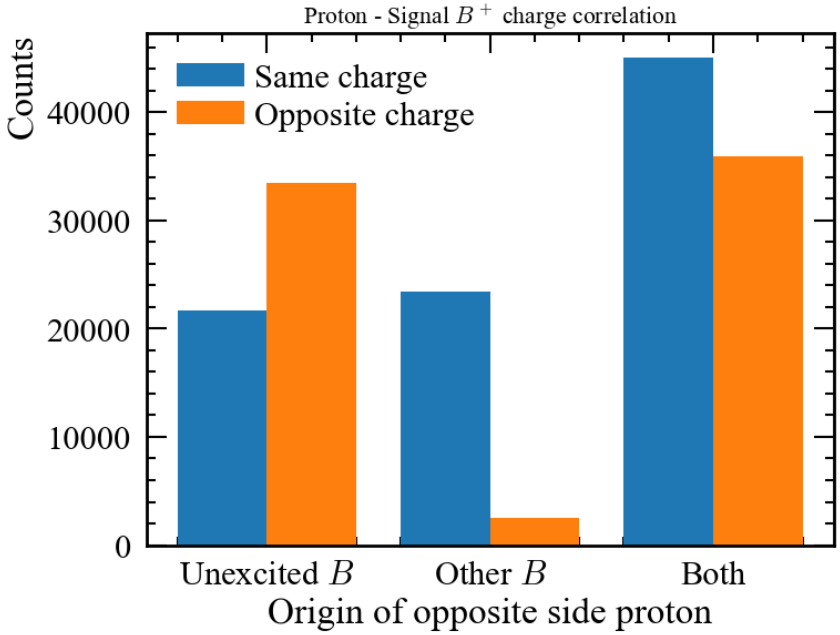

Figure 3.2: Number of OS protons for different origins with same charge and opposite charge correlation to the signal $B^{\pm}$ charge.

bars, events with multiple OS protons could be filtered out, since decays like $B \to p\bar{p}X$ do not retain flavour information, as there is no clear choice between which of the two protons a tagger would choose.

Looking at the pure OS proton sample again, events with 2 or more protons are removed and the new reduced numbers shown in figure 3.3 which is plotted on top of figure 3.2. An equal reduction in both charge correlations can be seen, with the majority of the removed protons originating from unexcited B mesons.

However, a filter like this is difficult to perform in real data, since the preselection does not select all OS protons, potentially missing events that should be removed, and also selecting a lot of background track, meaning events with only a single proton are removed due to an extra background track being selected. Additionally, this idea does not combat the opposing correlations between the two different types of origins and does not help identify these origins either, meaning further investigations have to be made.

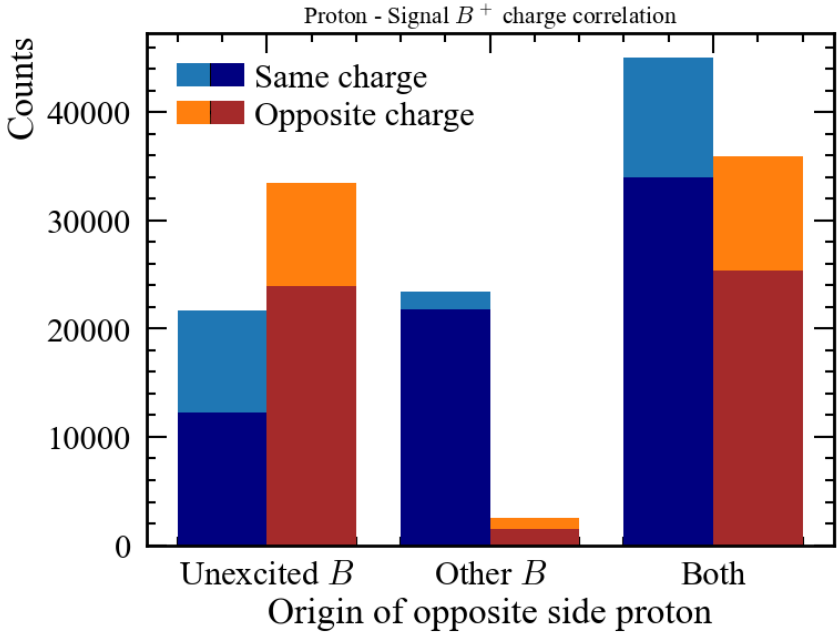

Figure 3.3: Reduced counts of protons in darker colour overlaid over the same bars from figure 3.2

3.3 The opposite side proton tagger

3.3.1 Estimated tagging power

An approximation for how well a tagger might perform is by simply calculating the tagging power based on the number of OSProtons and their charge correlation. Opposite side protons are chosen with the truth variables and then their charge is compared to that of the signal $B^{\pm}$ meson in the same event.
In total there are about 1.2 million events of which 70 thousand contain an OSP and a total of 81 thousand individual OSP tracks. Of these tracks, 45 thousand have the same charge as the signal B and 36 thousand the opposite charge. This lets us fill out the equations 2.4.4 & 2.4.5 to get an estimate for the tagging power an algorithm might be able to achieve.

$$\varepsilon_{\text{tag}} = \frac{69972}{1209157} = 5.8\%; \qquad \omega = \frac{35952}{81027} = 44.4\%; \qquad \varepsilon_{\text{eff}} = 0.058 \cdot (1-2\cdot 0.444)^2 = 0.07\% \quad (3.3.1)$$

As can be seen here, the tagging power is very low compared to any of the preexisting taggers in figure 2.11. Since there are not a lot of OS protons, the efficiency is quite low already but the problem with the charge correlations as mentioned in section 3.2.1 leads to a very high mistag, which together result in the very low estimated tagging power.
This can also be compared with the OS kaon:

$$\varepsilon_{\text{tag}} = \frac{351442}{1209157} = 29.1\%; \qquad \omega = \frac{132627}{351442} = 37.7\%; \qquad \varepsilon_{\text{eff}} = 0.291 \cdot (1-2\cdot 0.377)^2 = 1.75\% \quad (3.3.2)$$

The estimated OS kaon tagging power is slightly better than the preexisting OSK tagger, but approximately matches it.

3.3.2 Flavour tagger training

For this analysis, three different scenarios with differing influence from the *TRUEID* variable are explored.
First an OSP tagger for 2024 conditions, so with no truth information is trained. This is to establish a baseline for the later taggers to see if improvements are made when better particle identification is introduced.
For the next scenario a rough imitation of TORCH is used to train an OSP and OSK tagger. The dataset is split at a momentum cut of 10 GeV. For tracks with a lower momentum, the *TRUEID* variable is used, which is equivalent to perfect PID for low momentum, to simulate TORCH. Of course, the detector itself is not expected to be perfect, however, as mentioned before, simulations in Run 5 conditions are not available and a simulation of TORCH does not fit into the frame of this study. Since TORCH is expected to give good PID for low momenta, the approximation should still give valid insights. The OSK tagger serves as check to see if the program can perform well and if the momentum split does not impact it in unpredictable ways.
Lastly, as a comparison, *TRUEID* is used on the entire dataset, equivalent to perfect PID for all tracks, and an OSP tagger is trained.

3.3.3 Decision tree training

A preselection is performed in order to significantly reduce the number of unwanted background tracks. It should not only select the correct particle type but also the correct origin, so remove tracks that are unlikely to originate from the OS B decay. A decision tree has to be trained for each of the scenarios mentioned above. It has to identify tracks with the correct particles (protons or kaons) and also identify if these originate from the opposite side B decay.
This study uses the DecisionTreeClassifier module from the scikit python library [26]. Using the truth information, the dataset is labelled between the OSP (and OSK) and others and the classifier trained on all variables available (see appendix B). Then the best paths with the lowest gini impurity are selected. The preselection has to reject enough background for the NN training to recognize the OS protons, however a sufficient amount of data has to remain to have enough statistics for the NN to train on. The preselections here aim to contain around 20000-25000 OS protons, which is a quarter of all OSP tracks available.
The resulting decision trees and cuts used for the preselection are shown in appendix A, with explanations for the variables given in appendix B.
The trained decision tree for the first scenario, an OSP tagger in 2024 conditions, is shown in figure 3.4. The red node is the final selected node and the preselection follows the cuts along the path there. It is to note, that the signal B decay tracks are already filtered out prior to training,

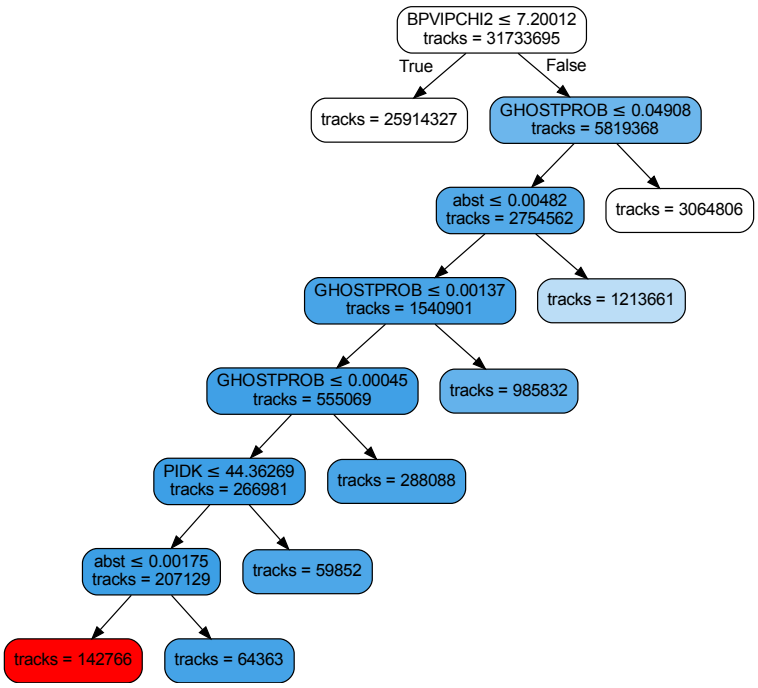

Figure 3.4: Decision tree schematic for an OSP tagger in 2024 conditions with the cuts given in each branch. Branches not leading to the final selected node are not shown. A darker blue means higher purity, with the final selected node in red.

since these are identified when reconstructing the signal B meson, and to speed up the DT training a preliminary cut of $PIDP > 5$ (see appendix B) is performed, which already removes a lot of non proton tracks. Tracks with missing values are also removed to prevent errors during training.

For the scenario with the momentum cut 3 decision trees are trained. A single DT is trained to distinguish between OS protons, OS kaons, and the rest for the tracks with momenta above 10 GeV. This DT does not have the preliminary *PIDP* cut, since that would remove kaons. Additionally, for the low momentum tracks a DT is trained individually for OS protons and OS kaons. Since the correct particles are already selected with *TRUEID*, their only job is to select the tracks originating for the opposite side B decay and not to also filter for the correct particle type. The resulting cuts are combined to give a preselection for the OSP tagger and the OSK tagger each. Lastly, for the perfect PID scenario, another DT is trained for the pure proton sample to identify the OS protons.

Statistics on the selected tracks for each preselection are given in table 3.2.

	2024	OSP with p cut	OSK with p cut	Perfect PID	Full dataset
Protons	84782	89937	4066	99931	9483435
Kaons	39963	618	361123		13541754
OS protons	22019	20704	278	26260	81027
OS kaons	22629	277	179418		481207
OSP fraction	27.2%	25.6%	0.3%	32.4%	100%
OSK fraction	4.7%	0.1%	37.3%		100%
Total	141905	91580	373727	99931	127441635

Table 3.2: Number of selected protons and kaons selected by the 4 preselections and the fractions of OSP and OSK selected out f all available.

The preselection in 2024 shows, as expected, a large kaon contribution and even contains more OS kaons than OS protons, which means a tagger with this preselection is a mixed OSP and OSK tagger rather than an OSP tagger.
With the second scenario, in which low momentum protons are selected with perfect PID, the amount of kaons falls off significantly and this problem does not exist anymore. Similarly, the OSK preselection has a low fraction of protons.
The last scenario has perfect PID so there are no kaons.
For all 3 OSP preselections, around 20-25 thousand OSP tracks are selected out of the 80 thousand tracks available.

3.3.4 Neural network training

With the preselections completed, neural networks can be trained to assign a tagging decision and predicted mistag for each track. In this study, the NN is given all 34 features available in the dataset, as listed in appendix B, as input variables. These range from kinematic variables, like the momentum of the track or angle relative the the signal B, information about the event, like the number of tracks or primary vertices, and variables for particle identification, like $PIDP$. The NN has 2 hidden layers with 8 nodes and 5 nodes respectively and a single output node, which is taken over from a previous study on OSK taggers. In order to define the model, perform training and read the output, the python package pytorch [27] is used.
The preselected data is labelled 0 or 1 via its charge correlation with the signal B meson and is then randomly split into three independent subsets: training, validation and calibration. The NN is trained on the training subset and the early stopping technique is used to combat overfitting, it is set to stop after the validation has not improved for 200 epochs.
The output is of the model transformed with the sigmoid function to give a value between 0 and 1. As a last step the output is calibrated using the python package ftcalib [28], which also computes the calibrated efficiency and tagging effectiveness of the tagger.
In order to find good values for the batch size and learning rate of the model a grid search is performed. An array for batch size (32, 48, 64, 80, 96) and for learning rates (0.0005, 0.001, 0.0025, 0.005, 0.0075, 0.01) is given and training performed for every combination of these values in a grid.

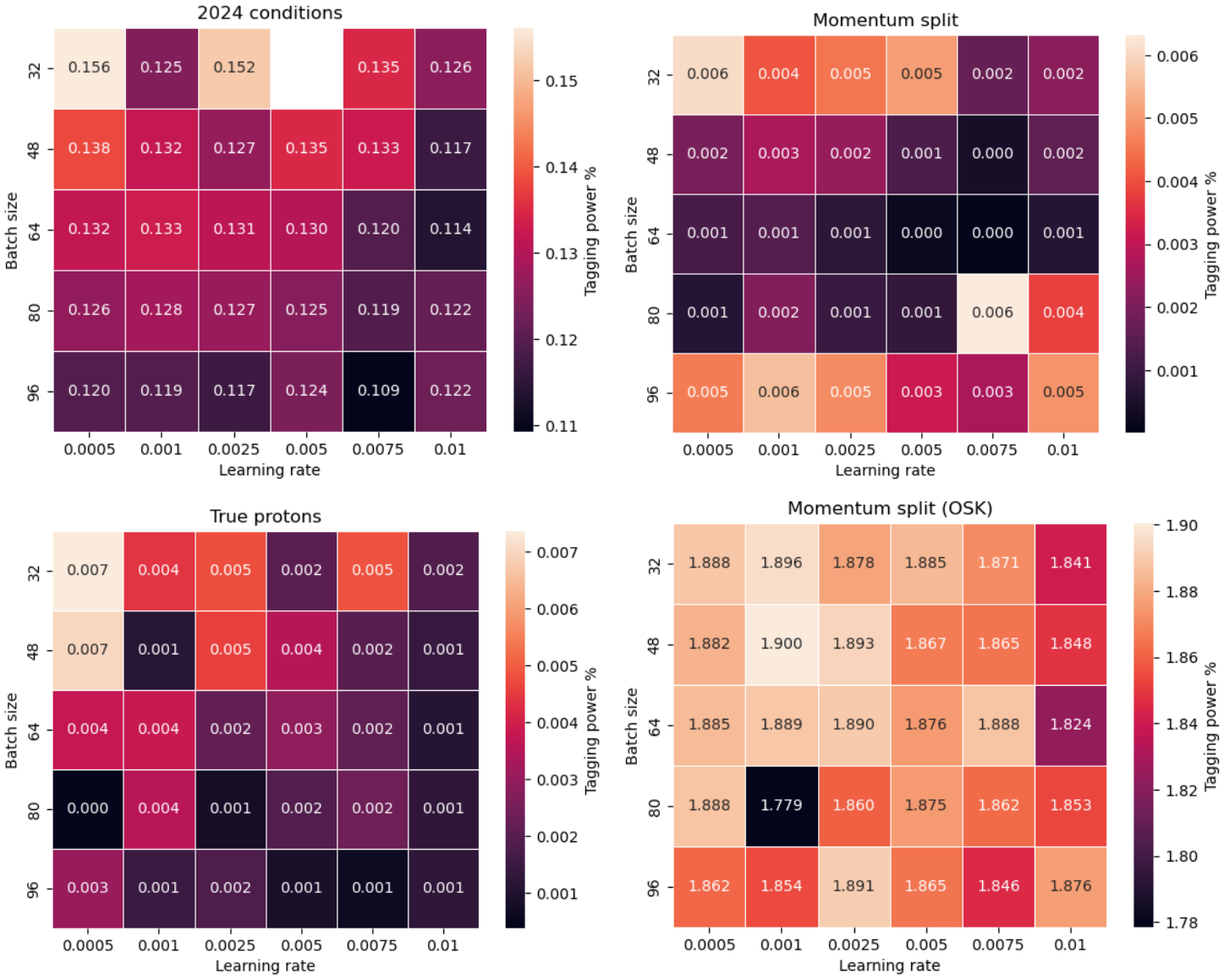

Figure 3.5: Calibrated tagging power in % for the OS proton grid searches in 2024 conditions (top left), with the momentum split selection (top right), with pure proton selection (bottom left)and the OS kaon grid search with the momentum split selection (bottom right).

	2024	OSP with p cut	Perfect PID	OSK with p cut
ε_{eff}[%]	9.3131	6.0269	7.437	23.9187
ω [%]	43.5266	48.381	48.4279	35.9062
ε_{tag} [%]	0.1561	0.0063	0.0074	1.9004

Table 3.3: Performances of the best OSP taggers in the three scenarios after calibration, as measured by tagging efficiency ε_{eff}, mistag ω and tagging power ε_{tag}.

Figure 3.5 shows the results from the grid searches on the 3 scenarios for the OS proton taggers and table 3.3 shows the tagging performance of the best tagger in each grid search. It is clear that all three algorithms performed poorly, with the best result in 2024 conditions having a tagging power of only 0.16% and the two other two being at basically zero. The grid search could go to lower parameters but limitations had to be made due to time constraints and for the second and third scenario, the performance would require a 100 fold increase to be considered good, which is not expected to happen by simply changing the hyper parameters.
The better performance of the tagger in 2024 conditions can most likely be attributed to the presence of OS kaons in the preselected sample, so this is a mixed OSP and OSK tagger rather than an OSP tagger.
As a comparison the OSK tagger is also trained. The results can be seen in figure **??**. Here the best result has a tagging power of 1.8%, which is comparable to previous OSK taggers (figure 2.11), meaning the training algorithm does not seem to be negatively impacted by use of TRUEID for the selection and is able to perform properly.
Not only do the OSP algorithms have a lower tagging efficiency, as would be expected due to the smaller contribution of protons, but the mistag, especially for the two taggers with the better proton selection, is very high compared to the OSK tagger. These numbers approximately fit the estimates from section 3.3.1.
The ROC curves, loss curves and calibration plots for the best taggers from each grid search can be seen in appendix C.
A clear difference between the OSK and OSP ROC curves and calibration plots can be seen, while the OSK curve is clearly separated from the diagonal, the OSP curves in the two improved PID scenarios are almost flat on the diagonal, meaning they have almost no statistical significance. Similarly, while the OSK calibration plot is close to the identity and has a relatively low error with spaced out points from a mistag of 0.3 to 0.5, the two OSP lines have very large errors and the points are all bunched at a mistag of 0.45 to 0.5. In both of these the OSP tagger in 2024 performs better, most likely due to the afforementioned kaon contribution.
One thing to point out here is that the NN seems to start overfitting quite early with the best validation epoch being 20 or less. It might be worth investigating if changing the architecture or other changes impact this and could result in better results. However, this does not fit into the scope of this thesis and looking at the OSK tagger performance, this does not seem to have much of an impact.

3.3.5 Neural network to separate by origin of correlation

The NNs trained in section 3.3.4 do not perform well, however this could be due to the presence of the background making the task too complex or other unknown factors. As discussed in section 3.2.1, the protons have different charge correlations based on their origin, so this might cause difficulties. In order to simplify the task of the neural network, a pure OS proton sample is used and the NN is only given the task of separating these protons by either their origin or their charge correlation with the signal B meson.

Two different architectures are used, the same as the tagger training in chapter 3.3.4 and a larger architecture with one hidden layer of 32 nodes and three more hidden layers with 16 nodes each. Since this is a test to see if the network can create visible distinction and due to time constraints, a grid search is not performed. Instead the learning rate is fixed at 0.001 and the batch size at 48, since these are the best values found in the OSK training.

The outputs of the trained neural networks can be seen in figure 3.6. There is no visible distinction in any of the four neural networks, which means there is most likely no exploitable information in the variables of single tracks. The separation of the proton origins or charge correlations requires a more complex or holistic approach beyond the analysis of individual tracks.

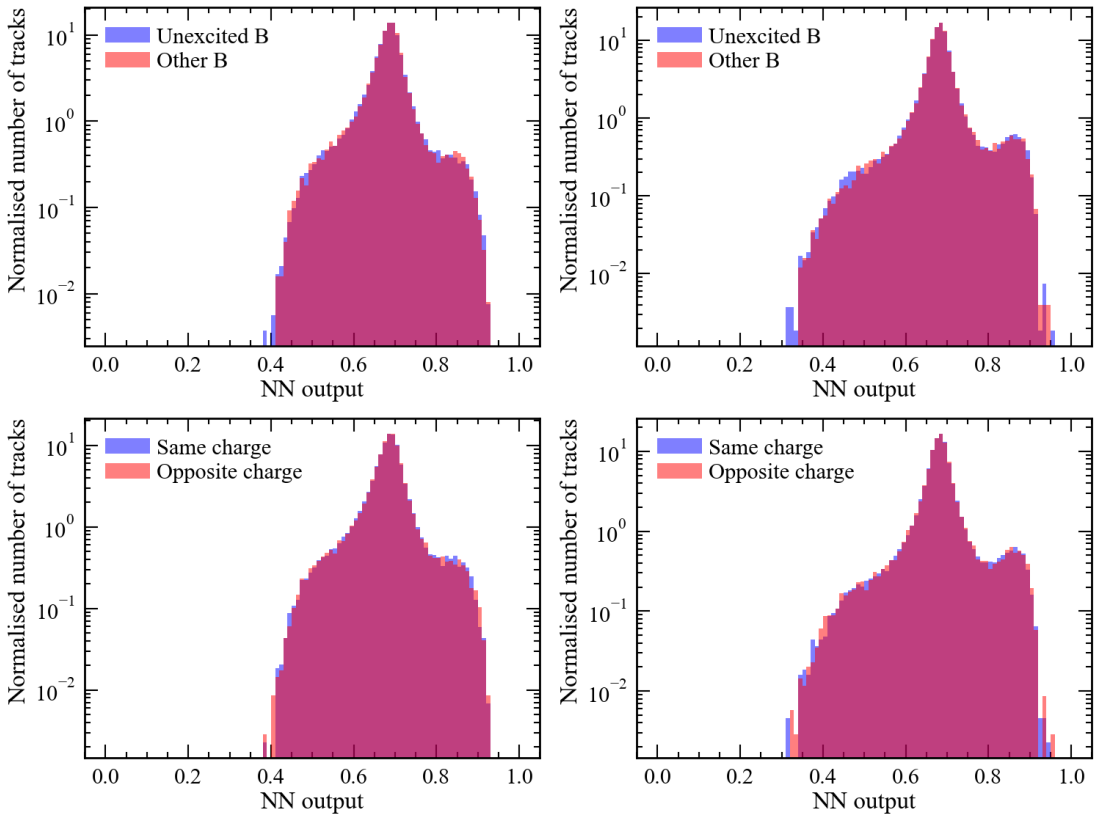

Figure 3.6: Normalised number of tracks on logarithmic scale for the neural network output, separated by the training label. Training on the origin (top) and the charge correlation (bottom) with the simple network (left) and complex network (right).

Chapter 4

Conclusion

The development of an opposite side proton tagger at the LHCb has not been considered so far due to the difficult separation between protons and kaons at low momenta. For the future Upgrade II of the LHCb detector a new time of flight detector has been proposed. It would provide better particle identification at low momenta, which is an issue for the current particle identification system. Especially proton and kaons would benefit from this due to their higher mass and thus lower velocity and the same momentum as lighter particles like pions and muons. Since the distinction between kaons and protons would improve, this might impact a potential OSP tagger.

The study presented in this thesis aims to understand the impact of TORCH on the identification of opposite side protons and whether this could enable the development of a new OSP tagger. Since simulations containing a TORCH detector are not available, this study focuses on training taggers using 2024 simulations across three different scenarios. The decision tree training in the first step shows that a perfect PID for tracks under 10 GeV momentum improves the selection of protons significantly. While keeping the total amount of selected OS protons similar, the preselection in 2024 conditions contains a comparable amount of OS kaons, whereas the addition of the improved PID removes almost all OS kaons from the selection. Assuming TORCH would have a similar PID performance, this shows it would indeed improve the OSP preselection.

However, the neural networks to determine tagging decision and mistag of the tracks perform poorly with the best results being in the scenario with the 2024 conditions, having a tagging power of 0.16%, which is almost 10 times worse than the OS kaon tagger in Run II. In the two scenarios with better PID, the taggers perform even worse with a negligible tagging power of 0.007% at best. This also leads to the conclusion, that the better tagging power in the first scenario can most likely be attributed to the large fraction of kaons in the OSP preselection.

Investigations on the origins of the OS protons and their charge correlation with the signal B meson, show that they are not dominated by Λ_b^0 decays, but actually are more likely to originate from an unexcited B meson decay. This is made worse by the fact that the dominant charge correlations between the proton and the signal B is reversed for the two origins. The protons from unexcited B mesons are more likely to have the opposite charge of the signal B meson, while the protons from Λ_b^0 and other decays tend to have the same charge. Since it is difficult to

distinguish between these protons, the charge correlations cancel each other, leading to a high estimated mistag of 44.4%. Filtering out events with multiple OS protons does not lead to a significant improvement.
In order to simplify the problem, further NN training is performed on a pure OS proton sample with the goal of separating these by either the origin or the charge correlation. In addition to this a larger NN architecture is trained to increase the level of abstraction. None of the trained NNs are able to visibly separate data. This leads to the conclusion, that an OSP tagger is not able to function solely on track information.
Further studies and ideas are required to develop more advanced methods, which look at multiple tracks or entire events to identify the origins of protons, thus discerning the correct charge correlation.

Appendix A

Preselection cuts

The schematics of the decision trees used for the preselection are shown in figures A.1, A.2 and A.3, with the final selected nodes in red. Branches which do not lead to this node are removed. The nodes to the final branch each have a variable and a threshold. If the variable is smaller or equal it goes left, else it goes right. The variables are explained in appendix B. The deeper the colour of a node the lower the impurity, so white means very impure.
The cuts for the preselections are also listed in table A.1

	2024	p: $p > 10\,\text{GeV}$	p: $p < 10\,\text{GeV}$	K: $p > 10\,\text{GeV}$	K: $p < 10\,\text{GeV}$	Pure P
Origin_Flag	$\neq 0$	$\neq 0$	$\neq 0$	$\neq 0$	$\neq 0$	$\neq 0$
absID			$= 2212$		$= 321$	$= 2212$
PIDP	≥ 5	> 27.34323		≤ 55.82233		
BPVIPCHI2	> 7.20012	> 6.20479	> 5.22847	> 6.20479	> 23.43348	> 4.8426
abst	≤ 0.00175	≤ 0.00098	≤ 0.00237		≤ 0.00459	≤ 0.00093
GHOSTPROB	≤ 0.00045	≤ 0.00085				
PIDK	≤ 44.36269	≤ 24.09752		> 45.57765		
MINIPChi2			> 3.4551			
MINIP (mm)			≤ 1.2075		≤ 2.01528	
BPVIP (mm)			≤ 0.6434			≤ 1.10176
Eta						≤ 4.38702
PT (MeV)						> 841.62769

Table A.1: Preselection cuts extracted from the decision trees. The samples from the selections with momentum cuts are combined together.

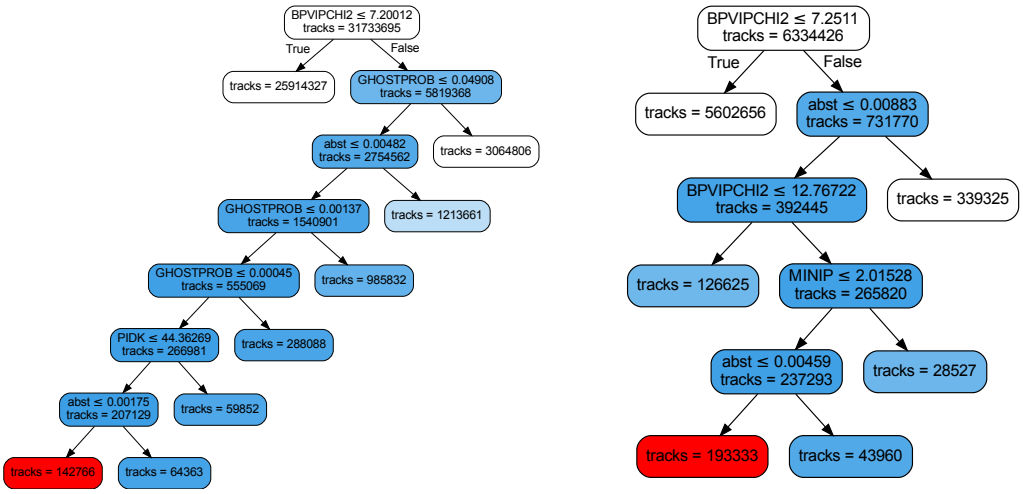

Figure A.1: Decision tree schema for 2024 conditions (left) and truth selected kaons with momenta under 10 GeV (right).

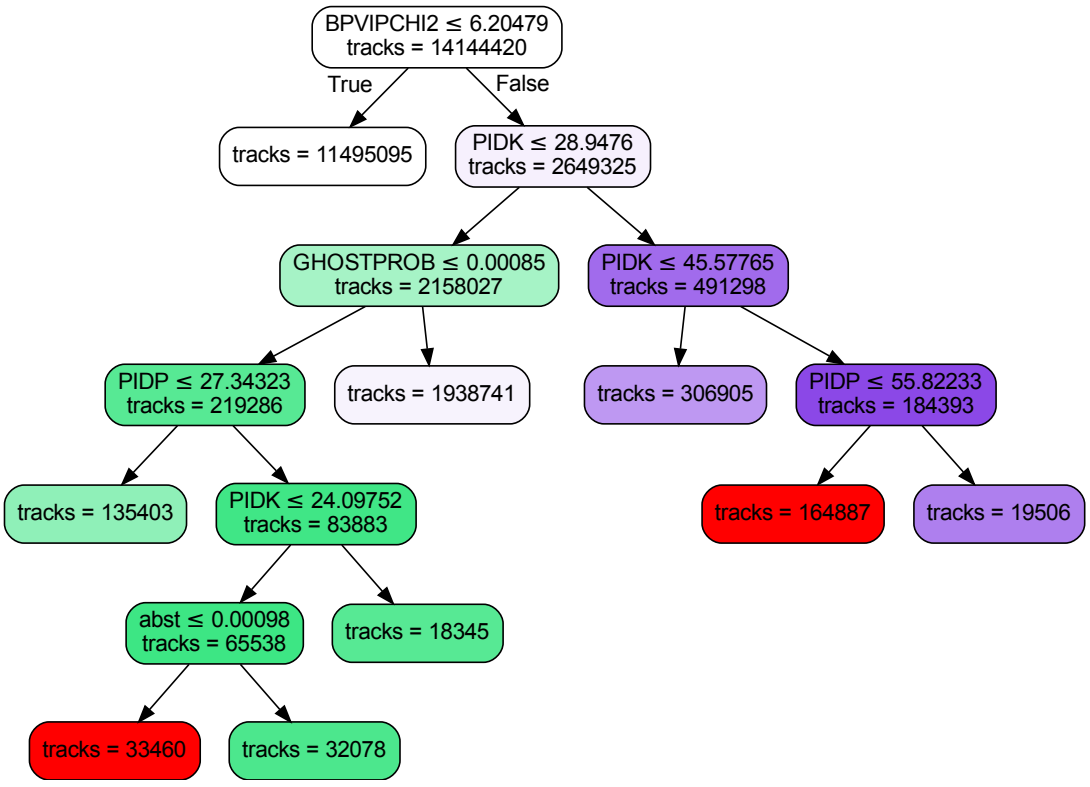

Figure A.2: Decision tree schema for tracks with momenta over 10 GeV. The proton selection is green (left), while the kaon selection is purple (right).

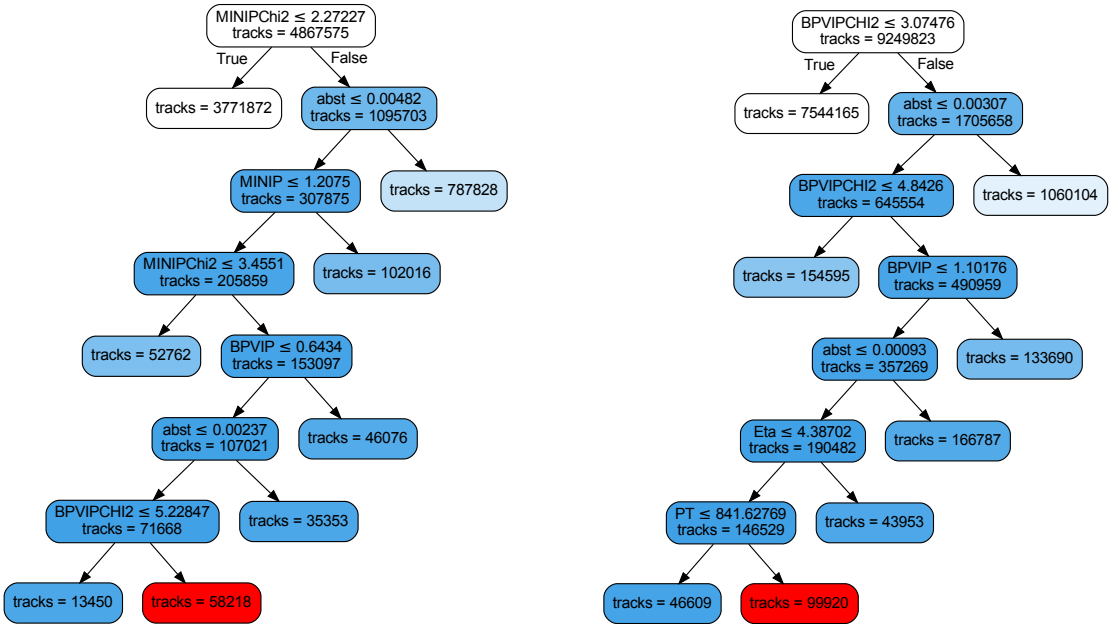

Figure A.3: Decision tree schema for truth selected protons with momenta under 10 GeV (left) and truth selected protons for all momenta (right).

Appendix B

Variables

Truth variables used in the analysis:

- **Origin_Flag:** Information about track origin. 0 = Signal; 2,3 = OS B decay.
- **TRUEID:** Particle ID following the Monte Carlo numbering scheme [25].
- **absID:** $absID = |TRUEID|$

Descriptions of variables used in the preselection and the neural network:

- **PIDX:** Log likelihood difference to be a particle X vs. a pion: $\ln \mathcal{L}(X) - \ln \mathcal{L}(\pi)$ (X: kaon, proton, electron, muon).
- **BPVIP:** Impact parameter (closest distance of the extrapolated track) to B primary vertex.
- **BPVPIPCHI2:** χ^2 of BPVIP fit.
- **CHI2DOF:** $\frac{\chi^2}{ndof}$ of track fit.
- **Eta:** $\eta = -\log(\tan(\theta/2))$. With the polar angle relative to the beam pipe θ.
- **MINIP:** Minimum impact parameter.
- **MINIPCHI2:** χ^2 of MINIP fit.
- **P:** Momentum.
- **PT:** Transverse momentum relative to beampipe.
- **Phi:** Azimuthal angle.
- **nPVs:** Number of primary vertices in the event.
- **nTracks:** Number of tracks in the event.
- **PhiDistance:** Azimuthal angle difference to signal B track.
- **cos_PhiDistance:** cosine of PhiDistance

- **absPhiDistance:** absolute value of PhiDistance
- **EtaDistance:** Polar angle difference to signal B track.
- **Delta_R:** EtaDistance^2 + PhiDistance^2
- **diff_P:** Momentum difference between signal B and track.
- **diff_PT:** Transverse momentum difference between signal B and track.
- **P_proj:** Scalar product of signal B 4 momentum and track 4-momentum.
- **abst:** $|t|$ with $t = \frac{\vec{y}^2 - \vec{y}\cdot\vec{x}}{\vec{y}\cdot\vec{p}}$, the signal B decay vertex $\vec{y}$, track vertex $\vec{x}$ and track 3 momentum $\vec{p}$. A helper variable for EVIP.
- **EVIP:** $\log\sqrt{\vec{x}^2 + 2t(\vec{x}\cdot\vec{p}) + t^2\vec{p}^2}$. A kinematic variable with no specific physical interpretation.
- **DeltaQ_Proton:** $\sqrt{(E + E_B)^2 - (\vec{p} + \vec{p_B})^2} - M_B - 938.272\,08\,\text{MeV}$ with the track energy E, 3 momentum $\vec{p}$ and the signal B energy E_B, 3 momentum $\vec{p_B}$ and mass M_B and the proton mass 938.272 08 MeV.
- **Signal_TagPart_PT:** $\sqrt{(px_B + px)^2 + (py_B + py)^2}$ with the momentum fractions px, py of the track and px_B, py_B of the signal B.
- **eoverP:** Track charge divided by the momentum.
- **PROBNNX:** Probability for track to be particle X using machine learning (X: electron, ghost, kaon, proton, pion).
- **GHOSTPROB:** Probability for track to be a ghost, a falsely reconstructed track, that has no particle associated to it.

Appendix C

Neural Network plots

The ROC curves of the neural networks are shown in figure C.1, the loss curves of the training are shown in figure C.2 and the calibration plots of the taggers are shown in figure C.3.

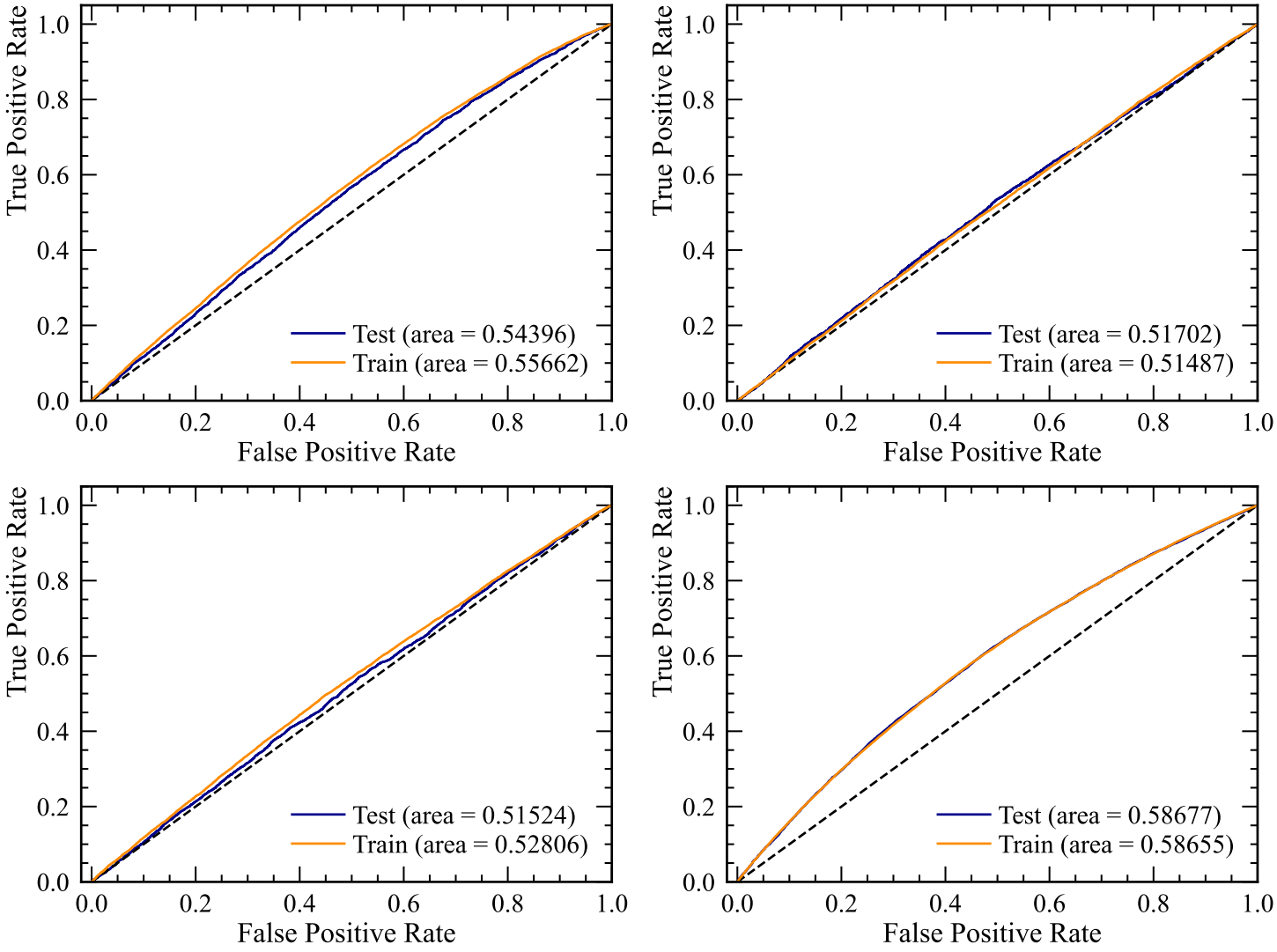

Figure C.1: ROC curves of the trained neural networks with the training sample (yellow) and the validation sample (blue). The dotted diagonal line represents a ROC curve with no statistical significance, equivalent to random guessing. Preselection in 2024 conditions (top left), preselection with additional low momentum perfect PID (top right), preselection with perfect PID (bottom left) and OS kaon tagger with low momentum perfect PID (bottom right).

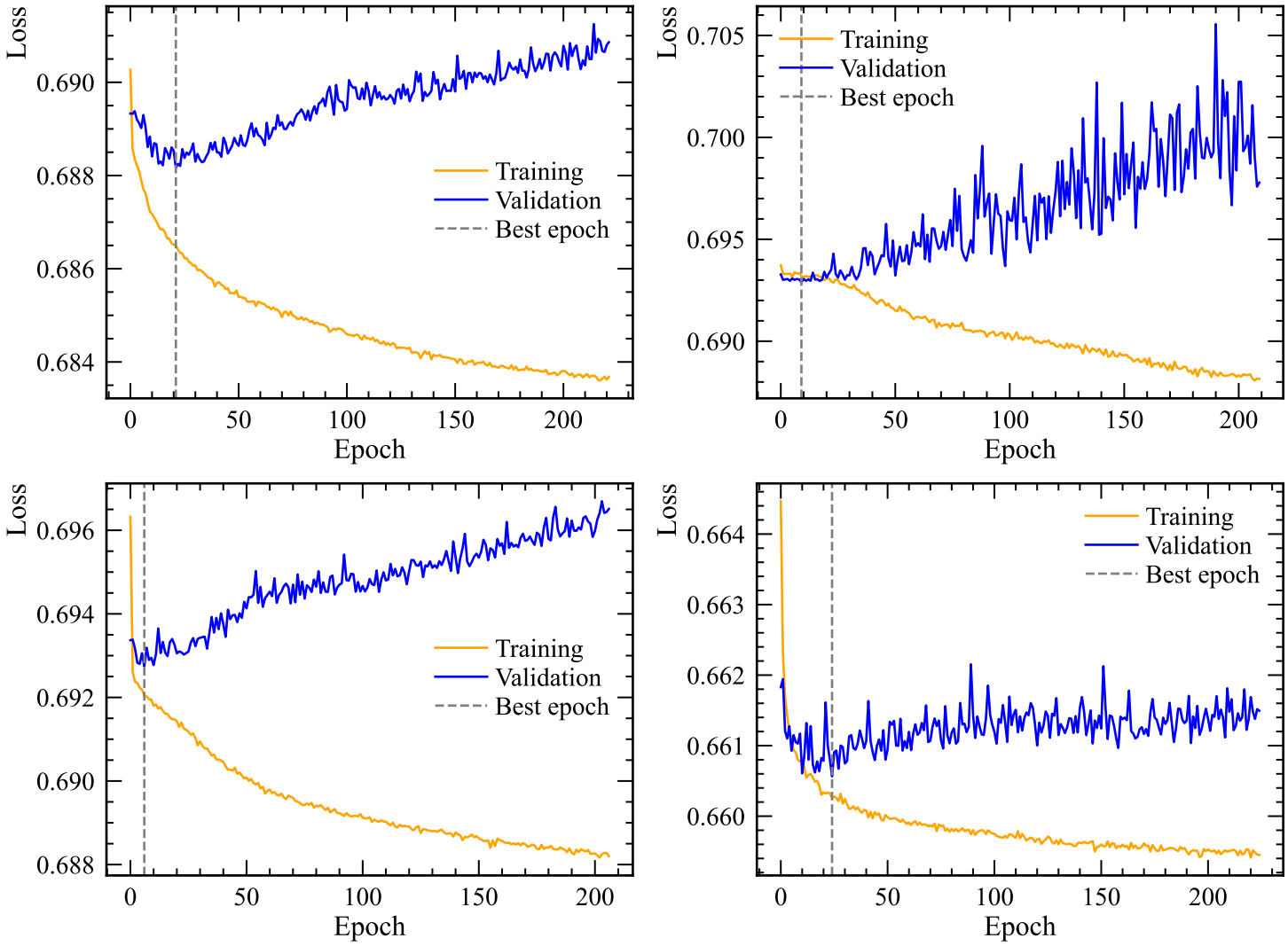

Figure C.2: Loss curves of the NN training with the training sample (yellow) and the validation sample (blue). The dotted line indicates the best epoch, which is then used as the final model. Preselection in 2024 conditions (top left), preselection with additional low momentum perfect PID (top right), preselection with perfect PID (bottom left) and OS kaon tagger with low momentum perfect PID (bottom right).

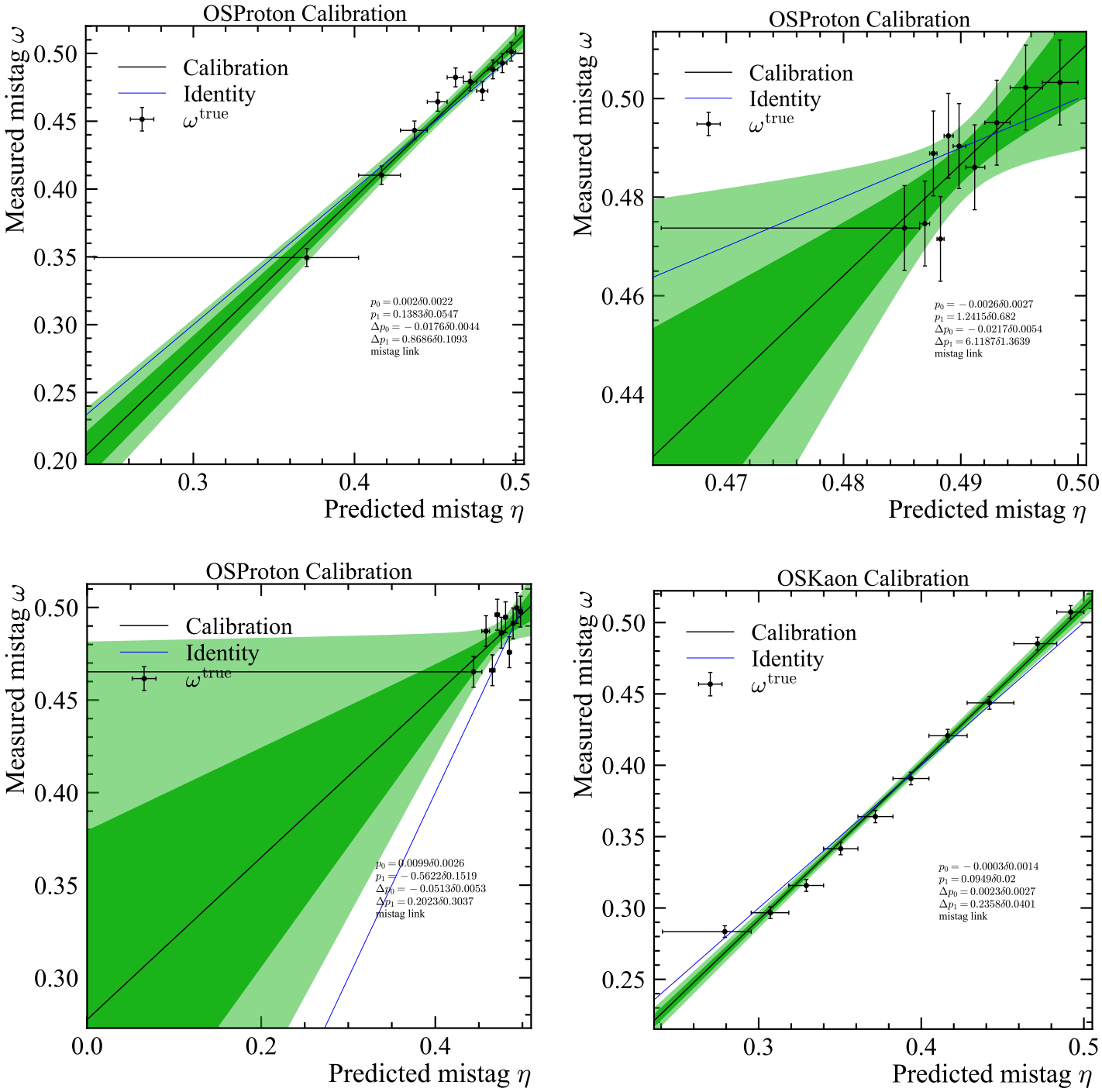

Figure C.3: Calibration plots of the trained taggers, the blue line is the identity $y = x$. The green shaded areas represent the 1-σ and 2-σ confidence intervals. Preselection in 2024 conditions (top left), preselection with additional low momentum perfect PID (top right), preselection with perfect PID (bottom left) and OS kaon tagger with low momentum perfect PID (bottom right).

Bibliography

[1] Mark Thomson. *Modern particle physics*. New York: Cambridge University Press, 2013. ISBN: 978-1-107-03426-6. DOI: 10.1017/CBO9781139525367.

[2] Lyndon R Evans and Philip Bryant. "LHC Machine". In: *JINST* 3 (2008). This report is an abridged version of the LHC Design Report (CERN-2004-003), S08001. DOI: 10.1088/1748-0221/3/08/S08001. URL: https://cds.cern.ch/record/1129806.

[3] A. Augusto Alves Jr. et al. "The LHCb Detector at the LHC". In: *JINST* 3 (2008), S08005. DOI: 10.1088/1748-0221/3/08/S08005.

[4] R. Aaij et al. "Precise determination of the B_s^0–$\overline{B}_s^0$ oscillation frequency. Precise determination of the B0s-B0sbar oscillation frequency". In: *Nature Phys.* 18.1 (2022). All figures and tables, along with any supplementary material and additional information, are available at https://cern.ch/lhcbproject/Publications/p/LHCb-PAPER-2021-005.html (LHCb public pages), pp. 1–5. DOI: 10.1038/s41567-021-01394-x. arXiv: 2104.04421. URL: https://cds.cern.ch/record/2764338.

[5] R. Aaij et al. "Opposite-side flavour tagging of B mesons at the LHCb experiment". In: *The European Physical Journal C* 72.6 (June 2012). ISSN: 1434-6052. DOI: 10.1140/epjc/s10052-012-2022-1. URL: http://dx.doi.org/10.1140/epjc/s10052-012-2022-1.

[6] The LHCb collaboration. *Framework TDR for the LHCb Upgrade II: Opportunities in flavour physics, and beyond, in the HL-LHC era*. Tech. rep. Geneva: CERN, 2021. URL: https://cds.cern.ch/record/2776420.

[7] Michal Kreps. "LHCb TORCH detector". In: *Talk at IAS HEP 2021*. Accessed: 2024-08-24. CERN. 2021. URL: https://indico.cern.ch/event/971970/contributions/4169857/attachments/2172021/3667173/torchIASHep2021.pdf.

[8] LHCb collaboration et al. *The LHCb upgrade I*. 2023. arXiv: 2305.10515 [hep-ex]. URL: https://arxiv.org/abs/2305.10515.

[9] M. Adinolfi et al. "Performance of the LHCb RICH detector at the LHC". In: *The European Physical Journal C* 73.5 (2013), p. 2431. ISSN: 1434-6052. DOI: 10.1140/epjc/s10052-013-2431-9. URL: https://doi.org/10.1140/epjc/s10052-013-2431-9.

[10] Neville Harnew et al. "TORCH: a large area time-of-flight detector for particle identification". In: *Nucl. Instrum. Methods Phys. Res., A* 936 (2019). 5 pages, 2 figures, Paper submitted to Nuclear and Methods A : Proceedings of Frontier Detectors for Frontier Physics - 14th Pisa Meeting on Advanced Detectors, pp. 595–597. DOI: 10.1016/j.nima.2018.10.099. arXiv: 1810.06658. URL: https://cds.cern.ch/record/2645864.

[11] I Belyaev et al. "Handling of the generation of primary events in Gauss, the LHCb simulation framework". In: *Journal of Physics: Conference Series* 331.3 (2011), p. 032047. DOI: 10.1088/1742-6596/331/3/032047. URL: https://dx.doi.org/10.1088/1742-6596/331/3/032047.

[12] Torbjörn Sjöstrand et al. "An introduction to PYTHIA 8.2". In: *Computer Physics Communications* 191 (2015), pp. 159–177. ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2015.01.024. URL: https://www.sciencedirect.com/science/article/pii/S0010465515000442.

[13] David J. Lange. "The EvtGen particle decay simulation package". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 462.1 (2001). BEAUTY2000, Proceedings of the 7th Int. Conf. on B-Physics at Hadron Machines, pp. 152–155. ISSN: 0168-9002. DOI: https://doi.org/10.1016/S0168-9002(01)00089-4. URL: https://www.sciencedirect.com/science/article/pii/S0168900201000894.

[14] S. Agostinelli et al. "Geant4—a simulation toolkit". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (2003), pp. 250–303. ISSN: 0168-9002. DOI: https://doi.org/10.1016/S0168-9002(03)01368-8. URL: https://www.sciencedirect.com/science/article/pii/S0168900203013688.

[15] Lucio Anderlini et al. *The PIDCalib package*. Tech. rep. Geneva: CERN, 2016. URL: https://cds.cern.ch/record/2202412.

[16] Amir Akbari, Lilian Ng, and Bruno Solnik. "Drivers of economic and financial integration: A machine learning approach". In: *Journal of Empirical Finance* 61 (Jan. 2021). DOI: 10.1016/j.jempfin.2020.12.005.

[17] Leo Breiman et al. *Classification and Regression Trees*. Wadsworth and Brooks/Cole, 1984.

[18] TikZ.net. *Adapted from: Neural Networks in TikZ*. https://tikz.net/neural_networks/. Accessed: 2024-09-07.

[19] Jan Pawlowski. *Physics and Machine Learning*. Lecture Notes. 2024.

[20] PyTorch. *BCELoss — PyTorch Documentation*. https://pytorch.org/docs/stable/generated/torch.nn.BCELoss.html. Accessed: 2024-09-07. 2024.

[21] S. Navas et al. "Review of particle physics". In: *Phys. Rev. D* 110.3 (2024), p. 030001. DOI: 10.1103/PhysRevD.110.030001.

[22] Wikimedia Commons. *Standard Model of Elementary Particles*. 2024. URL: https://w.wiki/Ayou.

[23] Julian Wishahi. *Flavour Tagging Plots for Conference*. https://twiki.cern.ch/twiki/bin/viewauth/LHCb/FlavourTaggingConferencePlots. Accessed: 2024-08-23. 2016.

[24] *Comparison of Flavour Tagging performances displayed in the ω-ε_{tag}-plane*. 2020. URL: https://cds.cern.ch/record/2707121.

[25] Particle Data Group. *Particle Identification*. http://pdg.lbl.gov/current/mc-particle-id. Accessed: 2024-08-29.

[26] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[27] Jason Ansel et al. "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation". In: *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*. ASPLOS '24. La Jolla, CA, USA: Association for Computing Machinery, 2024, pp. 929–947. ISBN: 9798400703850. DOI: 10.1145/3620665.3640366. URL: https://doi.org/10.1145/3620665.3640366.

[28] Q. Führing and V. Jevtić. *lhcb-ftcalib: A software package for the calibration of flavour-tagged LHCb data*. 2024. DOI: 10.5281/zenodo.12156328. URL: https://gitlab.cern.ch/lhcb-ft/lhcb_ftcalib.

Acknowledgements

First and foremost, I would like to express my gratitude to professor Stephanie Hansmann-Menzemer for providing me with the opportunity to work in her LHCb group and for her insightful feedback and advice throughout my research.
I am also deeply grateful to my two supervisors, Lennart Uecker and Sara Celani, for their patience in answering my many questions and their expert guidance throughout every step of this thesis.
I would also like to extend my thanks to the working group at the Physikalisches Institut and the LHCb collaboration for their warm welcome and support, which made this work possible.
Special thanks go to my family and friends for their support and understanding, especially my friend Yukai Zhao, who provided critical feedback and and helped answer many of my questions.
I would like to dedicate a special mention to my late grandmother Inge Peckys, who was not permitted to pursue advanced education herself, yet always took pride in our successes.
Lastly, my deepest and most heartfelt gratitude goes to my girlfriend, Kayla Emge. Her constant encouragement and support have been instrumental in helping me come this far.

Erklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, den 9. September 2024,

Anselm de Jonge