

Measurement of the decay $B \rightarrow KK\ell\nu$ with B2BII

Matic Lubej

Ljubljana, 2018

Changelog

- First submission of the note.

Contents

	Page
1 Introduction	1
2 Data and MC	2
2.1 MC	2
2.2 Data	2
3 B2BII conversion	3
3.1 Description	3
3.2 Validation	3
4 Event reconstruction	4
4.1 Final state particles selection	4
4.2 Combination of FSP particles	11
4.3 Rest of event clean-up	13
4.4 Loose neutrino reconstruction	14
4.5 q^2 calculation	17
4.6 Event categorization	19
4.6.1 Hadronic decay MVA training	20
4.7 Signal region definition	22
4.8 Selection summary	23
5 Rest of event clean-up	24
5.1 Setting up the MVA	24
5.2 Clusters clean-up	25
5.2.1 π^0 MVA training	25
5.2.2 γ MVA training	27
5.3 Tracks clean-up	29
5.3.1 Tracks from long-lived particles	31
5.3.2 Duplicate tracks	32
5.4 Belle clean-up	40
5.5 Clean-up results	41
5.6 ROE clean-up validation	44
6 Background suppression	45
6.1 Control decay and other resonant background	45

6.2	Continuum suppression	46
6.2.1	Characteristic variables	46
6.2.2	MVA training	49
6.3	$B\bar{B}$ suppression	50
6.3.1	Boosting to uniformity	52
6.4	Selection optimization	53
6.5	Data and MC agreement	56
7	Signal extraction	60
7.1	Fit templates	60
7.2	Adaptive binning algorithm	61
7.3	Signal MC fit results	62
7.3.1	Toy MC experiment	63
7.3.2	Toy MC linearity test	63
7.4	Control fit results	63
7.4.1	Fit yields and yield ratios	64
7.4.2	Double ratios	64
7.4.3	Branching ratio measurement and comparison	64
7.5	Signal fit to data	64
8	Systematics	65
8.1	Model uncertainty effects	65
8.2	PID efficiency correction	65
8.3	Bias	65

1 Introduction

Will add more info later

2 Data and MC

Will add more info later

2.1 MC

- Belle generic MC, on $Y(4S)$ resonance energy, converted with B2BII
- Belle generic MC, off resonance energy, converted with B2BII (for checks)
- Belle rare + ulnu MC, converted with B2BII
- Signal MC, produced with similar script to charged ulnu, converted with B2BII

2.2 Data

- Full Belle dataset, on resonance, converted with B2BII
- Full Belle dataset, off resonance, converted with B2BII (for checks)

3 B2BII conversion

3.1 Description

3.2 Validation

4 Event reconstruction

In this chapter the procedure for event reconstruction of the B meson decay $B \rightarrow KK\ell\nu$ is shown, starting with final state particle selection and then combining them to obtain B meson candidates.

4.1 Final state particles selection

Since the neutrino escapes detection, we can only reconstruct the charged tracks in the decay, which are the two kaons (K) and the light lepton, which is the electron (e) or muon (μ). These are some of the particles which are commonly referred to as final state particles (FSP). Final state particles have a long lifetime and are usually the particles that we detect when they interact with the material in the detector.

It is important to limit our selection of FSP particles in order to cut down the number of particle combinations, and consequentially computation time and file sizes.

Leptons

Figures 4.1 and 4.2 show the impact parameters d_0 and z_0 , the momentum in $\Upsilon(4S)$ center-of-mass system (CMS), and the PID information for true and fake electrons and muons, where an extra category for true electrons/muons from the signal decay is shown.

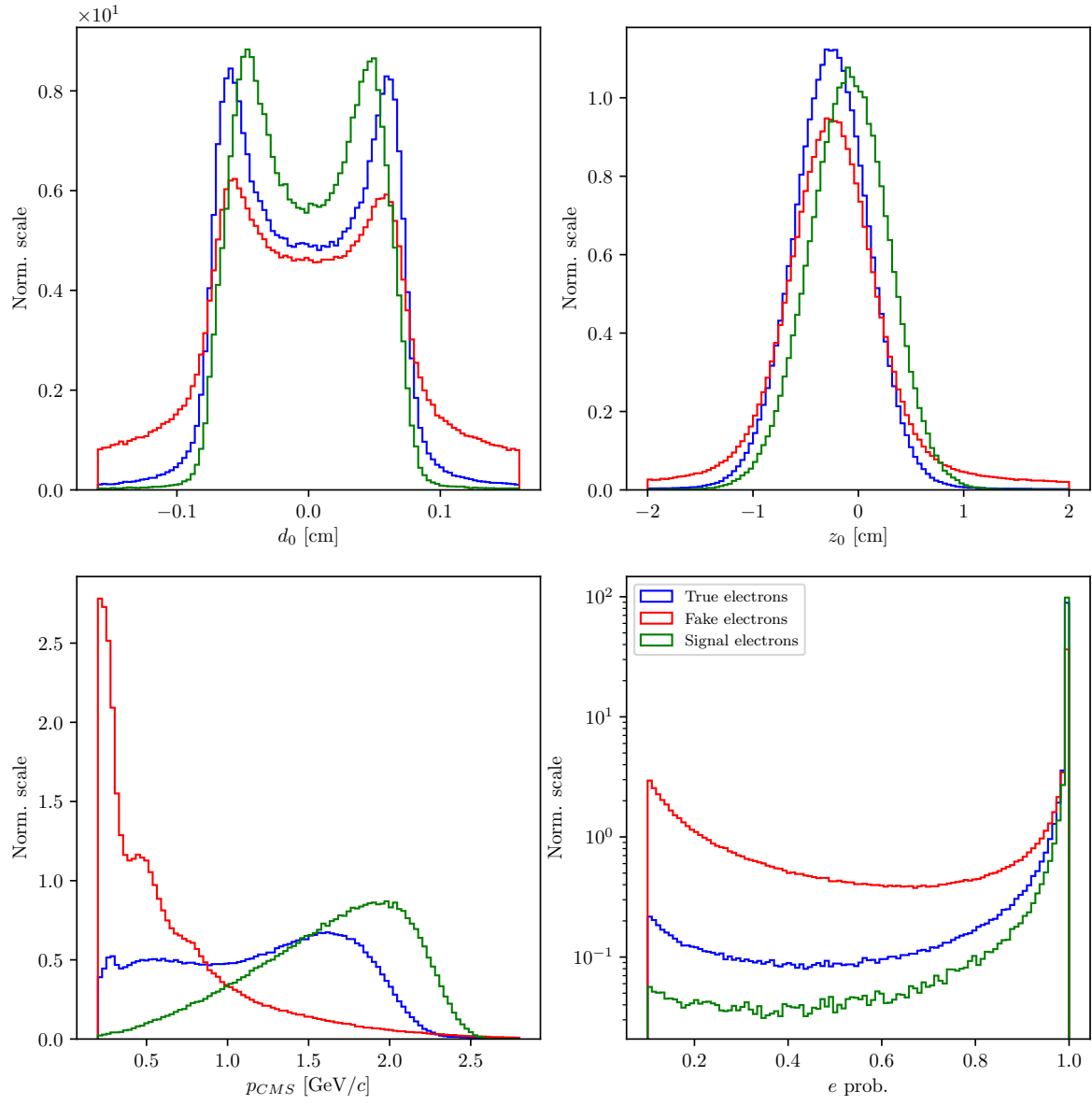


Figure 4.1: Normalized properties of true (blue), fake (red) and true electrons from signal decay (green)

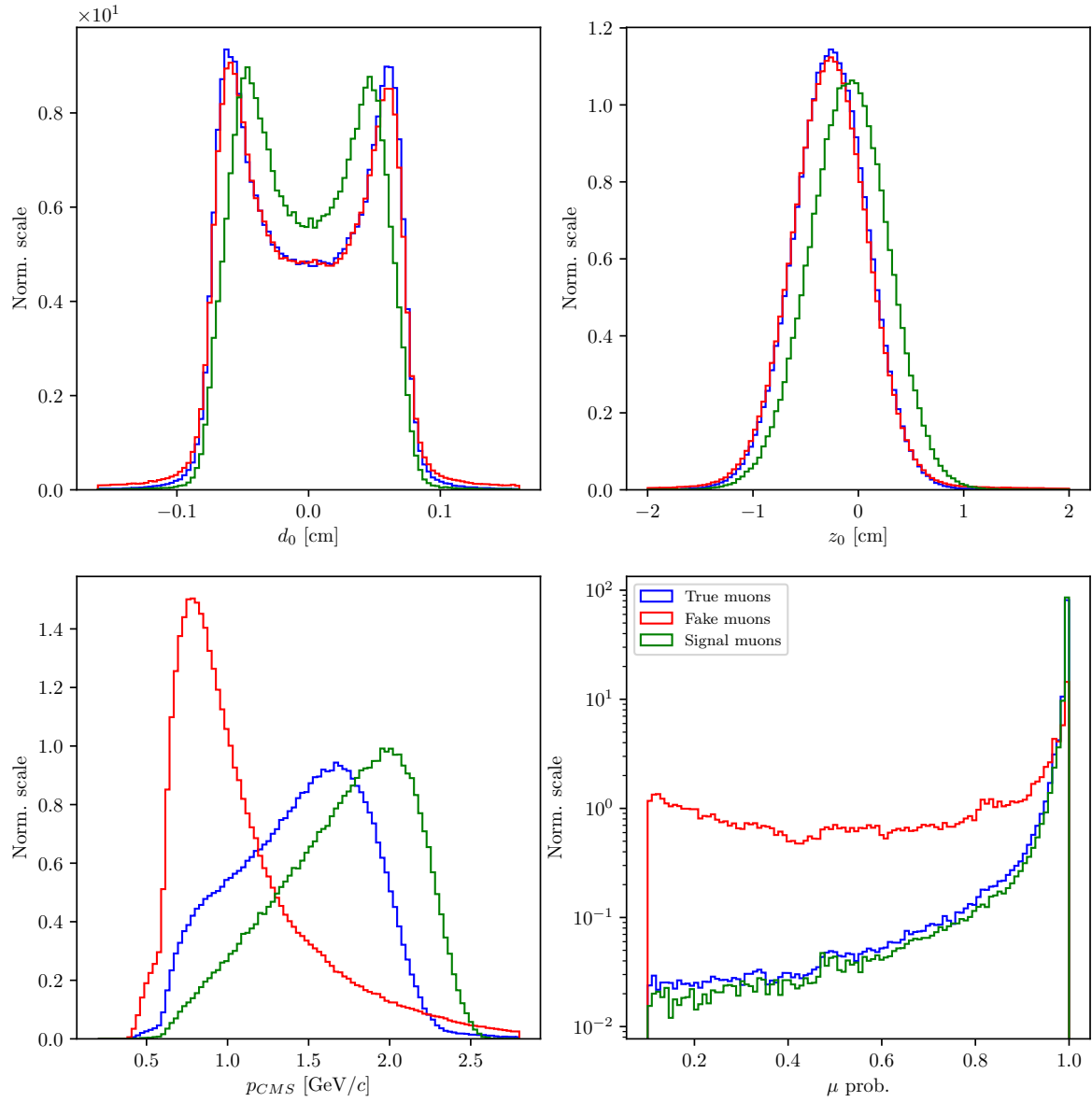


Figure 4.2: Normalized properties of true (blue), fake (red) and true muons from signal decay (green)

Based on these distributions, we can define a set of cuts

- $|d_0| < 0.1$ cm,
- $|z_0| < 1.5$ cm,
- $p_{CMS} \in [0.4, 2.6]$ GeV/ c for electrons,

- $p_{CMS} \in [0.6, 2.6]$ GeV/ c for muons.

After this selection we can determine the optimal PID cuts for electrons and muons, where we optimize by maximizing the standard definition of *figure of merit* (FOM), defined in Eq. (4.1)

$$FOM = \frac{S}{\sqrt{S+B}}, \quad (4.1)$$

where S represents number of signal and B the number of background candidates.

The FOM plot is shown in Figures 4.3 and 4.4. The cuts values are based on PID cuts used for PID efficiency calibration. The optimal value for the PID cuts is equal to the largest available value, regardless of the leptons coming from signal decays or not. The optimized PID cuts for leptons are

- e prob. > 0.9 for electrons,
- μ prob. > 0.97 for muons.

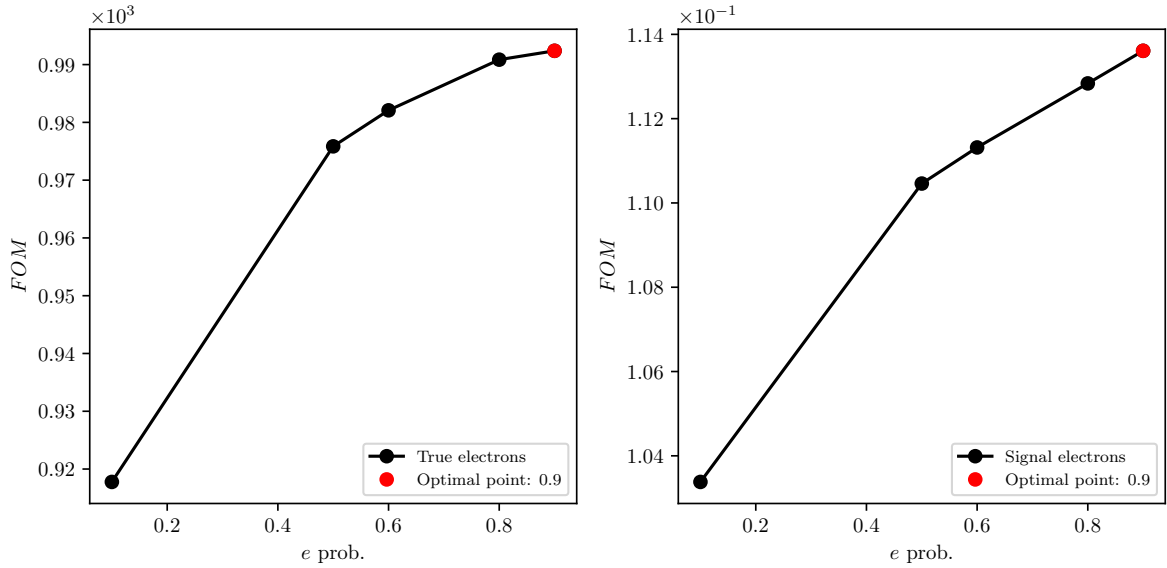


Figure 4.3: FOM optimizations of the PID probability cuts for true electrons (left) and true electrons from signal decays (right).

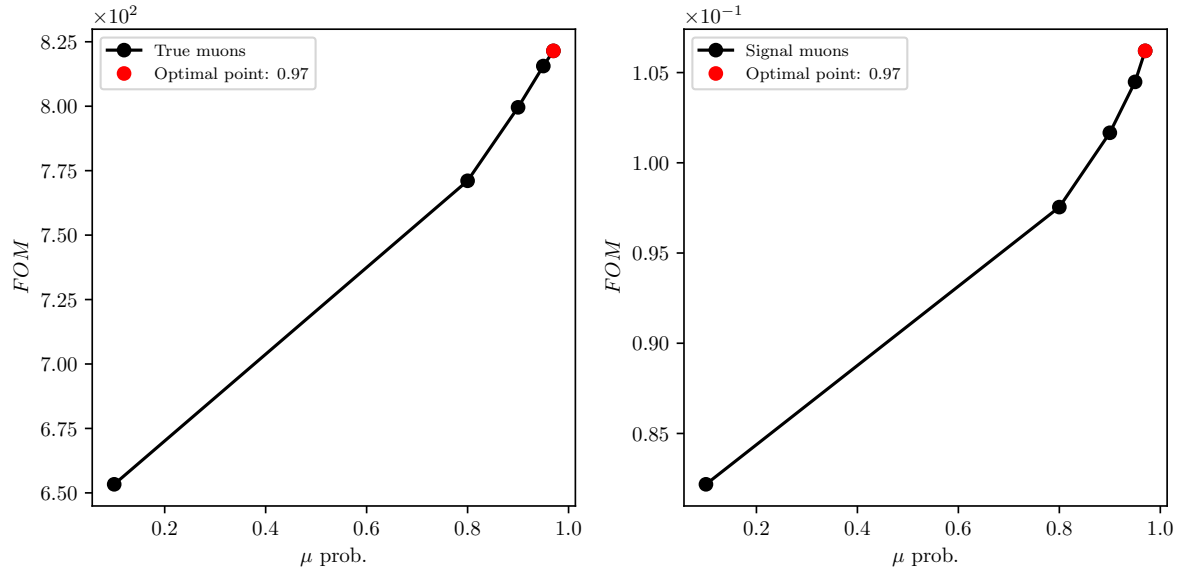


Figure 4.4: FOM optimizations of the PID probability cuts for true muons (left) and true muons from signal decays (right).

Kaons

We repeat the procedure for both kaons. Figure 4.5 shows the impact parameters d_0 and z_0 , the momentum in $\Upsilon(4S)$ center-of-mass system (CMS), and the PID information for true and fake kaons, where an extra category for true kaons from the signal decay is shown.

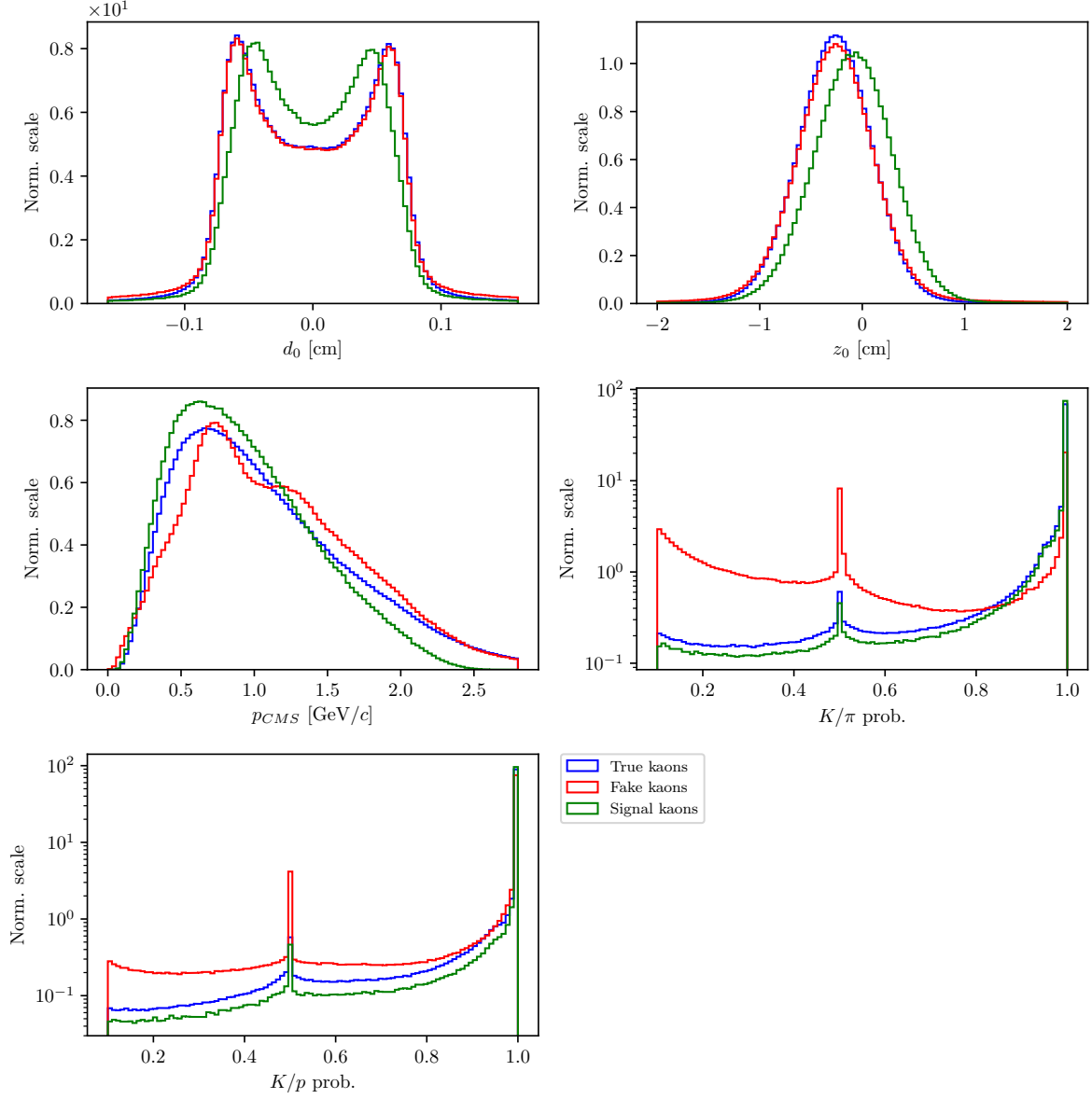


Figure 4.5: Normalized properties of true (blue), fake (red) and true kaons (green) from signal decay

We define the kaon cuts in the same manner as in the case for leptons

- $|d_0| < 0.15$ cm,
- $|z_0| < 1.5$ cm,
- $p_{CMS} \in [0, 2.5]$ GeV/ c .

The PID optimization in this case is taken in two steps. First we optimize the cut on K/π , and after that the K/p separation probability. Figure 4.6 shows the optimization procedure for PID cuts on kaon candidates.

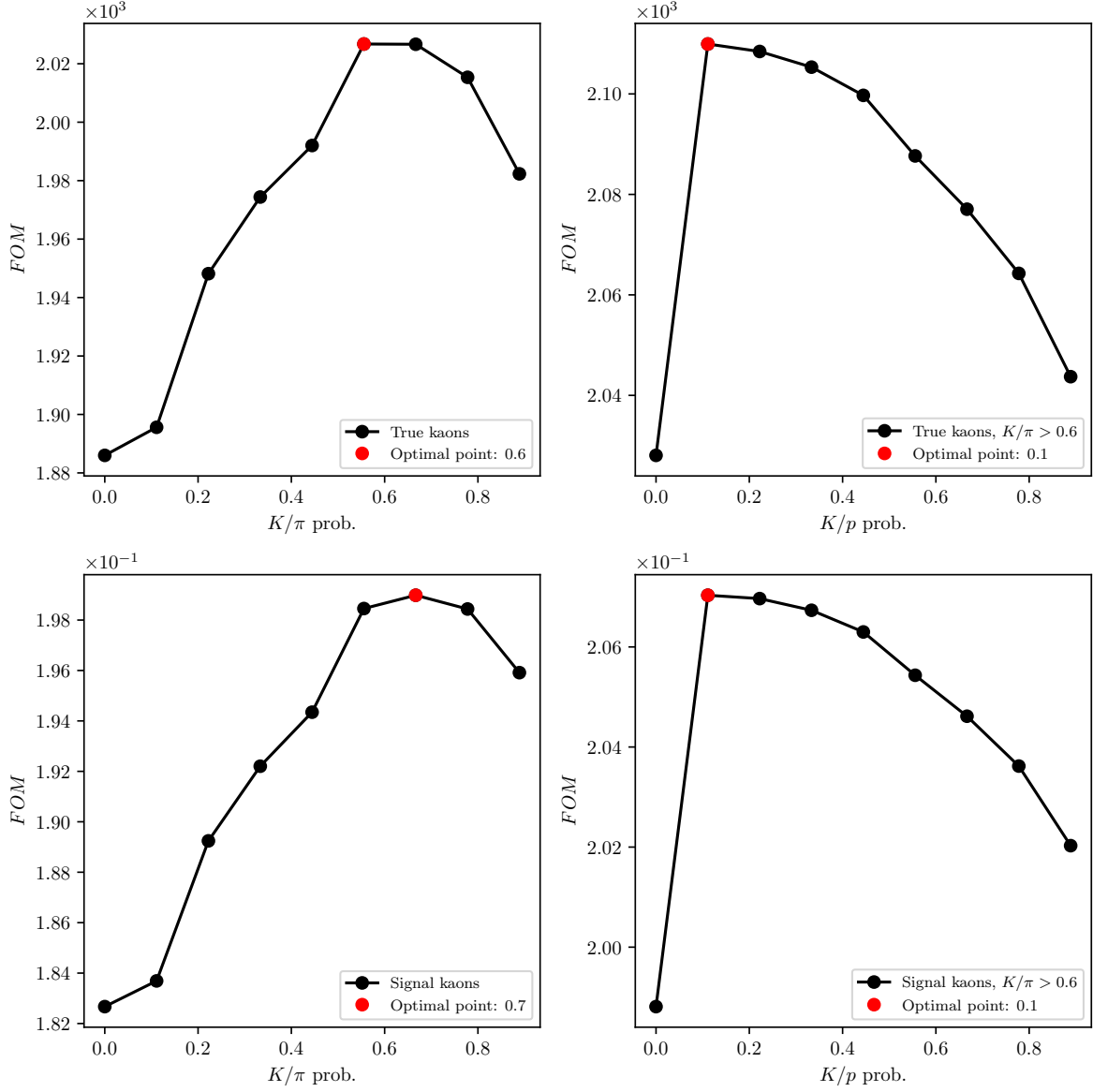


Figure 4.6: FOM optimizations of the PID probability cuts for true kaons (top) and true kaons from signal decays (bottom). The plots on the left show the optimization of the first step for the K/π probability cut, and the plot on the right the K/p probability cut.

The optimized PID cuts for kaons are

- $K/\pi > 0.6$,
- $K/p > 0.1$.

4.2 Combination of FSP particles

With the pre-selected kaon and lepton candidates we make combinations for potential B meson candidates. Due to the missing neutrino, we reconstruct the B mesons in the following two channels

$$\begin{aligned} B^+ &\rightarrow K^+ K^- e^+, \\ B^+ &\rightarrow K^+ K^- \mu^+, \end{aligned}$$

and similarly for B^- . When an arbitrary combination is obtained, we perform a vertex fit of the three tracks in order to discard combinations with a low probability of tracks coming from the same point. B mesons have a relatively long lifetime and decay along the z-axis of the detector in the direction of the boost, so the vertex fit is enforced with an *iptube* constraint, which constrains the vertex to an elongated ellipsoid along beam direction. We demand that the fit converged and apply a cut on the minimal fit probability. The fit probability for signal and background B meson candidates is shown in Figure 4.7 (left). We perform a *FOM* cut optimization of this variable, which is shown in Figure 4.7 (right).

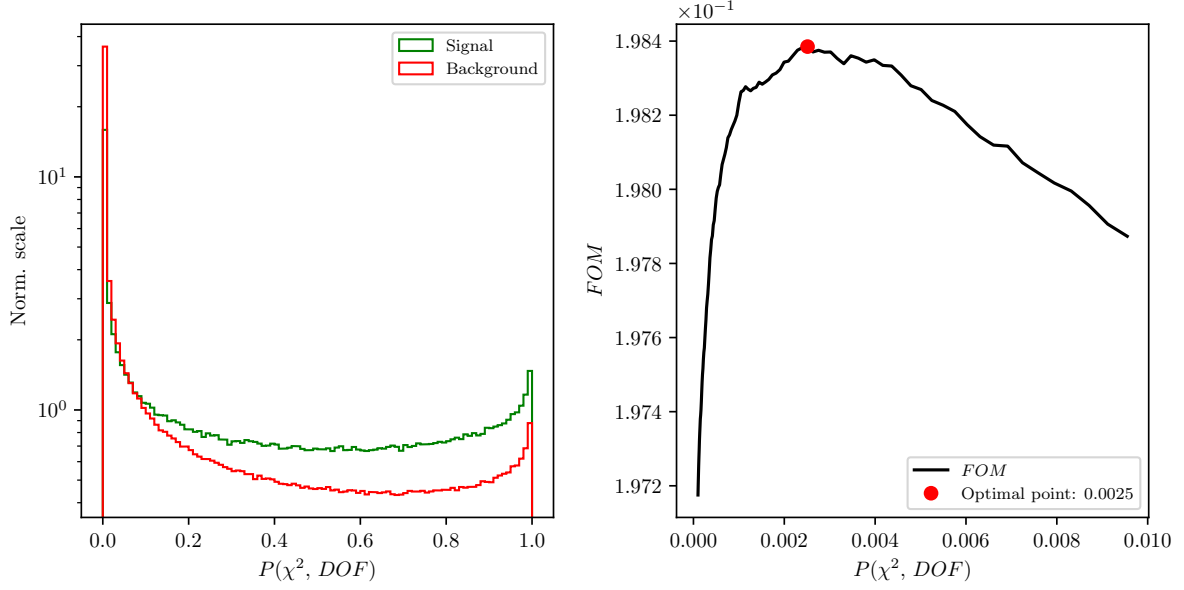


Figure 4.7: Normalized vertex fit probability distribution for signal and background B meson candidates in logarithmic scale (left) and FOM optimization of the vertex fit probability (right).

Even though that the vertex fit probability cut was optimized here, we choose a standard cut of

- $P(\chi^2, NDF) > 1.0 \times 10^{-3}$.

With the neutrino being the only missing particle on the reconstructed side, it is possible to determine the angle between the direction of the reconstructed B (denoted as $Y \rightarrow KK\ell$) and the true B , as

$$p_\nu = p_B - p_Y, \quad (4.2)$$

$$p_\nu^2 = m_\nu^2 = m_B^2 + m_Y^2 - 2E_BE_Y + 2\vec{p}_B \cdot \vec{p}_Y \approx 0, \quad (4.3)$$

$$\cos(\theta_{BY}) = \frac{2E_BE_Y - m_B^2 - m_Y^2}{2|\vec{p}_B||\vec{p}_Y|}, \quad (4.4)$$

where all the energy and momenta above are calculated in the CMS frame. The mass of the neutrino is equal to 0 to a very good precision, so we use it in Eq. (4.3). In addition, we can substitute the unknown energy and momentum magnitude, E_B and $|\vec{p}_B|$, of the B meson in Eq. (4.4), with quantities from the well known initial conditions

$$E_B = E_{CMS}/2, \quad (4.5)$$

$$|\vec{p}_B| = p_B = \sqrt{E_B^2 - m_B^2}, \quad (4.6)$$

where E_{CMS} is the total energy of the e^+e^- collision in the CMS frame and m_B is the nominal mass of the B meson.

For the correctly reconstructed candidates, this variable lies in the $[-1, 1]$ region, though only to a certain precision, due to the finite detector resolution. For background candidates, however, the values populate also the non-physical regions, as is shown in Figure 4.8 (left). We impose an optimized cut on this variable from Figure 4.8 (right) to discard a large amount of background.

- $|\cos(\theta_{BY})| < 1.0$.

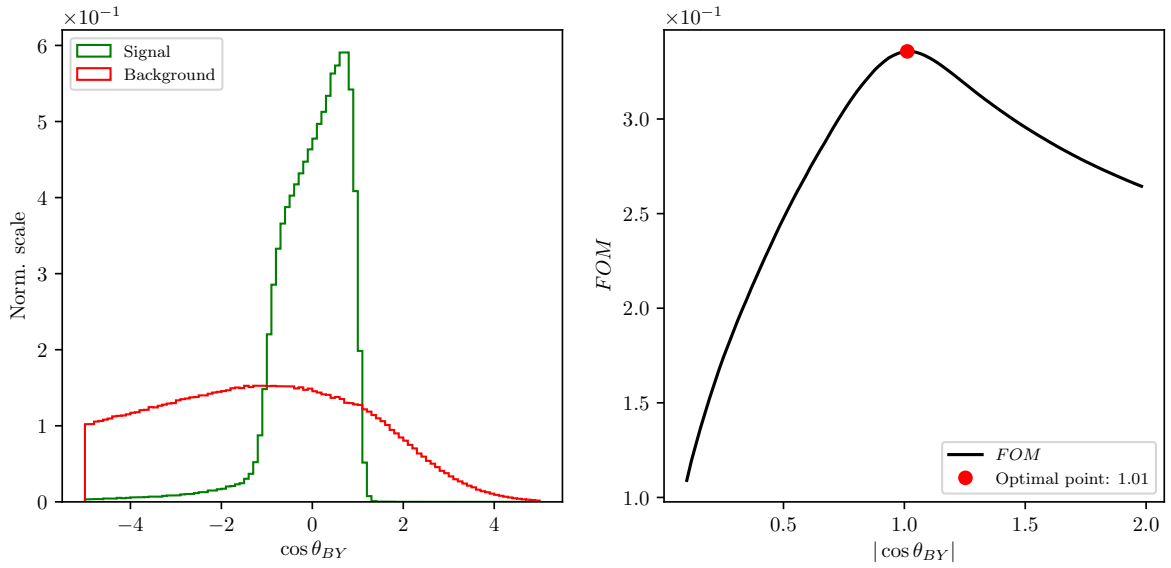


Figure 4.8: Normalized $\cos \theta_{BY}$ distribution for signal and background B meson candidates (left) and FOM optimization of the $\cos \theta_{BY}$ variable (right).

4.3 Rest of event clean-up

Due to the beam background in the detector, material interactions, or other processes, random tracks and clusters enter our event and get reconstructed as part of the physics process we want to study. These tracks and clusters are not interesting and further spoil the data we measure. In order to remedy this, we perform an extensive clean-up of the tracks and clusters in the ROE side before calculating the four-momentum of the missing part of the event. The clean-up procedure is performed separately on tracks and clusters and uses multiple steps with multivariate analysis (MVA) algorithms in order to separate good tracks and clusters from the bad ones, which populate the ROE. Then, for each ROE object, a ROE mask is created for tracks and clusters, which narrates the

use of this track or cluster in the final calculations of the ROE four-momentum.

In order to preserve the continuity of this chapter, a more detailed description of the ROE clean-up can be found in Chapter 5. From this point on we assume the ROE to be efficiently cleansed of extra tracks and clusters .

4.4 Loose neutrino reconstruction

The signal-side neutrino escapes detection, so we cannot directly determine it's four-momentum. However, due to the detectors geometry, which almost completely covers the full solid angle, and due to well known initial conditions of the $\Upsilon(4S)$ meson, it is possible to determine the kinematics of the missing neutrino via indirectly reconstructing the companion B meson by summing up all the FSP particles four-momenta in the ROE. This is known as the *untagged* method. Here we see the motivation for the ROE clean-up, since our signal candidate reconstruction depends on the other B meson.

The total missing four-momentum in the event can be determined as

$$p_{miss} = p_{\Upsilon(4S)} - \sum_i^{\text{Event}} (E_i, \vec{p}_i), \quad (4.7)$$

$$p_{miss} = p_{\Upsilon(4S)} - \left(p_Y - \sum_i^{\text{Rest of event}} (E_i, \vec{p}_i) \right), \quad (4.8)$$

where the summation runs over all charged and neutral particles in the defined set with

$$p_i^{\text{neutral}} = (p_i, \vec{p}_i) \quad \text{and} \quad p_i^{\text{charged}} = \left(\sqrt{m_i^2 + p_i^2}, \vec{p}_i \right), \quad (4.9)$$

where we assumed all neutral particles to be massless photons. For charged tracks in the ROE a mass hypothesis needs to be defined in order to determine the track's energy. After the ROE clean-up we make the following procedure of choosing the mass hypothesis

1. e , if e prob. $> \mu$ prob. and e prob. > 0.9 ,
2. otherwise μ , if μ prob. $> e$ prob. and μ prob. > 0.97 ,
3. otherwise K , if K/π prob. > 0.6 ,
4. otherwise π .

We define the square of the missing mass, m_{miss}^2 , which is zero, if signal-side neutrino is

the only missing particle in the event, as shown in Eq. (4.11).

$$\mathbf{p}_\nu = \mathbf{p}_{miss} = (E_{miss}, \vec{p}_{miss}), \quad (4.10)$$

$$m_{miss}^2 = p_{miss}^2 = p_\nu^2 = m_\nu^2 \approx 0. \quad (4.11)$$

Since the detector is not perfect, the distribution of the m_{miss}^2 variable has a non-zero width. Additionally, tails are introduced as soon as we have missing particles such as extra missing neutrinos, other neutral undetected particles such as K_L^0 , or simply missing tracks due to detection failure. Figure 4.9 shows the distribution of m_{miss}^2 as defined with the missing four-momentum in Eq. (4.10). Correctly reconstructed candidates, which come from events where the other B meson decayed via a hadronic decay mode, should peak at zero. If this is not the case, candidates are shifted to larger values of this variable. Due to this fact, we impose a cut on the m_{miss}^2 variable in order to partially discard candidates with spoiled properties, even if it was in principle a correct combination of FSP particles on the signal side

- $|m_{miss}^2| < 7 \text{ GeV}/c^2$.

This cut was not optimized, since the optimal case would result in a too strong threshold at this point in the analysis, since we still may want to retain as much signal candidates as possible, even if they are coming from events with semi-leptonic decays of the other B meson.

For further purposes in this analysis we also define a subset of all signal candidates, which come from events where the companion B meson decayed hadronically and all of its particles were taken into account correctly. We only allow for missing photons, since they are frequently irradiated due to brehmsstrahlung effects and don't have such a big impact on the 4-momentum of the final candidate. We denote this subset as *perfect* signal.

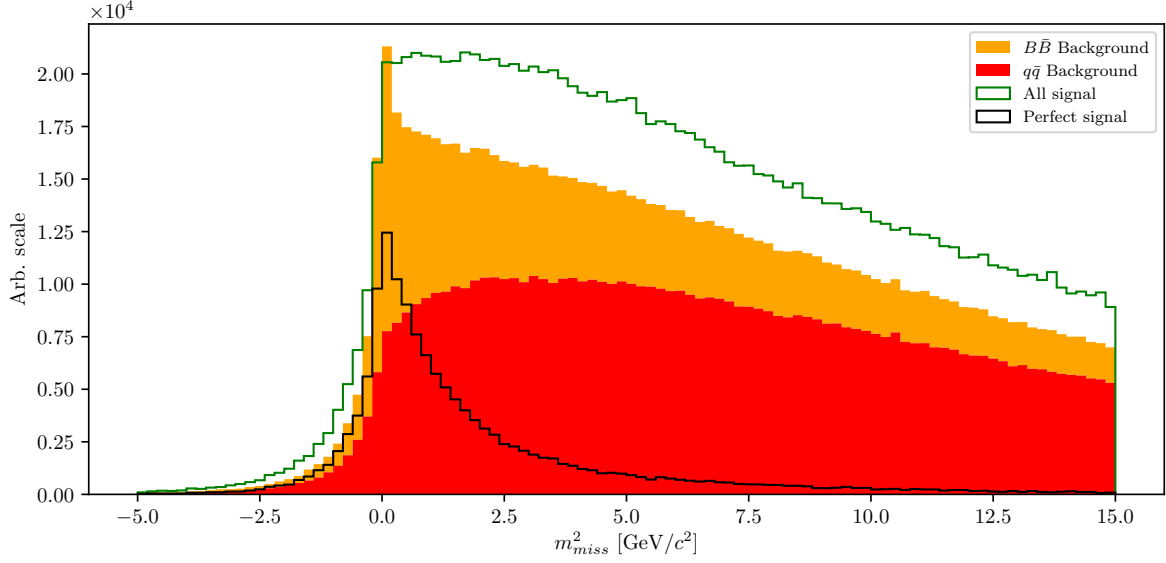


Figure 4.9: Squared missing mass distribution for $q\bar{q}$ and $B\bar{B}$ background in filled histograms. All signal (green) and perfect signal (black) are scaled up equally.

The main uncertainty in neutrino four-momentum, defined as in Eq. (4.10) comes from energy uncertainty. It is a common practice to substitute the missing energy with the magnitude of the missing momentum, since the momentum resolution from the measurement is much better, thus redefining the neutrino four-momentum to

$$p_\nu = (|\vec{p}_{miss}|, \vec{p}_{miss}), \quad (4.12)$$

which fixes the neutrino mass to 0 GeV/ c^2 .

The newly defined neutrino four-momentum can be added to the four-momentum of the $Y(KK\ell)$ candidate to obtain the full B meson four-momentum and calculate the traditional M_{BC} and ΔE variables

$$\Delta E = E_B - E_{CMS}/2, \quad (4.13)$$

$$M_{BC} = \sqrt{(E_{CMS}/2)^2 - |\vec{p}|^2}. \quad (4.14)$$

Since the final fit will be performed over ΔE and M_{BC} , we define the fit region

- $M_{BC} \in [5.1, 5.295]$ GeV/ c^2 ,
- $\Delta E \in [-1.0, 1.3]$ GeV.

Figure 4.10 shows the distributions of ΔE (left) and M_{BC} (right) for signal and major types of background after the pre-cuts. Both signal components are scaled up with

respect to the background components, but are in proper scale one to another. The effects of missing particles are clearly seen based on the shape difference between all and perfect signal.

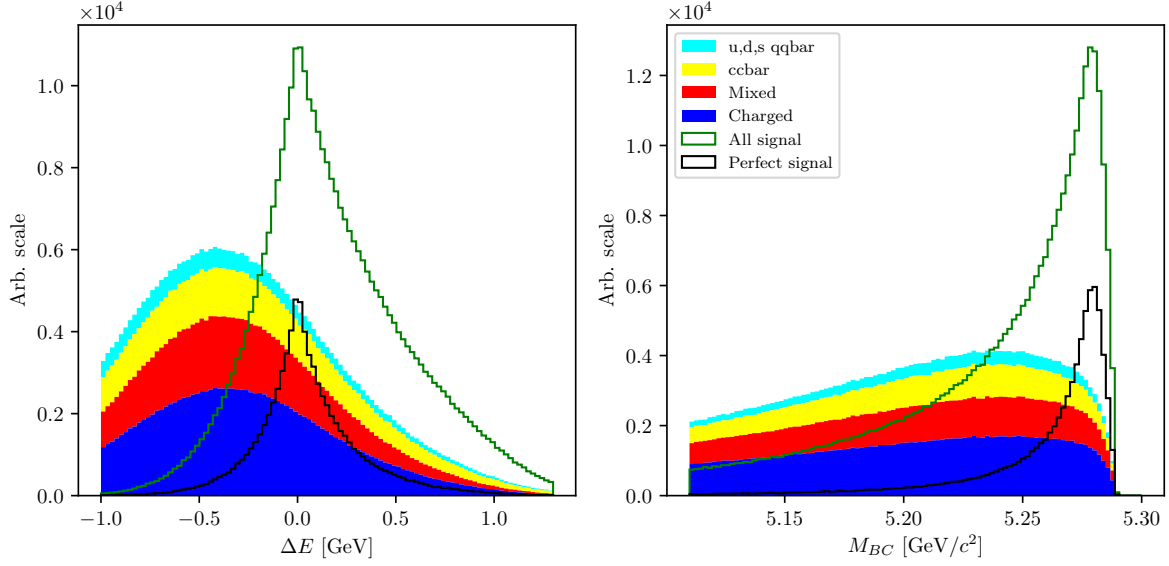


Figure 4.10: Distributions of ΔE (left) and M_{BC} (right) for signal and major types of background after the precuts. Both signal components are scaled up with respect to the background components, but are in proper scale one to another. The perfect signal has a much better resolution in both distributions, since the event is perfectly reconstructed.

4.5 q^2 calculation

Momentum transfer squared, q^2 , is the squared Lorentz invariant of the four-momentum which is transferred from the B meson to the W boson. There are several possible calculations of this variable which offer different resolution. The following describes the calculation of q^2 which follows the calculation from [] and yields the best resolution.

For correctly reconstructed events Eq. (4.13) satisfies the condition $\Delta E \approx 0$ within precision. It is possible to rescale the neutrino energy in such way that we fix ΔE to zero, meaning

$$\Delta E' = (E_Y + \alpha E_\nu) - E_{CMS}/2 = 0. \quad (4.15)$$

and calculate an adapted version of M_{BC}

$$M'_{BC} = \sqrt{(E_{CMS}/2)^2 - |\vec{p}_Y + \alpha \vec{p}_\nu|^2}. \quad (4.16)$$

The neutrino momentum resolution dominates the ΔE uncertainty [], so the correction factor α reduces this source of uncertainty.

A second correction can be applied by rotating the direction of the neutrino momentum by a small angle with respect to the reconstructed one. Such an angle is chosen in order to fix the value of M'_{BC} to the nominal mass of the B meson, m_B .

The corrected neutrino momentum is then solely used for the q^2 calculation, alongside the reconstructed lepton four-momentum, as

$$q^2 = q'^2 = (p_\ell + p_\nu)^2. \quad (4.17)$$

The q^2 distribution and its resolution are shown in Figure 4.11, along with additional versions of q^2 , with details of the calculation method in the caption.

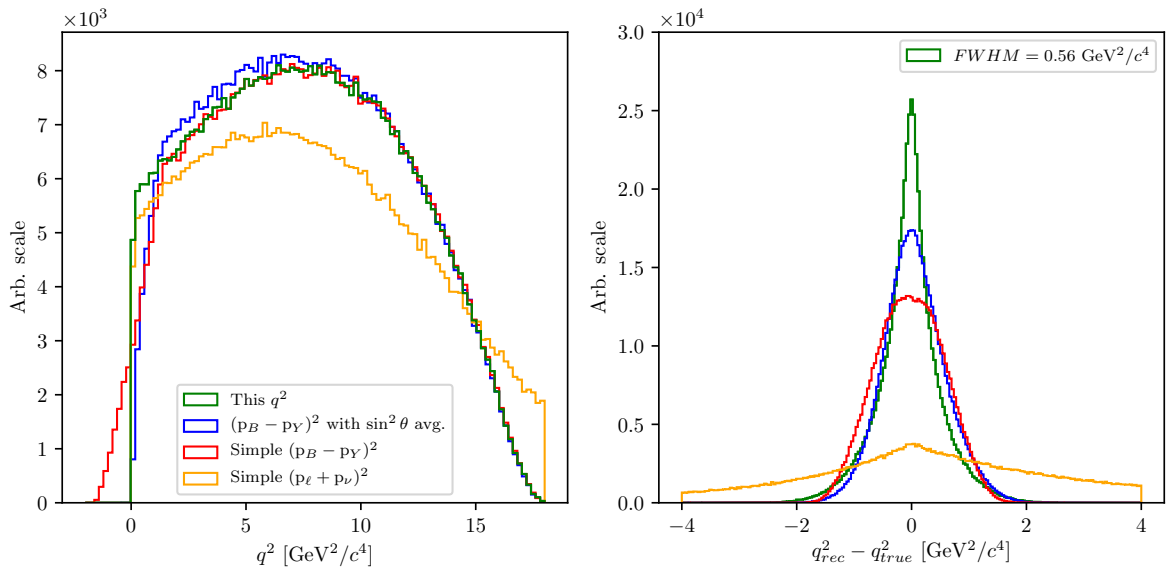


Figure 4.11: Distributions of q^2 (left) and q^2 resolution (right) for various methods of q^2 calculation. The blue distribution follows the procedure in [], whereas the red and the orange ones are straight-forward calculations with available information in the reconstruction. The q^2 calculation in red assumes a resting B meson in the CMS frame, and the calculation in orange uses the neutrino four-momentum from Eq. (4.10).

One must bare in mind that even though this calculation yields a precise result, this says nothing about the correctness of the q^2 model which was used (ISGW2) []. Since this decay has not been observed yet, we do not have a good description of the decay model, which is also a source of systematics in this analysis.

4.6 Event categorization

The missing information due to an escaping neutrino in our reconstructed channel is replaced by information from the companion B meson. Since this is an untagged reconstruction, the quality of the companion B meson affects the properties of signal candidate. Perfect reconstruction of a hadronically decaying companion B meson results in pronounced peaks at $\Delta E \approx 0$, $m_{miss}^2 \approx 0$ and $M_{BC} \approx m_B$, while imperfect reconstruction due to any kind of missing particles produces long tails and/or a shift from the desired value in the mentioned distributions. These effects are undesired, since they make it harder to separate signal from background.

To remedy this, we split our signal candidates into 4 categories and work only with the best category. For categorization we use splitting in two ways. First, we look at the charge product of the reconstructed B meson and the ROE object. For correctly reconstructed events, this should have a value of

$$q_{B_{sig}} q_{B_{comp}} = -1, \quad (4.18)$$

however, this value is distributed due to missing charged particles in the reconstruction. Secondly, we train an MVA classifier based on ROE object properties in order to recognize companion B mesons, which decayed hadronically. The details of the training are described in subsection 4.6.1.

We define the 4 categories in the following way

1. ([I] $q_{B^\pm} q_{B^\mp} = -1$ and $BDT_{had.} > 0.57$,
2. ([II] $q_{B^\pm} q_{B^\mp} \neq -1$ and $BDT_{had.} > 0.57$,
3. ([III] $q_{B^\pm} q_{B^\mp} = -1$ and $BDT_{had.} \leq 0.57$,
4. ([IV] $q_{B^\pm} q_{B^\mp} \neq -1$ and $BDT_{had.} \leq 0.57$.

Different categories for all signal candidates after the pre-cuts are shown in Figure 4.12. Category I has the best resolution, so we use it in our analysis from this point on.

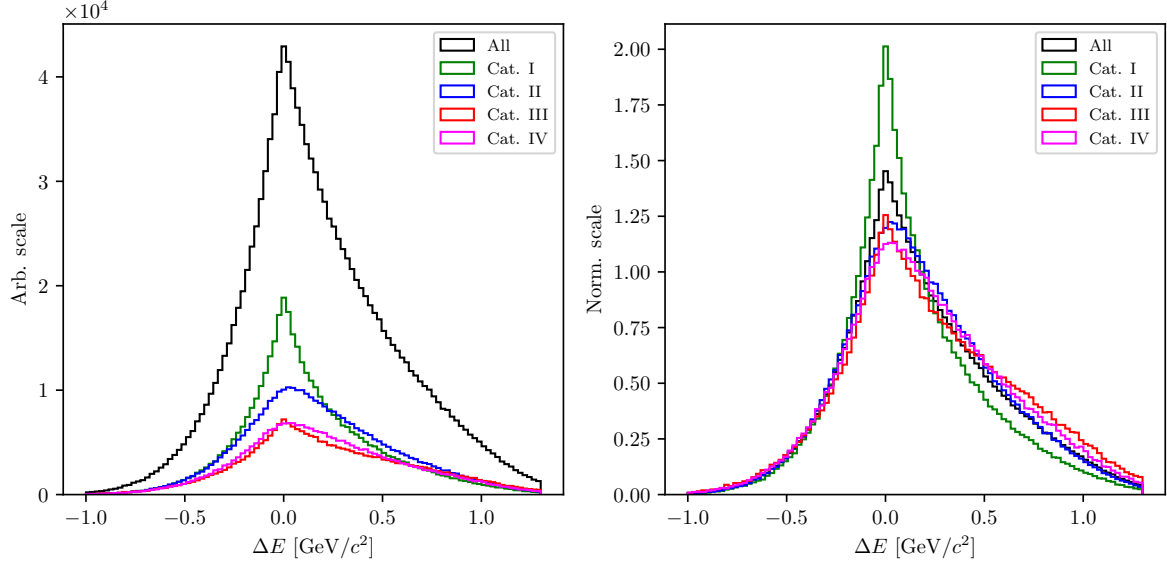


Figure 4.12: Categorization of signal candidates based on the charge product of both B mesons in the event and the MVA output for recognizing hadronic decays of the companion B meson. The plot on the left shows the distributions in an arbitrary scales, while the plot on the right shows the normalized distributions.

4.6.1 Hadronic decay MVA training

In order to train an MVA classifier to recognize events with hadronically decaying B mesons in the ROE side, we prepare a dataset of

- 249385 target candidates,
- 223681 background candidates,

where the definition of target is a ROE which corresponds to a hadronically decayed companion B meson.

The input variables used in this MVA are ROE specific and do not depend on the signal side. They are

- angle between tracks,
- track quantities
 - $P(\chi^2, DOF)$ of the tracks fit from the ROE side,

- K and ℓ FlavorTagger variables,
- Charge of the ROE,
- $\cos \theta$ of the ROE momentum in the CMS frame,
- Number of tracks in ROE,
- Number of distant tracks in ROE, which don't pass requirements from Section 4.8.

The classifier output is shown in Figure 4.13 (left). Candidates which populate the region with low values of the classifier output are more likely to come from semi-leptonic decays, so we want to discard those candidates. When optimizing the FOM , we redefine the S in Eq. (4.1) to a correctly reconstructed signal candidate with a hadronically decayed companion B meson. This FOM optimization, shown in Figure 4.13 (right), yields the optimal cut

- $BDT_{had.} > 0.57$.

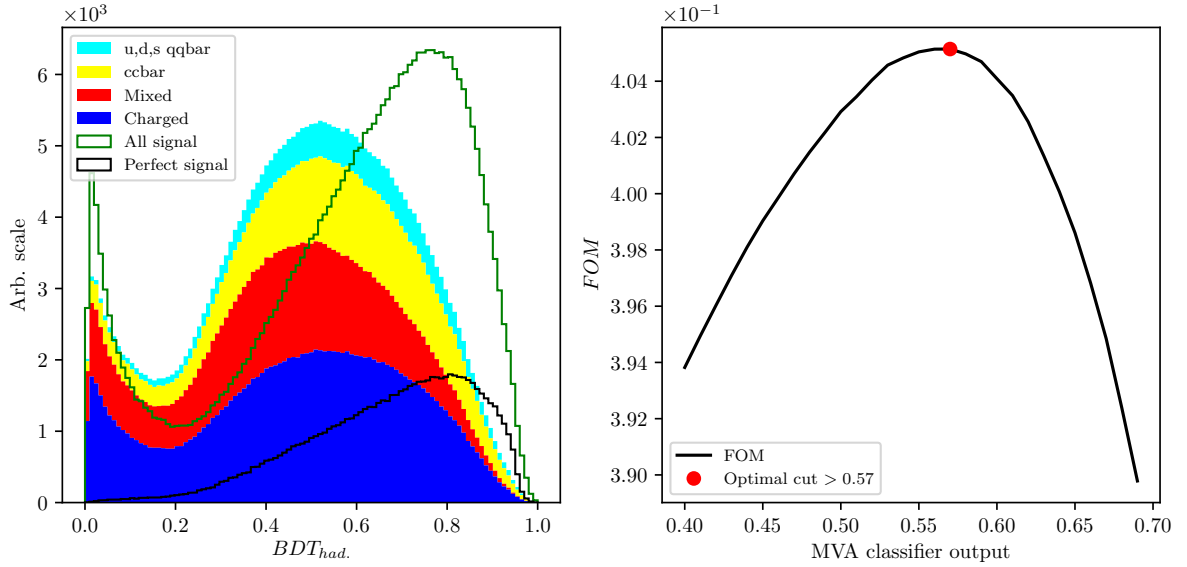


Figure 4.13: Hadronic MVA classifier output for major types of background and scaled-up signal (left), and FOM optimization on this cut for correctly reconstructed signal candidates with hadronically decayed companion B meson (right).

4.7 Signal region definition

Since signal candidates are now categorized, we can define a signal region, where most of our perfectly reconstructed candidates lie. We use this region for optimizations of all cuts in the following steps of background suppression in chapter 6. The 2D *FOM* optimization of the optimal M_{BC} and ΔE is shown in Figure 4.14. The signal region is defined as

- $M_{BC} > 5.270 \text{ GeV}/c^2$,
- $|\Delta E| < 0.166 \text{ GeV}$.

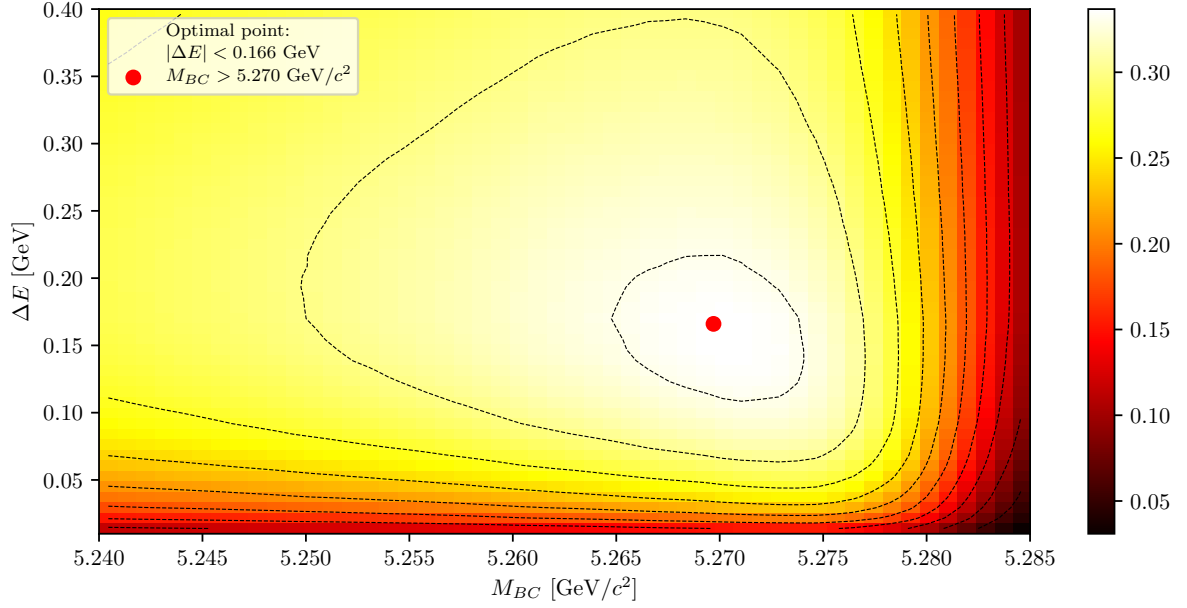


Figure 4.14: 2D *FOM* optimization of the signal region definition, where the signal in the optimization was represented by perfectly reconstructed candidates.

Lastly, we can tighten the cut on m_{miss}^2 , which we intentionally left loose before the signal categorization. With the *FOM* optimization of perfectly reconstructed candidates inside the signal region, shown in Figure 4.15, the optimal cut on m_{miss}^2 is

- $|m_{miss}^2| < 1.1 \text{ GeV}/c^2$.

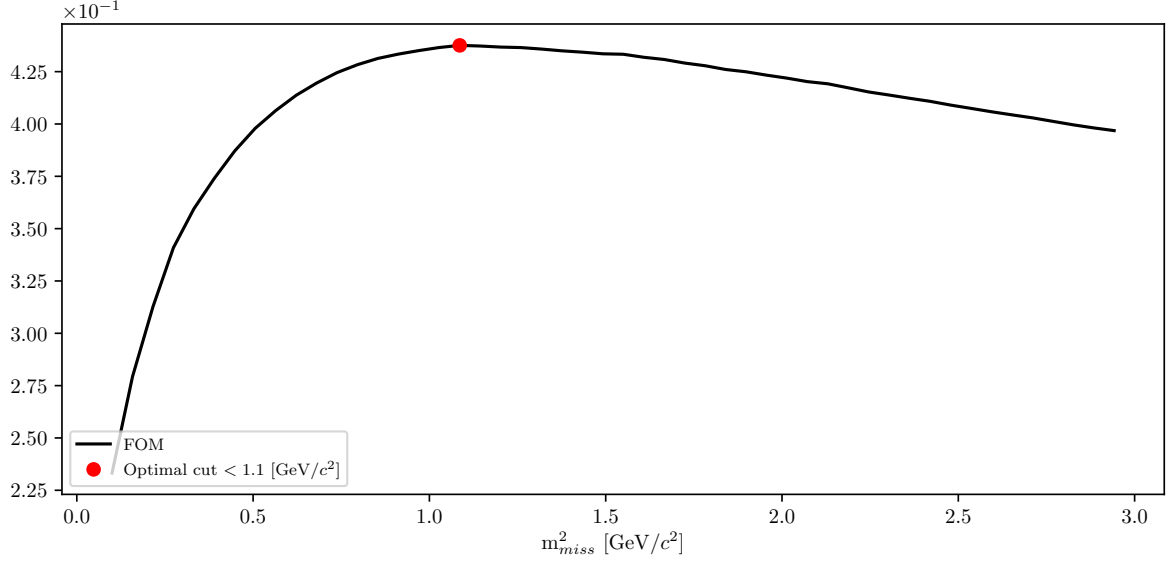


Figure 4.15: *FOM* optimization of the optimal m_{miss}^2 cut in the signal region.

4.8 Selection summary

In this section one can find the summary of all selection cuts in the event reconstruction, from FSP particles up to the B meson.

- FSP particles:
 - Electrons: $|d_0| < 0.1 \text{ cm}$, $|z_0| < 1, 5 \text{ cm}$, $p > 0.6 \text{ GeV}/c$,
 $p_{CMS} \in [0.4, 2.6] \text{ GeV}/c$, $eID > 0.9$,
 - Muons: $|d_0| < 0.1 \text{ cm}$, $|z_0| < 1, 5 \text{ cm}$, $p_{CMS} \in [0.6, 2.6] \text{ GeV}/c$,
 $\mu ID > 0.9$,
 - Kaons: $|d_0| < 0.15 \text{ cm}$, $|z_0| < 1, 5 \text{ cm}$, $p_{CMS} < 2.5 \text{ GeV}/c$,
 $K/\pi \text{ ID} > 0.6$, $K/p \text{ ID} > 0.1$,
- B meson candidates:
 - Before ROE clean-up: $P(\chi^2, DOF) > 1 \times 10^{-3}$, $|\cos \theta_{BY}| < 1$, $|m_{miss}^2| < 7 \text{ GeV}/c^2$,
 - After ROE clean-up: $\Delta E \in [-1.0, 1.3] \text{ GeV}$, $M_{BC} \in [5.1, 5.295] \text{ GeV}/c^2$,
 - After signal categorization: $q_{B_{sig}} q_{B_{comp}} = -1$, $BDT_{had.} > 0.57$, $|m_{miss}^2| < 1.1 \text{ GeV}/c^2$.

5 Rest of event clean-up

Continuing from Section 4.3, the description of the ROE clean-up process is described here.

5.1 Setting up the MVA

Training the MVA classifiers follows the same recipe for all the steps in this chapter. For each step we run B meson reconstruction on Signal MC with a generic companion B meson. This way the produced weight files are less likely to be signal-side dependent and can be used also for untagged analyses of other decays. For every correctly reconstructed signal B meson we save the necessary information for each MVA step (i.e. properties of ROE clusters). Only correctly reconstructed candidates are chosen here, to prevent leaks of information from the signal side to the ROE side.

For each clean-up step a dataset was prepared for the training of "signal" (target) against "background". The Fast-BDT (FBDT) [1] algorithm was used as the MVA classifier and the following hyper-parameters were chosen for optimization

- **nTrees**: the number of trees in the FBDT forest,
- **nLevels**: the number of levels in each FBDT tree.

Figure 5.1 shows a graphical interpretation of the FBDT forest with **nTrees** and **nLevels**. In all cases the hyper-parameters were optimized with a grid-search method in the hyper-parameter phase-space. Additionally, we perform a k -fold cross-validation ($k = 5$) for each hyper-parameter set, where we cycle through the 5 parts of the dataset and use 4 parts of the dataset for training, whereas the remaining part is used for validation. This minimizes the bias towards the validation sample when optimizing the hyper-parameters [1].

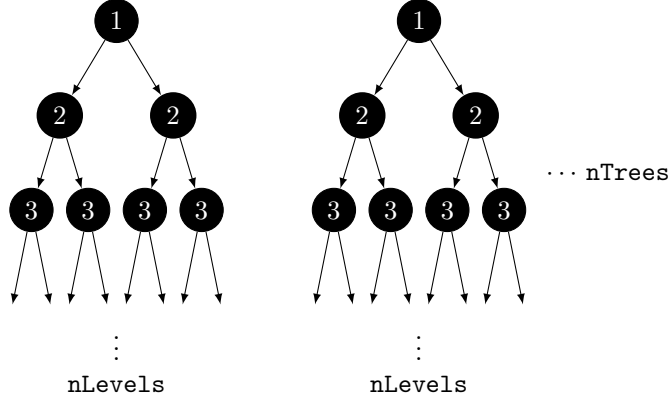


Figure 5.1: A schematic of a BDT forest with $nTrees$, each tree having a depth of $nLevels$.

5.2 Clusters clean-up

Photons originate from the IP region, travel to the ECL part of the detector in a straight line and produce a cluster. The direction of the photon is determined via the location of the cluster hit in the ECL and the energy of the photon is directly measured via the deposited energy. This way the four-momentum of photons is determined and used in Eq. (4.8).

Most of the photons in events with B mesons come from $\pi^0 \rightarrow \gamma\gamma$ decays. However, a lot of hits in the ECL are also created by photons coming from beam-induced background or secondary interactions with the detector material. Such photons add extra energy and momentum which spoils our measured quantities.

In the first step of the clusters clean-up we train an MVA which recognizes good π^0 candidates. The output of this classifier is then applied to photons and represents a sort of a π^0 origin probability, which is used as an additional classifier variable in the next step of the clean-up.

5.2.1 π^0 MVA training

The dataset of π^0 candidates contains

- 387125 target candidates,
- 416019 background candidates,

where the definition of target is that both photon daughters that were used in the reconstruction of the π^0 are actual photons and real daughters of the π^0 particle. We use π^0 candidates from the converted Belle particle list and select those with invariant mass in the range of $M \in [0.10, 0.16]$ GeV. After that we perform a mass-constrained fit on all candidates, keeping only the ones for which the fit converged.

The input variables used in this MVA are

- p and p_{CMS} of π^0 and γ daughters,
- fit prob. of the mass-constrained fit, invariant mass and significance of mass before and after the fit,
- angle between the photon daughters in the CMS frame,
- cluster quantities for each daughter photon
 - E_9/E_{25} ,
 - theta angle,
 - number of hit cells in the ECL,
 - highest energy in cell,
 - energy error,
 - distance to closest track at ECL radius.

The classifier output variable is shown in Figure 5.2.

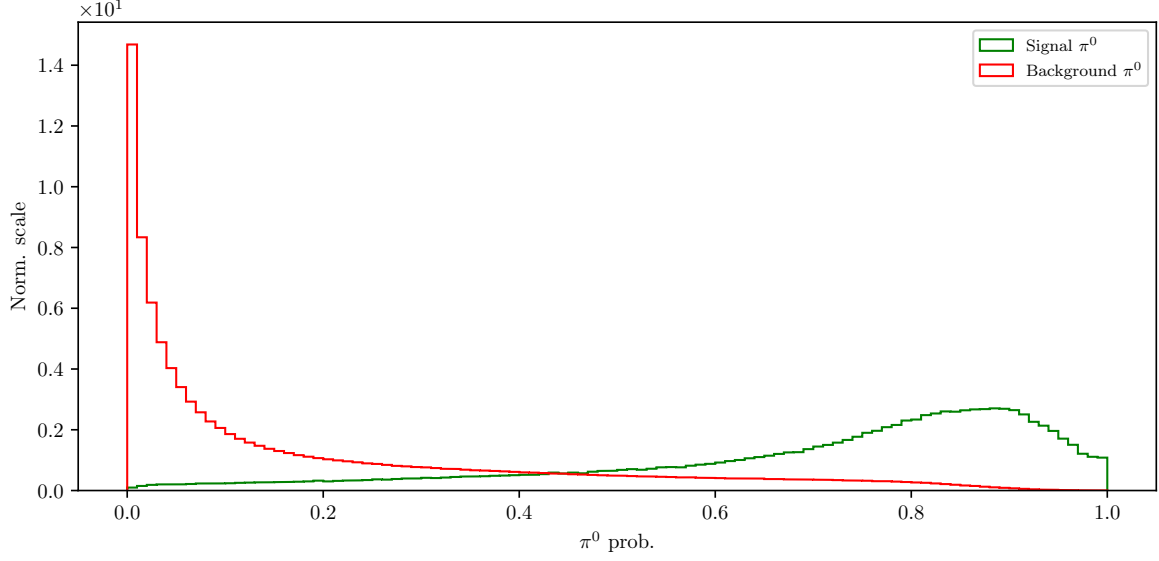


Figure 5.2: Classifier output of the π^0 training for signal and background π^0 candidates.

The distributions for all input variables and their correlations for signal and background candidates can be found in Appendix X for all steps of the ROE clean-up.

5.2.2 γ MVA training

In this MVA training we take the π^0 classifier output of the previous training as an input in order to train a classifier to distinguish between good and bad photons. The π^0 probability information from the previous step is applied to all photon pairs which pass the same π^0 cuts as defined in the previous step. Since it's possible to have overlapping pairs of photons, the π^0 probability is overwritten in the case of a larger value, since this points to a greater probability of a correct photon combination. On the other hand, some photon candidates fail to pass the π^0 selection, these candidates have a fixed value of π^0 probability equal to zero.

The dataset of γ candidates contains

- 324781 target candidates,
- 333353 background candidates,

where the definition of target is that the photon is an actual photon which is related to a primary MC particle. This tags all photon particles from secondary interactions as background photons. We use the converted γ candidates from the existing Belle particle

list.

The input variables used in this MVA are

- p and p_{CMS} of γ candidates,
- π^0 probability,
- cluster quantities
 - E_9/E_{25} ,
 - theta angle,
 - number of hit cells in the ECL,
 - highest energy in cell,
 - energy error,
 - distance to closest track at ECL radius.

The classifier output variable is shown in Figure 5.3.

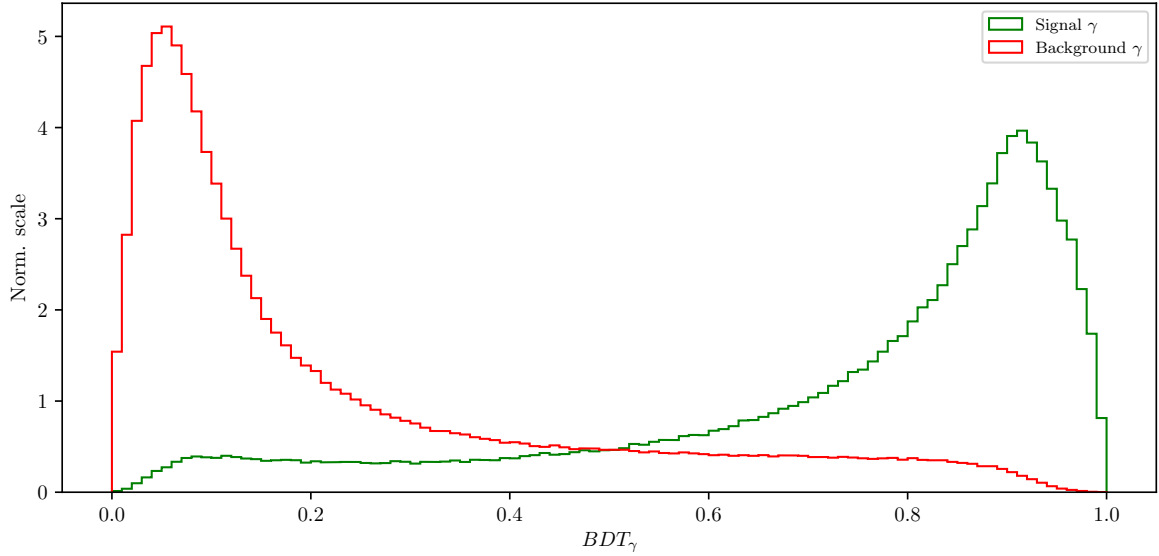


Figure 5.3: Classifier output of the γ training for signal and background γ candidates.

With the final weights for photon classification in hand, we apply them to the photon particle list. The cut optimization is shown in Figure 5.4 (left), with the optimal cut on

the γ classifier output at

- $BDT_\gamma > 0.519$.

Figure 5.4 (right) shows the LAB frame momentum of the photons before and after the cut in logarithmic scale. The signal efficiency and background rejection at this clean-up cut are

- Signal efficiency: $\epsilon_{SIG} = 82.6 \%$,
- Background rejection: $1 - \epsilon_{BKG} = 81.7 \%$.

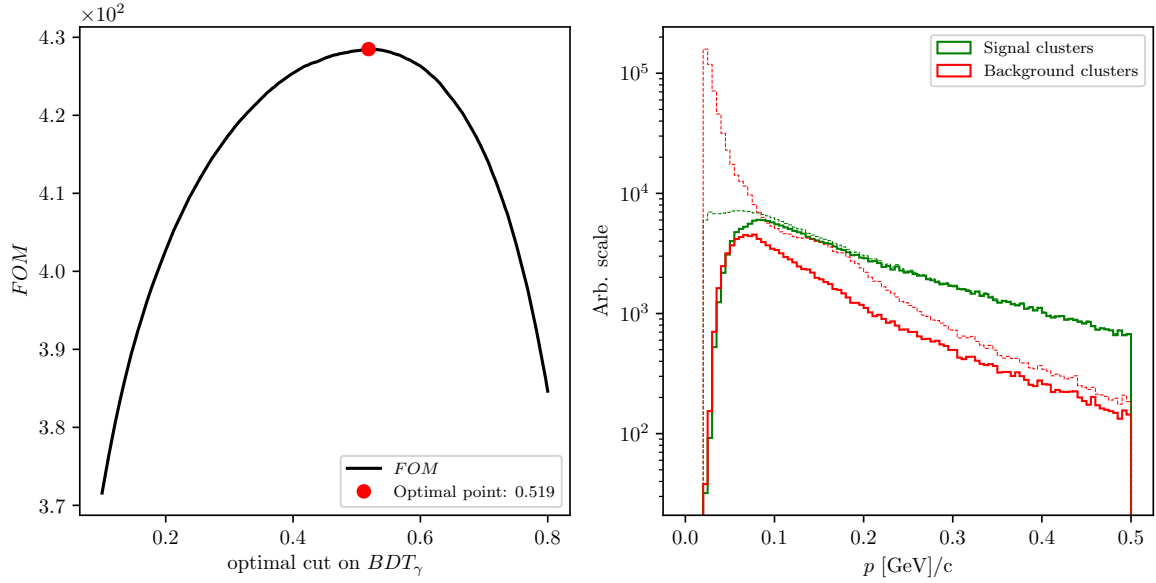


Figure 5.4: The FOM of the classifier output optimization (left) and momentum magnitude in the LAB frame of signal and background photon candidates before and after the optimal cut (right).

The event is now considered to be clean of extra clusters.

5.3 Tracks clean-up

Charged particles leave hits in the detector, which are then grouped into tracks by advanced tracking algorithms. The track is fitted and the track momentum is determined. With the help of particle identification information (PID), we are able to make an intelligent decision about the mass hypothesis of the particle and thus reconstruct the charged particle's four-momentum, which is then added in the loop in Eq. (4.8).

Most of the quality (good) tracks, which come from physics event of interest, come from the IP region, where the collisions occur. Cleaning up the tracks is a more complex procedure than cleaning up the clusters. The following facts need to be taken into account

- (a) good tracks can also originate away from the IP region, due to decays of long-lived particles, such as $K_S^0 \rightarrow \pi^+\pi^-$,
- (b) charged particles from background sources produce extra tracks, or duplicates,
- (c) low momentum charged particles can curl in the magnetic field and produce multiple tracks,
- (d) secondary interactions with detector material or decays of particles in flight can produce "kinks" in the flight directory, resulting in multiple track fit results per track.

Schematics of all the cases mentioned above are shown in Figure 5.5.

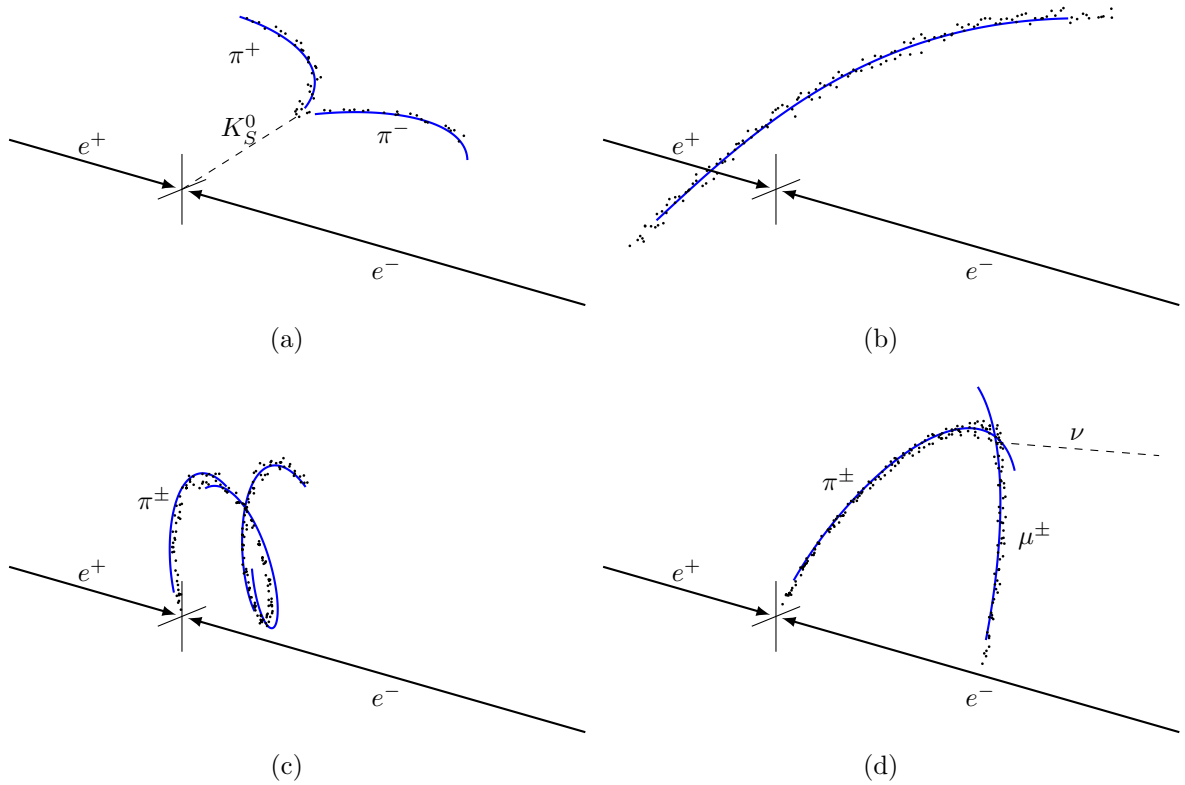


Figure 5.5: (a) Tracks from long-lived neutral particles, which decay away from the IP region, (b) Random tracks from background which are reconstructed, (c) Low-momentum particles which curl in the magnetic field, (d) in-flight decays of particles, which produce a kink in the trajectory.

It is obvious that tracks from the same momentum source should only be taken into account once, or, in case of background tracks, not at all. Such tracks will from this point on be denoted as *extra* tracks, because they add extra four-momentum to our final calculations in Eq. (4.8). At the same time, we have to take care that we don't identify *good* tracks as *extra* tracks. Both of these cases have negative impacts on the final resolution of all variables which depend on information from ROE.

5.3.1 Tracks from long-lived particles

The first step in tracks clean-up is taking care of tracks from long-lived particles. Here we only focus on K_S^0 , since they are the most abundant. This step is necessary because the π^\pm particles, coming from the K_S^0 decays, have large impact parameters, which is usually a trait of background particles. In order to minimize confusion from the MVA point-of-view, these tracks are taken into account separately.

We use the converted K_S^0 candidates from the existing Belle particle list and use a pre-trained Neural Network classifier result in order to select only the good K_S^0 candidates. Figure 5.6 shows the distribution of the K_S^0 invariant mass for signal and background candidates, before and after the classifier cut. The momentum of selected K_S^0 candidates is added to the ROE, while the daughter tracks are discarded from our set.

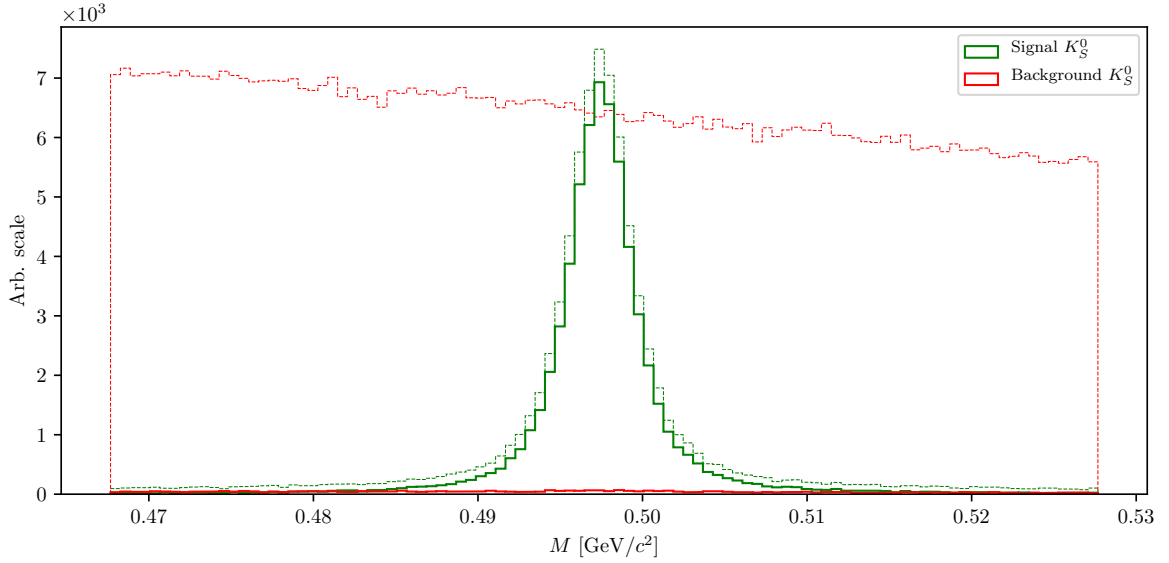


Figure 5.6: Invariant mass of the K_S^0 candidates before (dashed lines) and after (solid lines) the cut on the Neural Network classifier for signal (green) and background candidates (red). Signal peaks at nominal K_S^0 mass, while background covers a wider region.

The signal efficiency and background rejection for K_S^0 candidates after this cut and on the full range are

- Signal efficiency: $\epsilon_{SIG} = 80.7 \%$,
- Background rejection: $1 - \epsilon_{BKG} = 99.4 \%$.

5.3.2 Duplicate tracks

All good tracks at this point should be coming from the IP region, since we took care of all the good tracks from long-lived particle decays, therefore we apply a cut on impact parameters for all the remaining tracks

- $|d_0| < 10 \text{ cm}$ and $|z_0| < 20 \text{ cm}$

and proceed with the clean-up of track duplicates.

Defining a duplicate track pair

In this step we wish to find a handle on secondary tracks from low momentum curlers and decays in flight. The main property for these cases is that the angle between such two tracks is very close to 0° or 180° , since tracks deviate only slightly from the initial direction, but can also be reconstructed in the opposite way. Figure 5.7 shows the distribution of the angle between two tracks in a single pair for random track pairs and duplicate track pairs, where the latter were reconstructed as two same-sign or opposite-sign tracks.

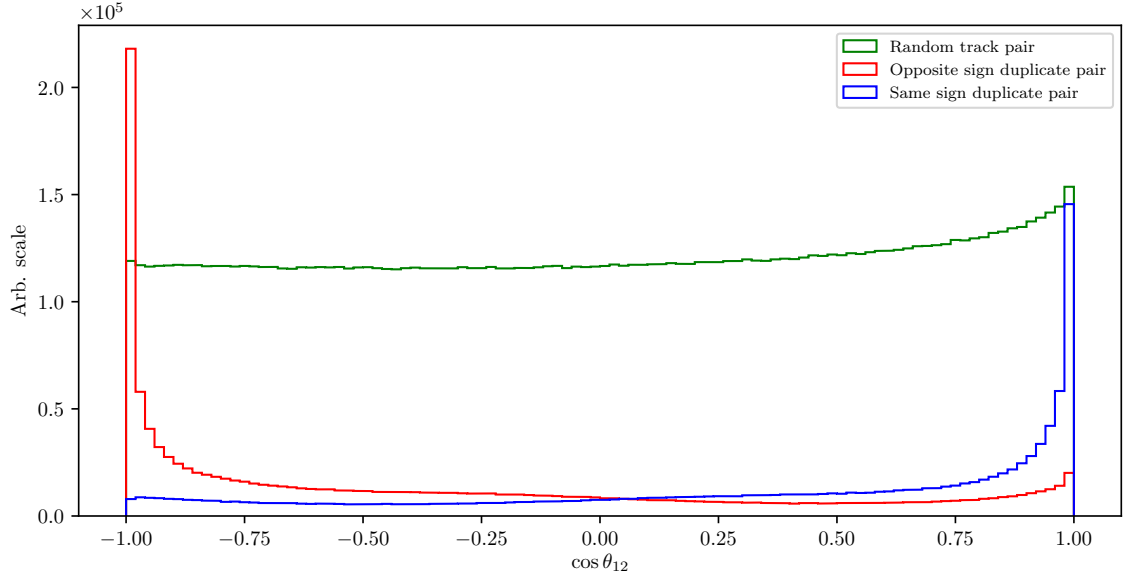


Figure 5.7: Distribution of the angle between two tracks in a single pair for random track pairs (green) and duplicate track pairs, where the latter were reconstructed as two same-sign (blue) or opposite-sign tracks (red).

If the particle decayed mid-flight or produced multiple tracks due to being a low-momentum curler, then, as the name suggests, these particles most likely had low momentum in the transverse direction, p_T . Since both tracks originate from the same initial particle, the momentum difference should also peak at small values. Figure 5.8 shows the momentum and momentum difference of tracks which belong to a random or a duplicate track pair.

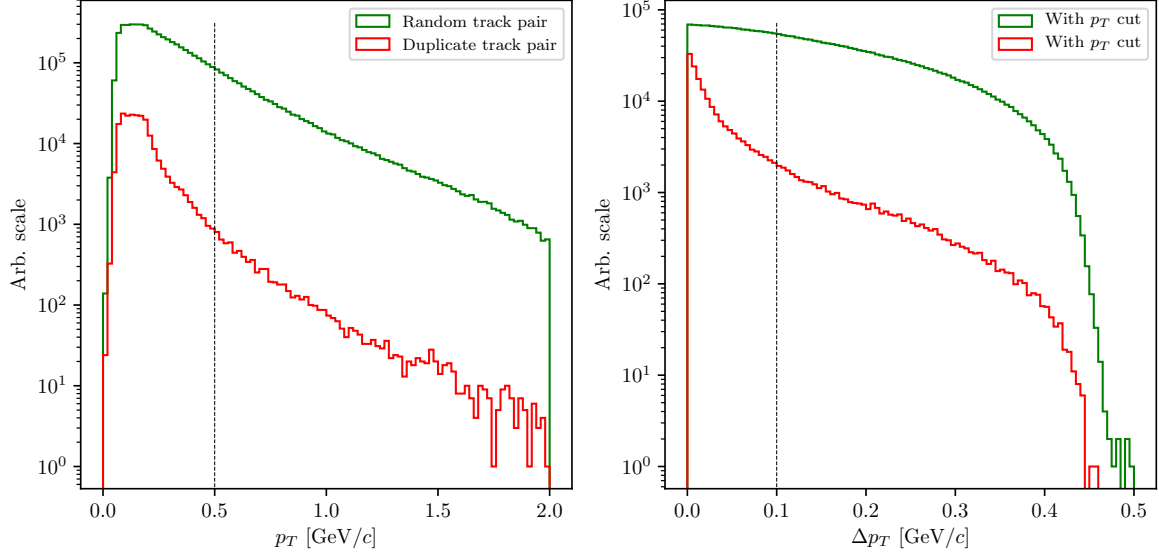


Figure 5.8: Distribution of transverse momentum p_T (left) and transverse momentum difference Δp_T (right) for all tracks coming from random (green) or duplicate track pairs (red). The plot on the right already includes the cut on p_T from the plot on the left.

We impose a cut of

- $p_T < 0.5 \text{ GeV}/c$,
- $|\Delta p_T| < 0.1 \text{ GeV}/c$,

in order to cut down the number of random track pairs, while retaining a high percentage of duplicate track pairs. After all the cuts defined in this chapter, the final distribution of the angle between two tracks is shown in Figure 5.9.

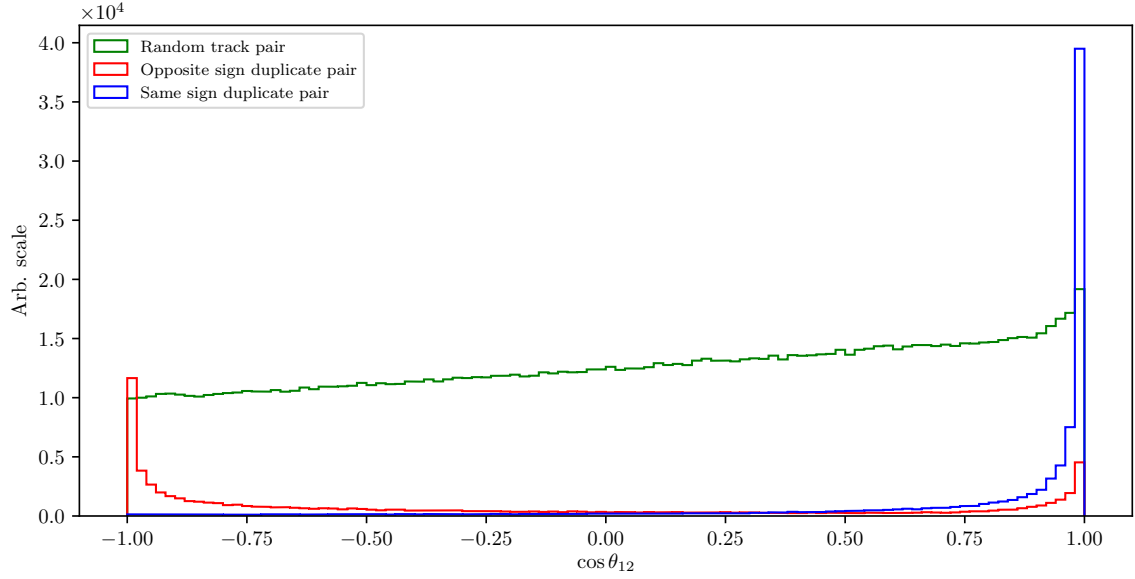


Figure 5.9: Distribution of the angle between two tracks in a single pair after applying the selection cuts defined in this subsection. The distributions are shown for random track pairs (green) and duplicate track pairs, where the latter were reconstructed as two same-sign (blue) or opposite-sign tracks (red).

Training the duplicate track pair MVA

This final sample of track pairs is now fed into an MVA, which is trained to recognize duplicate track pairs over random ones. The dataset contains

- 215601 target candidates,
- 311124 background candidates,

where the definition of target is that the track pair is a duplicate track pair.

The input variables used in this MVA are

- angle between tracks,
- track quantities
 - impact parameters d_0 and z_0 ,
 - transverse momentum p_T ,

- helix parameters and helix parameter errors of the track,
- track fit p -value,
- number of hits in the SVD and CDC detectors

The classifier is able to distinguish between random and duplicate track pairs in a very efficient manner, as shown in Figure 5.10.

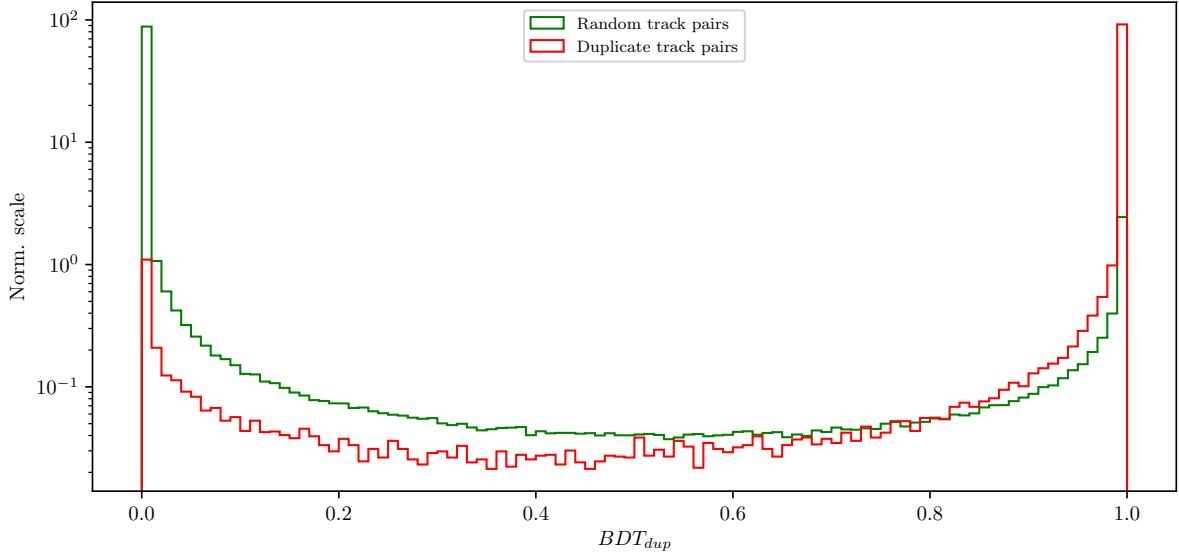


Figure 5.10: Classifier output of the track pair training for random track pairs and duplicate track pairs.

The FOM function for optimal cut selection is shown in Figure 5.11 (left), along with the angle between the two tracks before and after the optimal cut (right). The optimal cut for duplicate track selection is

- $BDT_{duplicate} > 0.99915$.

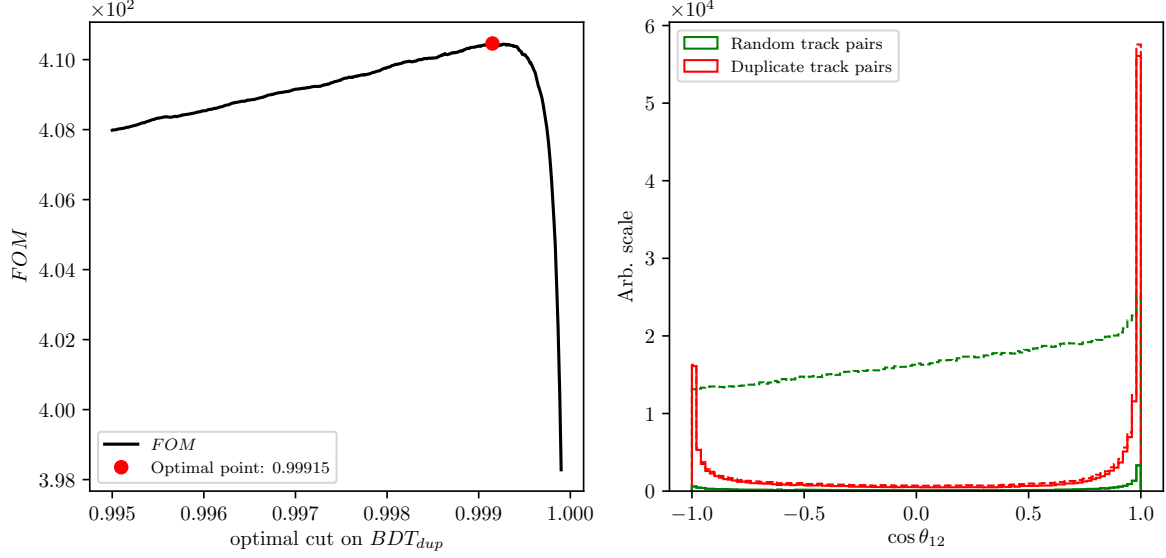


Figure 5.11: The optimization of the FOM function for the cut on classifier output (left) and distribution of the angle between two tracks in a single pair before (dashed) and after (solid) applying the optimal cut on the output classifier for random and duplicate track pairs (right).

Defining duplicate tracks

What remains now is to decide which track from the duplicate track pair to keep and which to discard. For this purpose we apply duplicate pair-level information to each track in the pair in the form of

$$\Delta f = f_{this} - f_{other}, \quad (5.1)$$

where f is an arbitrary variable from the list of track quantities in Subsection 5.3.2. From the point-of-view of *this* track, a track is more *duplicate*-like if the following is true

- $\Delta d_0, \Delta z_0 > 0$ (*this* track further away from the IP region),
- $\Delta p_T, \Delta p_Z < 0$ (*this* track has lower momentum),
- $\Delta N_{SVD}, \Delta N_{CDC} < 0$ (*this* track has less hits in the SVD and CDC),

Additionally we define an MC truth variable

$$\Delta\chi^2 = \chi_{this}^2 - \chi_{other}^2, \quad \chi^2 = \sum_{i=x,y,z} \frac{(p_i - p_i^{MC})^2}{\sigma(p_i)^2}, \quad (5.2)$$

where we compare all components of track momentum to the true values. If $\Delta\chi^2 > 0$, then *this* track has a higher probability of being a duplicate track and should be discarded.

However, it turns out that solving this problem is not as simple as discarding one track and keeping the other one. An additional complication here is that we can have more than one extra track from the same initial particle, which leads to track pairs where both tracks are track duplicates. For example, if we have the following case

$$\begin{aligned} t_1 &: \text{good track,} \\ t_2 &: \text{extra track,} \\ t_3 &: \text{extra track,} \\ \text{pair}_1 &: (t_1, t_2), \\ \text{pair}_2 &: (t_1, t_3), \\ \text{pair}_3 &: (t_2, t_3), \end{aligned}$$

where t_1 is the original track and t_2 and t_3 are extra tracks, with t_3 being even more duplicate-like with respect to t_2 . Here tracks t_2 and t_3 should be discarded while t_1 should be kept. We can achieve this if we overwrite existing pair-level information in the tracks for cases where the variable difference Δf is more duplicate-like. If we follow the same example, we could fill information about the property f in six different orders.

$$\begin{aligned} 1. & (t_1, t_2^*) \rightarrow (t_1, t_3^*) \rightarrow (t_2^*, t_3^*), \\ 2. & (t_1, t_2^*) \rightarrow (t_2^*, t_3^*) \rightarrow (t_1, t_3^*), \\ 3. & (t_1, t_3^*) \rightarrow (t_2, t_3^*) \rightarrow (t_1, t_2^*), \\ 4. & (t_1, t_3^*) \rightarrow (t_1, t_2^*) \rightarrow (t_2^*, t_3^*), \\ 5. & (t_2, t_3^*) \rightarrow (t_1, t_3^*) \rightarrow (t_1, t_2^*), \\ 6. & (t_2, t_3^*) \rightarrow (t_1, t_2^*) \rightarrow (t_1, t_3^*), \end{aligned}$$

where the "*" symbol denotes when a track is recognized as a duplicate track with respect to the other track. We see that no matter the order, both t_2 and t_3 get recognized as duplicate tracks correctly.

Training the duplicate track MVA

The training procedure is similar as before. The sample of tracks from duplicate track pairs is now fed into an MVA, which is trained to distinguish duplicate tracks from good tracks. The dataset contains

- 160146 target candidates,
- 128568 background candidates,

where the definition of target is that the track is a duplicate track.

The input variables used in this MVA are

- theta angle of the track momentum,
- track quantities
 - impact parameters d_0 and z_0 ,
 - momentum components p_T and p_z
 - number of hits in the SVD and CDC detectors
 - track fit p -value,
- pair-level information
 - $\Delta d_0, \Delta z_0, \Delta N_{CDC}, \Delta N_{SVD}, \Delta p_T, \Delta p_z$.

The classifier is shown in Figure 5.12. The weights from this training are applied to the tracks, where now each track has a certain probability of being a duplicate track. We now compare these values between both tracks in each track pair as

$$\Delta BDT_{final} = BDT_{final}^{this} - BDT_{final}^{other}, \quad (5.3)$$

which is again applied to all track pairs and overwritten for tracks which are more duplicate-like.

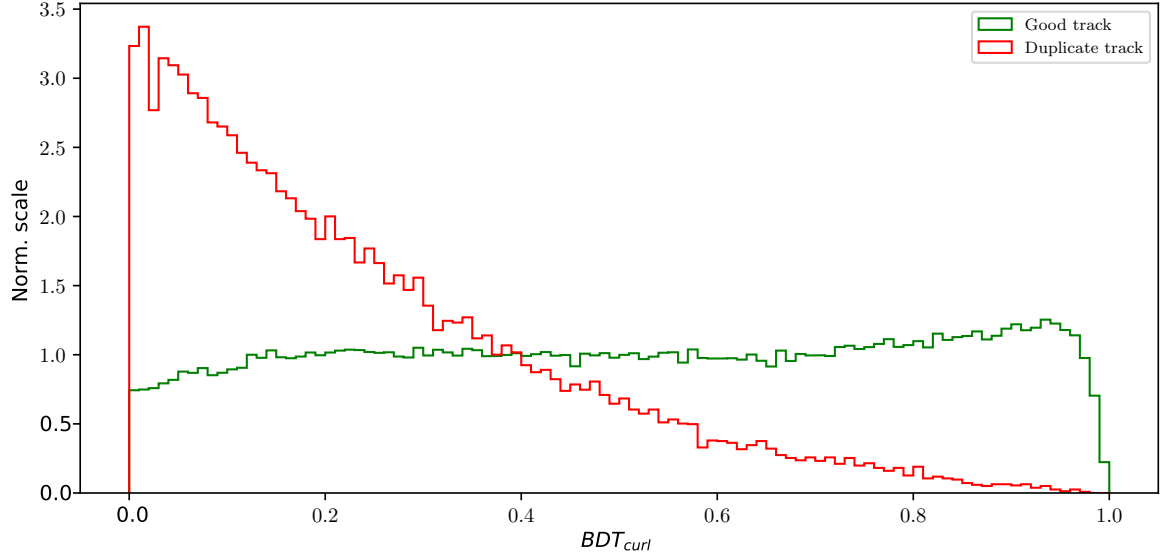


Figure 5.12: Classifier output of the MVA training for curling track recognition.

Finally, we select all duplicate tracks which survive the cut

$$\Delta BDT_{final} > 0 \quad (5.4)$$

and discard them from our ROE. We can check the performance of our duplicate track classifier by applying the procedure on a sample of duplicate track pairs and comparing the predicted result with the truth, based on Eq. (5.2). Table 5.1 shows the performance of the duplicate track recognition in the form of percentages of correctly and incorrectly identified duplicate and original tracks. The model seems to perform well and the event is now considered to be clean of duplicate tracks.

	Predicted duplicate track	Predicted good track
Duplicate track	84.63 %	20.95 %
Good track	15.37 %	79.05 %

Table 5.1: Ratios of correctly classified and misclassified tracks.

5.4 Belle clean-up

The clean-up, used standardly at Belle, is much simpler and relies only on a set of rectangular cuts for neutral particles as well as charged ones. In case of photons, a

single cut on photon energy is applied, depending on the region where the photon hit the relevant part of the detector. The photon cuts are summarized in Table 5.2.

	$17^\circ < \theta < 32^\circ$	$32^\circ < \theta < 130^\circ$	$130^\circ < \theta < 150^\circ$
E_γ	$> 100 \text{ MeV}$	$> 50 \text{ MeV}$	$> 150 \text{ MeV}$

Table 5.2: Photon selection for the Belle clean-up procedure. Different cuts are applied on photons in different parts of the detector

In case of tracks, pairs are selected which satisfy the following criteria:

- $p_T < 275 \text{ MeV}/c$,
- $\Delta p = |\mathbf{p}_1 - \mathbf{p}_2| < 100 \text{ MeV}/c$,
- $\cos \theta(\mathbf{p}_1, \mathbf{p}_2) < 15^\circ$ for same sign,
- $\cos \theta(\mathbf{p}_1, \mathbf{p}_2) > 165^\circ$ for opposite sign.

Of the two tracks, the one with a larger value of formula in Eq. 5.5 is discarded. The remaining tracks in the event then need to satisfy the conditions described in Table 5.3.

$$(\gamma|d_0|)^2 + |z_0|^2, \quad \gamma = 5. \quad (5.5)$$

	$p_T < 250 \text{ MeV}/c$	$250 \text{ MeV}/c < p_T < 500 \text{ MeV}/c$	$p_T > 500 \text{ MeV}/c$
$ d_0 $	$< 20 \text{ cm}$	$< 15 \text{ cm}$	$< 10 \text{ cm}$
$ z_0 $	$< 100 \text{ cm}$	$< 50 \text{ cm}$	$< 20 \text{ cm}$

Table 5.3: Photon selection for the Belle clean-up procedure. Different cuts are applied on photons in different parts of the detector

5.5 Clean-up results

In this section the results of the ROE clean-up are shown. It is clear that cleaning up the event affects the shape of various distribution, especially ΔE and M_{BC} , which we are most interested in. Since the reconstruction procedure includes cuts on the cleaned-up variables, this also results in an efficiency increase of the reconstructed sample.

We compare the clean-up setup, defined in this analysis, to the standard clean-up used by Belle, and to a default case, where no clean-up was applied at all. Figure 5.13 shows distributions of ΔE and M_{BC} for various clean-up setups after applying all pre-cuts up to the point in Section 4.3. We see an improvement in resolution as well as an increase in

efficiency in both observed variables for the case where the ROE clean-up is performed. Table 5.4 shows ratios of efficiencies and $FWHM$'s of ΔE with respect to the default case. While both the Belle and ROE clean-up increase efficiency, only ROE clean-up improves the resolution.

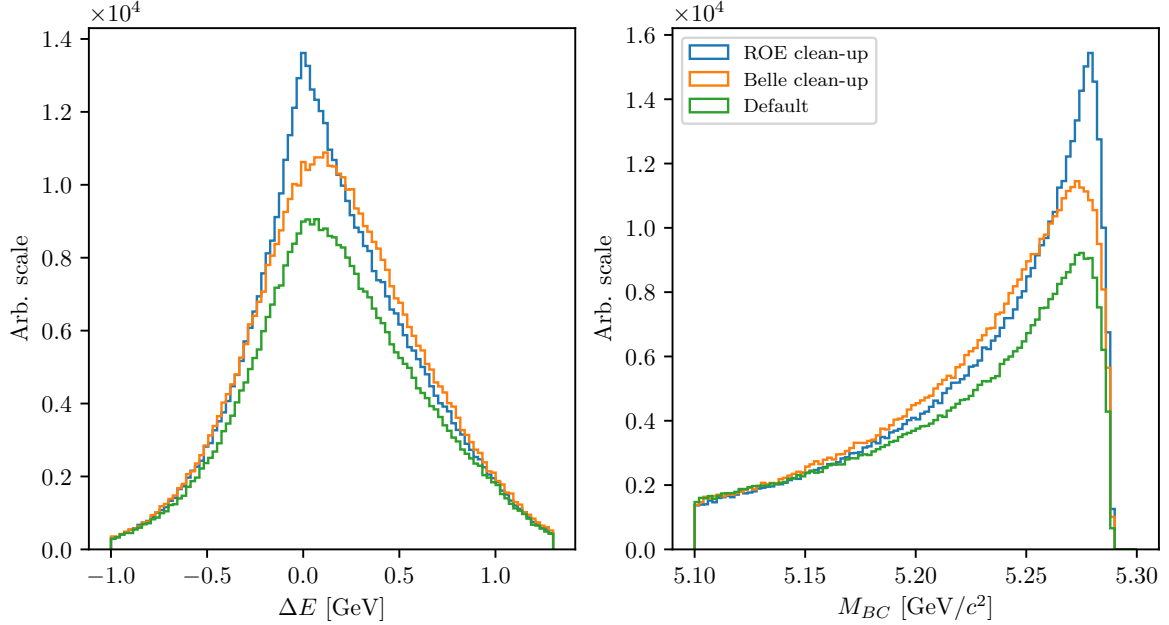


Figure 5.13: ΔE and M_{BC} distributions for various types of clean-up procedures. For ROE clean-up, the procedure seems to improve resolution as well as increase efficiency, compared to the default case.

	Efficiency ratio	FWHM ratio
Belle clean-up	121.90 %	102.60 %
ROE clean-up	123.40 %	77.92 %

Table 5.4: Comparison of efficiencies and $FWHM$'s of ROE and Belle clean-up setups with respect to the default case (no clean-up).

Another variable which heavily depends on the clean-up is the charge product of the signal and companion B meson candidate, already defined in Eq. (4.18), shown in Figure 5.14 for various clean-up procedures. As a cross-check, we can also look at ΔE and M_{BC} variables for each value of the charge product. These plots are shown in Figure 5.15 and they show a clear increase in efficiency and improvement in resolution for the correct value of the charge product in the case of the ROE clean-up. For other values of the charge product there also seems to be an increase in efficiency for both cases of clean-up, but neither help improve the resolution. This supports our choice of signal

categorization, defined in Section 4.6, where we select only candidates with the correct value of the charge product.

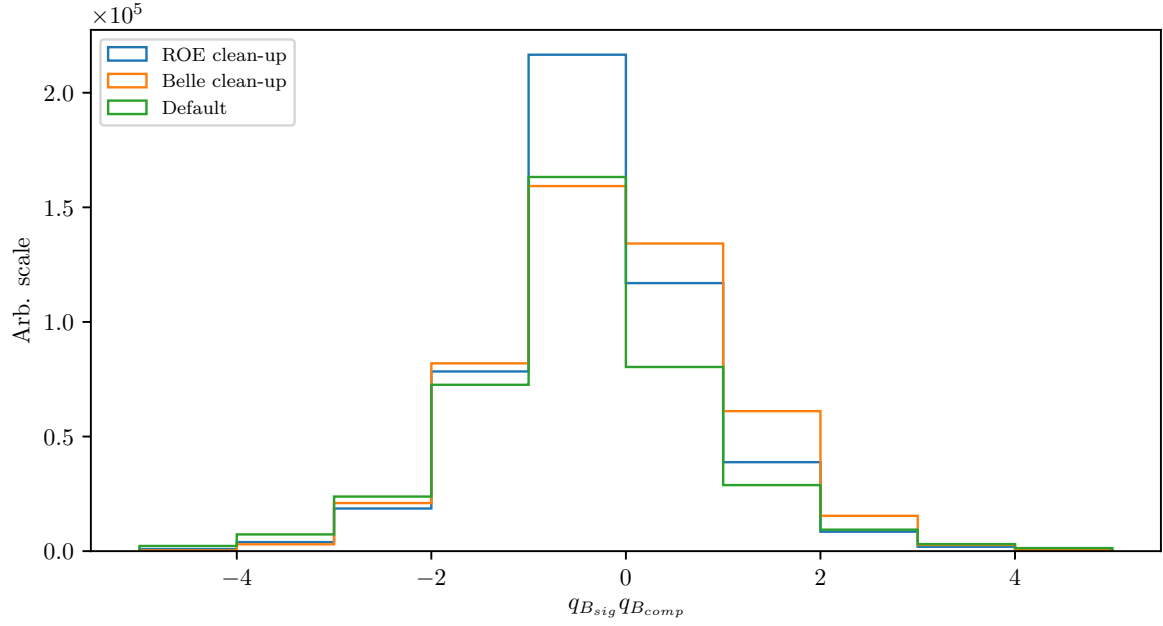


Figure 5.14: Distribution of B_{sig} and B_{comp} charge product for various types of clean-up procedures. For ROE clean-up, the procedure seems to increase the number of candidates with the correct value of the product.

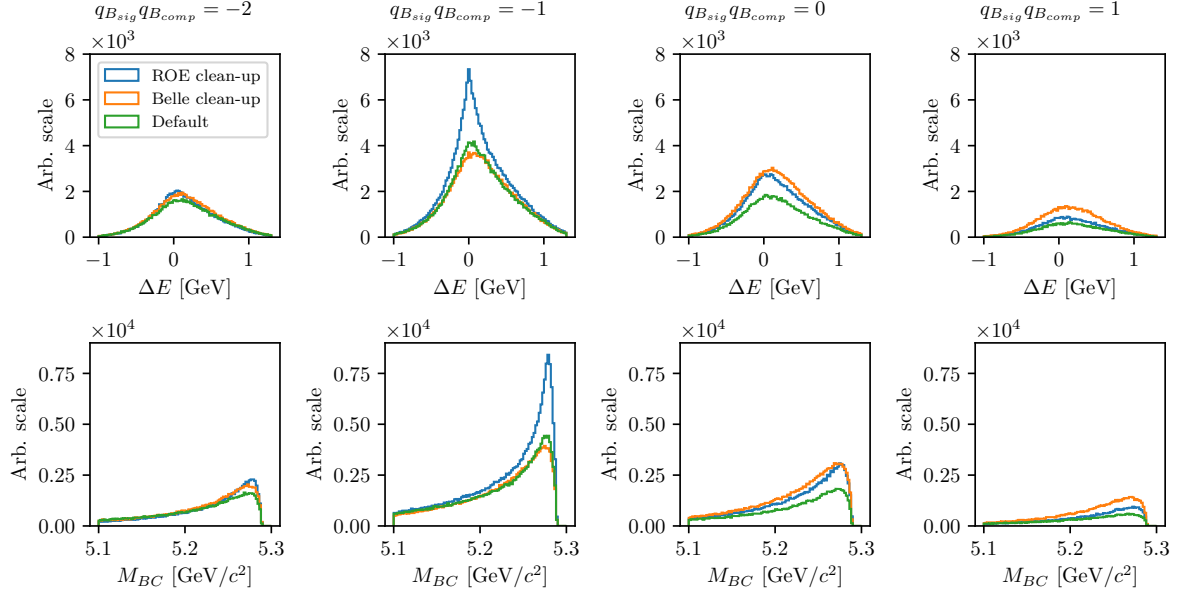


Figure 5.15: Distributions of ΔE (top) and M_{BC} (bottom) for various types of clean-up procedures, split by specific values of the charge product. While there seems to be an increase in efficiency for both clean-up procedures in general, ROE-cleanup makes the most improvement to resolution and efficiency for the correct value of the charge product.

5.6 ROE clean-up validation

The ROE clean-up seems to perform well, but we have to keep in mind that we are testing the procedure on simulated, not real data. Therefore we have to make sure that this procedure performs on simulated as well as real data in the same manner.

Will add more later.

6 Background suppression

This chapter shows the procedure in suppressing various kinds of background by applying cuts on MVA classifier outputs.

6.1 Control decay and other resonant background

In this analysis we study decays with kaons in the final state. This means that standard procedures in $b \rightarrow u$ analyses in order to suppress $b \rightarrow c$ backgrounds, such as K -veto, are not possible. As a consequence, our final sample consists of combinations of K pairs coming from $b \rightarrow c$ sources, such as $D^0 \rightarrow K^+ K^-$. Such candidates usually have resonance-like properties in the two-kaon invariant mass spectrum. Figure 6.1 shows this invariant mass spectrum of two kaons, m_{KK} , where obvious resonant structures are present from sources like

- $\varphi \rightarrow K^+ K^-$ (sharp resonance at $\sim 1.019 \text{ GeV}/c^2$),
- $D^0 \rightarrow K^+ K^-$ (sharp resonance at $\sim 1.864 \text{ GeV}/c^2$),
- $D^0 \rightarrow K^+ \pi^-$ (wide, shifted resonance, due to kaon miss-identification).

An opportunity for a control decay choice presents itself as

$$B^+ \rightarrow D^0 \ell^+ \nu, \quad D^0 \rightarrow K^+ K^-,$$

which has very similar properties as the studied signal decay, is much more abundant, and, most importantly, is easy to suppress, since it only populates a very narrow region in the m_{KK} invariant mass spectrum. In order to suppress these resonant backgrounds while studying the signal or control decay, we impose a set of the following cuts

- Signal decay: $|m_{KK} - m_\varphi| > \Delta_\varphi$, $|m_{KK} - m_{D^0}| > \Delta_{D^0}$, $|m_{K\pi} - m_{D^0}| > \Delta_{D^0}$,
- Control decay: $|m_{KK} - m_\varphi| > \Delta_\varphi$, $|m_{KK} - m_{D^0}| \leq \Delta_{D^0}$, $|m_{K\pi} - m_{D^0}| > \Delta_{D^0}$,

where $m_\varphi \approx 1.019 \text{ GeV}/c^2$ and $m_{D^0} \approx 1.864 \text{ GeV}/c^2$ are nominal masses of the φ and D^0 mesons, and $\Delta_\varphi \approx 8 \times 10^{-3} \text{ GeV}/c^2$ and $\Delta_{D^0} \approx 1.2 \times 10^{-2} \text{ GeV}/c^2$ are symmetric cut widths around the nominal mass values for the φ and D^0 mesons, respectively.

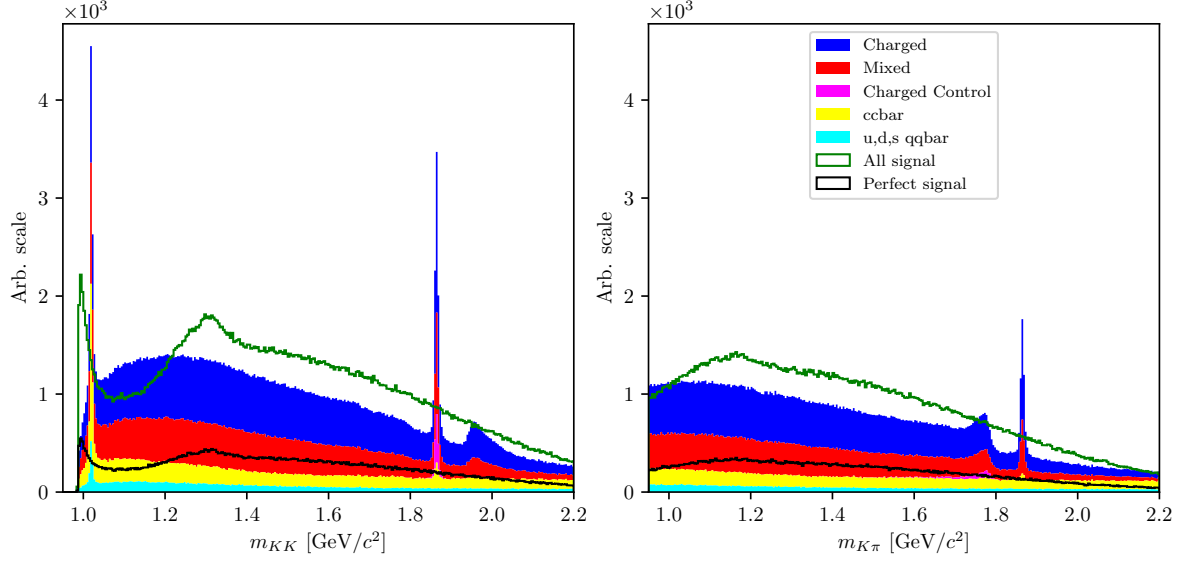


Figure 6.1: Invariant mass of two correctly reconstructed kaons (left) and invariant mass of two kaons, where one was misidentified as a pion (right).

6.2 Continuum suppression

Continuum background are physics processes where continuum states are produced in electron and positron collisions

$$e^+e^- \rightarrow q\bar{q},$$

where $q = u, d, s$ or c , and are a sizable contribution to $B\bar{B}$ events. Additionally to kinematic constraints to separate $e^+e^- \rightarrow \Upsilon(4S) \rightarrow B\bar{B}$ decays from $e^+e^- \rightarrow q\bar{q}$, properties of the "event shape" are also often used, because phase-space distributions of decayed particles differ for these two processes. Continuum background events are generated in a back-to-back way in the CMS frame, so hadrons produced in the quark fragmentation possess only a small transverse momentum compared to the initial momentum magnitude. This leads to a spatially confined, jet-like structure. On the other hand B mesons from $B\bar{B}$ events are produced almost at rest in the CMS frame. Their decay products from an isotropic distribution in the detector, which yields a spherical event shape.

6.2.1 Characteristic variables

Information on the phase-space distribution of decay particles is obtained in a number of different ways. In this subsection different characteristic variables are presented which

are used in the MVA training. They all focus on kinematic and shape differences between the two processes, which we want to discriminate.

Thrust and related variables

It is possible to define a thrust axis \mathbf{T} for a collection of N momenta p_i as a unit vector along which their total projection is maximal. Thrust axis \mathbf{T} can be obtained by maximizing the expression

$$\mathbf{T} = \frac{\sum_i |\mathbf{T} \cdot \mathbf{p}_i|}{\sum_i |\mathbf{p}_i|}. \quad (6.1)$$

In this case, a related variable is $|\cos \theta_T|$, where θ_T is the angle between the thrust axis of the momenta from B meson decay particles, and the thrust axis of all particles in the ROE. Since both B mesons in $B\bar{B}$ events are produced at rest, their decay particles, and consequentially their thrust axes, are uniformly distributed in the range $[0, 1]$. On the other hand, decay particles from continuum events follow the direction of the jets in the event. As a consequence, the thrusts of both the B meson and the ROE are strongly directional and collimated, which results in a large peak at $|\cos \theta_T| \approx 1$. Additionally, one can also use the variable $|\cos \theta_{T,B}|$, which is the thrust axis between the B candidate and the beam axis. For B candidates from $B\bar{B}$, this distributions is again uniformly distributed, while for candidates from continuum events this distribution follows the distribution of the jets with the function $1 + \cos^2 \theta_{T,B}$. Figure 6.2 shows the distributions of $|\cos \theta_T|$ (left) and $|\cos \theta_{T,B}|$ (right) for different B meson candidates.

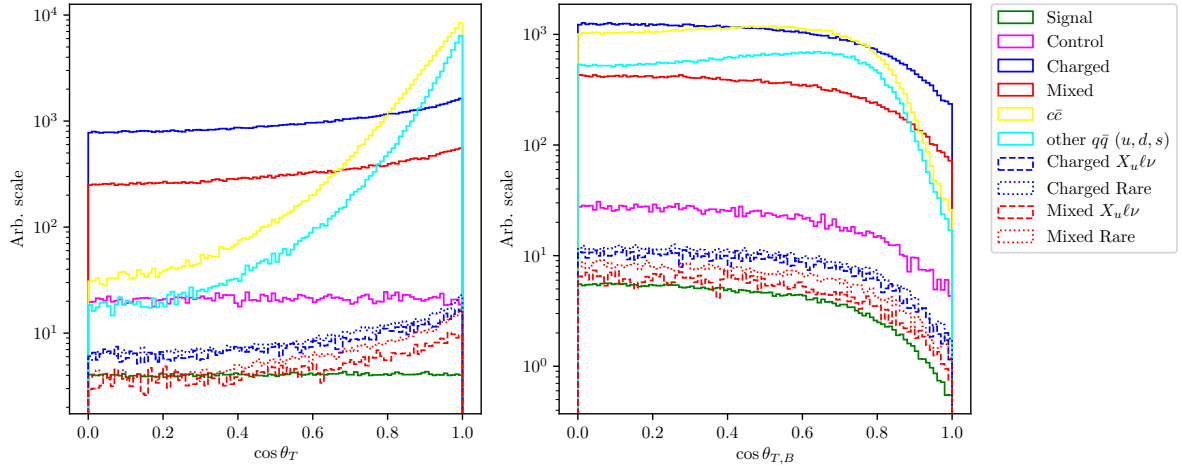


Figure 6.2: Distributions of $|\cos \theta_T|$ (left) and $|\cos \theta_{T,B}|$ (right) for different B meson candidates.

CLEO cones

CLEO cones have been introduced by the CLEO collaboration [1] and are an additional specific tool to provide optimal background discrimination. They are nine variables corresponding to the momentum flow around the thrust axis of the B meson candidate, binned in nine cones of 10° around the thrust axis, as illustrated in Figure 6.3.

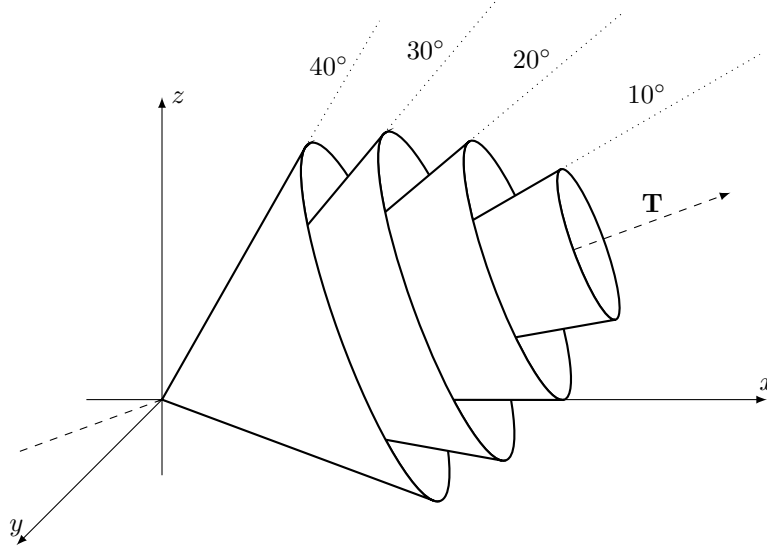


Figure 6.3: Concept of CLEO cones. \mathbf{T} denotes the thrust axis of the B meson candidates in the event. Each variable corresponds to a momentum flow around the thrust axis in steps of 10° .

KSFW moments

Fox-Wolfram moments are another useful parametrization of phase-space distribution of energy flow and momentum in an event [2]. For a collection of N momenta p_i , the k -th order normalized Fox-Wolfram moment R_k is defined as

$$R_k = \frac{H_k}{H_0} = \frac{1}{H_0} \sum_{i,j} |p_i| |p_j| P_k(\cos \theta_{ij}), \quad (6.2)$$

where θ_{ij} is the angle between p_i and p_j , and P_k is the k -th order Legendre polynomial. For events with two strongly collimated jets, R_k takes values close to 0 (1) for odd (even) values of k , so these moments provide a convenient discrimination between $B\bar{B}$ and continuum events.

Belle developed a refined generation of Fox-Wolfram moments, called Kakuno-Super-Fox-Wolfram (KSFW) moments to further suppress the continuum background. There

are 17 different KSFW moments which are grouped into R_k^{so} , R_k^{oo} and R_k^{ss} []. The latter ones are excluded due to correlations with B meson specific variables.

6.2.2 MVA training

Most of the characteristic variables, described in subsection 6.2.1, were taken together in order to train a single MVA classifier for continuum suppression. All characteristic variables were checked for possible q^2 correlation. Variables with significant correlation or complex shapes in the 2D plot were discarded from the training set, since they would have introduced unwanted dependence on the unreliable model, ISGW2, used for signal MC generation. Additionally, all of the characteristic variables in our set do not depend on the signal mode, they only differ in the kinematic and topological aspects of $B\bar{B}$ and continuum background events.

The training dataset consisted of 2×10^5 candidates, where 50 % of the candidates are correctly reconstructed signal events, 25 % are $u\bar{u}$, $d\bar{d}$ and $s\bar{s}$ background with expected proportions, and 25 % is $c\bar{c}$ background. Since the full Belle dataset is experiment dependent, we construct the training dataset by proportionally sampling each MC dataset, corresponding to the appropriate experiment number.

The training variable set consisted of

- B meson direction and thrust related variables
 - magnitude of thrust axes of B and ROE ,
 - cosine of angle between thrust axis of B and thrust axis of ROE ,
 - cosine of angle between thrust axis of B and beam direction,
 - reduced Fox-Wolfram moment R_2 ,
- all 9 CLEO Cones
- KSFW Moments
 - R_{01}^{so} , R_{02}^{so} , R_{03}^{so} , R_{04}^{so} ,
 - R_{10}^{so} , R_{12}^{so} , R_{14}^{so} ,
 - R_{20}^{so} , R_{22}^{so} , R_{24}^{so} ,
 - R_0^{oo} , R_1^{oo} , R_2^{oo} , R_3^{oo} , R_4^{oo} ,
- FlavorTagging variables

- qp of e, μ, ℓ ,
- qp of intermediate e, μ, ℓ ,
- qp of $K, K/\pi$, slow pion, fast hadron,
- qp of maximum P^*, Λ , fast-slow-correlated (FSC),
- Other
 - $\Delta z, \Delta t$.

Figure 6.4 shows the classifier output for various types of background, all in expected MC proportions. B meson candidates from continuum background are dominant at lower values, while candidates from $B\bar{B}$ events populate the region with higher values.

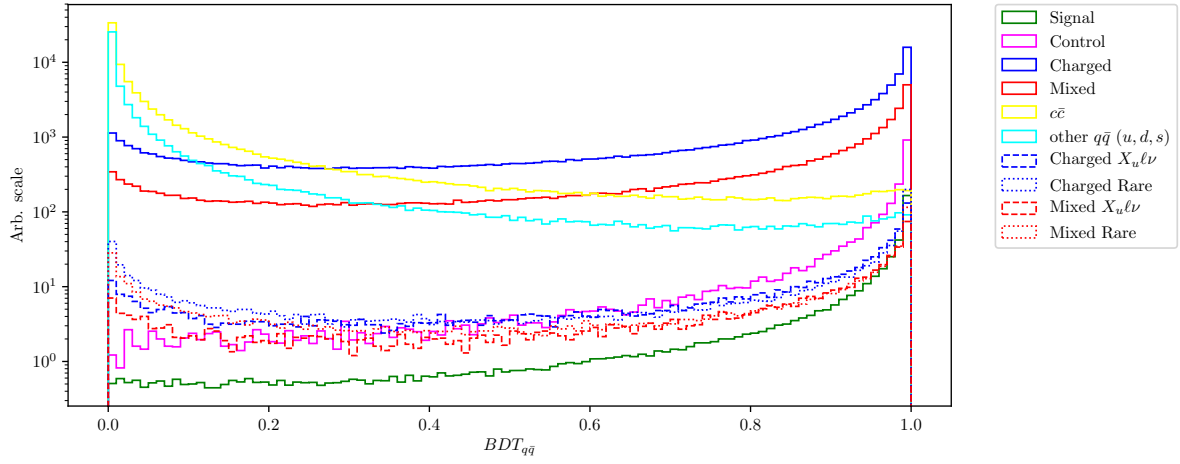


Figure 6.4: Continuum suppression classifier output for signal and various types of background. B candidates from continuum events dominate the lower region, while candidates from $B\bar{B}$ dominate in the upper region of the classifier output.

6.3 $B\bar{B}$ suppression

After separating continuum background from $B\bar{B}$ events, the next step is to train an MVA classifier to recognize our signal candidates amongst the candidates of other $B\bar{B}$ background. $B\bar{B}$ background consists of

- $b \rightarrow c\ell\nu$ background,
- $b \rightarrow u\ell\nu$ background,

- Other rare decays (radiative, penguin, rare 2- and 3-body decays, ...).

Similarly as before, the training dataset for this classifier consisted of 2×10^5 candidates, where 50 % of the candidates are correctly reconstructed signal events. The remaining part of the training dataset consists of all background, not including the control sample, because we are not interested in suppressing it directly. The background part of the dataset consists of 75 % charged and neutral $B\bar{B}$ events in equal proportions, whereas the remaining 25 % is equally populated with charged and neutral $B\bar{B}$ events from $b \rightarrow u\ell\nu$ and other rare decays. The training dataset was proportionally sampled in the same manner as described in subsection 6.2.2.

In order to separate this kind of background, we must be careful not to introduce correlations with the fit variables (ΔE , M_{BC}) or any kind of model dependence (correlation with q^2). This means that we can not use any information of the decay particles or the candidate, which is of kinematics nature, such as decay particles momenta, decay angles or other variables with such behavior.

The training variable set consisted of

- fit probability of $P(\chi^2, DOF)$ of the signal candidate and the ROE side, separately,
- $\cos \theta_{BY}$ from Eq. (4.4),
- \cos of the angle between momentum and vertex of X , where $X \in [KK, KK\ell, KK\ell\nu]$,
- FlavorTagger variables for the two signal-side kaons,
- number of kaons, tracks and distant tracks in ROE,
- θ angle of the ROE momentum in CMS frame,
- ξ_Z from []
- Δz ,
- m_{miss}^2 from Eq. (4.11),
- $B \rightarrow D^*\ell\nu$ veto variables,

where distant tracks are all tracks in ROE which satisfy the condition of $|d_0| > 10.0$ cm or $|z_0| > 20.0$ cm. The last entry, $B \rightarrow D^*\ell\nu$ veto variables, are a set of variables where we partially reconstruct the D^* candidate 4-momentum via a linear combination of the π_s^\pm 4-momentum in the $D^* \rightarrow D\pi_s^\pm$ decay. It helps discard the most dominant $B \rightarrow D^*\ell\nu$ background. It is most efficient in the $B^0 \rightarrow D^{*-}\ell^+\nu$ decay, where D^{*-} further decays via $D^{*-} \rightarrow \bar{D}^0\pi_s^-$. Other decays do not contain a charged π_s particle and are harder to reconstruct with good precision. This results in larger suppression

of the neutral $B\bar{B}$ background only. Figure 6.5 shows the veto variable with a partial reconstruction of a charged π_s^\pm .

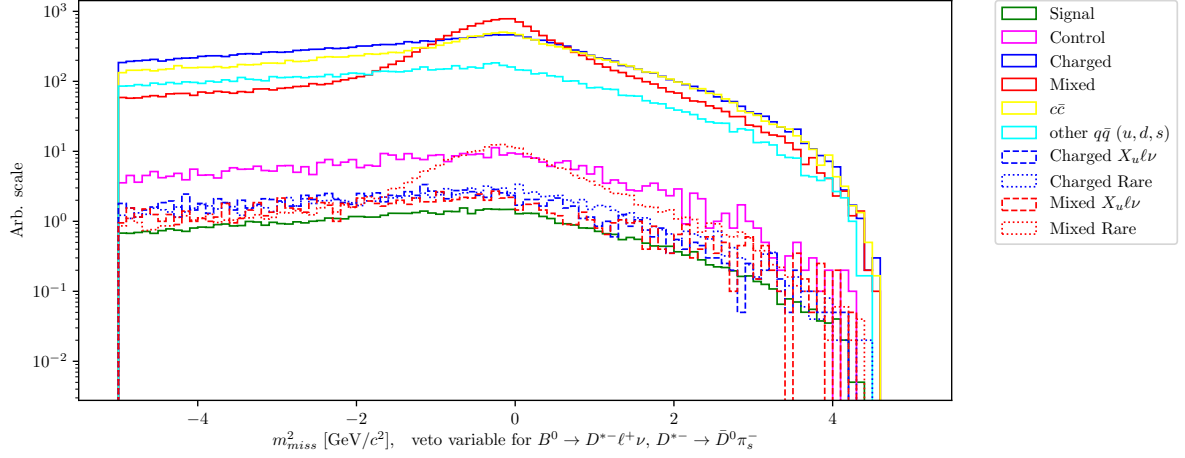


Figure 6.5: Distribution of m_{miss}^2 for partially reconstructed $B^0 \rightarrow D^{*-}\ell^+\nu$ decays, which serves as a veto.

When the training is finished and hyper-parameters of the classifier are optimized, the classifier output, shown in Figure 6.6 (left), can be used for background suppression. B meson candidates from $B\bar{B}$ background are dominant at lower values, while candidates from $B\bar{B}$ events populate the region with higher values. Since the differences between signal and background $B\bar{B}$ events are smaller than $B\bar{B}$ and $q\bar{q}$ events, the resulting classifier has a smaller separation power than in previous section.

6.3.1 Boosting to uniformity

The selection approach with standard classifiers is optimal for counting experiments, as it by construction produces the optimal selection for observing an excess of signal over background events. Today's BDT algorithms, which work in this way, produce non-uniform selection efficiencies and may, as a consequence, shape background distributions to look like signal. In order to minimize such behavior, it is possible to discard variables, which are correlated with the variable of interest (in our case ΔE and M_{BC}), from the training set. This, however, decreases the classifiers discriminating power. Another approach is to use a novel boosting method, uBoost, which is trained to optimize an integrated FOM under the constraint that the BDT selection efficiency for the desired class must be uniform. The uBoost algorithm balances the biases to produce the optimal uniform selection [1].

The training set used in this training is the same as described in the beginning of this chapter, along with the same set of training variables. It will be seen later that the

standard BDT classifier shapes the background to look like signal mostly in the M_{BC} picture, therefore we train the uBoost classifier with a uniformity constraint in the M_{BC} variable for background candidates. The resulting classifier output is shown in Figure 6.6 (right). For this classifier, the separation power between signal and background seems worse, however, the shapes of backgrounds can differ and the final conclusion will be shown only after the signal extraction.

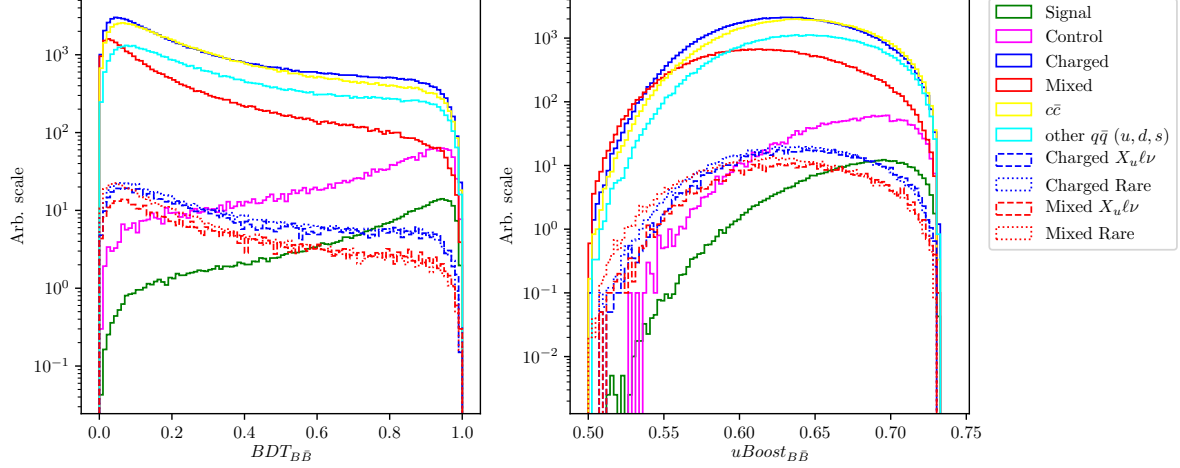


Figure 6.6: $B\bar{B}$ suppression classifier output for signal and various types of background for the standard BDT classifier (left) and the uBoost classifier (right). B candidates from $B\bar{B}$ background events dominate the lower region, while signal and control candidates dominate in the upper region of the classifier output.

6.4 Selection optimization

Instead of two separate $q\bar{q}$ and $B\bar{B}$ FOM optimizations, it is more efficient to do a simultaneous 2D FOM optimization, since the two classifiers are not completely uncorrelated. In the same manner as before, FOM is optimized for perfectly reconstructed signal candidates in the signal window, after all the pre-cuts, signal categorization, and after cutting out the background resonances and the control decay. The FOM plot with the optimal point for both $B\bar{B}$ MVA classifiers is shown in Figure 6.7.

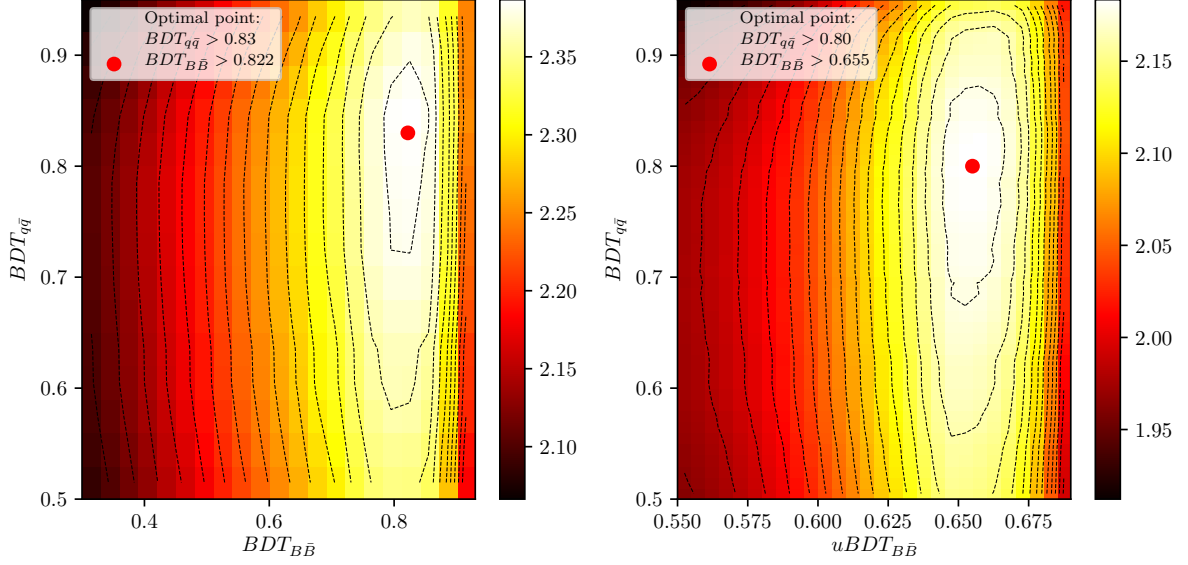


Figure 6.7: 2D *FOM* optimization of continuum suppression classifier and the standard BDT (left) and uBoost (right) $B\bar{B}$ suppression classifier.

We can compare signal and major background distributions of ΔE and M_{BC} after the 2D *FOM* optimization for both classifiers. Figure 6.8 shows the arbitrary (left) and normalized scale (right) for ΔE (top) and M_{BC} (bottom) for the final sample optimized with the standard BDT classifier, while Figure 6.9 shows similarly for the final sample optimized with uBoost classifier. We can see that there is considerably more background in the latter case, however, also shapes of background and signal distributions differ greatly. The biggest change seems to be in the shape of the M_{BC} distribution, where the background component is much more signal like in the final sample optimized with the standard BDT classifier than in the other case. The total numbers of expected signal candidates and the signal-to-noise ratios for both classifiers are:

- Standard BDT: $N_{sig} = 152.82$, $N_{sig}/N_{bkg} = 3.70$ %,
- uBoost: $N_{sig} = 241.56$, $N_{sig}/N_{bkg} = 1.09$ %.

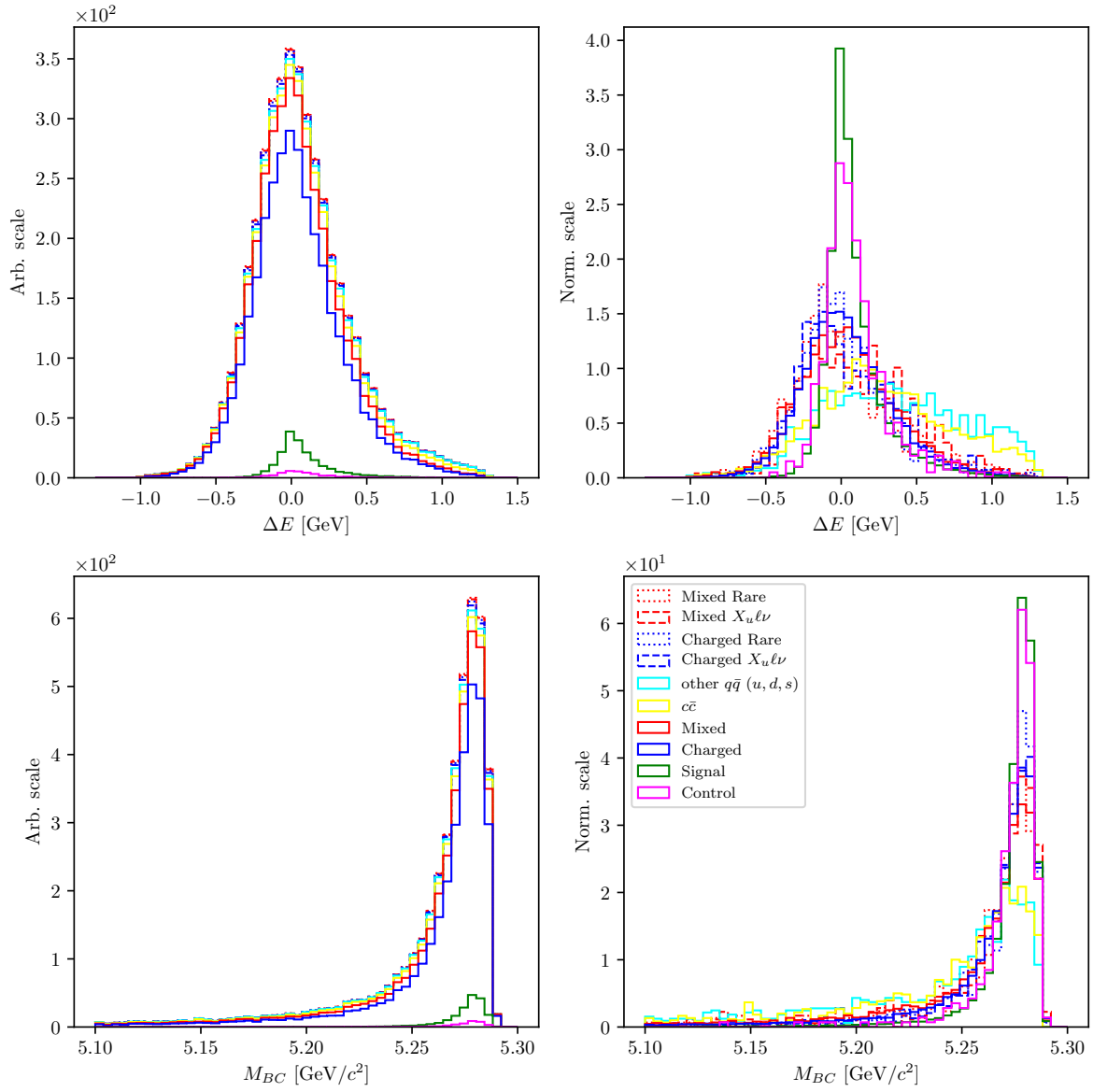


Figure 6.8: Arbitrary (left) and normalized scale (right) for ΔE (top) and M_{BC} (bottom) for the final sample optimized with the standard BDT classifier.

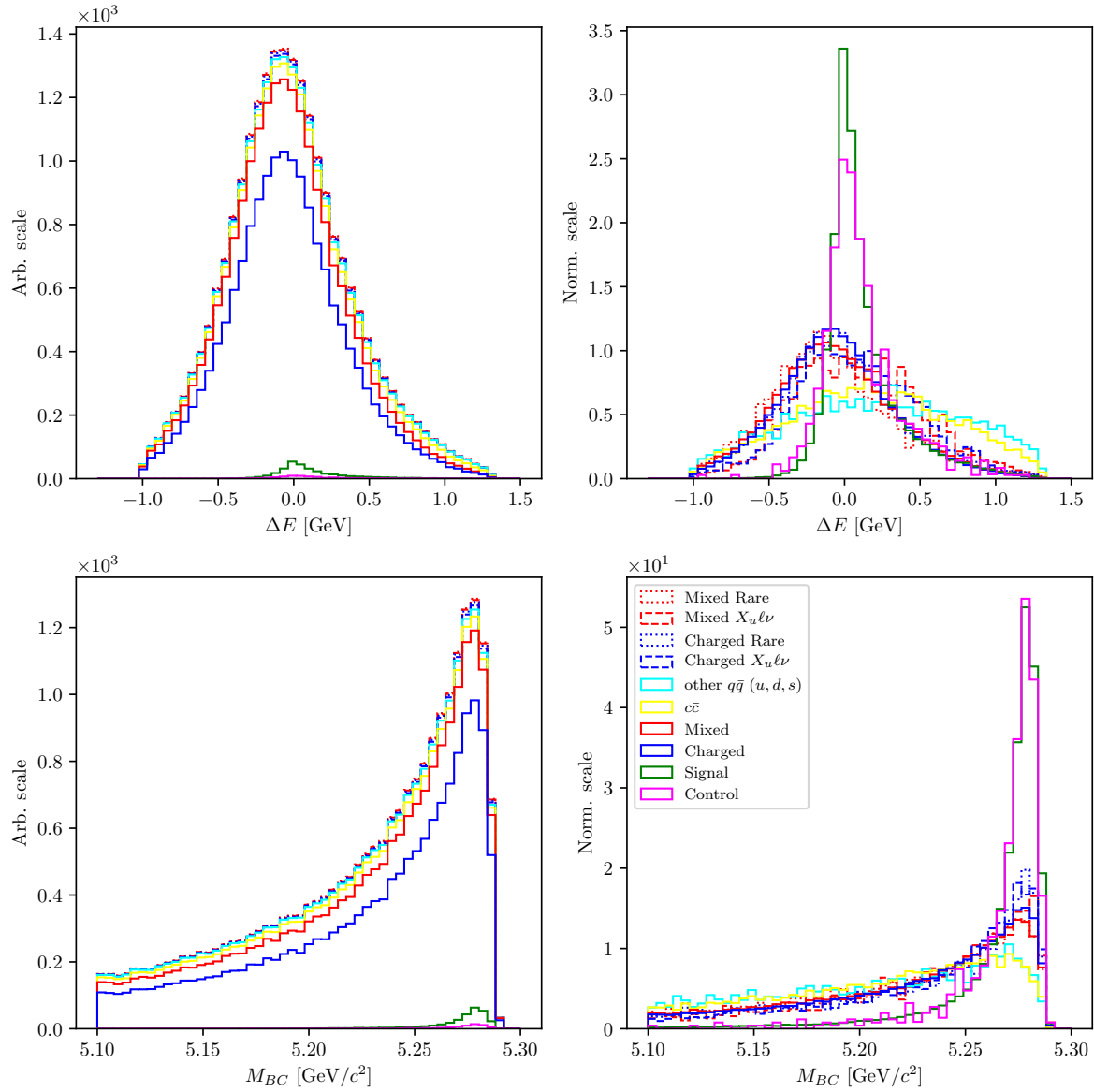


Figure 6.9: Arbitrary (left) and normalized scale (right) for ΔE (top) and M_{BC} (bottom) for the final sample optimized with the uBoost classifier.

6.5 Data and MC agreement

With the final selection in place, we can check the data and MC agreement by checking the control decay region in on- and off- resonance data. Off-resonance samples provide the ability to check the agreement of the $q\bar{q}$ background component, while on-resonance samples can be used to check the validity of the control MC sample and, consequentially,

the signal MC sample.

The off-resonance data were collected at 60 MeV below the $\Upsilon(4S)$ resonance peak energy in order to determine the non- $B\bar{B}$ background. It therefore offers a direct view of the $q\bar{q}$ background data sample, which we can compare to the off-resonance MC sample. Figure 6.10 shows ΔE , M_{BC} and the $q\bar{q}$ classifier output, $BDT_{q\bar{q}}$, for off-resonance data and MC in the control region, before any MVA cuts, where the MC sample was scaled down by a factor of 6, due to having more streams of MC. Figures show a slight and a consistent overestimation of $q\bar{q}$ MC. However, looking at the shapes of the distributions, the only difference seems to be the normalization factor, so the distribution shapes can be left as they are, since the normalization will be automatically set in the template fit. There also seems to be a difference in the classifier performance for the continuum background suppression on data and MC. This leads to further differences between data and MC in the $q\bar{q}$ sample after the classifier cut, but we estimate that these differences are negligible, since a relatively small amount of continuum background passes the selection, compared to other background types.

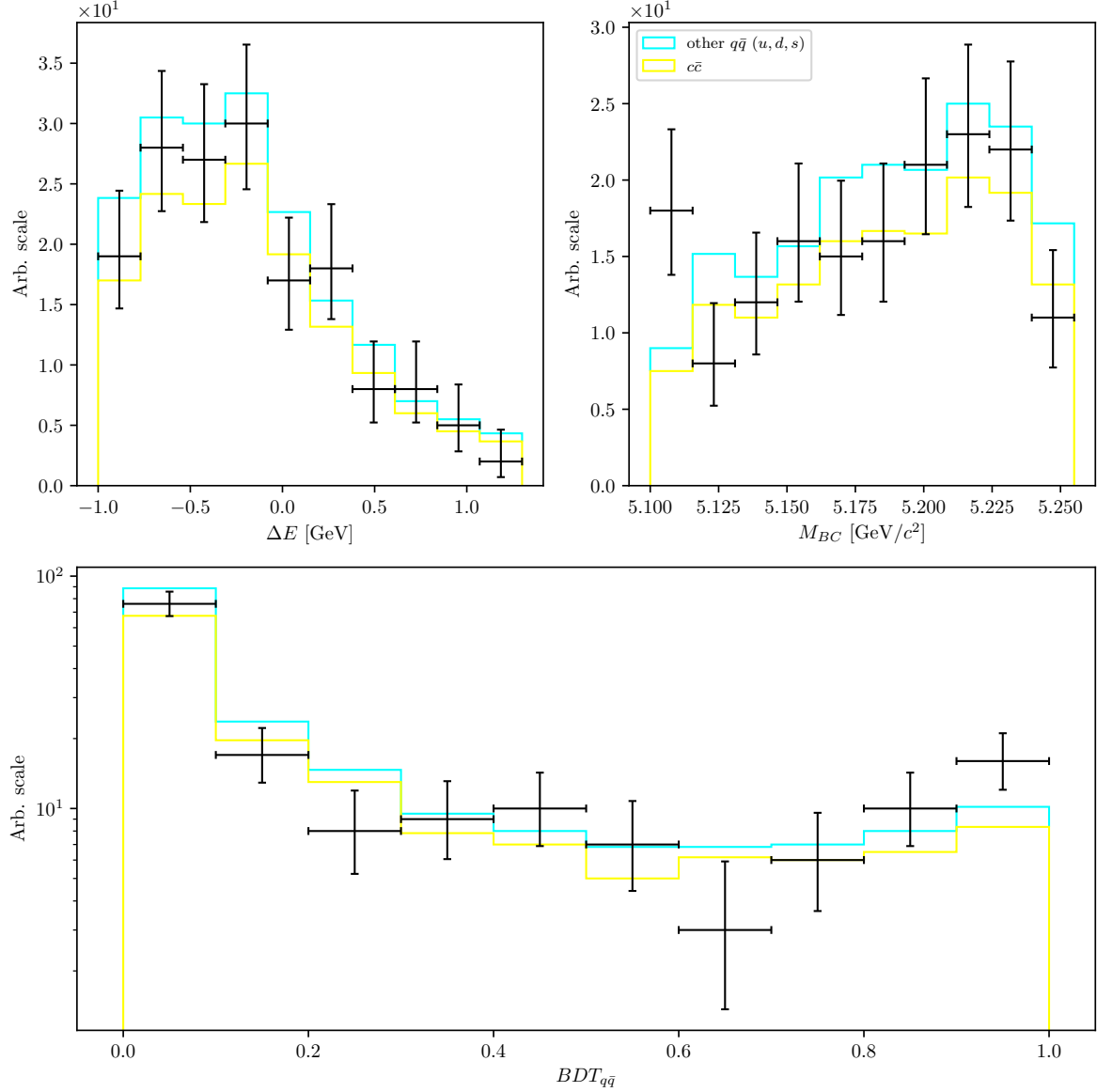


Figure 6.10: ΔE (left), M_{BC} (right) and the $q\bar{q}$ classifier output (bottom), for off-resonance data and MC in the control region before any MVA cuts.

We can repeat the check on on-resonance data. Figure 6.11 shows ΔE , M_{BC} and $BDT_{q\bar{q}}$, where one can see inconsistencies between data and MC on the lower spectrum, where continuum background is dominant. On the other hand, data and MC seem to agree well in the upper part of the spectrum, where $B\bar{B}$ events are dominant, which confirms our statement about data and MC differences in continuum background being negligible from before. Overall, data and MC seem to agree very well already off-the-shelf after all the pre-cuts and without any corrections. This means that the modeling of this MC

sample is very precise in this particular region of data and that there are no significant differences between data and MC for the control sample and the signal sample.

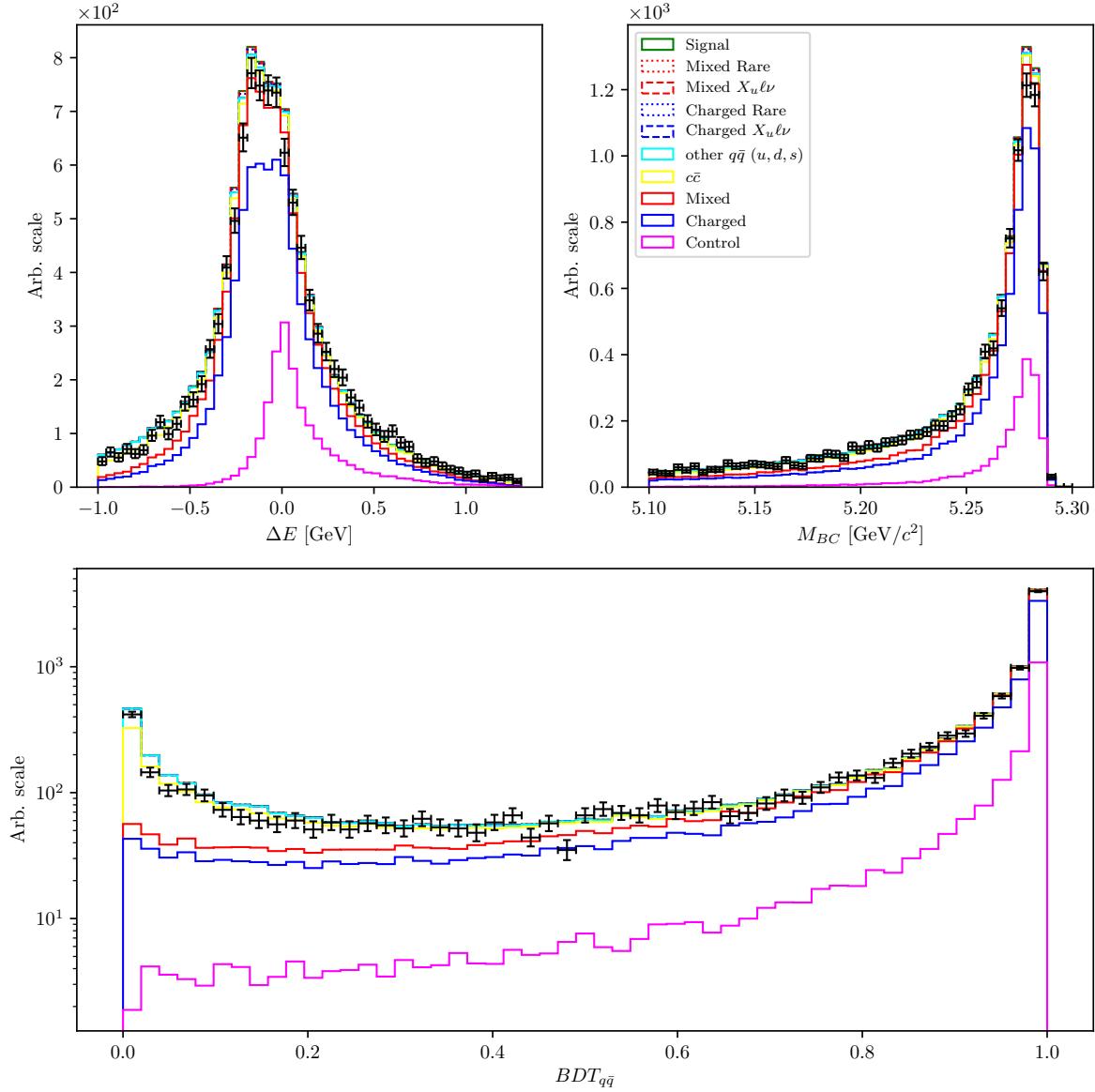


Figure 6.11: ΔE (left), M_{BC} (right) and the $q\bar{q}$ classifier output (bottom), for on-resonance data and MC in the control region before any MVA cuts.

7 Signal extraction

In this chapter we present the procedure for signal yield extraction, using the framework of RooFit. We define 2D histogram templates in ΔE and M_{BC} , based on MC, for signal and several types of background. Using these templates, the independent full sample is fitted with a binned maximum likelihood (ML) fit, so that the individual template ratios and their sum describe the fitted sample as best as possible. For each fit, asymmetric errors are calculated using MINOS [1].

7.1 Fit templates

For fit template construction, the following MC samples are used:

- Signal MC (produced in abundance),
- 10 streams of **charged** and **mixed** $B\bar{B}$ background,
- 6 streams of $c\bar{c}$ (**charm**) and other $q\bar{q}$ (**uds**) background,
- **ulnu** sample, corresponding to $20\times$ integrated luminosity of the full Belle dataset,
- **rare** sample, corresponding to $50\times$ integrated luminosity of the full Belle dataset.

For MC, we perform 10 fits, where the fitted sample was not used in the creation of the templates. In case of the $B\bar{B}$ background, each stream was fitted once, whereas in the case of the $q\bar{q}$ background, the streams were cycled back to the beginning after using the initial 6 streams. **rare** and **ulnu** samples were sampled randomly in appropriate sizes. In case of data, all available MC was used for creating the templates.

The full signal MC sample was used for the signal template definition in case of MC and data fits.

The following histogram templates were used in the fits

- signal template,
- control template,

- $B\bar{B}$ template,
- $q\bar{q}$ template,

where the $B\bar{B}$ template consists of appropriate contributions of `charged`, `mixed`, `rare` and `ulnu` samples. All templates shapes are fixed and all template yields are left as floating parameters in the fit, except in the case of the control template. The yield of the control template was fixed to the expected MC value.

All four 2D templates for both final selections are shown in Figure X. One can see that the defined fit region includes regions of low statistics, which are populated with bins of zero content. This might produce problems, and the solution for them is described in the next section.

PLOT

7.2 Adaptive binning algorithm

Bins with zero content are a direct consequence of having a finite MC sample and represent a liability in ML fits. Due to low statistic in the edge regions, the locations of these empty bins can vary for the templates and the fitted sample. A problem occurs if all templates have an empty bin where the fitted sample does not. In the scope of ML fits, this effectively means that there are entries where there is zero probability that there would be any. We will call such bins *problematic*, because in these cases the fit does not converge.

The ideal solution for this problem is to increase the MC statistics. Since that is not an option, we pursue other solutions, such as decreasing the number of bins. While this has a good potential to work, the drawback of it is a decrease in the template resolution in densely populated regions, where good resolution is most needed. The optimal solution here seems to be a choice of variable bins, with fine binning in the densely populated regions and larger bins in the regions with low statistics.

We have devised an algorithm, which compares the templates and the fitted sample, and defines a variable binning so that there are no more problematic bins in the end. Figure X shows two examples of how the procedure works. The algorithm takes an argument for the initial number of uniform bins in each dimension, and does the following

1. define uniform binning in both dimensions with the provided argument,
2. create a 2D histogram from MC templates with expected yields,
3. define an *optimal* region, where most of the 2D integral is contained and where all

bins have a non-zero content (this region does not change throughout the process),

4. compare the histograms for the expected and the fitted sample, find the problematic bins,
5. loop until all problematic bins disappear
 - a) find problematic bin, which is nearest to the maximum bin,
 - b) change the binning from N to $N - 1$ from that bin and in the direction away from the maximum bin.

PLOT

An additional problem seems to be the plotting of the fitted templates and the sample with variable binning. It would seem that RooFit does not take the bin widths into account when plotting, while everything works as expected for the fit itself. This was bypassed by extracting the fitted yields and applying them to templates and samples with uniform binning, which were then used for drawing.

7.3 Signal MC fit results

The fit setup was used on the final samples obtained with both versions of the final selection (standard BDT and uBoost). The choice of the initial uniform binning is not obvious, so we perform 10 fits for each initial binning choice and plot to compare fitted and expected yield difference, pulls and significance for both final samples, shown in Figure X. Fits to final sample, obtained with the uBoost classifier, generally seem to have a lower bias, pull distributions closer to the normal distribution and better significance than the fits in the case of the standard BDT final sample. This fixes our choice of the final selection. It is also possible to determine the optimal initial binning choice to be somewhere in the region of X bins in each dimension, since a higher number of initial bins yields biased results, while a lower one yields results with poorer significance.

PLOT

Figure X shows the fit result and the fitted sample of an arbitrary stream for ΔE and M_{BC} , for the fit region and for the signal region. The fits seem stable and under control. Fit results for all streams of MC are shown in Figure X, along with the global fit over all streams. The results seem to describe the expected value in a precise manner, with the bias much smaller than the average statistical error. The normalized χ^2 value of the global fit is X, while the average significance of the fits is X.

PLOT

PLOT

7.3.1 Toy MC experiment

Toy MC experiments allow us to study the yields, errors and the pulls of the signal fit by generating our own pseudo datasets, according to the MC, and not producing it in the standard way. In order to test the fit behavior, a toy MC study is performed where all available MC was used for template creation. We constructed X pseudo datasets, where each dataset was generated according to the expected size of each template category, distributed according to the Poisson distribution. All fits were performed with the optimal initial uniform binning of $X \times X$ bins in ΔE and M_{BC} .

Figure X shows distributions of the fit yields, errors and the pull distribution of all pseudo fits. The fits seem to be under control, as the pulls follow a normal distribution with a mean of X and standard deviation of X .

7.3.2 Toy MC linearity test

Linearity test is used for determining sensitivity of the fit to the amount of signal in the fitted sample. Since this is a first measurement of this decay channel, MC could be very different from data, so we need to perform this test in order to determine our sensitivity to smaller, as well as larger amounts of signal candidates, relative to the expected amount.

The pseudo datasets are generated in the same way as in the previous subsection. X steps from $[0.1, 10]$ in the logarithmic scale are taken for factors of signal amount and for each factor X pseudo datasets are generated according to Poisson statistics.

Figure X shows the average fit yield, standard error and the (TODO check) pulls for X fits at each signal fraction value. The expected MC result is shown at fraction value 1. The plots show no bias with regard to the signal fraction, while the pulls seems to be described by the normal distributions throughout the fraction range. At expected value we are just below 3σ significance, as already pointed out at the beginning of this section.

7.4 Control fit results

The same fit procedure as in Section 7.3 can be applied for the control decay. This enables us to compare the fit behavior on MC as well as data, and see if the former describes the latter in a precise enough manner. For control decay, data and MC are

compared at various selection stages to compare yield ratios and cut efficiencies on data and MC in the form of double ratios. The fits were made for the following cut scenarios

- a) sample after pre-cuts,
- b) signal category (I) sample,
- c) b) + $BDT_{q\bar{q}}$ cut,
- d) b) + $BDT_{B\bar{B}}$ cut,
- e) b) + $BDT_{q\bar{q}} + BDT_{B\bar{B}}$ cut.

First, the binning scan was performed in order to determine optimal binning for each cut scenario above. The results for control fit yield, pulls and significance are shown in Figure X.

7.4.1 Fit yields and yield ratios

7.4.2 Double ratios

7.4.3 Branching ratio measurement and comparison

7.5 Signal fit to data

8 Systematics

8.1 Model uncertainty effects

8.2 PID efficiency correction

8.3 Bias

Addendum A: ROE clean-up π^0 training

Correlations

Hyper-parameter optimization

Importance

Validation

Addendum B: ROE clean-up γ training

Correlations

Hyper-parameter optimization

Importance

Validation

Addendum C: ROE clean-up duplicate pair training

Correlations

Hyper-parameter optimization

Importance

Validation

Addendum D: ROE clean-up duplicate training

Correlations

Hyper-parameter optimization

Importance

Validation

Addendum E: $q\bar{q}$ suppression training

Correlations

Hyper-parameter optimization

Importance

Validation

Addendum F: $B\bar{B}$ suppression training

Correlations

Hyper-parameter optimization

Importance

Validation