Angular Analysis Of The $B \to K^*\ell\ell$ Decay In Long Shutdown 1 Data At Belle II

Dissertation submitted
for the award of the title
"Doctor of Natural Sciences"
to the
Faculty of Physics, Mathematics
and Computer Science (FB 08)
of the
Johannes Gutenberg-Universität Mainz

Martin Simon Sobotzik
born in Worms

Mainz, January 2025

Angular Analysis Of The $B \to K^* \ell\ell$ Decay In Long Shutdown 1 Data At Belle II

1. Reviewer:
2. Reviewer:

Date of Defense: February 7th, 2025

Martin Simon Sobotzik
Belle II Collaboration
Institut für Kernphysik
Johannes Gutenberg-Universität
Johann-Joachim-Becher-Weg 45
D-55128 Mainz
sobotzik@uni-mainz.de

JOHANNES GUTENBERG UNIVERSITÄT MAINZ

"Enough or not . . . it will have to do."

Leo Tolstoy, *Anna Karenina* (1878)

Abstract

In recent years, the search for physics beyond the Standard Model has become more and more sophisticated. Rare decays of the B meson are an ideal probe for phenomena beyond the Standard Model. These decays are driven by the $b \to s\ell\ell$ quark transition, a flavor changing neutral current, which is forbidden on tree level in the Standard Model and can only be realized using higher-order loop diagrams, such as electroweak penguin or W^+W^- box diagrams resulting in highly suppressed branching fractions ($\sim 1 : 10^6$). Extensions of the Standard Model may include effective operators inducing this transition, but with different Lorentz structure resulting in enhanced branching fractions or changes to the angular distribution.

Previous studies performed by LHCb, CMS and ATLAS at CERN using proton-proton collision data showed deviations in a variable describing such a decay of up to $3.6\,\sigma$ to the prediction of the Standard Model. Performing such an angular study is only possible with a significant amount of data. Another environment to produce these large amounts of B mesons are the B-factories. These factories let electrons and positron collide at the center-of-mass energy corresponding to the invariant mass of the $\Upsilon(4S)$ which decays almost exclusively to pairs of B mesons. Since 2019, SuperKEKB with the Belle II experiment has been operating as such a factory and was able to collect a total integrated luminosity of $364\,\text{fb}^{-1}$ corresponding to a data sample of $\sim 400 \cdot 10^6\ B\bar{B}$ pairs.

This thesis will present comprehensive studies of a workflow to analyze the $B \to K^*\ell\ell$ decay at Belle II. This includes the development of *snapshot ensemble deep neural networks* for classification, which aim to maximize the efficiency and purity of the data samples for each possible charge configuration of the decay channel. These networks allow the two dimensional discriminating fit on the uncorrelated M_{bc} and ΔE variables to reliably determine the signal and non-signal contributions. In the next step, as a novel approach, another deep neural network is developed to model the detector response function following the *function approximation theorem*. In order to deal with remaining non-signal events on a statistical level, the *sWeights* method is studied and used. By applying this method it is possible to determine the true signal shape of the *target* variables (the kinematic variables to fully describe the decay) by calculating weights from the supplied PDF models of the uncorrelated *discriminating* variables (M_{bc} and ΔE). Using this method, the background contribution within the target variables has no longer be determined by simulation.

In order to test the developed workflow, the differential decay rate is extracted in the three angular dimensions in four bins of q^2, the di-lepton invariant mass squared. Exploiting transformation symmetries in the differential decay rate, the number of free parameters of the differential decay rate is reduced from eight down to three (F_L, S_3 and P_5') which are then determined on both simulation and real data.

The available dataset of the Belle II experiment contains $B \to K^*\ell\ell$ signal events in the single to low double digits in each q^2 bin. With this amount, it is not possible

to provide meaningful results for F_L, S_3 and P_5'. However, Belle II is expected to collect data until 2035 when a total integrated luminosity of $50\,\text{ab}^{-1}$ is recorded. This increase in available data by a factor of over 100 will enable a more competitive angular analysis that can be performed following the workflow that is presented in this thesis.

Contents

Chapter 1

Introduction

"No, no!" said the Queen.
"Sentence first - verdict afterwards."

Lewis Carroll, *Alice in Wonderlands* (1865)

Humans have sought to understand the natural world since ancient times, beginning with questions like *How to tame fire?* to "*[...], was die Welt im Innersten zusammenhält*" (trans.: *[...] what holds the world together in its innermost elements*) [Faust I, Vers 382 f / Goethe]. In the realms of mathematics, physics and technology, there have been vast improvements over the last centuries resulting in very successful and yet still incomplete models. The model trying to find a scientific answer to the philosophical question posed in Goethe's Faust is the Standard Model of particle physics. This model describes the interaction and structure of matter based on experimental observations and their implications. It is extremely successful in describing subatomic phenomena within the boundaries of the field of experimental particle physics. However, as mentioned, being a model, it is incomplete and not a *theory of everything* since it is not able to describe observations mainly coming from cosmology. Nowadays, a problem of the Standard Model is not the lack of competing models but rather the lack of uncharted areas where it breaks down and physics beyond the Standard Model could be found within the context of particle physics.

There are two possible approaches to reach out to these new uncharted areas. The high energy approach used by experiments at the Large Hadron Collider at CERN, reaches such high energies that the direct production of exotic heavy particles is possible. Studying these particles gives insight in the structure of matter in general. The other approach are high precision measurements which try to find indirect signatures of physics beyond the Standard Model. These effects could emerge as new particles in intermediate steps of decays. One of the most promising experiments of this frontier is the Belle II experiment located in Tsukuba, Japan. This experiment specializes on the measurement of events related to the decay of B mesons produced in a e^+e^--collider experiment operated mostly on the $\Upsilon(4S)$-resonance center-of-mass energy allowing the direct production of B meson pairs in a clean experimental environment. It is a rather new experiment and only started collecting data in 2019 and will continue to do so until 2035 when it is expected to have recorded a total integrated luminosity of $\int \mathcal{L}\,\mathrm{d}t = 50\,\mathrm{ab}^{-1}$.

Recent measurements in the last years showed intriguing discrepancies between the prediction of the Standard Model and experimental observations in particle decays undergoing a $b \to s\ell\ell$ transition. These flavor changing neutral currents cannot occur directly according to the Standard Model but only as higher order transitions

involving heavy bosons resulting in a high suppression of these decays compared to direct (*tree level*) transitions. Therefore, effects arising from physics beyond the Standard Model or new physics contributions could be measured and identified. However, due to the highly suppressed branching ratio of these $b \to s\ell\ell$ decays, the signal yield is very small and background events dominate, unless sophisticated background suppressions are applied. Furthermore, the remaining background has to be treated with care. Previous studies relied on simulation to determine the background contribution. However, this strategy only works with good modeling and is therefore prone to errors.

This thesis is divided into three separate parts. The first part presents the theoretical background needed to understand the underlying processes of this thesis. Also, the experimental setup and methods which will be used throughout the thesis will be introduced.

The second part will focus on the analysis of the angular distributions of the $B \to K^*\ell\ell$ decays with $\ell = e, \mu$ as they became a matter of interest in the past years due to the discrepancies found between the Standard Model prediction and measurements done at CERN, which might hint to physics beyond the Standard Model. This part will present the developed workflow to perform such an analysis of this angular distribution at Belle II. For this, the Long Shutdown 1 dataset (2019 to 2022) of Belle II with a total integrated luminosity of $\int \mathcal{L}_{\text{LS1}}\,\mathrm{d}t = 364\,\text{fb}^{-1}$ alongside Monte-Carlo generated data with $\int \mathcal{L}_{\text{MC}}\,\mathrm{d}t = 4 \cdot \int \mathcal{L}_{\text{LS1}}\,\mathrm{d}t$ is used. This workflow includes the applications of new strategies like using neural networks for classification and modeling of the detector response function to perform such an angular analysis. Furthermore, a rather new technique called *sWeights* is used and studied in order to obtain a clean signal shape in the angular distributions without relying on simulation to determine the background contribution. Finally, the complete workflow will be tested and studied on simulation as well as real data provided by the Belle II collaboration.

There will be an additional part in the appendix providing the documentation of the software which is developed for the pixel detector of Belle II (PXD) to find single event upsets (SEU) during online data taking. The PXD is the innermost detector of the Belle II experiment. Due to its closeness to the collision region of the experiment it has to withstand high radiation rates and ionizing particles. These particles can cause a random bit status change in the detector (SEU) stopping the detector from working as designed. Identifying these SEUs as soon as possible is therefore crucial for ensuring that the detector works as expected. Originally, these SEUs had to be found by the operator responsible for monitoring the detector while data taking and it will be shown that not all SEUs were found by them as soon as they showed up. This developed software will therefore increase the quality of Belle II data in the future.

Part I

Introduction and Methods

Chapter 2

Theoretical Foundations

> All models are wrong, but some are useful.
>
> George Box

2.1 The Standard Model

The Standard Model (SM) is the result of decades of measurements, discoveries and predictions. It is the widely accepted theory of particle physics. The fundamental idea of the SM is that everything is made out of particles with no internal structure. Mathematically it is formulated as a gauge theory.

2.1.1 Particles of the Standard Model

The particles of the SM can be separated by their spin (s) in categories of fermions (*matter* particles; $s = 1/2$) and bosons (*force* particles; $s = 0, 1$). The fermions can be considered as the building blocks of nature which can interact with each other by the exchange of three interacting-particles carrying the strong, weak and electromagnetic force. These forces are described by the gauge theory which links internal symmetries of the interacting particles to conserved charges.

The particles of the SM are depicted in Fig. 2.1. Fermions are colored in purple and green while bosons are indicated by red and yellow. Fermions themselves are further divided into leptons (green) and quarks (purple). Each of them appear in three generations of tuples, where in each tuple, both partners differ by one unit of electric charge[1]. Quarks can be further categorized in up-type (u, c, t quarks) and down-type (d, s, b quarks) particles. While quarks can interact with the electromagnetic (photon), strong (gluon) and weak (W and Z) force, leptons can only interact with the weak force, and the electromagnetic force if the lepton itself carries an electric charge.

2.1.2 The Electromagnetic Interaction

The interaction of fermions with the photon is based on the principles of the $U(1)$ gauge symmetry and field theory. This $U(1)$ group is based on the idea that the laws

[1] Our everyday world consists almost exclusively of 1st generation fermions. u and d quarks together form protons and neutrons (strong interaction) which can combine to heavier atoms alongside electrons (electromagnetic interaction).

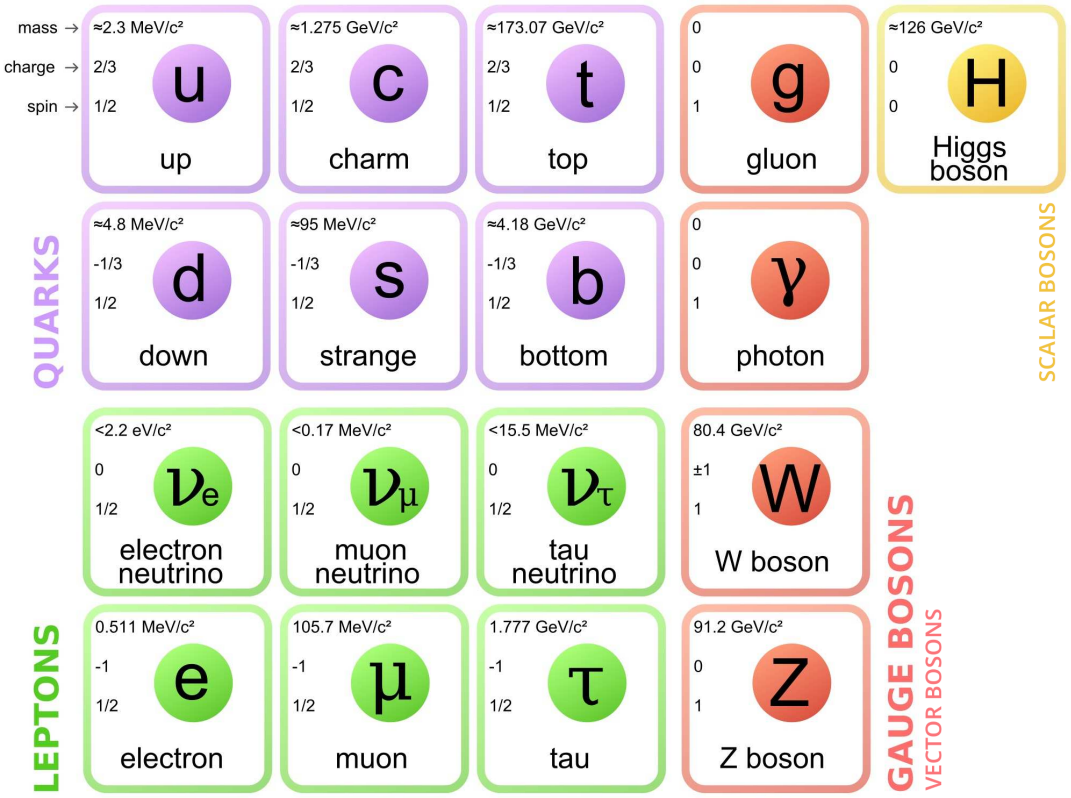


FIGURE 2.1: Particles of the SM. The three generations of leptons and quarks make up the fermions. The gauge boson are the force particles. The Higgs boson breaks the electroweak symmetry giving particles their mass [1].

of physics are independent on the choice of the local phase of the quantum field. The scalar field Lagrangian can be written as:

$$\mathcal{L} = (D^\mu \psi)^\dagger (D_\mu \psi) - m^2 \psi^\dagger \psi - \frac{1}{4} F_{\mu\nu} F^{\mu\nu} \tag{2.1}$$

where $\psi(x)$ is the wave function of a fermion with the mass m, $A_\mu(x)$ is a gauge field, $F_{\mu\nu}$ is the field strength tensor and $D_\mu \psi = (\partial_\mu \psi + iqA_\mu \psi)$. This is a locally gauge invariant quantity under the simultaneous transformations

$$\psi(x) \to e^{i\theta(x)} \psi(x) \qquad \text{and} \qquad A_\mu \to A_\mu - \frac{1}{q} \partial_\mu \theta(x). \tag{2.2}$$

As a result, the transformation $(D^\mu \psi)^\dagger (D_\mu \psi) \to (D^\mu \psi)^\dagger (D_\mu \psi) + O(\theta^2)$ is locally invariant. Furthermore, A_μ can be identified as the electromagnetical 4-potential.

2.1.3 Weak Interaction

The Lagrangian for two sorts of different fermions, each with a mass m, can be written as

$$\mathcal{L} = \bar{\Psi}(i\gamma^\mu D_\mu - m)\Psi - \frac{1}{4} G_{\mu\nu} G^{\mu\nu} \tag{2.3}$$

with

$$\Psi = \begin{pmatrix} f \\ g \end{pmatrix} \qquad \text{and} \qquad \bar{\Psi} = \begin{pmatrix} \bar{f} & \bar{g} \end{pmatrix} \tag{2.4}$$

and γ being a Dirac γ-matrix. This Lagrangian is gauge invariant under a local $SU(2)$ transformation

$$\Psi \to e^{\frac{i}{2}\sigma\theta}\Psi \qquad \text{and} \qquad \bar{\Psi} \to e^{-\frac{i}{2}\sigma\theta}\bar{\Psi} \tag{2.5}$$

with $\sigma = (\sigma_x, \sigma_y, \sigma_z)$ being the Pauli matrices. As a result, the Noether currents arise from this symmetry. In this theory, the conserved charge-like quantity is the weak isospin $\hat{T} = \int \mathrm{d}^3x \hat{\Psi}^\dagger \frac{\sigma}{2} \hat{\Psi}$. Furthermore, the interaction of three W particles (their isospin charge is one unit each) is possible in a gauge theory with local $SU(2)$ symmetry. In contrast, the photon itself carries no electromagnetic charge and, as a result, direct photon-photon interactions are not possible.

2.1.4 Electroweak Interaction

Both the electromagnetic $U(1)$ and the weak $SU(2)$ field theory can be unified to form the so-called *electroweak* interaction. This was done by Glashow in 1968 [2].

$$SU(2)_L \otimes U(1)_\mathrm{Y} \tag{2.6}$$

L represents the *left* handedness of the fermions. In general, this handedness can be obtained by using projectors:

$$\psi_L = P_L = \frac{1}{2}(1 - \gamma_5)\psi \tag{2.7}$$

$$\psi_R = P_R = \frac{1}{2}(1 + \gamma_5)\psi \tag{2.8}$$

where ψ is the fermionic field, $\gamma_5 = i\gamma_0\gamma_1\gamma_2\gamma_3$ with γ_i being a Dirac γ-matrix. The photon interacts with particles regardless of their handedness, while the weak interactions with the W and Z bosons only interact with left handed particles. Consequently, the weak isospin T, which relates the electrically charged part of the weak interaction, is zero for right handed particles.

$U(1)$ is the symmetry group of the weak hypercharge Y, which is defined by the formula:

$$Q = T_3 + \frac{\mathrm{Y}}{2} \tag{2.9}$$

This equation relates the electric charge Q of the particle to the third component of the weak isospin T_3 and the hypercharge Y. The charged $W^\pm$ boson carry $T_3 = \pm 1$, as a result they allow transitions between left handed up- and down-type particles.

Shortly after the Big Bang, the Universe was more symmetrical. The lepton fields are experiencing an internal $SU(2) \otimes U(1)$ symmetry. Leptons and their neutrinos were massless. Once the Universe cooled down to a temperature of below 10^{16} K a symmetry breaking phase transition took place.

$$SU(2)_L \otimes U(1)_\mathrm{Y} \xrightarrow{SSB} U(1)_\mathrm{EM} \tag{2.10}$$

Eq. 2.10 shows this spontaneous symmetry breaking (SSB) which is introduced by the Higgs mechanism. Due to this mechanism, the originally massless gauge bosons

mix into the massless γ and the heavy weak mediators Z and $W^{\pm}$. This also causes the fermions to have a mass [3].

2.1.5 Quantum Chromo Dynamics

The strong interaction is described by the local symmetry gauge group $SU(3)$. The boson mediating this strong force is the gluon. The color charge which arises from the symmetry of this group is conserved in strong interactions. Both quarks and these gluons carry a color charge, and hence they participate in strong interactions only with each other. Since the gluons themselves carry color charge, they are also able to interact with each other. The self-interacting of this gauge field provides an explanation of the asymptotic freedom. The color charge becomes stronger at larger distances allowing the gauge field to decay into a pair of gluons.

2.1.6 Quantum Numbers

The SM combines the electromagnetic and weak force with the electroweak interaction described by the Glashow-Weinberg-Salam model (GWS) and the Quantum Chromo Dynamics (QCD) which describes the interactions of the strong force [2] [4].

$$SU(3)_C \otimes SU(2)_L \otimes U(1)_Y \tag{2.11}$$

In the SM, left- and right handed fermionic particles are arranged into doublet and singlet states of the weak isospin for the weak interaction, respectively. The quantum numbers of the fermions in the SM are listed in Tab. 2.1.

TABLE 2.1: Quantum numbers of the fermions within the SM. T is the weak isospin with T_3 being its third component. Y is the hypercharge and Q the charge. The subscript L stands for left handed and R for right handed. Anti-particles are not shown in this table.

		Generation			Quantum Numbers			
	Field	(1)	(2)	(3)	T	T_3	Y	Q
Quarks	Q_L	$\begin{pmatrix} u \\ d' \end{pmatrix}_L$	$\begin{pmatrix} c \\ s' \end{pmatrix}_L$	$\begin{pmatrix} t \\ b' \end{pmatrix}_L$	$1/2$	$+1/2$ $-1/2$	$1/3$	$+2/3$ $-1/3$
	U_R	u_R	c_R	t_R	0	0	$+4/3$	$+2/3$
	D_R	d_R	s_R	b_R	0	0	$-2/3$	$-1/3$
Leptons	L_L	$\begin{pmatrix} \nu_e \\ e \end{pmatrix}_L$	$\begin{pmatrix} \nu_\mu \\ \mu \end{pmatrix}_L$	$\begin{pmatrix} \nu_\tau \\ \tau \end{pmatrix}_L$	$1/2$	$+1/2$ $-1/2$	-1	0 -1
	E_R	e_R	μ_R	τ_R	0	0	-2	-1

2.1.7 Charged Currents in the Quark Sector

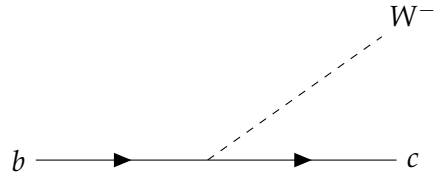


FIGURE 2.2: Feynman diagrams for weak transitions between generations.

Fig. 2.2 shows the Feynman diagram for the flavor transition between generations of quarks. This is only possible with the exchange of a charged W boson. As mentioned before, transitions from up- to down-type particles and vice versa are possible. Conventionally, the transition is described for the down-type particles. The flavor eigenstates, which are a superposition of the mass eigenstates, are tagged with a prime (e.g. d'). Cabibbo introduced this principle for two generations of quarks in 1963 [5] and it was expanded to three generations ten years later by Kobayashi and Maskawa [6] resulting in the so-called CKM matrix

$$V_{\mathrm{CKM}} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \tag{2.12}$$

which describes the transition as

$$\begin{pmatrix} d \\ s \\ b \end{pmatrix}_{\mathrm{weak}} = \begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = V_{\mathrm{CKM}} \begin{pmatrix} d \\ s \\ b \end{pmatrix}_{\mathrm{mass}} . \tag{2.13}$$

The CKM matrix is complex and, in order to conserve probability, it is constructed to be unitary.[2] It can be fully described by three mixing angles and one complex phase, resulting in a total of four free parameters. The entries of the CKM matrix (V_{ij}) appear at each charged current vertex. The indices i and j represent the flavors of the corresponding quarks. The complex-conjugated elements V_{ij}^* account for the couplings in $\mathcal{CP}$-conjugated processes. The complex phase leads to the relation

$$V_{ij} \neq V_{ij}^* \tag{2.14}$$

among the elements with $i \neq j$. This relation is the only known mechanism which leads to $\mathcal{CP}$ violation [7] [8].

For all generations i, j, the unitarity of the CKM matrix can also be expressed by

$$\sum_i V_{ij} V_{ik}^* = \delta_{jk} \qquad \text{and} \qquad \sum_j V_{ij} V_{kj}^* = \delta_{ik}. \tag{2.15}$$

When this relation is given, it can be visualized as triangles in the complex plane. It is common to use the triangle constructed by the following relation

$$V_{ud} V_{ub}^* + V_{cd} V_{cb}^* + V_{td} V_{tb}^* = \delta_{db} = 0 \tag{2.16}$$

since almost all elements can be determined by measurements related to B decays. By dividing each side of this equation by the experimentally best known term $V_{cd} V_{cb}^*$, the so-called Unitarity Triangle can be constructed. The three angles of the triangle are defined as

$$\alpha = \Phi_2 = \arg\left(-\frac{V_{td} V_{tb}^*}{V_{ud} V_{ub}^*} \right) \tag{2.17}$$

$$\beta = \Phi_1 = \arg\left(-\frac{V_{cd} V_{cb}^*}{V_{td} V_{tb}^*} \right) \tag{2.18}$$

[2] If this 3×3 matrix is a sub-matrix (i.e. there are more than three quark generations) then the CKM matrix would not be unitary.

$$\gamma = \Phi_3 = \arg\left(-\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*}\right) \tag{2.19}$$

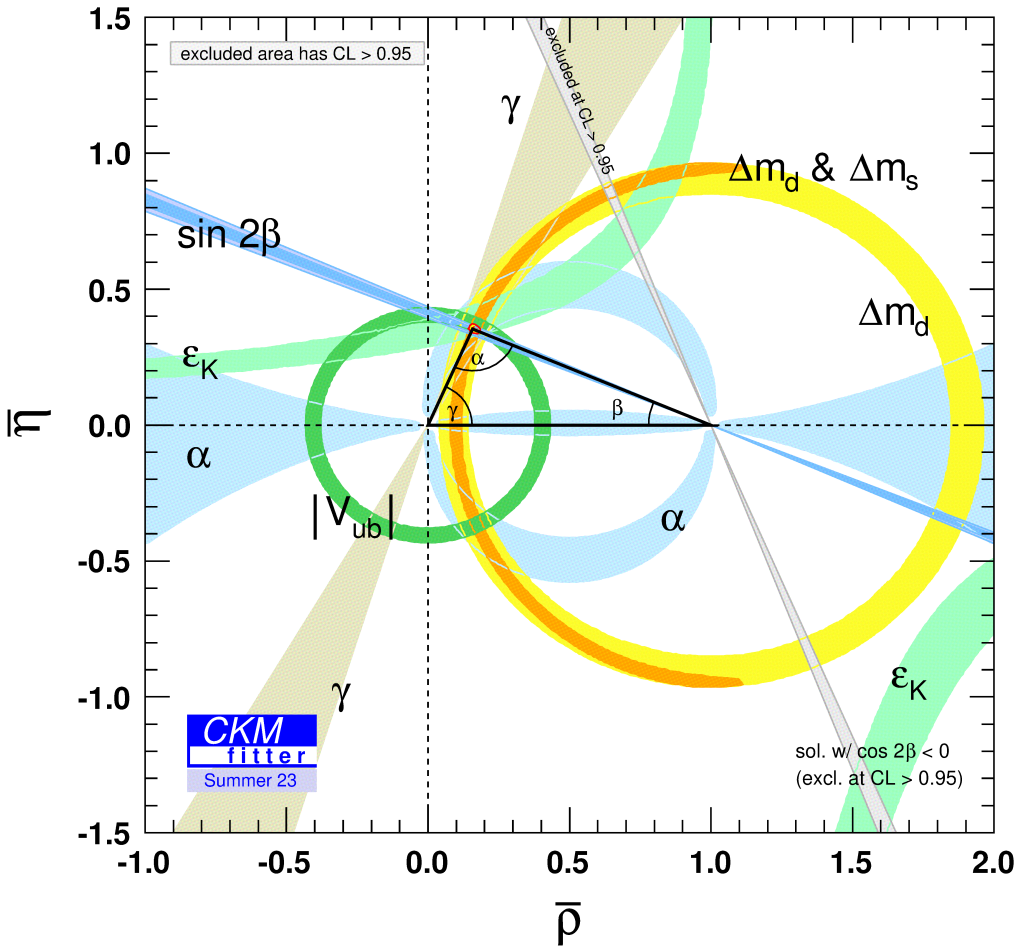


FIGURE 2.3: Latest constraints of the unitarity triangle provided by the CKM fitter group [9]. The parameter ϵ_K is determined by measuring the amount of $\mathcal{CP}$ violation observed in the mixing of neutral kaons. Δm_s and Δm_d describe the oscillation frequency of the B_s and B_d meson, respectively.

The latest constraints of these angles are depicted in Fig. 2.3. Here, the unitarity triangle is shown in the $\bar{\rho}\bar{\eta}$ plane with

$$\bar{\rho} + i\bar{\eta} = -\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*}. \tag{2.20}$$

The flavor transitions within one generation in the CKM matrix are represented by the diagonal elements of the CKM matrix. These magnitudes are close to one. Non-diagonal entries are much smaller in comparison.

$$|V_{ij}| \approx \begin{pmatrix} 0.974 & 0.225 & 0.003 \\ 0.225 & 0.973 & 0.041 \\ 0.009 & 0.040 & 0.999 \end{pmatrix} \tag{2.21}$$

As a result, some flavor changing transitions are suppressed by the CKM mechanism. For example, the quark transitions $b \to c$ and $b \to u$ have both very small CKM elements (see Eqs. 2.12 and 2.21) which leads directly to the long lifetime of B mesons[3] [9].

2.1.8 Flavor Changing Neutral Currents

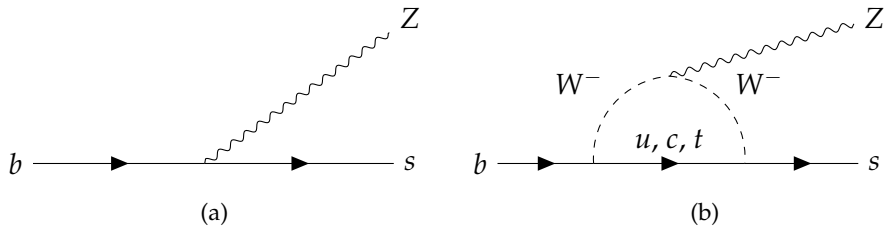


FIGURE 2.4: (a) This flavor changing neutral current is forbidden within the SM. (b) allowed Feynman diagram for flavor changing neutral currents within the SM. This decay is highly suppressed due to the higher order.

In the SM, flavor changing neutral currents (FCNC) are forbidden at tree level, as depicted in Fig. 2.4a according to the GIM (Glashow–Iliopoulos–Maiani) mechanism. One could imagine that the K^0 meson ($\bar{s}$ and d quarks) decays electroweak to two μ as depicted in Fig. 2.5. However, this decay is extremely suppressed and, as a consequence, the GIM mechanism was introduced. Without the GIM mechanism this decay would have a branching fraction in the order of $\sim 10^{-3}$. However, an upper limit of the branching fraction of only $2.1 \cdot 10^{-10}$ is determined [10].

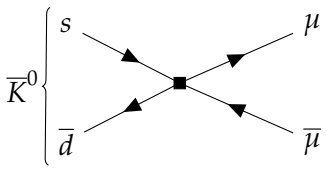


FIGURE 2.5: The decay $K^0 \to \mu\bar{\mu}$ is extremely suppressed and resulted in the introduction of the GIM mechanism which forbids flavor changing neutral currents on tree level in the SM.

The GIM mechanism was proposed at a time when only the three lightest quarks (u, d, s) were discovered. However, this mechanism requires the existence of the fourth quark (c). The prediction of the charm quark is therefore credited to Glashow, Iliopoulos and Maiani [11].[4]

However, these FCNC transitions can appear via higher order penguin or box diagrams. An example for a FCNC penguin is given in Fig. 2.4b. For B mesons, the most dominant decay mode is the $b \to c$ transition. The transition $b \to u$ is already suppressed by a factor of $V_{ub}/V_{cb} \approx 0.01$ compared to the former mode. Thus, these FCNCs are very rare compared to tree level decays, making them very well suited for the search for physics Beyond the SM (BSM) which would not be masked by a large SM contribution. Additionally, heavy particles can appear within the loop.

[3] A B meson contains one b anti-quark and one first generation quark.

[4] A bit more context about the prediction of the fourth quark through the GIM mechanism can be found in App. A.1.

This enables indirect sensitivity for high energy scales where BSM could appear as well.

2.1.9 Fermis Golden Rule

In first order, Fermis Golden Rule describes transition probabilities or decay rates between the initial $|i\rangle$ and final $|f\rangle$ energy eigenstate of a quantum system according to perturbation theory.

$$\Gamma_{i\to f} = \frac{2\pi}{\hbar} |\langle f|H'|i\rangle|^2 \rho(E_f) \tag{2.22}$$

In Eq. 2.22, $\langle f|H'|i\rangle$ is the matrix element of the perturbation H' between the final and initial state and $\rho(E_f)$ is the density of final states at an energy of E_f. Since decay rates can be measured in experiments, they can be directly related to the matrix element.

2.1.10 Effective Hamiltonian

This thesis will study the decay of B mesons. Due to the underlying mediators, these decays take place at variety of energy scales. The energy scale of the electroweak force is set by the mass of the heavy W and Z bosons. The interaction distance is inversely proportional to the mass of its propagator. For hadronization, the energy scale lies in the order of $\delta x \propto 1/m_b$. Simultaneous calculations of decay amplitudes in several energy scales are rather complicated in theoretical frameworks. As a result, long distance processes are separated from short distance processes, which can be computed by the use of perturbation techniques. The result is an effective theory where the heavy field gets integrated out. This leaves the hadronic field separated.

$$\mathcal{A}(M \to F) = \langle F | \mathcal{H}_{eff} | M \rangle = \frac{G_F}{\sqrt{2}} \sum_i V^i_{\mathrm{CKM}} \mathcal{C}_i(\mu) \langle F | \mathcal{O}_i(\mu) | M \rangle \tag{2.23}$$

Eq. 2.23 shows the calculation of the matrix element for the quark transition from state M to F. In this equation, $\mathcal{H}_{eff}$ is the effective Hamiltonian for this transition, G_F is the Fermi constant, V^i_{CKM} is the CKM matrix element of this transition and μ is the renormalization scale. $\mathcal{C}_i(\mu)$ is the so-called Wilson coefficient which contains the short distance effects. The long distance effects are described within the $\mathcal{O}_i(\mu)$ operators which are handled by non-perturbative theory. These coefficients can be calculated with the effects of BSM included. Therefore, they are of high interest in the search for BSM since the predictions of the SM can be studied with high precision and one can differentiate the impact between BSM scenarios.

2.1.11 Effective Hamiltonian for the $b \to s\ell\ell$ Transition

The $b \to s$ process can take place via an electromagnetic, gluonic and weak penguin and box diagrams. The effective Hamiltonian is then given by the operator product expansion for one-loop processes by:

$$\mathcal{H}_{eff} = -\frac{4G_F}{\sqrt{2}} V_{tb} V^*_{ts} \sum_{i=1}^{10} \mathcal{C}_i(\mu) \mathcal{O}_i(\mu) \tag{2.24}$$

In this equation, $V_{qq'}$ is the CKM matrix element according to this transition.

TABLE 2.2: Wilson coefficient and their corresponding process. The corresponding Feynman diagrams are depicted in Fig. 2.6

Wilson Coefficient	Operator Process
$\mathcal{O}_{1,2}$	current-current
$\mathcal{O}_{3-6}$	Penguin
$\mathcal{O}_7$	Electromagnetic
$\mathcal{O}_8$	Chromomagnetic
$\mathcal{O}_9$	Vector component of the electroweak penguin
$\mathcal{O}_{10}$	Axial vector component of the electroweak penguin

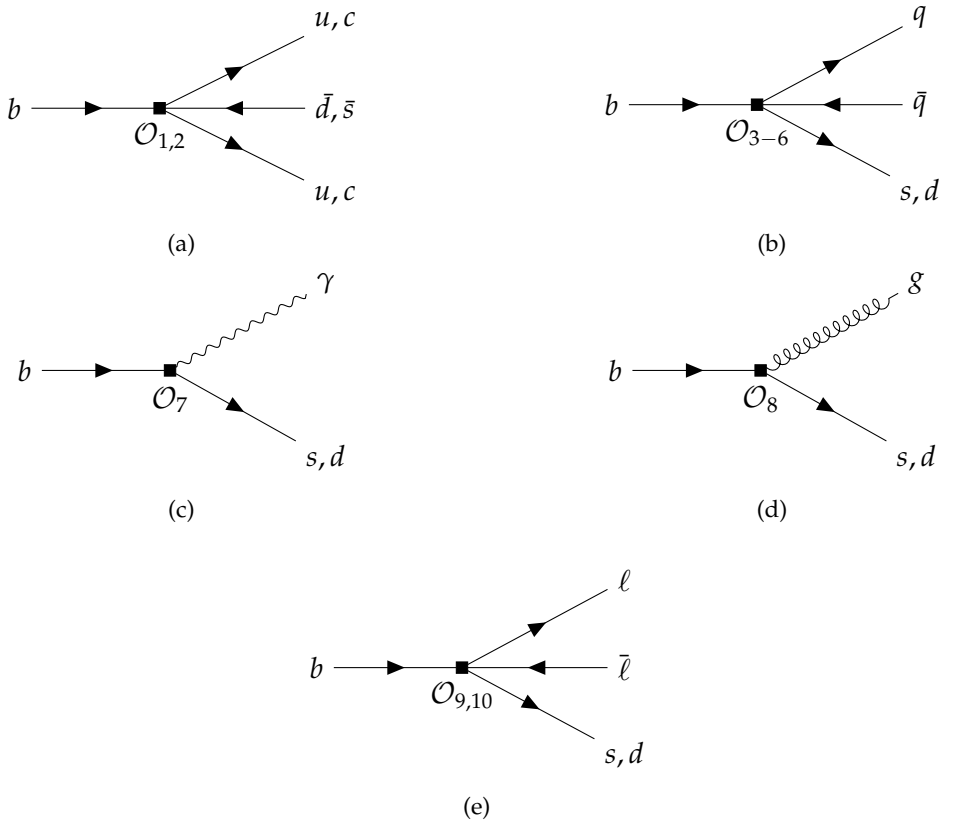


FIGURE 2.6: The underlying processes of the Wilson operators with a b quark in the initial state. (a) represents the underlying processes of $\mathcal{O}_{1,2}$, (b) of $\mathcal{O}_{3-6}$, (c) of $\mathcal{O}_7$, (d) of $\mathcal{O}_8$ and (e) of $\mathcal{O}_{9,10}$, depicted as Feynman diagrams.

The Wilson coefficients are described in Tab. 2.2 and the underlying process is depicted as Feynman diagrams in Fig. 2.6. At a leading order, only the operators $\mathcal{O}_7$, $\mathcal{O}_9$ and $\mathcal{O}_{10}$ contribute in the $b \to s\ell\ell$ transition.

$$\mathcal{O}_7 = \frac{e}{16\pi^2}\bar{s}_\alpha \sigma_{\mu\nu}\left(m_s P_L + m_b P_R\right) b_\alpha F^{\mu\nu} \tag{2.25}$$

$$\mathcal{O}_9 = \frac{e^2}{16\pi}\bar{s}_\alpha \gamma^\mu P_L b_\alpha \bar{\ell}\gamma_\mu \ell \tag{2.26}$$

$$\mathcal{O}_{10} = \frac{e^2}{16\pi}\bar{s}_\alpha \gamma^\mu P_L b_\alpha \bar{\ell}\gamma_\mu\gamma_5\ell \tag{2.27}$$

In these equations the Wilson coefficients for left handed operators are shown. Here, α is the color index, e is the electromagnetic coupling constant, P_L and P_R are the projection operators $1/2(1-\gamma_5)$ and $1/2(1+\gamma_5)$, respectively and $\sigma^{\mu\nu} = [\gamma^\mu, \gamma^\nu]$. Furthermore, the fields for a strange quark, a bottom quark, and a lepton are represented by s, b and ℓ, respectively. Finally, $F^{\mu\nu}$ is the electromagnetic field strength tensor. To obtain the Wilson coefficient for the right handed operators, denoted as $\mathcal{O}'_7$, $\mathcal{O}'_9$, and $\mathcal{O}'_{10}$, P_L and P_R need to be exchanged with each other.

2.2 The Differential Decay Rate

For the $B \to K^*\ell\ell$ decay, three Feynman diagrams contribute to the amplitude in the lowest order. These three diagrams are depicted in Figs. 2.7a to 2.7c. They contain two penguin diagrams exchanging a Z boson or γ (Figs. 2.7a, 2.7b) and a box diagram with a W boson loop (Fig. 2.7c) [12]. Figs. 2.7d and 2.7e show processes which are not part of the SM. In Fig. 2.7d a super-symmetric charged Higgs and in Fig. 2.7e a leptoquark occurs to suppress or enlarge the amplitude of the decay. These changes in the amplitude could be measured in an experiment paving the way for BSM.

2.2.1 The $B \to K^*\ell\ell$ Decay Topology

BSM could not only change the amplitude of the decay, short distance interaction would also be noticeable in a change of the angular distribution. To achieve this, the decay needs to be described by independent kinematic variables. In this thesis, the decay of B mesons will be analyzed. These mesons consist of a b and a first generation quark and can undergo the $B \to K^*(\to K\pi)\ell\ell$ decay with $\ell = e, \mu$. In order to describe this decay completely, five variables are sufficient. The most popular choice is $q^2 = M^2_{\ell\ell}$, $M_{K\pi}$ and three angles $\cos\theta_\ell$, $\cos\theta_K$ and ϕ, which are shown in Fig. 2.8. In this thesis, the angle θ_ℓ is defined as the angle between the direction of the lepton ℓ^+ (ℓ^-) and the direction of the B ($\bar{B}$) meson in the rest frame of the dilepton system. The angle θ_K is defined as the angle between the direction of the kaon K ($\bar{K}$) and the direction of the B ($\bar{B}$) meson in the rest frame of the K^* meson. The angle ϕ is defined as the angle between the planes created by the dilepton system ($\ell\ell$) and the dihadron system ($K\pi$) in the rest frame of the B meson [12].

In the following equations, $p_\alpha^{(\beta)}$ is the direction of the momentum of particle α in the rest frame of particle or system β resulting in for both the B^0 and B^+ (containing $\bar{b}$ quark) in:

$$\cos\theta_\ell = \left(\hat{p}_{\ell^+}^{(\ell^+\ell^-)}\right)\cdot\left(\hat{p}_{\ell^+\ell^-}^{(B)}\right) = \left(\hat{p}_{\ell^+}^{(\ell^+\ell^-)}\right)\cdot\left(-\hat{p}_B^{(\ell^+\ell^-)}\right) \tag{2.28}$$

$$\cos\theta_K = \left(p_K^{(K^*)}\right)\cdot\left(p_{K^*}^{(B)}\right) = \left(p_K^{(K^*)}\right)\cdot\left(-p_B^{(K^*)}\right) \tag{2.29}$$

The angle ϕ is calculated by:

$$\cos\phi = \left(\hat{p}_{\ell^+}^{(B)}\times\hat{p}_{\ell^-}^{(B)}\right)\cdot\left(\hat{p}_K^{(B)}\times\hat{p}_\pi^{(B)}\right) \tag{2.30}$$

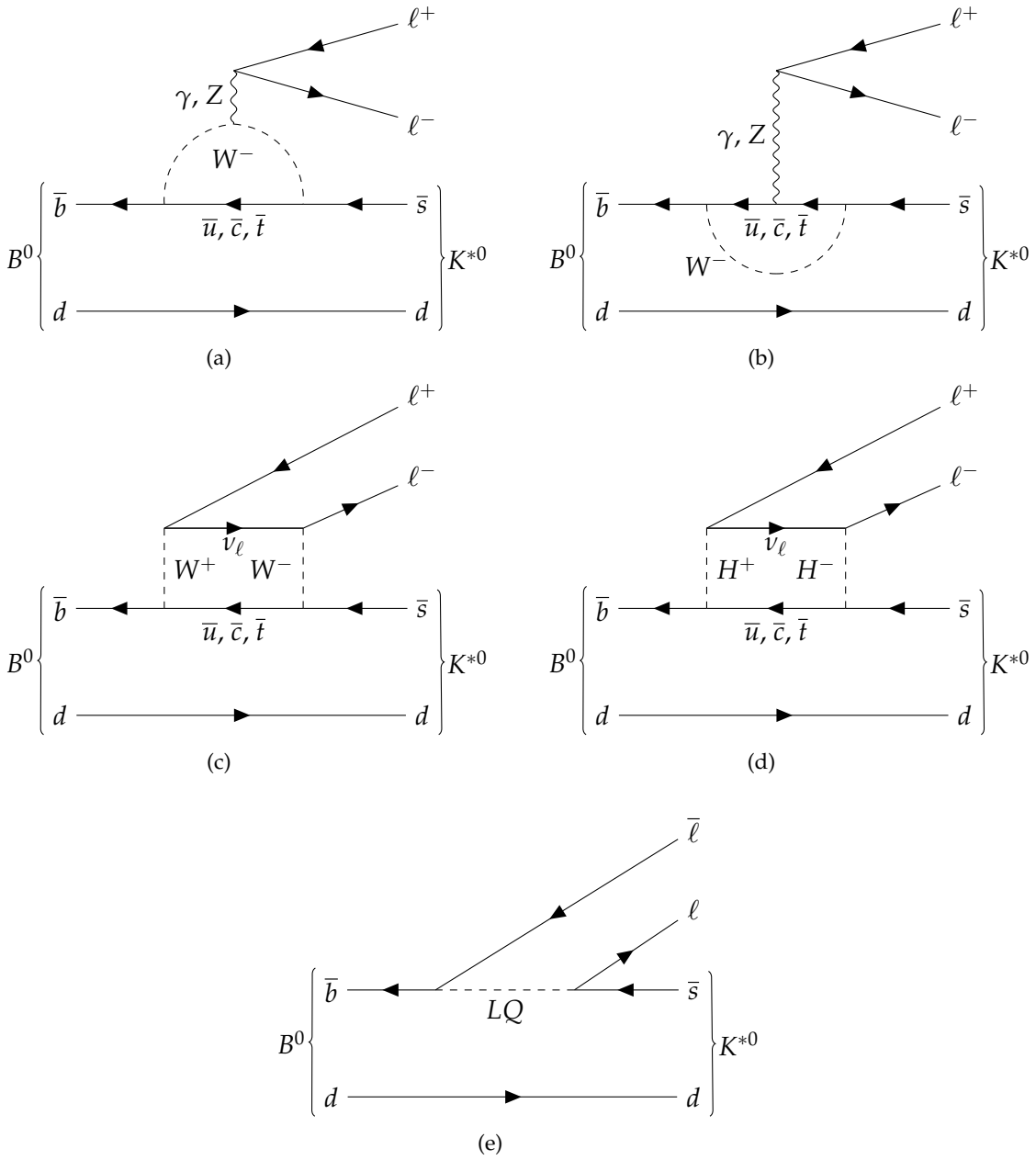


FIGURE 2.7: Feynman diagrams (a) and (b) show penguin and (c) a box process for the decay $B^0 \to K^{*0}\ell^+\ell^-$ within the SM. (d) shows a scenario outside of the SM. Charged Higgs are replacing the W boson loop. (e) shows the same process but the boson of the interaction is a leptoquark. This particle is also not part of the SM. In order to obtain the $B^+ \to K^{*+}\ell^+\ell^-$ Feynman diagram, the spectator d quark has to be replaced by a u quark.

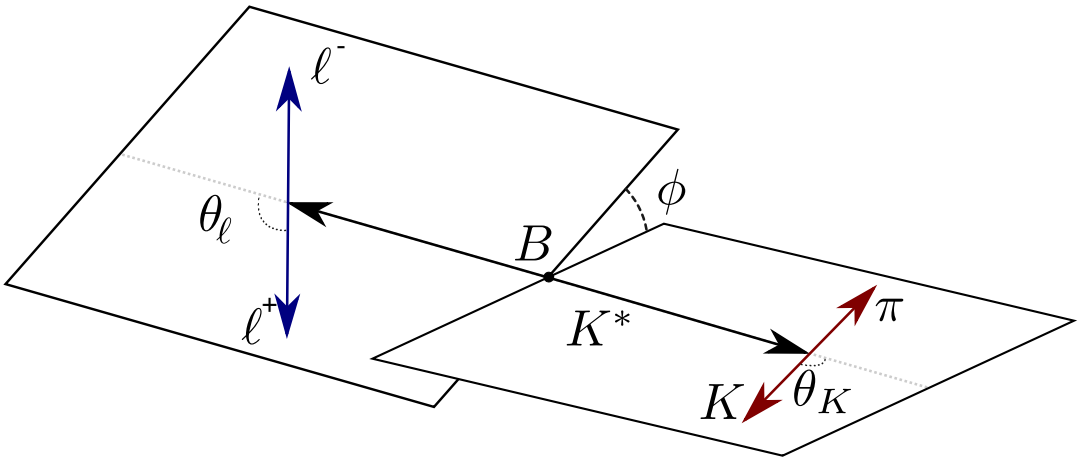


FIGURE 2.8: Definition of the observables for the decay $B \to K^* \ell\ell$.

$$\sin\phi = \left[\left(\hat{p}_{\ell^+}^{(B)} \times \hat{p}_{\ell^-}^{(B)}\right) \times \left(\hat{p}_K^{(B)} \times \hat{p}_\pi^{(B)}\right)\right] \cdot \hat{p}_{K^*}^{(B)} \tag{2.31}$$

With $\bar{B}^0$ or B^- particles (containing a b quark), these equations will change to:

$$\cos\theta_\ell = \left(\hat{p}_{\ell^-}^{(\ell^+\ell^-)}\right) \cdot \left(\hat{p}_{\ell^+\ell^-}^{(\bar{B})}\right) = \left(\hat{p}_{\ell^-}^{(\ell^+\ell^-)}\right) \cdot \left(-\hat{p}_{\bar{B}}^{(\ell^+\ell^-)}\right) \tag{2.32}$$

$$\cos\theta_K = \left(\hat{p}_K^{(K^*)}\right) \cdot \left(\hat{p}_{K^*}^{(\bar{B})}\right) = \left(\hat{p}_K^{(K^*)}\right) \cdot \left(-\hat{p}_{\bar{B}}^{(K^*)}\right) \tag{2.33}$$

$$\cos\phi = \left(\hat{p}_{\ell^-}^{(\bar{B})} \times \hat{p}_{\ell^+}^{(\bar{B})}\right) \cdot \left(\hat{p}_K^{(\bar{B})} \times \hat{p}_\pi^{(\bar{B})}\right) \tag{2.34}$$

$$\sin\phi = \left[\left(\hat{p}_{\ell^-}^{(\bar{B})} \times \hat{p}_{\ell^+}^{(\bar{B})}\right) \times \left(\hat{p}_K^{(\bar{B})} \times \hat{p}_\pi^{(\bar{B})}\right)\right] \cdot \hat{p}_{K^*}^{(\bar{B})} \tag{2.35}$$

2.2.2 The Differential Decay Rate for $B \to K^* \ell\ell$

Since the lifetime of the K^* is in the order of 1×10^{-23} s [13], only the daughters of the $K^* \to K\pi$ can be detected. By measuring the angle between these daughters, information about the polarization of the mother particle can be gained. To obtain the differential decay rate of the $B \to K^* \ell\ell$ channel, one has to square the matrix element, sum over all spins of the final state particles ($K\pi\ell\ell$) and restrict the kinematics of the four-body decay. A detailed description of this method can be found in [14]. Due to the low expected available number of events the S-wave contribution of from the $K^*(892)$ will not be included. As a result, the variable $M_{K\pi}$ will be considered as a fixed value and the number of variables to describe the decay fully goes down to four. The result is the following equation:

$$\frac{\mathrm{d}^4\Gamma}{\mathrm{d}\cos\theta_\ell \mathrm{d}\cos\theta_K \, \mathrm{d}\phi \mathrm{d}q^2} = \frac{9}{32\pi} I\left(q^2, \theta_\ell, \theta_K, \phi\right) \tag{2.36}$$

with

$$
\begin{aligned}
I\left(q^2, \theta_\ell, \theta_K, \phi\right) = & \left(I_1^s \sin^2\theta_K + I_1^c \cos^2\theta_K\right. \\
& + \left(I_2^s \sin^2\theta_K + I_2^c \cos^2\theta_K\right) \cos 2\theta_\ell \\
& + I_3 \sin^2\theta_K \sin^2\theta_\ell \cos 2\phi \\
& + I_4 \sin 2\theta_K \sin 2\theta_\ell \cos\phi \\
& + I_5 \sin 2\theta_K \sin\theta_\ell \cos\phi \\
& + I_6 \sin^2\theta_K \cos\theta_\ell \\
& + I_7 \sin 2\theta_K \sin\theta_\ell \sin\phi \\
& + I_8 \sin 2\theta_K \sin 2\theta_\ell \sin\phi \\
& \left. + I_9 \sin^2\theta_K \sin\theta_\ell \sin 2\phi\right) .
\end{aligned} \tag{2.37}
$$

In Eq. 2.37, $I_i^{(a)}$ are angular coefficients as functions of q^2 and they are written in terms of the K^* transversity amplitudes [14]. An advantage of this notation is that the angular variables are separated from the q^2 dependencies. All these coefficients $I_i^{(a)}$ are complete physical observables and contain the full information obtained from the experimental measurements. Since they are functions of the Wilson coefficients, short-distance effects are contained within them and BSM can influence them directly. The definitions used in Eq. 2.37 are only valid for the decay $B^0 \to K^{*0}\ell\ell$. The corresponding expression for the $\mathcal{CP}$ conjugated decay $\bar{B}^0 \to \bar{K}^{*0}\ell\ell$ is:

$$
\frac{\mathrm{d}^4\bar{\Gamma}}{\mathrm{d}\cos\theta_\ell \mathrm{d}\cos\theta_K\, \mathrm{d}\phi \mathrm{d}q^2} = \frac{9}{32\pi} \bar{I}\left(q^2, \theta_\ell, \theta_K, \phi\right) \tag{2.38}
$$

The function $\bar{I}\left(q^2, \theta_\ell, \theta_K, \phi\right)$ can be obtained from Eq. 2.37 by using the following modifications [14]:

$$
I^{(a)}_{1,2,3,4,7} \to I^{(a)}_{1,2,3,4,7} \qquad \text{and} \qquad I^{(a)}_{5,6,8,9} \to -I^{(a)}_{5,6,8,9} \tag{2.39}
$$

The resulting differential decay rate after combining Eqs. 2.36 and 2.38 for the measurements of B^0 and $\bar{B}^0$ is:

$$
\frac{\mathrm{d}^4(\Gamma + \bar{\Gamma})}{\mathrm{d}\cos\theta_\ell \mathrm{d}\cos\theta_K\, \mathrm{d}\phi \mathrm{d}q^2} = \frac{9}{32\pi} \sum_{i=1}^{9} \left(I_i + \bar{I}_i\right) f_i\left(\cos\theta_\ell, \cos\theta_K, \phi\right) \tag{2.40}
$$

For this thesis, all measurements will be conducted for averaged $\mathcal{CP}$ quantities only. The $\mathcal{CP}$ averaged $I_i^{(a)}$ terms are called S_i for the symmetric and A_i for the asymmetric $\mathcal{CP}$ ones.

$$
S_i^{(a)} = \frac{I_i^{(a)} + \bar{I}_i^{(a)}}{(\Gamma + \bar{\Gamma})} \tag{2.41}
$$

$$
A_i^{(a)} = \frac{I_i^{(a)} - \bar{I}_i^{(a)}}{(\Gamma + \bar{\Gamma})} \tag{2.42}
$$

2.2.3 Definition of Related Observables

In an angular measurement of $B \to K^*\ell\ell$, three variables are often used. These three variables are the forward-backward asymmetry A_{FB}, the longitudinal polarization of the K^* F_L and the transverse polarization asymmetry $A_T^{(2)}$. They are defined as:

$$A_{FB} = \frac{3}{4}\frac{I_6}{I_1^c + 4I_2^s} = \frac{3}{4}S_6 \tag{2.43}$$

$$F_L = \frac{I_1^c}{I_1^c + 4I_2^s} = S_2^c = S_2^s - 1 \tag{2.44}$$

$$A_T^{(2)} = \frac{2I_3}{4I_2^s} = \frac{2S_3}{1 - F_L} \tag{2.45}$$

In order to cancel out as many theoretical hadronic uncertainties on $I_i^{(a)}$ as possible, a different set of combinations are used as observables. In particular, the functions P_i' defined as follows are considered to be mostly free from form-factor uncertainties [14].

$$P'_{i=4,5,6,8} = \frac{S_{j=4,5,7,8}}{\sqrt{F_L\,(1 - F_L)}} \tag{2.46}$$

2.2.4 Reducing the Number of Observables

In principle, it is already possible to determine all S_i observables experimentally by fitting the probability density function (PDF; Sec. 4.1.1) to data. However, with extremely rare B decays, the signal yield is often very low and the fit would be very unstable. It is therefore necessary to reduce the number of observables by integrating over one or two angles, and to only consider the projection of the differential decay rate of the remaining angle. The equations for the possible projections are then:

$$\frac{1}{\Gamma}\frac{\mathrm{d}^2\Gamma}{\mathrm{d}\cos\theta_\ell \mathrm{d}q^2} = \frac{3}{4}F_L\left(1 - \cos^2\theta_\ell\right) + \frac{3}{8}\left(1 - F_L\right)\left(1 + \cos^2\theta_\ell\right) + A_{FB}\cos\theta_\ell, \tag{2.47}$$

$$\frac{1}{\Gamma}\frac{\mathrm{d}^2\Gamma}{\mathrm{d}\cos\theta_K\,\mathrm{d}q^2} = \frac{3}{2}F_L\cos^2\theta_K + \frac{3}{4}\left(1 - F_L\right)\left(1 - \cos^2\theta_K\right), \tag{2.48}$$

$$\frac{1}{\Gamma}\frac{\mathrm{d}^2\Gamma}{\mathrm{d}\phi\mathrm{d}q^2} = \frac{1}{2\pi}\left(1 + \frac{1}{2}\left(1 - F_L\right)A_T^{(2)}\cos 2\phi + A_{Im}\sin\phi\right), \tag{2.49}$$

with A_{Im} defined as:

$$A_{Im} = \frac{I_9}{I_1^c + 4I_2^s} \tag{2.50}$$

2.2.5 Improving the Statistical Sensitivity

The complete differential decay rate of $B \to K^*\ell\ell$ with $\mathcal{CP}$ averaged observables can be obtained by using the definitions from above and from reference [14]. The resulting equation is [15]:

$$\begin{aligned}\frac{1}{\mathrm{d}\Gamma/\mathrm{d}q^2}\frac{\mathrm{d}^4\Gamma}{\mathrm{d}\cos\theta_\ell \mathrm{d}\cos\theta_K\,\mathrm{d}\phi \mathrm{d}q^2} =& \frac{9}{32\pi}\Big[\frac{3}{4}(1-F_L)\sin^2\theta_K + F_L\cos^2\theta_K \\ &+ \frac{1}{4}(1-F_L)\sin^2\theta_K\cos 2\theta_\ell \\ &- F_L\cos^2\theta_K\cos 2\theta_\ell \\ &+ S_3\sin^2\theta_K\sin^2\theta_\ell\cos 2\phi \\ &+ S_4\sin 2\theta_K\sin 2\theta_\ell\cos\phi \\ &+ S_5\sin 2\theta_K\sin\theta_\ell\cos\phi \\ &+ S_6\sin^2\theta_K\cos\theta_\ell \\ &+ S_7\sin 2\theta_K\sin\theta_\ell\sin\phi \\ &+ S_8\sin 2\theta_K\sin 2\theta_\ell\sin\phi \\ &+ S_9\sin^2\theta_K\sin^2\theta_\ell\sin 2\phi\Big]\end{aligned} \tag{2.51}$$

There are a total of eight free parameters. All of them can be determined by a direct fit to data. For Belle II, the expected statistic is not sufficient enough to perform an eight-dimensional fit to the data, as of now. To enhance statistical sensitivity, a transformation technique will be applied. This will decrease the number of dimensions thereby, improving the statistical sensitivity. A detailed description of this transformation can be found in reference [16] and [17]. The basic idea is to apply a transformation to certain regions in the three-dimensional angular space and to then exploit the symmetries of cosine and sine, canceling out terms in the equation. The result is a reduction of the free parameters in the fit without the negative effect of losing experimental sensitivity. After applying the following transformations to the dataset, it is possible to be independently sensitive to the observable of interest:

$$P_4', S_4 : \begin{cases} \phi \to -\phi & \text{for } \phi < 0 \\ \phi \to \pi - \phi & \text{for } \theta_\ell > \pi/2 \\ \theta_\ell \to \pi - \theta_\ell & \text{for } \theta_\ell > \pi/2 \end{cases} \tag{2.52}$$

$$P_5', S_5 : \begin{cases} \phi \to -\phi & \text{for } \phi < 0 \\ \theta_\ell \to \pi - \theta_\ell & \text{for } \theta_\ell > \pi/2 \end{cases} \tag{2.53}$$

$$P_6', S_7 : \begin{cases} \phi \to \pi - \phi & \text{for } \phi > \pi/2 \\ \phi \to -\pi - \phi & \text{for } \phi < -\pi/2 \\ \theta_\ell \to \pi - \theta_\ell & \text{for } \theta_\ell > \pi/2 \end{cases} \tag{2.54}$$

$$P_8', S_8 : \begin{cases} \phi \to \pi - \phi & \text{for } \phi > \pi/2 \\ \phi \to -\pi - \phi & \text{for } \phi < -\pi/2 \\ \theta_K \to \pi - \theta_K & \text{for } \theta_\ell > \pi/2 \\ \theta_\ell \to \pi - \theta_\ell & \text{for } \theta_\ell > \pi/2 \end{cases} \tag{2.55}$$

Using these transformations, all terms of Eq. 2.51 vanish except for the first five and the corresponding S_i and P_j' term. As a consequence, the number of free parameters for each transformed decay rate is reduced to three. Namely, F_L, S_3 and one of the observables corresponding to the transformation $S_{4,5,7,8}$ or $P'_{4,5,6,8}$ [18].

Since this thesis will focus on the variable of P_5', only the transformation according to Eq. 2.53 will be applied. First ϕ is transformed with $\phi \to -\phi$ for $\phi < 0$. This will transform Eq. 2.51 to:

$$\begin{aligned}\frac{1}{\mathrm{d}\Gamma/\mathrm{d}q^2}\frac{\mathrm{d}^4\Gamma}{\mathrm{d}\cos\theta_\ell \mathrm{d}\cos\theta_K\,\mathrm{d}\phi \mathrm{d}q^2} =& \frac{9}{16\pi}\Big[\frac{3}{4}(1-F_L)\sin^2\theta_K + F_L\cos^2\theta_K \\ &+ \frac{1}{4}(1-F_L)\sin^2\theta_K\cos 2\theta_\ell \\ &- F_L\cos^2\theta_K\cos 2\theta_\ell \\ &+ S_3\sin^2\theta_K\sin^2\theta_\ell\cos 2\phi \\ &+ S_4\sin 2\theta_K\sin 2\theta_\ell\cos\phi \\ &+ S_5\sin 2\theta_K\sin\theta_\ell\cos\phi \\ &+ S_6\sin^2\theta_K\cos\theta_\ell\Big]\end{aligned} \tag{2.56}$$

Next, θ_ℓ will be folded following $\theta_\ell \rightarrow \pi - \theta_\ell$, which, together with Eq. 2.46, will finally lead to:

$$\begin{aligned}\frac{1}{\mathrm{d}\Gamma/\mathrm{d}q^2}\frac{\mathrm{d}^4\Gamma}{\mathrm{d}\cos\theta_\ell \mathrm{d}\cos\theta_K \mathrm{d}\phi \mathrm{d}q^2} =& \frac{9}{8\pi}\Big[\frac{3}{4}(1-F_L)\sin^2\theta_K + F_L\cos^2\theta_K \\ &+ \frac{1}{4}(1-F_L)\sin^2\theta_K\cos 2\theta_\ell \\ &- F_L\cos^2\theta_K\cos 2\theta_\ell \\ &+ S_3\sin^2\theta_K\sin^2\theta_\ell\cos 2\phi \\ &+ \sqrt{F_L(1-F_L)}P_5'\sin 2\theta_K\sin\theta_\ell\cos\phi\Big]\end{aligned} \tag{2.57}$$

Studies concerning the improvement of statistical sensitivity of this transformation were performed with data created by the workflow presented in this thesis and the results of these studies can be found in [19].

2.3 Recent Measurements

Measurements by LHCb ignited interest in the variable P_5'. This experiment was the first to provide a full angular measurement in all three angles. As a result, they were able to extract the variables A_{FB}, F_L, $A_T^{(2)}$, A_T^{Re}, S_3, S_9 and A_9 [12]. In 2013, LHCb extracted $P'_{4,5,6,8}$ together with F_L and $A_T^{(2)}$ in a different analysis of the same dataset. This analysis was performed in six bins of q^2 [16]. Among the 24 measurements, one anomaly was found. This $3.6\,\sigma$ deviation from the SM became to be known as the P_5' anomaly in the q^2-region $4.30\,\mathrm{GeV}^2/c^4 < q^2 < 8.68\,\mathrm{GeV}^2/c^4$.

Since then, the P_5' variable was measured by multiple experiments. In 2016, the Belle collaboration conducted a P_5' analysis with a dataset corresponding to a total integrated luminosity of $\int \mathcal{L}_{\mathrm{Belle}}\,\mathrm{d}t = 711\,\mathrm{fb}^{-1}$. The results are shown in Fig. 2.9. The measured value of P_5' were in agreement with the previously found anomaly by LHCb. However, due to the small number of measured signal decays, the errors of the Belle measurement were quite large.

The most recent and precise measurement was done by CMS in 2024. Using an integrated luminosity of $140\,\mathrm{fb}^{-1}$ of proton-proton collisions of the 2015 to 2018 dataset at centre-of-mass energies of 13 TeV. Their study confirmed a previously found deviation between the predicted and measured P_5' variable. This result can be

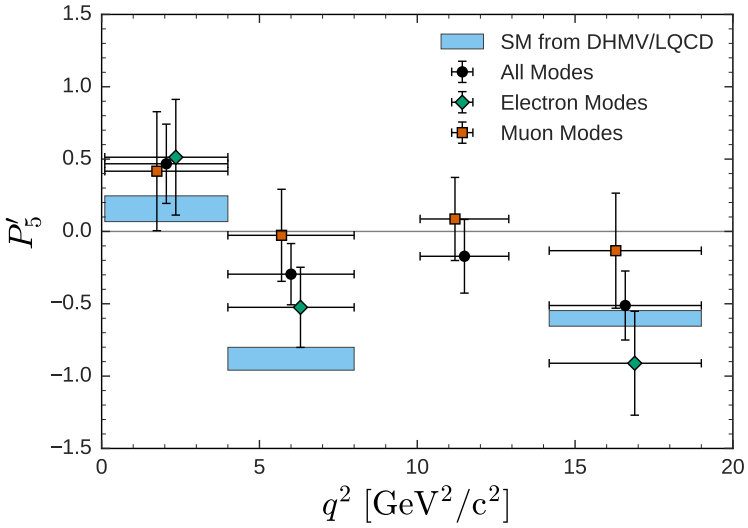


FIGURE 2.9: Measurement of the P_5' variable as a function of q^2 performed by Belle [20], compared to SM calculations from DHMV [21] and LQCD [22].

found in Fig. 2.10 [23]. A collection of the previous measurements done by ATLAS [24], LHCb [25], CMS [23] and Belle [26] can be found in Fig. 2.11.

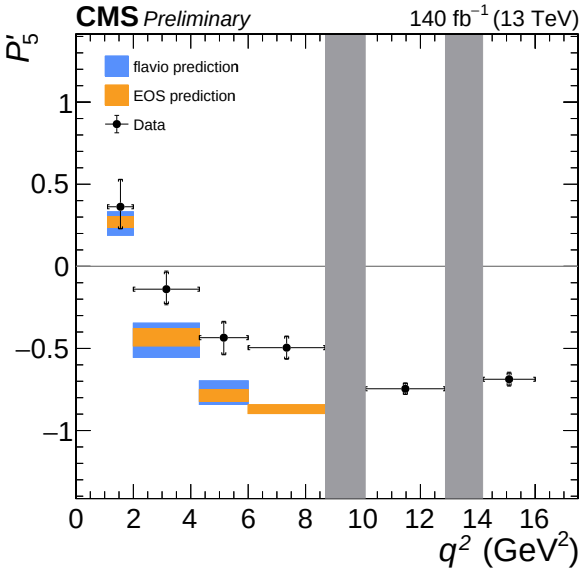


FIGURE 2.10: Most recent measurements of the P_5' variable done by CMS [23]. This plot also shows the flavio [27] and EOS [28] SM prediction.

2.4 Physics Beyond the Standard Model

Despite the great success of the SM at the electroweak scale, it is not a complete theory and therefore it is not able to explain all phenomena observed. As a result, it is considered to be an approximation of a more general theory at this scale.

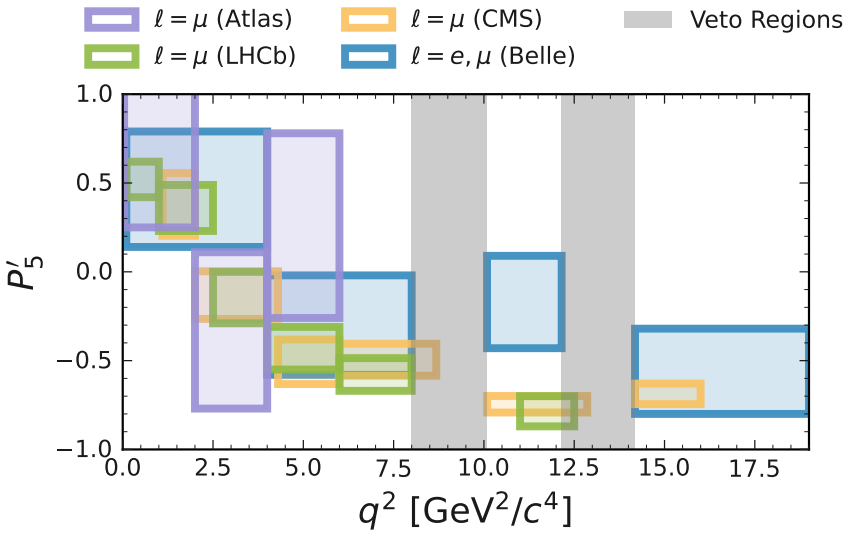


FIGURE 2.11: Most recent measurements of the P_5' variable. Results are from Atlas [24], LHCb [25], CMS [23] and Belle [26].

The SM consists of a total of 19 free parameters which are listed in Tab. 2.3. Those parameters are determined by experiment. In a global all-encompassing theory, it would be ideal to derive measurable constants directly from theoretical predictions.

TABLE 2.3: The 19 free parameters of the SM

Parameters	Amount
Quark Masses	6
Lepton Masses[5]	3
Mixing Angles of the CKM Matrix	3
$\mathcal{CP}$-Violating Phase	1
Gauge Coupling Constants	3
QCD Vacuum Angle	1
Higgs Vacuum Expectation Value	1
Higgs Mass	1

Furthermore, the largest discrepancies between SM predictions and observed values arise in cosmology. The most popular examples are *dark matter* and *dark energy*. Observations of galactic rotation curves have revealed that galaxies rotate faster than expected based on their visible matter content, suggesting that they are not in gravitational equilibrium. One possible explanation is that the masses of galaxies are far greater than the visible mass.[6] As a consequence, an unknown form of gravitationally acting matter was introduced called dark matter. This matter does not interact with the electromagnetic force and is therefore not visible to us. Observed effects of (for example) gravitational lensing in collisions of galaxies support the theory of dark matter.

[5] The neutrinos also must have masses since neutrino flavor oscillation can be observed.

[6] Another possible explanation is *Modified Newtonian Dynamics* (MOND). This theory modifies the way gravitation behaves on large scales.

Through the observations of supernovae as standard candles, the acceleration of the expansion of the universe can be determined and is found to be increasing.[7] Therefore, an additional form of energy is hypothesized, which could explain the accelerated expansion of the universe. Experiments show that more than 95% of the energy in the universe consists of this dark matter and dark energy [29]. The SM does not provide a possible particle candidate for these phenomena.

Another unexplained phenomenon is the imbalance between the amount of matter and anti-matter found in the universe. According to the SM, after the Big Bang, an equal amount of matter and anti-matter would have been produced. However, no significant amount of naturally occurring anti-matter has been found in the universe. Only a small fraction of this discrepancies can be explained by $\mathcal{CP}$ violation.

An unsatisfying aspect of the SM is the fine-tuning. This is more an aesthetic problem than a physical one, but it still leaves a lot of physicists displeased. The problem is that higher order loop contributions are divergent. As a result, according to theory, the Higgs boson has to have an infinite mass. However, the Higgs mass was determined to be relatively low by experiment. Within the SM, this mass can only be finite by fine-tuning the tree-level and loop-contributions.

Obviously, the SM is not able to explain all of these phenomena and BSM is needed. The effective approach, as presented in this thesis, is one of the more promising approaches. It is not necessary to know exactly how BSM contributions are affecting the processes as long as it is possible to determine the differences in the measured Wilson coefficients compared to the SM.

[7] A universe containing only gravitationally attracted matter would have a decreasing acceleration of its expansion.

Chapter 3

Experimental Setup at Belle II

> God help us, we're in the hands of engineers.
>
> Jeff Goldblum, *Jurassic Park (1993)*

SuperKEKB is a two-ring, asymmetric, electron-positron collider, located at KEK (*High Energy Accelerator Research Organization*) in Tsukuba, Japan. The electrons are accelerated to an energy of 7 GeV and the positrons to 4 GeV, respectively, hence the term asymmetric. The center-of-mass energy is about $\sqrt{s} = 10.58\,\text{GeV}$, which is close to the mass of the $\Upsilon(4S)(b\bar{b})$ resonance, which is above the B meson production threshold. This meson decays, almost exclusively, via two B mesons. For this reason SuperKEKB is also called B-factory. These B mesons decay further and their decay products can be detected by the Belle II detector in order to study the properties of the mother B meson. Since the electrons and positrons annihilate and only a $\Upsilon(4S)$ is created, the events detected by Belle II are very clean, meaning that there are only a handful of final state particles in each event. Also, since the inertial momenta and energies are known, it is possible to perform precise studies on decays with a neutrino in the final state.[1]

In this chapter, the experimental setup at the Belle II experiment will be presented. It will start with a brief introduction to the SuperKEKB accelerator. Then the detector will be described, starting with the innermost subdetector up to the outermost. Finally, there will also be a brief introduction to the simulated Monte Carlo and recorded datasets provided by Belle II.

3.1 SuperKEKB

The SuperKEKB accelerator is an upgraded version of the KEKB collider at the same facility. Between 1998 and 2010, KEKB was operating as a B-factory. During its time, KEKB achieved an instantaneous luminosity of $\mathcal{L} = 2.11 \times 10^{34}\,\text{cm}^{-2}\,\text{s}^{-1}$ which was a world record for its time. After more than ten years of operation, KEKB was discontinued, to be upgraded to SuperKEKB [30].

In Fig. 3.1 the schematic layout of the SuperKEKB accelerator is shown. The source of the electron beam is at the *low emittance gun*. Here, the electrons are emitted and are then accelerated in a linear particle accelerator. After the second linac stage, the electrons hit a positron production target, where the positrons are created.

[1] It is not possible to detect neutrinos with Belle II directly. However, knowing the initial momenta and energies and reconstructing all of the final state particles, the neutrino must carry the missing energy of the event.

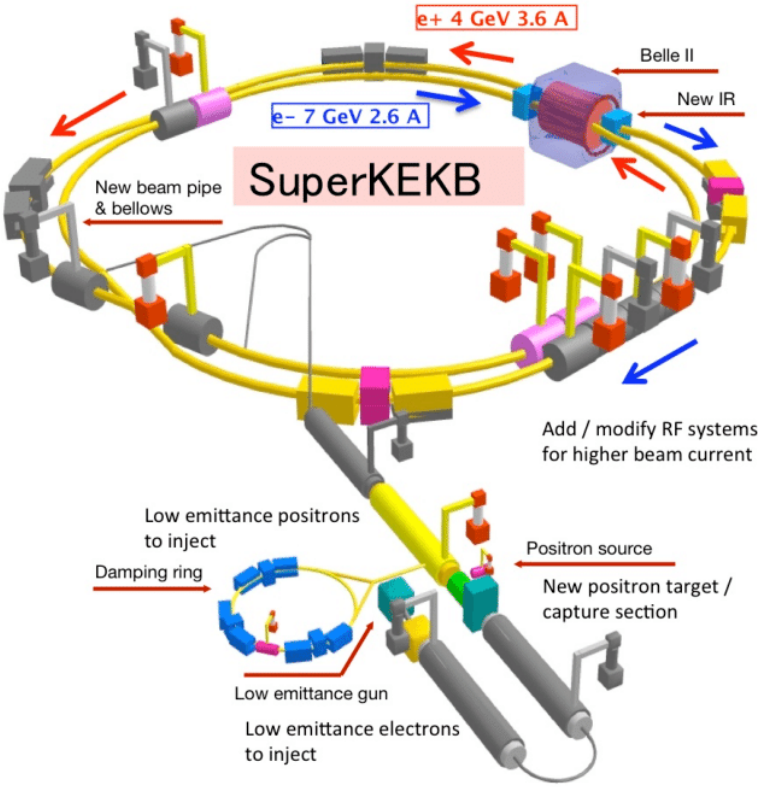


FIGURE 3.1: Schematic representation of the SuperKEKB collider [31].

Afterwards, there are more accelerating stages before the two beams are finally injected into their independent storage rings, *high-energy ring* for the electrons and *low-energy ring* for the positrons, respectively. Both of these rings have a circumference of about 3 km. The beams are brought to collision in the center of the Belle II detector. This point of collision is also called *interaction region* (IR) or *interaction point* (IP).

One of the main upgrades of SuperKEKB compared to KEKB is, that a smaller beam energy asymmetry is used. The advantage of this is that higher beam currents and better focusing magnets can be used which will ultimately lead to a luminosity 30 times higher than previously [32]. The difference between the beam energies also enables the study of $\mathcal{CP}$ violation in B mesons. This can only be done when the boost of these mesons is high and their position and tracks close to the IP is measured which opens the door to time dependent measurements. The Lorentz boost at SuperKEKB can be calculated by:

$$\beta\gamma = \frac{E_{\text{Her}} - E_{\text{LER}}}{\sqrt{s}} = \frac{7\,\text{GeV} - 4\,\text{GeV}}{10.58\,\text{GeV}} = 0.284 \tag{3.1}$$

This boost can be brought in relation with the distance the particle travels in the time frame Δt:

$$\Delta z = c\beta\gamma\Delta t \tag{3.2}$$

Using this relation, alongside a spatial point resolution of the Belle II detector in the order of $\sim 10\,\mu\text{m}$ will enable time dependent measurements.

Another goal of SuperKEKB is to reach an integrated luminosity of $\int \mathcal{L}\,\mathrm{d}t = 50\,\mathrm{ab}^{-1}$ by 2035. For the sake of clarification, the concept of luminosity will be described with more detail in the next section.

3.1.1 Luminosity

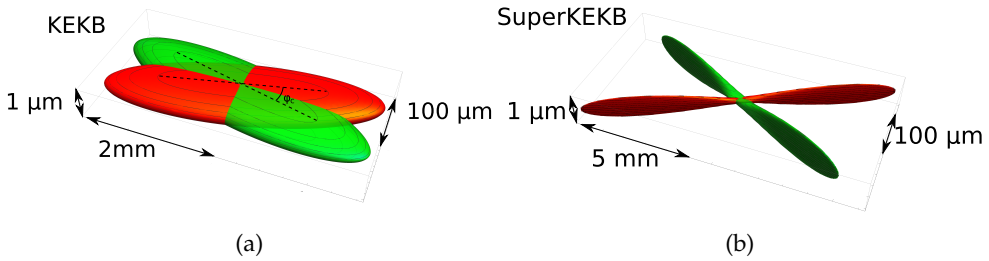
The instantaneous luminosity $\mathcal{L}$ is an indicator for the performance of a collider. Knowing $\mathcal{L}$ and the cross-section σ, it is possible to estimate the expected rate of events per seconds by calculating:

$$\frac{\mathrm{d}N}{\mathrm{d}t} = \mathcal{L}\sigma \tag{3.3}$$

This rate can only be increased for a given process by increasing the luminosity since the cross-section is a constant provided by nature. The instantaneous luminosity in a storage ring such as SuperKEKB can be calculated by:

$$\mathcal{L} = \frac{N_{e^-} N_{e^+} f_c}{4\pi\sigma_x\sigma_y} \cdot S \tag{3.4}$$

With σ_x and σ_y being the horizontal and vertical size of the beam assuming a Gaussian profile. N_{e^-} (N_{e^+}) is the number of electrons (positrons) in each bunch. Due to the way particles are accelerated in an accelerator, the particles do not form a continuous beam but are chopped in bunches. f_c is the average crossing rate with $f_c = n \cdot f_r$, where n is the number of bunches and f_r is the revolution frequency. S is a correction factor which takes geometrical effects linked to the finite cross-section and bunch length into account [33]. By increasing the number of bunches and the number of particles within each bunch, SuperKEKB is able to double the luminosity compared to KEKB. In addition to that, the size of the interaction region of SuperKEKB is reduced to one fifteenth the size of KEKB, resulting in a vertical beam size of just $\sigma_y \approx 80\,\mathrm{nm}$, which is depicted in Fig. 3.2b. As a consequence, the luminosity will be increased 30-fold [32] [34] [35]. The current status of the integrated luminosity at Belle II will be described in more detail in Sec. 3.11.



(a) (b)

FIGURE 3.2: The beam crossing of KEKB in (a) and SuperKEKB in (b) [36].

3.2 The Coordinate System at Belle II

For the sake of clarification, the coordinate system at Belle II will be discussed briefly. This coordinate system is sketched in Fig. 3.3. The z-axis is aligned with the direction of the magnetic field and will be referred to as the forward direction. The electrons and positrons are moving in opposite directions along the z-axis with a small angle

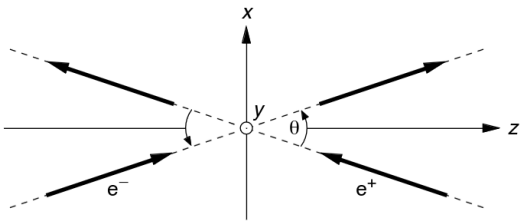


FIGURE 3.3: The coordinate system at Belle II. The y-axis is pointing towards the viewer. The angle between the electron and positron beam is $\theta = 4.75°$ [37].

of $^{4.75°}/_{2}$. The x-axis points to outside of the SuperKEKB storage rings and the y-axis to the top of the experiment.

3.3 The Belle II Detector

Belle II is an upgraded version of the Belle detector, which was a solid-angle magnetic spectrometer located at the interaction region of KEKB. A sketch of the Belle II detector is shown in Fig. 3.4. In the following sections, each sub-detector of Belle II will be presented and described briefly, starting from the innermost to the outermost sub-detector.

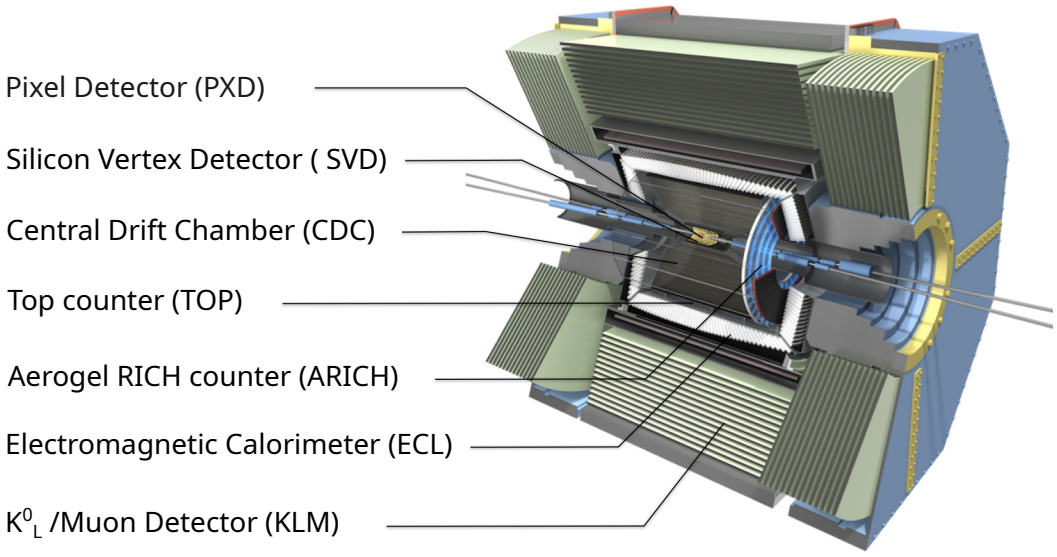


FIGURE 3.4: The Belle II detector with its subdetectors [38].

3.4 Vertex Detector

The vertex detectors (VXD), which consists of the pixel vertex detector (PXD) and the silicon vertex detector (SVD), are the innermost sub-detectors of Belle II. They are able to make precise measurements of the particle tracks close to the interaction region. This enables the reconstruction of decay vertices of long-lived particles by

determining the distance and spatial resolution of the first measured hit and the effect of multiple scattering. A sketch of the VXD can be found in Fig. 3.5.

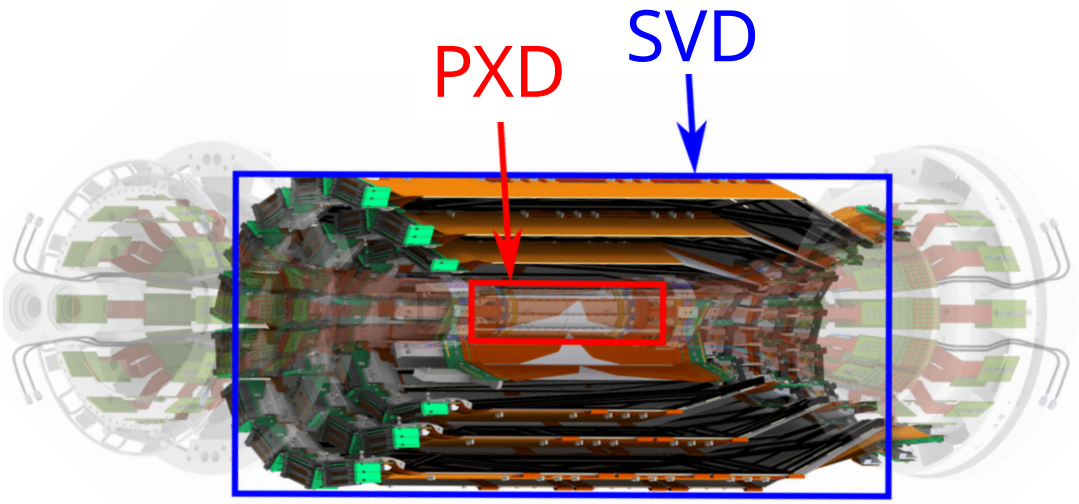


FIGURE 3.5: Sketch of the vertex detectors. The innermost sub-detector (PXD) is surrounded by the silicon-vertex detector (SVD). [39]

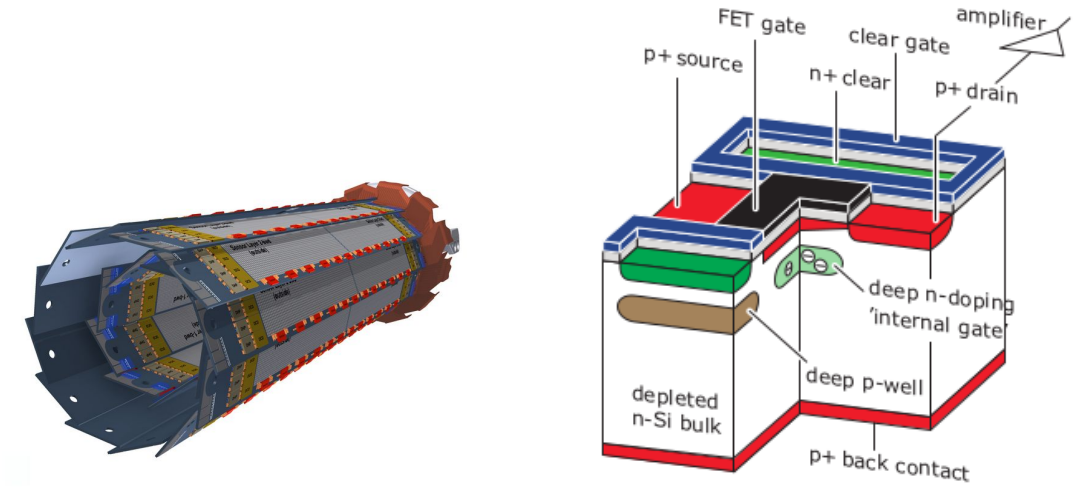
3.4.1 Pixel Vertex Detector

Since the PXD is a topic of discussion in the appendix (Ch. L), this sub-detector will be described with a bit more detail.

The main purpose of the PXD is to determine the spatial position of the decay vertices of B and D mesons and the τ lepton. To achieve this, the *Depleted P-channel Field-Effect Transistor* (DePFET) technology is used, which allows very thin ($\approx 50\,\mu m$) sensors. Additionally, due to the proximity of the PXD to the interaction region, it has to withstand a high radiation caused by the QED background [34] [40]. This high radiation can cause random bit status flips while taking data, so-called *Single Event Upsets* (SEUs). These random changes in a state, stop the PXD from working as expected. Therefore, a way has to be found to identify these SEUs and recover the bit status. Ch. L will provide more details about these glitches and how to detect them.

A sketch of the PXD with two complete layers can be found in Fig. 3.6a. The inner layer consists of eight planar sensors (ladders) at a radius of 14 mm. Each of these ladders has a width of 15 mm and an effective length of 90 mm. The outer layer, at a radius of 22 mm, is made of 12 ladders of the same kind. The single hit spatial resolution of the PXD is determined to be about 10 µm [34]. It is important to note, that before the first long shutdown of Belle II (2019 to 2022), only two of the 12 outer ladders were installed due to production issues.[2] During the first long shutdown in 2022 to 2024, a new and complete PXD (called PXD2) was installed in Belle II.

[2] A sketch of this installed PXD is depicted in Fig. B.1.



(a) (b)

FIGURE 3.6: (a) Sketch of the complete PXD detector [41]. (b) Illustration of the DePFET technology used in the PXD [42].

DePFET

DePFET is a concept for semiconductor detectors combining detection and amplification in a single device. It was developed in 1987 by J. Kemmer and G. Lutz of the MPI for Physics in Munich [34].

A cross-section of a pixel with this concept in shown in Fig. 3.6b. The main material of a DePFET cell is fully depleted silicon. Within this silicon substrate, depleted by a high negative voltage, a *p*-channel MOSFET (*metal oxide semiconductor field effect transistor*) or a JFET (*junction field effect transistor*) is integrated. Both field effect transistors act as a first pre-amplification.

Whenever an ionizing particle hits the detector, electron-hole pairs are created, which are separated by the potential field of the sidewards depletion. The holes propagate to the back contact and the electrons are accumulated in the potential minimum, called the *internal gate*. The DePFET concept has the advantage that the amplification of the signal happens right above the position of its generation. As a result, lateral charge transfer where losses occur are avoided. However, the most important feature is the low noise performance of the DePFET concept even at room temperature. This is a result of the very small capacitance of the internal gate.

At Belle II, these DePFET pixels are 75 µm thick and vary in size between $50 \times 50\,\mu m^2$ up to $50 \times 85\,\mu m^2$. Each module consists of 250×768 pixels and each ladder of the PXD is made out of two modules. A complete read out of all pixels takes 20 µs with a dead time of only 100 ns and the detector covers a polar angle of $17° < \theta < 150°$.

3.4.2 Silicon Vertex Detector

The SVD is a four layered, doubled-sided strip detector of six-inch wafers surrounding the PXD. Due to cost it is not possible to cover this volume of detector with the DePFET technology. A sketch of the SVD can be seen in Fig. 3.5. Similar to the PXD, it covers the same θ polar angle. Also, the fundamental working principle is similar. Whenever a particle travels through the sensors, it creates electron-hole pairs along its path by ionization. These pairs are separated and produce a signal, which can be measured. As a result, the path which the particle took, can be reconstructed [34].

3.5 Central Drift Chamber

The central Drift Chamber (CDC) surrounds the SVD. The volume of the CDC is filled with a 50 % helium and a 50 % ethane gas mixture. The detector itself consists of 14 366 sense wires arranged in 56 layers, alternating between a stereo and an axial arrangement as depicted in Fig. 3.7. In addition to these sense wires, there are 42 240 aluminum field wires, which create an electric field in order to separate electron-ion pairs that are created when a charged particles loses energy due to ionization while passing through the CDC. One task of the CDC is to reconstruct the momenta and tracks of charged particles [34]. The polar angle coverage of the CDC is the same as for the VXD ($17° < \theta < 150°$).

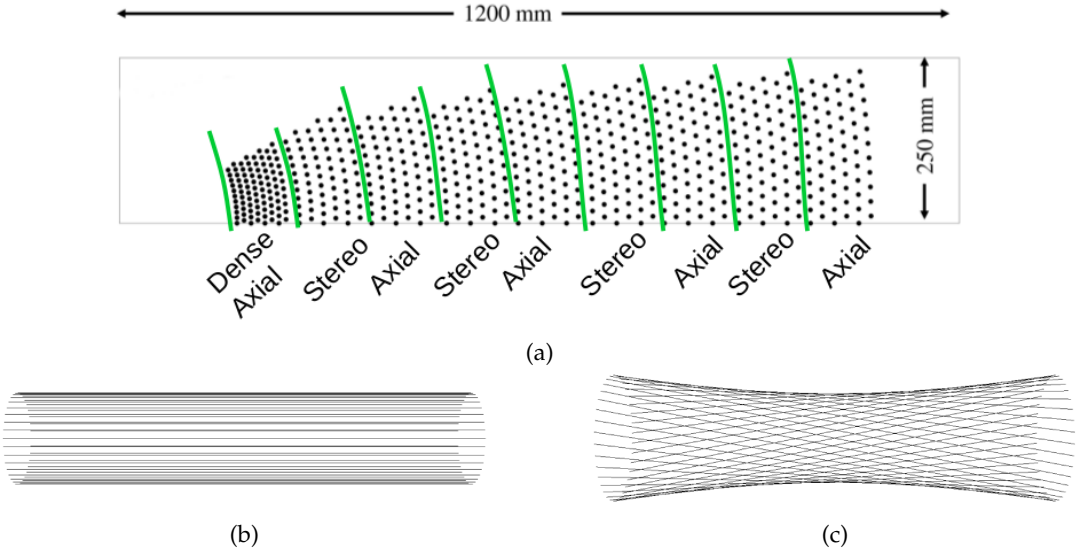


FIGURE 3.7: (a) Cross-section of a small fraction of the CDC. Each dot represents a wire. In addition, the separation of the different superlayers are indicated by the green lines. The axial and stereo arrangement is depicted in (b) and (c), respectively [43].

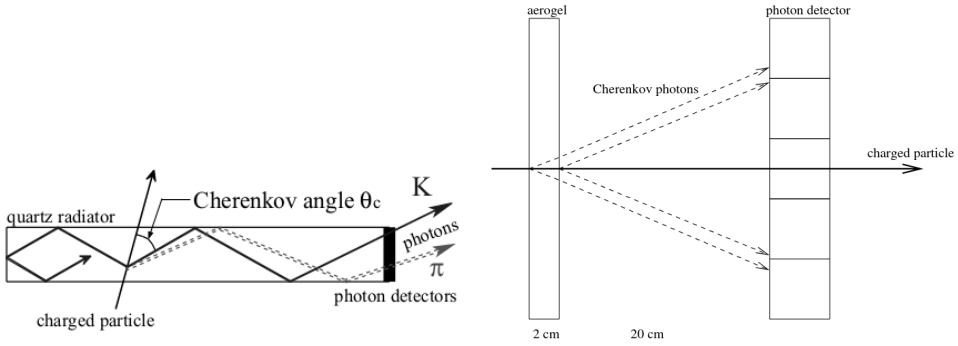
3.6 Particle Identification Detectors

In order to improve the particle identification, two additional detectors are installed, the Time-Of-Propagation (TOP) and the Aerogel Ring-Imaging Cherenkov detector (ARICH).

When the velocity of a charged particle β is higher than the speed of light within this medium $c_n = c_0/n$ with n being the refractive index, Cherenkov light is emitted in a cone around the direction of the particle momentum. The angle θ_C of this light to the particle momentum can be calculated by [44]:

$$\cos(\theta_C) = \frac{1}{n\beta} \tag{3.5}$$

The working principle of the TOP detector can be seen in Fig. 3.8a. The emitted Cherenkov light is reflected within the crystal until it reaches the photon detectors at the end. The photons are then focused by mirrors and are finally detected by



(a) (b)

FIGURE 3.8: (a) Operating mode of a TOP and (b) of the ARICH detector. Kaons and pions can be distinguished by measuring the Cherenkov angle [34].

Microchannel-Plate Photo Multiplier (MCP-PMTs). By measuring the time difference of these photons, the position of transit of the charged particle can be determined. Depending on the angle of the photons, different MCP-PMTs are hit and the original angle can be determined [34].

The ARICH on the other hand is a close proximity Ring-Imaging Cherenkov detector. The Cherenkov photons escape the first medium and are then detected by a photo detector which is placed 20 cm behind it. The Cherenkov angle of these photons can then be determined by reconstructing the ring created by them.

While the TOP is located in the barrel region, the ARICH is located in the forward end-cap region of the Belle II detector.

3.7 Electromagnetic Calorimeter

The main task of the Electromagnetic Calorimeter (ECL) is the detection of photons, and charged particles alongside their energy and angular coordinates with high efficiency. The detector itself can be divided in three regions. One barrel region and two end-cap regions, resulting in a total of $6624 + 2 \cdot 2112 = 10\,848$ Thallium doped Cesium iodide CsI(Tl) crystal elements, each with a length of about 30 cm corresponding to about 16.1 radiation lengths X_0 for photons and electrons. These crystals are all read out individually by photodiodes [34].

High energetic electrons (positrons), pions and photons produce electromagnetic showers in the scintillating crystals by creating bremsstrahlung and undergoing pair-production. The higher the energy of the incoming particle the larger the shower becomes which can spread across several crystals. Using algorithms, the measured energy of each crystal can be combined and the initial energy of the incoming particle can then be reconstructed. The ECL covers a polar angle of $12.4° < \theta < 155.1°$ [32]. Electrons (positrons), photons and pions are completely stopped by the ECL while muons and neutral kaons can pass it depositing only a fraction of their energy.

3.8 K_L^0 and Muon Detector

Since both the K_L^0 and the muon pass through the ECL without getting stopped, an additional detector has to surround the ECL. This K_L^0 and muon detector (KLM)

consists of an alternating sandwich structure of 4.7 cm thick iron plates and resistive plate chambers in between [32].

K_L^0 and muons create clusters in both ECL and the KLM. However, since muons are charged, they create tracks in the CDC and they can therefore be distinguished from the neutral K_L^0.

3.9 Particle Identification

Before an analysis can start, one must be able to reconstruct the particle type. However, it is not possible to calculate the rest-mass of particles by individual detectors. Instead, first individual hits in the detectors are combined to tracks. Then the momentum and the energy of the particle can be determined by precise measurements of the momentum and rest-mass under a particle hypothesis. Using information like the energy loss in the CDC ($\mathrm{d}E/\mathrm{d}x$), particle discrimination information from TOP and ARICH and cluster size and position in the ECL with corresponding KLM clusters, it is possible for the particle identification system (PID) to assign probabilities to each measured track to be an electron, muon, kaon and pion. In order to separate different particle hypotheses, likelihood ratios are introduced with:

$$\mathrm{PID}_{\alpha \text{ vs. } \beta} = \frac{\prod_i \mathcal{L}_\alpha^i}{\prod_i \mathcal{L}_\alpha^i + \prod_i \mathcal{L}_\beta^i} \tag{3.6}$$

with $\prod_i \mathcal{L}_\alpha^i$ being the likelihood for a particle kind α in detector component i.

3.10 Trigger System and Data Acquisition

In Tab. 3.1, the cross sections and trigger rates for some physical processes at a luminosity of $\mathcal{L} = 6 \times 10^{35}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ are shown.

TABLE 3.1: Cross-section and trigger rate for a luminosity of $\mathcal{L} = 6 \times 10^{35}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ at Belle II [34]. The numbers are adjusted to correspond to the current instantaneous luminosity goal of Belle II.

Physics Process	Cross-Section [nb]	Rate [Hz]
$\Upsilon(4S) \to B\bar{B}$	1.2	720
Hadron Production from Continuum	2.8	1650
$\mu^+\mu^-$	0.8	480
$\tau^+\tau^-$	0.8	480
Bhabha ($\theta_{\mathrm{lab}} \geq 17°$)	44	260[3]
$\gamma\gamma_{\mathrm{lab}} \geq 17°$	2.4	14[3]
2γ Process ($\theta_{\mathrm{lab}} \geq 17°$, $p_t \geq 0.1\,\mathrm{GeV}$)	≈ 80	≈ 11200
Total	≈ 130	≈ 15000

In order to handle these high rates, a multilayer hierarchy trigger system is needed. Each subdetector provides trigger information from the corresponding subdetector to the global decision logic (GDL). This global trigger then decides whether or not the event should be written out [34]. It fulfills the following conditions:

[3] Due to their distinct signatures it is possible to pre-scale these very high cross sections by a factor of 100

0. high efficiency for continuum and hadronic events from $\Upsilon(4S) \to B\bar{B}$
1. a peak average trigger rate of 30 kHz
2. a stable latency of 5 µs
3. a timing resolution of less then 10 ns
4. a minimum two-event separation of 200 ns
5. a robust and flexible trigger configuration

Once the GDL decided that the event should be stored, the data acquisition system (DAQ) reads out the detector signals. Starting with the front-end electronics, the DAQ passes the signals through multiple layers of data processing before writing it to the storage system [34].

3.11 Data on Tape

Belle II started data taking in 2019 with the goal to collect a total integrated luminosity of about 50 ab^{-1} by 2035. This schedule is depicted in Fig. 3.9. The majority of this integrated luminosity will be taken at a center-of-mass energy corresponding to the mass of the $\Upsilon(4S)$ [45].

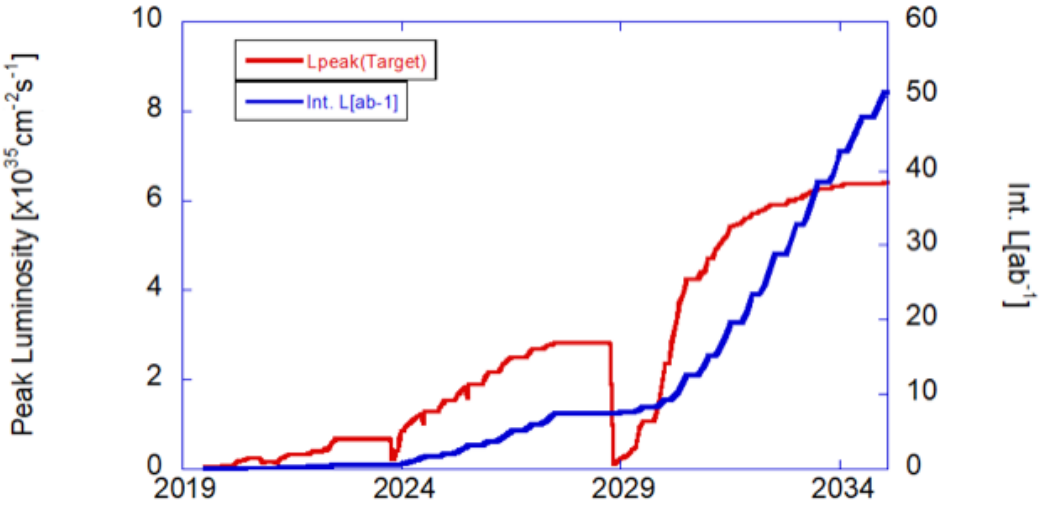


FIGURE 3.9: Goal of the integrated and peak luminosity of Belle II [45].

In this thesis, data collected up to July 2022 is analyzed. This data set corresponds to a total integrated luminosity of 426.86 fb^{-1} including 364.41 fb^{-1} of on-resonance data. More details about the collected data, also called Long Shutdown 1 (LS1) dataset[4, 5], can be seen in Tab. 3.2 and Fig. 3.10. For the analysis of this thesis only *good* runs are taken into account. As a result, the integrated luminosity of the dataset for this analysis is $\int \mathcal{L}_{\mathrm{LS1}}\, \mathrm{d}t = 364\,\mathrm{fb}^{-1}$.

[4]The first large data taking phase took place from 2019 to 2022, followed by a shutdown in order to upgrade and repair parts of the detector. Hence the name Long Shutdown 1 dataset.

[5]During 2024, Belle II started to continue collecting data. Fig. B.2 shows the most recent collected (integrated) luminosities.

TABLE 3.2: Integrated luminosities recorded by Belle II during LS1. These integrated luminosities do not include bad runs [46].

Center-of-Mass Energy	Integrated Luminosity [fb^{-1}]
$\Upsilon(4S)$	364.41 ± 1.64
$\Upsilon(4S)_{\text{Offres}}$	42.64 ± 0.20
$\Upsilon(4S)_{\text{Scan}}$	0.079 ± 0.001
$\Upsilon(5S)_{\text{Scan}}$	19.73 ± 0.12
Total	426.9 ± 1.7

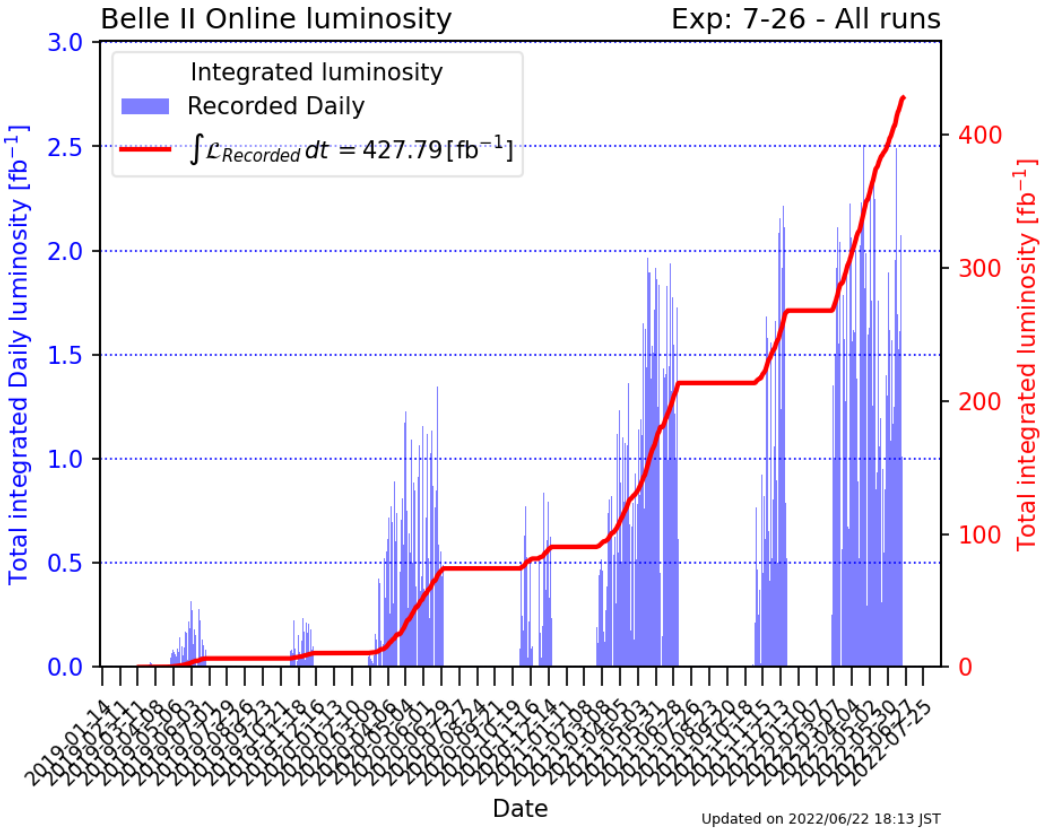


FIGURE 3.10: Total (integrated) luminosity at Belle II. This luminosity also includes all center-of-mass energies at Belle II and bad data runs which will not be used for analysis [45].

3.12 Monte Carlo Data

A large portion of this thesis, will be done using simulated data, also called Monte Carlo (MC) data. This MC data will help to test and establish analysis procedures in order to improve them with real data. The software packages to generate these MC data are EvtGen [47] and PYTHIA [48]. These software packages simulate the decay chain and determine the final state particles. After this, another package, called PHOTOS [49], calculates the final state radiation. Finally, the detector response is simulated by the software package GEANT4 (Geometry and Tracking) [50].

3.12.1 Generic Monte Carlo

In order to study the backgrounds, all possible $b \to q$ transitions are simulated with their corresponding branching ratio. Depending on the spectator quark of the B meson, there are two kinds of generic MC: charged ($e^+e^- \to B^+B^-$) and mixed ($e^+e^- \to B^0\bar{B}^0$) MC. In addition to these generic MC, generic continuum ($e^+e^- \to q\bar{q}$ with $q = u, d, s, c$) and tau-pair ($e^+e^- \to \tau\bar{\tau}$) MC are also provided.

There are two major subsets of these background MC samples, *run-independent* (*ri*) and *run-dependent* (*rd*) MC. *rd* MC has the exact detector setup simulation which was present when the data was recorded. This means, that if, for example, some subdetector was partially deactivated during a run then the same conditions were created during the simulation. Belle II provides four streams of independently created *rd* MC data sets. On the other hand, during the creation of *ri* MC, the exact configurations of the detector conditions are not taken into account. Resulting in slightly less realistic MC samples which are easier to generate. There are $1\,\text{ab}^{-1}$ of generic *ri* continuum and tau-pair MC and $3\,\text{ab}^{-1}$ of generic mixed and charged MC. The Monte Carlo iteration used in this thesis is MC15. More information can be found in Tab. 3.3.

3.12.2 Signal Monte Carlo

Since the $B \to K^*\ell\ell$ decays are very rare, it is crucial for the analysis that way more than the expected amount of these signal events are generated and provided for analysis. For that reason, Belle II provides for each of the possible signal processes $B \to K^*\ell\ell$ 10 million events, generated in the same manner as *ri* MC. A summary of all the available data used in this thesis is shown in Tab. 3.3.

TABLE 3.3: Signal and generic MC with the underlying process and the amount of available MC data. *4 Streams* corresponds to four times the total integrated luminosity on tape ($\int \mathcal{L}_{4\,\text{Streams}}\,\text{d}t = 4 \cdot 364\,\text{fb}^{-1}$).

	Name	Process	Amount *rd*	Amount *ri*
Signal	mixed	$B^0 \to K^{*0}\ell\ell$ ($\ell = e, \mu$)	/	10^7 events
	charged	$B^+ \to K^{*+}\ell\ell$ ($\ell = e, \mu$)	/	10^7 events
Generic	mixed	$e^+e^- \to \Upsilon(4S) \to B^0\bar{B}^0$	4 Streams	$3\,\text{ab}^{-1}$
	charged	$e^+e^- \to \Upsilon(4S) \to B^+B^-$	4 Streams	$3\,\text{ab}^{-1}$
	continuum	$e^+e^- \to u\bar{u}, d\bar{d}, s\bar{s}, c\bar{c}$	4 Streams	$1\,\text{ab}^{-1}$
	tau-pair	$e^+e^- \to \tau\bar{\tau}$	4 Streams	$1\,\text{ab}^{-1}$

3.13 Skims

As already mentioned, Belle II will take a lot of data. However, for an analyst it is not feasible to process every recorded event during the analysis since this would take a lot of time and consume a lot of resources. For this reason, the collaboration provides centrally produced preselected datasets in order to reduce their file size while keeping relevant events. These so-called skimmed datasets are provided for both data on tape and MC data [51]. This analysis will use a skim called 12160200 ($B \to X\ell\ell$ (no LFV)). More information about this skim can be found in Tab. 6.2 in Sec. 6.5.

Chapter 4

Analysis Tools

Before enlightenment:
Chop wood, carry water
After enlightenment:
Chop wood, carry water.

Zen Kōan

This chapter will start with a brief overview about the concept of probability. After this, the fundamentals of correlation and how to find them will be introduced. It will conclude with the description of some parameter estimation techniques that used throughout this thesis and a brief introduction to the metric which is chosen to analyze the neural network classifiers (see Ch. 5).

4.1 A Brief Introduction to Probability

Three axioms form the foundation of probability. Let $\mathcal{P}(E)$ be the probability of an event E and Ω be a sample space of all possible events.

The first axiom states that the probability is a non-negative real number.

$$\mathcal{P} \in \mathbb{R}, \mathcal{P}(E) \geq 0 \qquad \forall E \in \mathcal{P}(\Omega) \tag{4.1}$$

The second axiom states that from everything possible at least one thing will occur.

$$\mathcal{P}(\Omega) = 1 \tag{4.2}$$

This axiom is sometimes referred to as the unitarity of probabilistic theory.

The third axiom states that if the individual probabilities of $\mathcal{A}$ and $\mathcal{B}$ are independent of each other then the probability that either $\mathcal{A}$ or $\mathcal{B}$ will happen is the sum of both probabilities [52].

$$\mathcal{P}(\mathcal{A} \cup \mathcal{B}) = \mathcal{P}(\mathcal{A}) + \mathcal{P}(\mathcal{B}) \quad \text{if} \quad \mathcal{A} \cap \mathcal{B} = \emptyset \tag{4.3}$$

These three axioms describe how probabilities behave and should be handled. However, they do not describe how the probabilities are obtained in the first place. Basically, there are two complementary different approaches: *Bayesian* and *frequentist*.

Frequentist Approach

The definition of a probability according to the frequentists is that if there are n distinguishable events, while none of them is preferred over one the other, and an event

can be realized in each one of them, and there are k realizations with the attribute $\mathcal{A}$, then the probability for the appearance of $\mathcal{A}$ is:

$$\mathcal{P}(\mathcal{A}) = \frac{k}{n} \tag{4.4}$$

This definition of probability seems circular but often there are underlying symmetries which allow this kind of definition. In all other cases, the empirical definition of probability is chosen: If the attribute $\mathcal{A}$ is carried by k of n observations, then the probability for the presence of $\mathcal{A}$ is

$$\mathcal{P}(\mathcal{A}) = \lim_{n\to\infty} \frac{k}{n}. \tag{4.5}$$

Again, one could argue that it is impossible to reach the limes of infinity. However, this probability often converges fast to a specific value in practical situations. A major disadvantage of the frequentist approach is that for example it fails to answer simple questions like: *Will it rain tomorrow?* since there is only one tomorrow and the exact event can not be repeated.

Bayesian Approach

Another popular definition of probability is based on the Bayesian approach. The Bayesian definition of probability is the degree to which one believes that attribute $\mathcal{A}$ will occur in an observation.[1] The belief in a hypothesis is expressed by a prior probability, which is updated as new and relevant data become available. This Bayesian definition of probability approaches the Frequentists empirical definition in the limes of infinite number of observations. The advantage of the Bayesian approach is that it still can give a probability to occurrences even if there is no underlying symmetry and no way of repeating the event. Another powerful feature of the Bayesian approach is Bayes theorem:

$$\mathcal{P}(\mathcal{A}|\mathcal{B}) = \mathcal{P}(\mathcal{B}|\mathcal{A}) \cdot \frac{\mathcal{P}(\mathcal{A})}{\mathcal{P}(\mathcal{B})} \tag{4.6}$$

This equation gives the probability for the occurrence of $\mathcal{A}$ under the condition that $\mathcal{B}$ happened ($\mathcal{A}$ given $\mathcal{B}$).

To know how to handle probabilities, especially Bayes theorem, is very important in the realms of particle physics.[2] With its help, it is possible to determine the probability of observing a signal event in a given data set.

The probability $\mathcal{P}(\text{signal}|\text{data})$ can be determined with the help of simulated data. As described in Sec. 3.12, the mixture of signal and background is known in a Monte Carlo data set. The ultimate objective is to draw conclusions about the probability for a signal event considering the measured data $\mathcal{P}(\text{signal}|\text{data})$. This can easily be done using Bayes theorem and a correct prior probability.

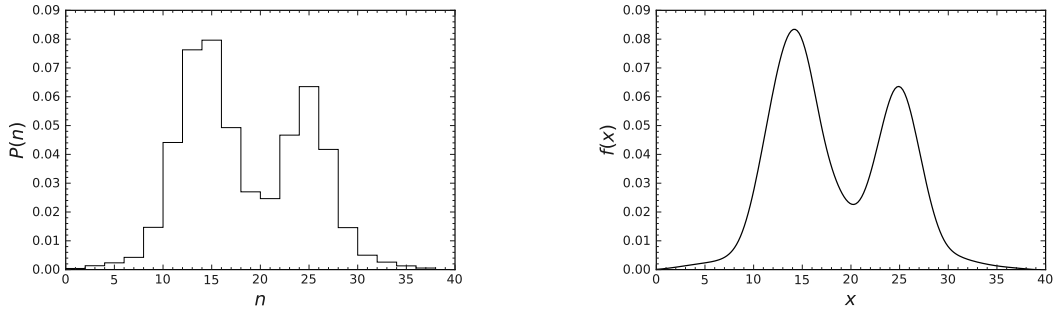
[1] A common analogy of the Bayesian method is how much of your money are you willing to bet on the occurrence of attribute $\mathcal{A}$ in an observation.

[2] Despite the success of the Bayesian approach, this statistical method to give probabilities, and therefore weight to evidence, is blocked in UK courts (*Regina v. T.*). It is argued that the subjectivity is often misunderstood and it is difficult to present Bayesian arguments in a way that everyone can understand them [53].

4.1.1 Probability Distribution

For a given discrete *random* variable n, each individual possible value of n is assigned with a probability.

$$P(n) = P(\{n\}) \tag{4.7}$$



(a) (b)

FIGURE 4.1: Example for a probability distribution of a discrete variable n (a) and a continuous variable x (b).

An example of a discrete probability distribution is shown in Fig. 4.1a. Given a continuous variable, Eq. 4.7 needs to be modified since the probability of each individual value is zero ($P(\{x\}) = 0$). In this case, a so-called *probability density function* (PDF) is used to properly describe the probability content of an interval with $x_1 < x < x_2$

$$\frac{\mathrm{d}P(x)}{\mathrm{d}x} = f(x) \qquad \text{with} \qquad P(x_1 < x < x_2) = \int_{x_1}^{x_2} f(x)\,\mathrm{d}x. \tag{4.8}$$

As a consequence, the PDF is a non-negative and normalized function.

$$f(x) \geq 0 \qquad \text{and} \qquad \int_{-\infty}^{\infty} f(x)\,\mathrm{d}x = 1 \tag{4.9}$$

In physics it is more common to encounter so-called *continuous variables* like the reconstructed mass or lifetime of a particle. An example for a distribution with a continuous variable x can be found in Fig. 4.1b.

The expected value or mean ($\mathbb{E}$) of a PDF is defined as:

$$\mathbb{E}[x] = \langle x \rangle = \int_{-\infty}^{\infty} x f(x)\,\mathrm{d}x \tag{4.10}$$

For a constant a, $\mathbb{E} = \langle ax \rangle = a \langle x \rangle$ is fulfilled. The variance of x is defined as:

$$\mathbb{V}[x] = \langle (x - \langle x \rangle)^2 \rangle = \langle x^2 \rangle - \langle x \rangle^2 = \mathbb{E}[(x - \langle x \rangle)^2] \tag{4.11}$$

The standard deviation of x is expressed by the square root of the variance.

$$\sigma_x = \sqrt{\mathbb{V}[x]} = \sqrt{\langle (x - \langle x \rangle)^2 \rangle} \tag{4.12}$$

Eqs. 4.8 to 4.12 can be modified to describe n random variables in m dimensions [54], resulting in the following PDF

$$\frac{\mathrm{d}^n P(x_1,\ldots,x_n)}{\mathrm{d}x_1,\ldots,\mathrm{d}x_n} = f(x_1,\ldots,x_n) \qquad \text{and} \qquad P(E) = \int_E f(x_1,\ldots,x_n)\mathrm{d}^n x \tag{4.13}$$

with the expected value

$$\langle x_i \rangle = \mathbb{E}[x_i] = \int_{-\infty}^{\infty} x_i f(x_1,\ldots,x_n)\,\mathrm{d}x_1\ldots\mathrm{d}x_n \tag{4.14}$$

and the variance

$$\mathbb{V}[x_i] = \int (x_i - \langle x \rangle)^2 f(x_1,\ldots,x_n)\,\mathrm{d}x_1\ldots\mathrm{d}x_n. \tag{4.15}$$

Furthermore, given two random variables x_i and x_j, the covariance between them can be calculated with

$$\begin{aligned}\mathbb{C}\mathrm{ov}[x_i,x_j] &= \langle (x_i - \langle x_i \rangle) \rangle \langle (x_j - \langle x_j \rangle) \rangle \\ &= \int (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) f(x_1,\ldots,x_n)\,\mathrm{d}x_1\ldots\mathrm{d}x_n.\end{aligned} \tag{4.16}$$

This covariance matrix will be important later in the analysis to estimate the error dependencies among the fit variables.

4.2 Correlations

An important aspect to keep in mind when fitting multidimensional PDFs are correlations since these fits will only deliver meaningful results when the underlying correlations of its variables are understood and treated correctly.

4.2.1 Linear Correlation / Pearson's Correlation Coefficient

The most common variant of a correlation is called linear correlation. It describes the correlation between two features of the same dataset. The coefficient r describing this correlation is called Pearson's correlation coefficient.

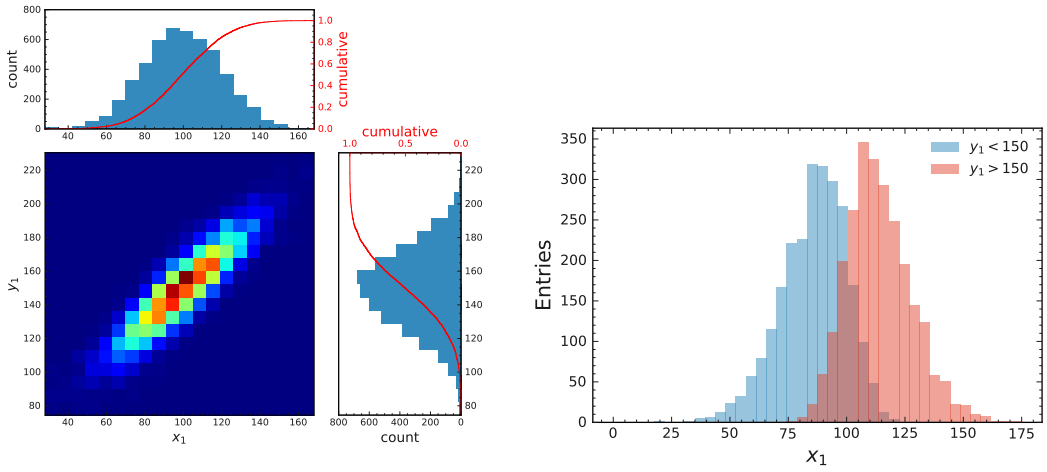
$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{4.17}$$

with

$$\bar{x} = \frac{1}{N}\sum_{j=1}^{N} x_j \qquad \text{and} \qquad \bar{y} = \frac{1}{N}\sum_{i=j}^{N} y_j \tag{4.18}$$

where x and y are the two features of the dataset with N entries and $\bar{x}$ and $\bar{y}$ are the arithmetic means. By construction, r can only take values within $[-1,1]$. This range indicates the correlation between both variables. $r = 1$ corresponds to a strong correlation, $r = -1$ to a strong anti-correlation and $r = 0$ to no correlation. An example for $r \approx 0.9$ can be found in Fig. 4.2a. This strong correlation is also visualized in Fig. 4.2b. Here, a cut on variable x_1 is applied, as a consequence, the shape of variable y_1 changes. Therefore, as expected, both variables are correlated. However, Pearson's correlation coefficient has its disadvantages since it is not able to find all non-linear correlations. This can be seen in Figs. 4.3a and 4.4a. Both examples have

a linear correlation coefficient of $r \approx 0$ indicating that there is no correlation between the variables x_2 and y_2 and x_3 and y_3, according to Pearson's coefficient. Despite that, Fig. 4.3b shows that the shape of x_2 depends on the values of y_2. As a consequence, these two variables are correlated. In Fig. 4.4b the shape of x_3 is independent of y_3. Therefore, both of these variables are truly uncorrelated. These examples show the weakness of the linear correlation coefficient. Especially for maximum likelihood fits (Sec. 4.3), it can be crucial to find and treat variables with correlations correctly. The method used to find these non-linear correlations will be presented with more detail in the next sections.



(a) (b)

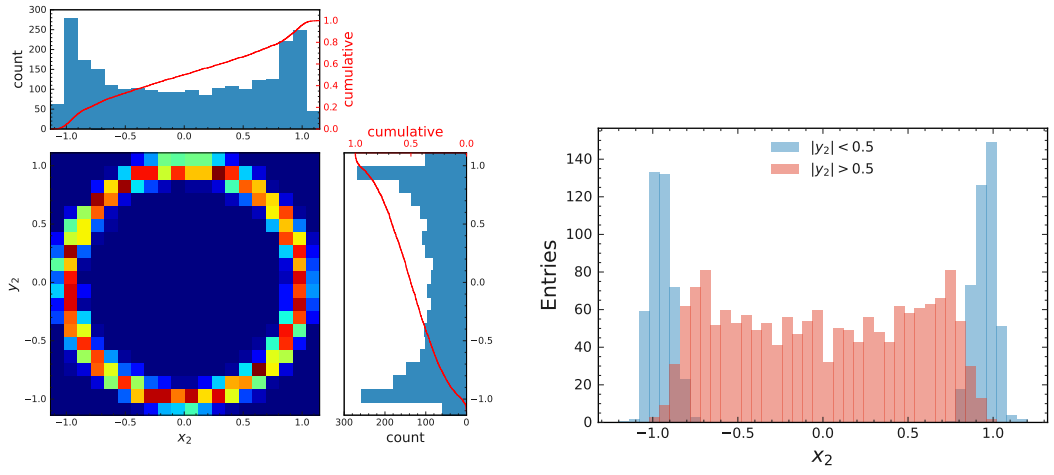
FIGURE 4.2: Distributions of correlations between two observables. (a) The linear correlation has been calculated to be $r \approx 0.9$. In (b) a cut on y_1 results in a different distribution in x_1. Both variables are correlated.

4.2.2 Flat Distributions

It can be advantageous to transform a non-discrete distribution x to a flat distribution x_f. This advantage is obvious if x has large tails and one wants to perform a histogram division. Due to statistical fluctuations, it is possible that some bins are left empty. Using a flat distribution, all bins are filled with the same number of entries and divisions by zero do not occur.

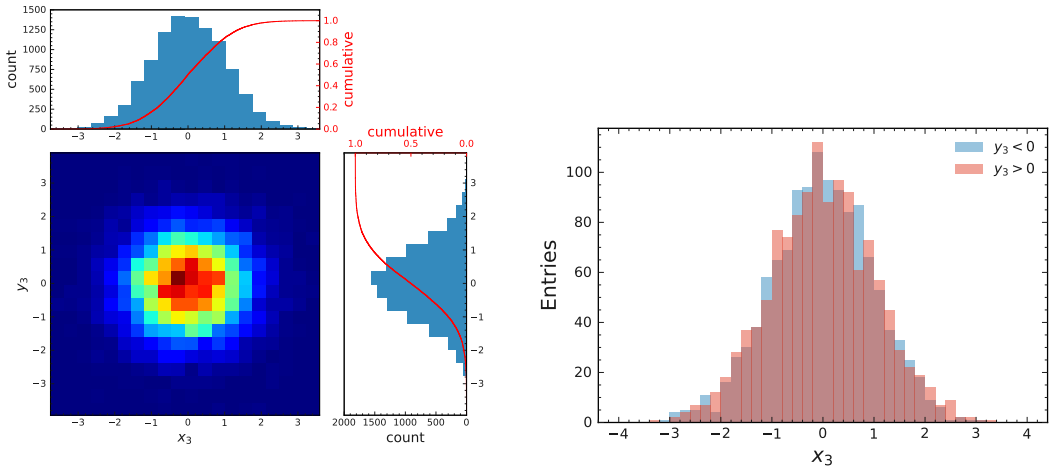
The easiest way to create a flat distribution is to change the binning in such a way that each bin contains the average of N/n_{bins} entries, with N being the size of the data set and n_{bins} the number of bins. The value of the bin edges can be chosen by calculating percentiles of x in order to prevent numerical instabilities.

Another way is to transform each value x_i of the dataset in such a way that the resulting distribution is flat ($x \to x_f$). This is done by transforming the initial distribution of the data set with its Cumulative Distribution Function (*CDF*). In the case of an unknown CDF, it can be approximated by creating a histogram h_{orig} with n_{bins} bins. Each bin i is filled with n_i^{orig} events. After this, the bin content of the new



(a) (b)

FIGURE 4.3: Distributions of correlations between two observables. (a) The linear correlation has been calculated to be $r \approx 0.0$ indicating that there is no correlation. However, a cut on y_2 results in a different distribution in x_2. This can be seen in (b). Both variables are correlated.



(a) (b)

FIGURE 4.4: Distributions of correlations between two observables. (a) The linear correlation has been calculated to be $r \approx 0.0$. In (b) a cut on y_3 does not change the distribution in x_3. Both variables are truly uncorrelated.

CDF histogram h_{CDF} can be calculated by successively adding up all the previous bin contents to each bin

$$n_i^{\text{CDF}} = \frac{1}{N} \sum_{j=1}^{j \leq i} x_j^{\text{orig}} \tag{4.19}$$

with $i = 1, 2, \ldots, n_{\text{bins}}$. In general, the normalization for h_{CDF} is $1/N$. However, it can be useful to chose a different normalization like $100/N$ in order to extract the

percentiles of x. The CDF can then be estimated by performing a spline fit (Sec. 4.3.4) on the values of h_{CDF}.[3] The flat distribution can be obtained by transforming $x^f = \text{CDF}(x)$. This procedure is demonstrated in Fig. 4.5.

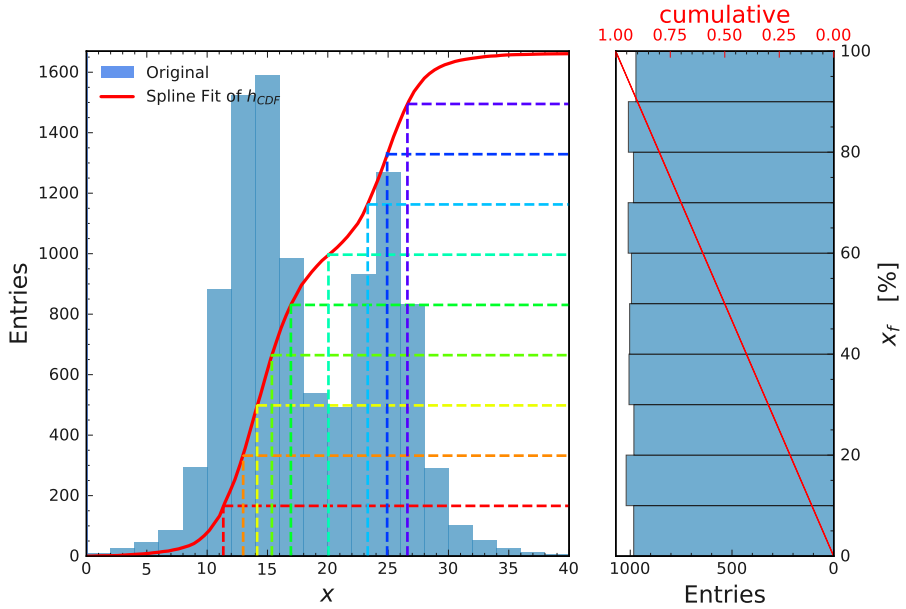


FIGURE 4.5: Flattening of a distribution with its CDF. On the left the original distribution can be seen. The red line is a spline fit performed over the CDF of the distribution. About 10% of this distribution is in the interval between zero and the vertical red line. Another $\approx$ 10% are between the vertical red and the vertical orange line and so on. The resulting flattened distribution is shown on the right.

4.2.3 Finding Correlations

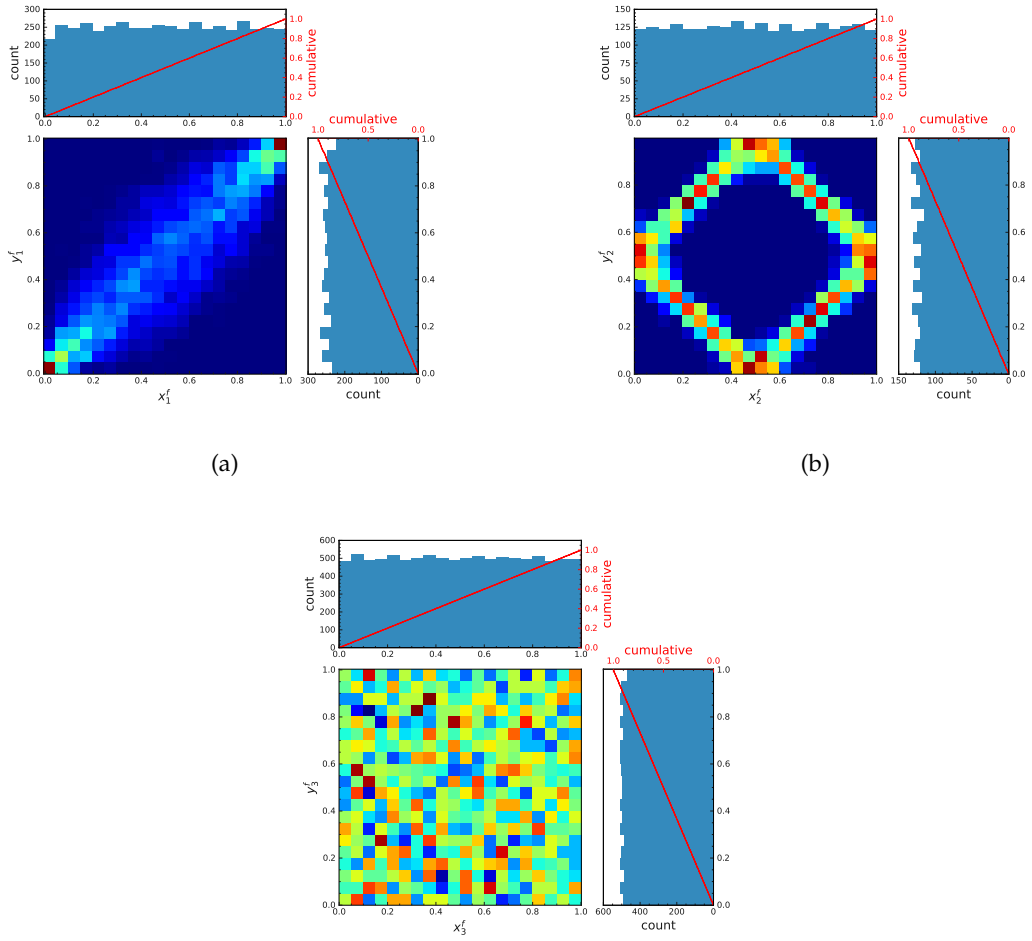
An important aspect is to find correlations between two variables x and y in a dataset with N entries. This can be done by transforming both variables to their corresponding flat distribution x^f and y^f. To study the correlation between both of these variables, a $n \times n$ histogram H with equal bin size can be constructed. In the case of an uncorrelated dataset, the scattered distribution is expected to be uniform across the two dimensional plane. Therefore, one expects that each bin of the histogram is filled with $n_{\text{exp}} = {}^{N}/_{n^2}$. If the condition $n_{\text{exp}} \gtrsim 25$ is fulfilled, the statistical error of each bin can be estimated with the Poisson error $\sigma_{n_{\text{exp}}} = \sqrt{N/n^2}$. Using this average, a χ^2 test-statistic can be calculated:

$$\chi^2 = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{(n_{ij} - n_{\text{exp}})^2}{\sigma^2_{n_{\text{exp}}}} \tag{4.20}$$

with n_{ij} being the bin content of the histogram $H(i,j)$. A χ^2 distribution with $n^2 - (2n-1)$ degrees of freedom [4] describes the probability p that the two distributions

[3] The spline fits for the examples above can be found in the appendix in Figs. C.1 (Line), Fig. C.2 (Circle) and Fig. C.3 (2D Gaussian).

[4] One degree of freedom is removed due to the constraints for the number of events in each row/column during the flat transformation.



(a) (b)

(c)

FIGURE 4.6: Flattened distributions for the variables (x_1, y_1) ((a) *line*; Fig. 4.2a), (x_2, y_2) ((b) *circle*; Fig. 4.3a) and (x_3, y_3) ((c) *2D Gaussian*; Fig. 4.4a).

are uncorrelated [55]. This procedure is shown for three examples in Figs. 4.6a (*line*), 4.6b (*circle*) and 4.6c (two dimensional Gaussian). The resulting flat distribution can be seen in Fig. 4.7.

The expected yield n_{exp} in these plots is indicated by the green color. The color axis is in units of the standard deviation σ with respect to the statistical error $\sigma_{n_{\text{exp}}}$. Bins colored in red contain more events then one would expect if both variables are uncorrelated. The opposite is true for the blue bins. Here, less entries then expected are filled in the bins. Using this method, it is possible to decide if both variables are correlated or not. This technique will be used in multiple occasions throughout the thesis.

4.3 Parameter Estimation

In physics, the true values of parameters of a physics problem are unknown. As a consequence, the estimated value $\hat{a}$ of a set of parameters may be different by statistical and systematical errors compared to the true values a_0.

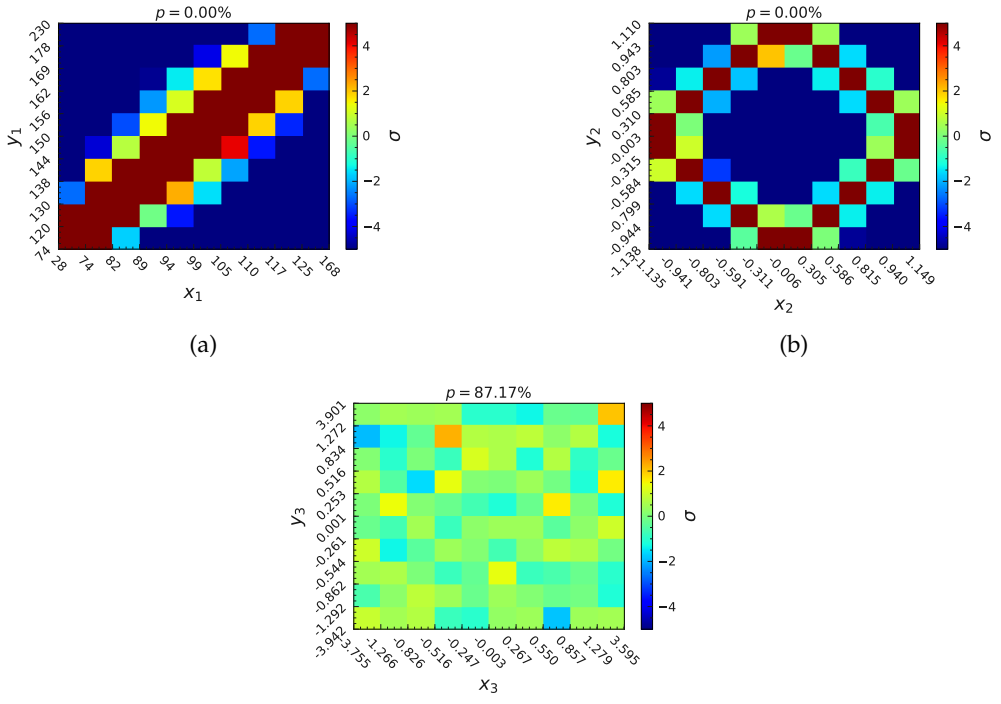


FIGURE 4.7: Flat correlation distributions for the three examples with the deviation from a flat expectation expressed in standard deviations. (a) shows a linear correlation. In (b) a lot of entries with $|\sigma| > 5$ can be observed. Indicating a strong correlation between both variables. In addition a structure can be seen. Furthermore, both, (a) and (b), have a probability p of about 0. In (c) no entry has a high deviation from the expected value. In addition to that, no structure can be seen and p is quite high. As a result, x_3 and y_3 are considered uncorrelated.

Oftentimes, the measured data can be considered as a set of random variables $\vec{x}$. These variables follow a probability density function $f(\vec{x}|a_0)$. The goal of an estimator process is to determine the underlying parameters $\hat{a}$ and their errors. This estimator process has to fulfill several criteria [56]:

Consistency: $\lim_{N\to\infty} \hat{a} = a_0$

Unbiasedness: $E[\hat{a}] = a_0$

Efficiency: The variance of $\hat{a}$ must be as small as possible

Robustness: There must be no effect on $\hat{a}$ by wrong data or hypotheses

4.3.1 Maximum Likelihood Method

An example for this estimator process is the measurement and determination of the mean value of the lifetime τ of an atomic state. In general, the method of choice is the so-called *maximum likelihood method*. In the context of a dataset following a normal distribution, this method corresponds to the χ^2-test. In a broader sense, the goal of the maximum likelihood method is to select a set of parameters $\theta = \theta_1, \ldots, \theta_n$

for a sample of random variables $\mathbf{x} = \vec{x}_1, \ldots, \vec{x}_n$ following a function $f(\vec{x}_i|\theta)$. This function is chosen to be a probability density function $\mathcal{P}(\vec{x}_i|\theta)$ according to Sec. 4.1.1. For all possible parameters of θ it is positive and normalized within their boundaries Ω.

$$\int_\Omega \mathcal{P}(\vec{x}|\theta)\,\mathrm{d}\vec{x} \equiv 1 \qquad \forall\theta \tag{4.21}$$

$$\mathcal{P}(\vec{x}_i|\theta) \geq 0 \tag{4.22}$$

The likelihood function $L(\vec{x}|\theta)$ of an ensemble of N independent data points is the product of each data point's probability.

$$L(\vec{x}_1, \ldots, \vec{x}_n|\theta) = \prod_{i=1}^{n} f(\vec{x}_i|\theta) \tag{4.23}$$

The estimated parameters $\hat{\theta}$ are then chosen in such a way that the maximum likelihood method maximizes $L(\vec{x}|\theta)$:

$$L(\vec{x}_1, \ldots, \vec{x}_n|\hat{\theta}) \geq L(\vec{x}_1, \ldots, \vec{x}_n|\theta) \qquad \forall\theta \tag{4.24}$$

As a convention and for numerical stability in computational algorithms, the logarithm of $L(\theta)$ is calculated and its result is multiplied by -1. This function is then minimized. In addition, it is possible that the data contains several categories, each having an individual number of expected events and their own PDF. As a result, the parameters are then obtained by extending the maximum likelihood fit:

$$\mathcal{L}(\theta) = -\ln L(\theta) = \sum_{j=1}^{n}\left\{\sum_{i=1}^{\mathcal{C}} N_i \mathcal{P}_i(\vec{x}_j|\theta)\right\} - \sum_{i=1}^{\mathcal{C}} N_i \tag{4.25}$$

Here, $\mathcal{C}$ is the number of categories, $\mathcal{P}_i$ is the PDF of the ith category and N_i is the number of signal events within the category.

4.3.2 zFit

A maximum likelihood fit will be used within this thesis to determine the observables from the angular distribution of the $B \to K^*\ell\ell$ decay. For this, the python library called *zFit* will be used. This module is able to perform unbinned maximum likelihood fits[5], create plots and generate data following a provided PDF (*toy Monte Carlo* samples). It was primarily designed as a particle physics data analysis toolkit [57].

4.3.3 sWeights

Later on in this analysis, it will be necessary to extract the true signal shape from a dataset containing both, signal and non-signal. This can be done with a statistical technique called ${}_s\mathcal{P}lot$. In a naive approach, after performing a fit to determine the yields N_i for all contributing PDFs f_i on the discriminating variable y, one can define the weight

$$\mathcal{P}_n(y_e) = \frac{N_n f_n(y_e)}{\sum_{k=1}^{N_s} N_k f_k(y_e)} \tag{4.26}$$

[5] It is important to be an unbinned fit since the binning of an histogram has a strong impact on the resulting observables. An unbinned fit circumvents this problem.

for all events. With $f_i(y_e)$ being the value taken by the PDFs f_i for event e. These PDFs are associated with a set of values y_e for the set of discriminating variables. Eq. 4.26 can then be used to build a histogram with the distribution $\tilde{M}_n$ of the target variable x by:

$$N_n \tilde{M}_n(\bar{x})\delta x \equiv \sum_{e \subset \delta x} \mathcal{P}_n(y_e) \tag{4.27}$$

Here, the right side of the equation sums over $N_{\delta x}$ events with x_e being the value taken by the variables x for event e which is in the x-bin centered around $\bar{x}$ that has a total width of δx [58]. However, this naive approach has the disadvantage that the PDFs of x enter implicitly in the definition of the weight. This can be avoided by extending the method which is done by the ${}_s\mathcal{P}lot$ *technique*. Here the sWeights are defined as:

$${}_s\mathcal{P}_n(y_e) = \frac{\sum_{j=1}^{N_s} V_{nj} f_j(y_e)}{\sum_{k=1}^{N_s} N_k f_k(y_e)} \tag{4.28}$$

with the inverse of the covariance matrix V_{nj} being defined as:

$$V_{nj}^{-1} = \sum_{e=1}^{N} \frac{f_n(y_e) f_j(y_e)}{(\sum_{k=1}^{N_s} N_k f_k(y_e))^2} \tag{4.29}$$

Using Eq. 4.28 it is now possible to obtain a ${}_s\mathcal{P}lot$ histogram with:

$$N_n \, {}_s\tilde{M}_n(\bar{x})\delta x \equiv \sum_{e \subset \delta x} {}_s\mathcal{P}_n(y_e) \tag{4.30}$$

This histogram reproduces on average the true distribution of the target variable x [58]. The technique requires the discriminating and target variables to be uncorrelated. Otherwise, Eq. 4.30 is no longer valid and a more sophisticated approach using a Monte Carlo simulation is needed to obtain the expected distribution.

An example for sWeights can be found in Fig. 4.8. Here, the uncorrelated variables M_{bc} and ΔE are used. A signal/background fit on the discriminating variable M_{bc} (Fig. 4.8a) was performed to obtain the PDF model and yield. With this model it is possible to compute the signal shape in the uncorrelated target variable ΔE without determining the background contribution in this variable. The obtained ΔE shape can be seen in Fig. 4.8d. The errors for the bin in this figure are calculated by:

$$\sigma = \sqrt{\sum_{i=1}^{n} w_i^2} \tag{4.31}$$

where w_i is the signal component sWeight of the ith entry and n is the total number of entries in the bin.

Unfortunately, the sWeight method has one disadvantage. It is not possible to calculate the sWeights on an already weighted dataset. This is not a problem, as long as the weights are uncorrelated to the target variables. In this case, the weights can be applied after the fit and the sWeights can then calculated.

This technique is already implemented in the python library sPlot [59] which will be used in the analysis in order to obtain the true shapes of the kinematic variables describing the decay. It has the big advantage that the background contribution has no longer be determined by simulation. It will be referenced to in this thesis as sWeights.

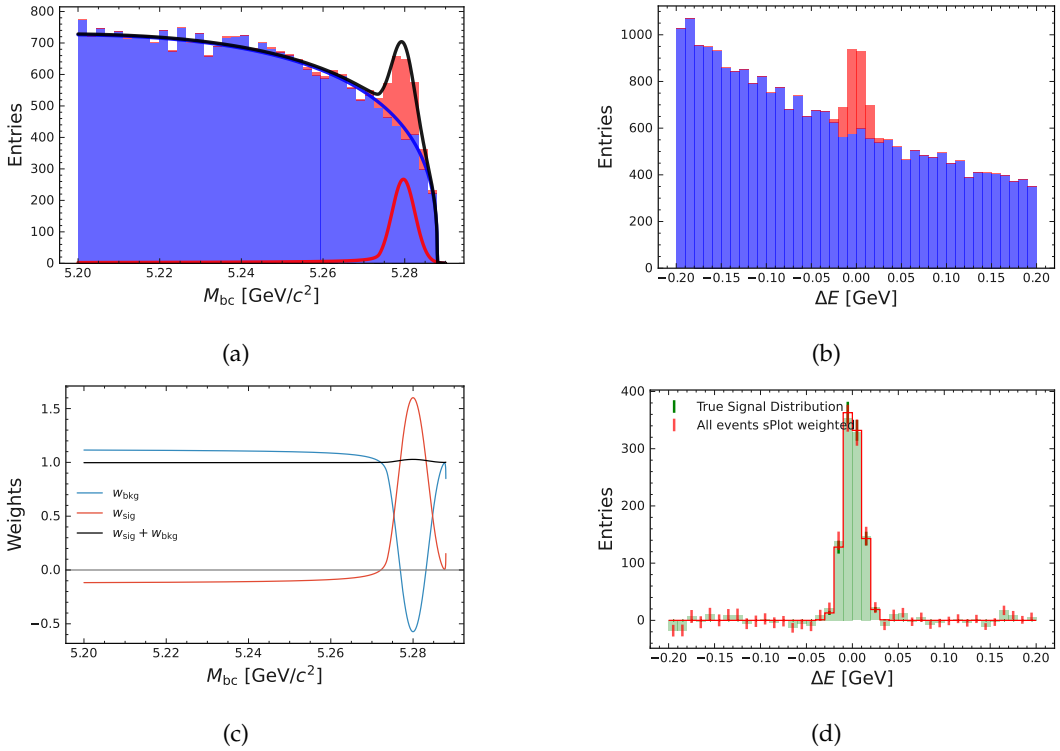


FIGURE 4.8: Example for sWeights method. (a) and (b) show the background and signal events in the variables M_{bc} and ΔE, respectively. The PDF model obtained from (a) is used to calculate the sWeights shown in (c). Here, the sWeights components for signal and background weights and their sum are shown as a function of M_{bc}. Using the signal weights component it is possible to obtain the true generated signal shape of ΔE shown in (d). The dataset was created with a zfit PDF model.

4.3.4 B-Spline

The B-spline (basis spline) interpolation is a numerical technique of piece-wise polynomial interpolation. The idea is to fit small degree polynomials to a small subset of the available values instead of fitting a high degree polynomial to all the values. These lower degree polynomials are combined at equidistant so-called knots. The adjustable parameters are therefore the amount of knots and the degree of the polynomials [60].

In Fig. 4.9, examples for spline fits are shown. The data in this example are created by adding random noise to the values of $f(x) = \sin(x)$ (black points). The purple line represents a line created by a B-spline fit with a smoothness $s = 0$. In contrast, the red line is the result of a B-spline fit with a smoothness equal to the number of available data points. As already mentioned, the technique of B-spline-interpolation will be used throughout this thesis. It will be referred to as *spline* fit.

4.4 Analyzing Classifiers / Metrics

One of the most difficult tasks is to make decisions. This is also true in physics. Analysts have to decide which events are discarded and which will be used for further

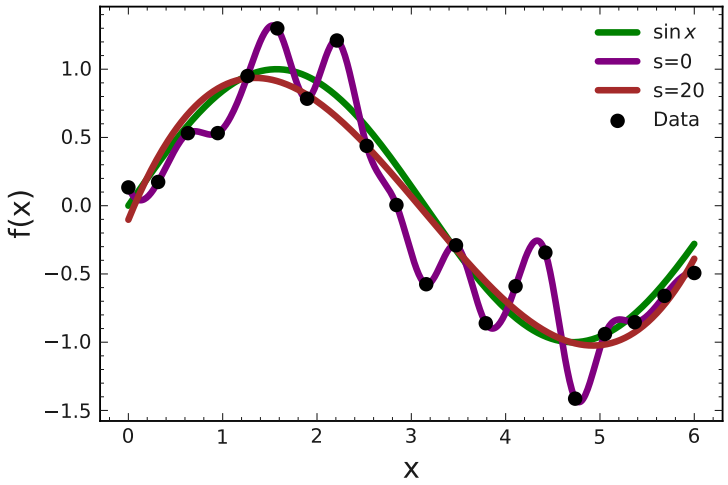


FIGURE 4.9: Example for spline interpolation with different amount of smoothing s. The data points are created by adding random noise to the values of the function $f(x) = \sin x$.

analysis. Some ways which help to make these decisions will be introduced briefly in the following sections.

4.4.1 Receiver Operating Characteristic

The *receiver operating characteristic* (*ROC*) is a method to rate and optimize analysis strategies. The general idea is to choose a feature describing a property of the events and a specific value (also called *threshold*). Everything on, for example, the right side[6] of this threshold will be considered signal and everything on the other side non-signal. This separation introduces four possible combinations which are listed in the so-called *confusion matrix* in Tab. 4.1.

TABLE 4.1: Confusion matrix for classifiers.

		True	
		1	0
Predicted	1	True Positive	False Positive
	0	False Negative	True Negative

True signal events labeled as signal are considered as *True Positive* (*TP*). When these true signal events are labeled as non-signal then this is the *False Negatives* (*FN*) case. The non-signal events on the other hand can be label as non-signal (*True Negative; TN*) and signal (*False Positive; FP*) events, respectively.

The ROC curve itself shows the efficiency (ϵ) vs. *1-background rejection* ($1 - \eta$). The points on the curve are calculated by:

$$\epsilon = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{4.32}$$

$$\eta = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}} \tag{4.33}$$

[6] Of course this depends on the analyzed variable. It can also be the other way round

Using these equations, it becomes obvious that the *best* case is reaching the point (1,1) in the ROC. This is only possible if there are no *False Negative* and *False Positive* events. However, this is often not possible since most of the time there is no feature which separates signal from non-signal events perfectly. Therefore, first off, a way has to be found to determine the best classifying feature, then the best threshold value of a feature dividing signal from non-signal has to be found. Both of these tasks can be solved using information of the ROC curve. The features can be ranked by calculating the area under the curve (*AUC*) itself. As a rule of thumb, the larger the AUC-value the better the feature works as a classifier. Finally, the threshold will be determined by calculating the point on the ROC curve with the smallest distance to the point (1,1). Of course, this is not the only way one can determine the threshold as it depends heavily on the goal of the analysis. If a more pure sample is preferred over a sample with a high efficiency, the threshold value should be adjusted accordingly.

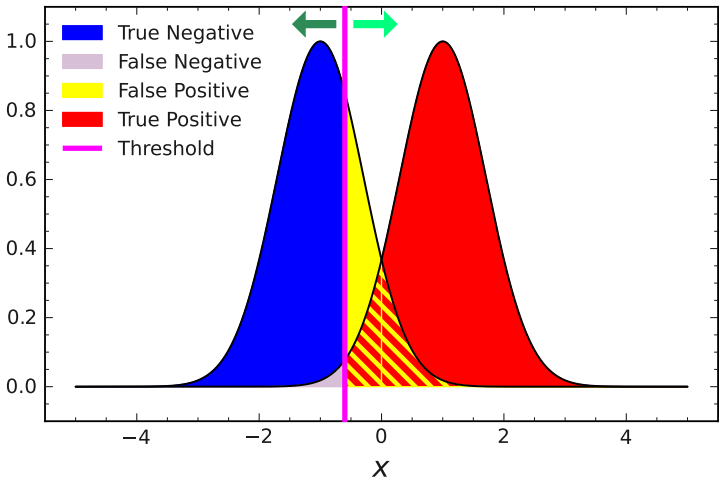


FIGURE 4.10: Example for ROC input is given. Once a threshold is chosen everything on the right side of it will be considered signal. Everything else would then not be included in further analysis.

An example for the threshold (*pink line*) and its effect on the input data is depicted in Fig. 4.10. Here, signal and non-signal events are shown. A value for the threshold is chosen and the four possible combinations mentioned in the confusion matrix are colored in.

The corresponding ROC curve is shown in Fig. 4.11. Since the threshold does not go through the point of the ROC with the closest distance (red point) to (1,1) (green point), the value for the threshold is not optimized [61] [62].

Imbalanced Data

Another aspect one needs to consider is the balance between the number of signal and non-signal events. Depending on the ratio a corresponding metric (Sec. 4.4) needs to be chosen. In general, if the ratio is one or higher, the ROC curve is probably the best classifier since both ϵ and η are independent from each other, meaning that adding more signal events won't change the value of η.

If the ratio is smaller than one (there are more non-signal than signal events) then another metric called *precision-recall curve* (PRC) is probably the better choice. *Recall* corresponds to the efficiency ϵ and *precision* ξ (also called *purity*) is defined by:

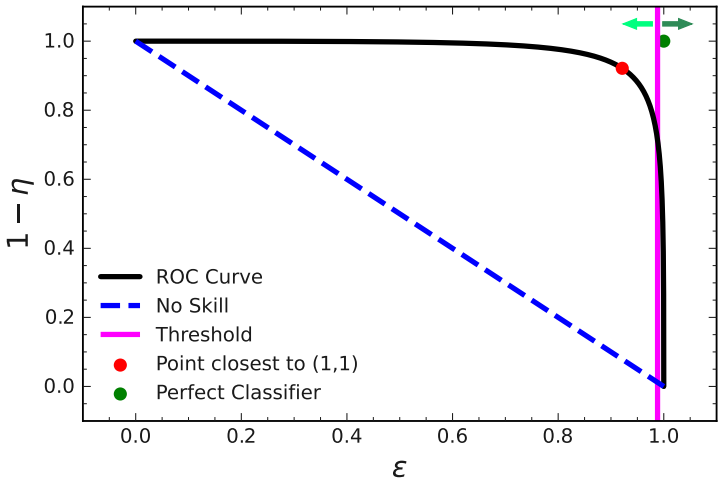


FIGURE 4.11: ROC curve for the example in Fig. 4.10. The pink line represents the threshold chosen in the former plot. Obviously, this threshold is not the best one since there are points on the ROC curve which are closer (red point) to the point (1,1) (green point). In this case, the best threshold is at $x = 0.0$. The green arrows show the effect of changing the threshold. The dashed line represents classifier with no separation power. A variable with this kind of ROC curve is not suitable to separate signal from non-signal.

$$\xi = \frac{TP}{TP + FP} \tag{4.34}$$

Precision is more focused on the positive class since it is not affected by large numbers of negative samples. Similar to the ROC, the best feature and threshold can be chosen by calculating the area under the curve and picking the one with the highest value [63].

4.4.2 Figure of Merit

Another way to determine the best threshold is the so-called *figure of merit* (*FOM*) which can be calculated by:

$$\text{FOM} = \frac{TP}{\sqrt{TP + FP}} \tag{4.35}$$

A FOM curve can then be calculated just like the ROC curve. Features can be ranked by calculating the maximum value of the FOM curve and the best threshold is then determined by the position of this maximum in each feature. The advantage of the FOM curve is that one can easily see the threshold in the FOM plot.

An example for a FOM plot is given in Fig. 4.12. This example uses the same data and threshold as Fig. 4.10. In the FOM example the best threshold at $x = 0$ can be read from the plot directly.

However, the FOM technique has a disadvantage. It can only be applied on a dataset where the signal and non-signal relation is close to the expected in the experiment. That being said, this can lead to problems when training a neural network (Sec. 4.4.1) where far more than the expected signal is loaded for training. For this reason, the ROC will be used as the metric of the classifying neural network.

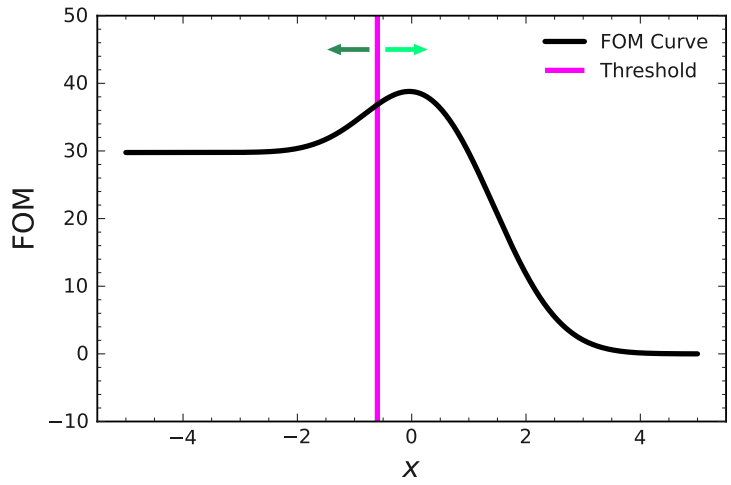


FIGURE 4.12: Example for a FOM curve. The maximum of the curve is reached at $x = 0.0$. Therefore, the best threshold value would also be at $x = 0.0$.

Chapter 5

Introduction to Neural Networks

There are no shortcuts.
Everything is reps, reps, reps.

Arnold Schwarzenegger, *Total Recall: My Unbelievably True Life Story* (2012)

Feed-Forward Neural Networks (from here on simply called neural networks) are algorithms based on multiple (*hidden*) layers of highly interconnected neurons.[1] The structure of these networks are inspired by the biological brain of animals, meaning, the neurons in a neural network mimic the way the neurons in a brain pass signal to one another [65]. The learning process of a neural network is similar to that of for example a human child.[2] In order to use neural networks, they need to be *trained* (*Deep Learning*[3] in the case of a deep neural network) by feeding them input data and providing the desired output. By adjusting the connections during the training, the network will be able to provide a more and more accurate prediction for each input. As a result, a neural network, for example, can be trained to recognize and distinguish between a shoe, shirt, pants etc. (Classifying neural network).

Overall, neural networks are used in all areas of modern society, from trying to predicting stock market movements, through predicting the demand of vegetables in a grocery store in order to reduce food waste to deep learning based speech synthesis and even generating digital pictures in any desired style [66]. In early 2023, a chat bot called ChatGPT, developed by the company OpenAI, reached public interest and sparked debates about how artificial intelligence will influence our daily life [67]. Nowadays, it becomes more and more difficult to imagine a world without the use of neural networks.

For the analysis of this thesis, neural networks based on TensorFlow 2.0 will be used in two major use cases. First, a classifying neural network is developed in order to distinguish between signal ($B \to K^* \ell\ell$) and non-signal (*continuum*, τ*-pair*, *self-cross feed* (a signal decay is present but the reconstruction failed), *non-signal B-events*) events. In the next step, a neural network will approximate the detector response function in order to correct for effects of the detector and reconstruction.

[1] In contrast to Feed-Forward Neural Networks there are also Hopfield Neural Networks which only consist of one layer of interconnected neurons in total. They are designed to store the information from the training within the network's connections itself [64].

[2] If the network really *understands* what it learned is a rather philosophical question. The same is true for the art generation. The first training must be done with art created by humans. However, it is also unlikely that a human will create art without ever seeing art by other humans in the first place.

[3] It's called deep learning since it contains one or more *hidden layers* between the input and the output layer. Therefore, it forms a complex structure. If a neural network only contains an input and an output layer, it is called *learning* instead of *deep learning*.

This chapter will start with a brief introduction to TensorFlow 2.0. After this, the basic functionality of neural networks will be described starting with the fundamental structure and the mathematics behind it. Finally, some techniques will be presented which can improve the performance of neural networks further.

5.1 About TensorFlow 2.0

TensorFlow 2.0 is a free open source end-to-end platform for machine learning developed by Google Brain. It was released in September 2019 as a successor of TensorFlow. Significant improvements included cross-compatibility and a major gain in the computing performance on GPU (*Graphics Processing Unit*). The ecosystem of tools and libraries provided by TensorFlow 2.0 are comprehensive and flexible and therefore let researchers and developers easily build and deploy machine-learning applications [68].

TensorFlow 2.0 supports a very user friendly application programming interface (API) called Keras which is used in this thesis. This allows the direct use of the Python language for neural networks.

5.2 Basic Functionality of Neural Networks

The fundamental functionality of a neural network has been known since the 1950s described by Frank Rosenblatt [69]. The general working principle of neural networks is rather simple and will be discussed briefly in this section. First, the general structure and then some mathematical basics will be sketched.

5.2.1 Structure of a Neural Network and Nodes

Neural networks consist of multiple (at least two) node layers (one input and one output layer) as depicted in Fig. 5.1. These layers contain highly interconnected nodes (also called neurons)[4]. Each node represents a mathematical function which has one or more signal inputs. The incoming signal is then processed and an output is produced which will be sent to every connected node of the next layer. The connection between the nodes themselves each carry an individual weight. These weights can be interpreted as the strength of the connection. Each node also carries a bias which is a constant value that is added to the sum of all outputs. All in all, a node can be interpreted as a model of linear regression consisting of *input*, *weights* (the connection between the nodes) and *bias* following Eq. 5.1 with z being the value of the node, w_i the weight of the ith input connection and x_i the ith input.

$$z = \sum_{i=1}^{m} w_i x_i + \text{bias} \tag{5.1}$$

5.2.2 Activation Function

Before the value of a node is sent to the next layer, it is processed by an activation function. These functions play a crucial role in introducing non-linearity into the neural network. Without non-linearity, these networks would be limited to modeling linear relationships between inputs and outputs. This restriction significantly

[4] If each node in each layer is connected to each node in the adjacent layers, then the network is a so-called convolutional neural network

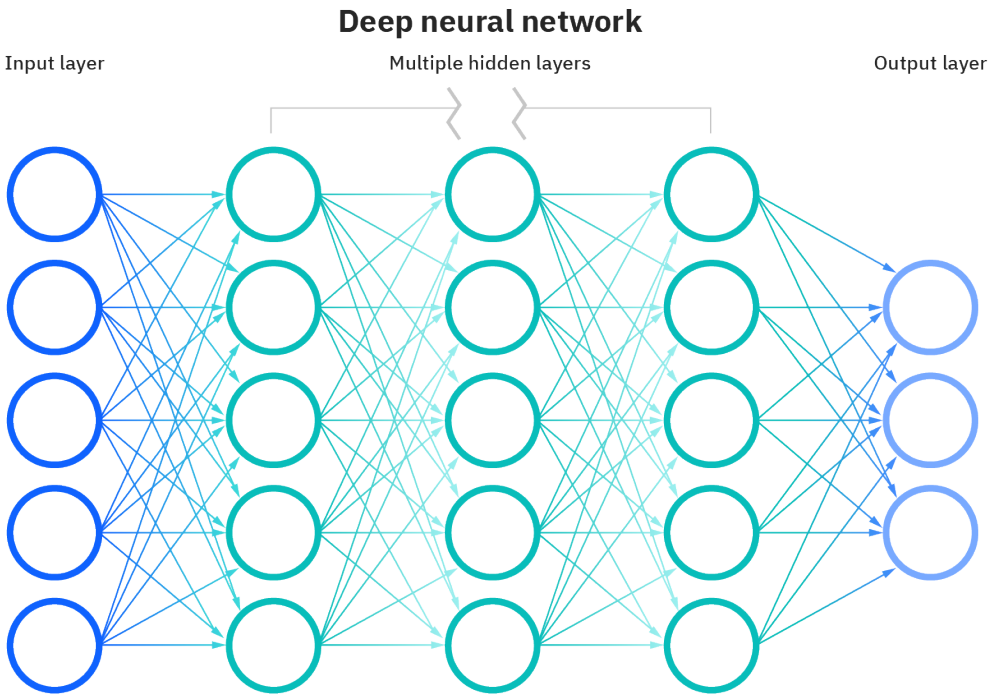


FIGURE 5.1: Artistic representation of a deep neural network containing one input, three hidden and one output layer. Each circle represents a neuron. The connections between the layers are shown by the colored arrows [65].

hinders their ability to capture the complex patterns and relationships often present in real-world data. The most popular activation functions for neural networks include *linear*, *ReLU*, *sigmoid* and *hyperbolic tangent* (*tanh*) which are shown in Fig. 5.2.

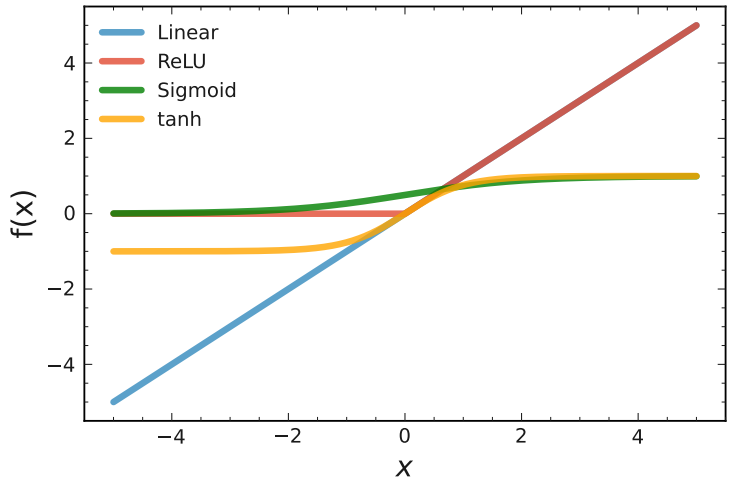


FIGURE 5.2: The four most popular activation functions.

linear $f(x) = x$

Used when linearity is expected. They are interpretable, computationally efficient and robust to outliers.

ReLU $f(x) = \max(0, x)$

Similar to linear, except that the output becomes 0 when the input value is negative. This property makes it less affected by the vanishing gradient problem that can obstruct the training.

sigmoid $f(x) = e^x / (1 + e^x)$

Often used in binary classification problems. The output values are in the range between zero and one, effectively converting it into a probability-like value.

tanh $f(x) = \tanh(x)$

Similar to sigmoid. However, this function also returns negative values and the slope is steeper around the origin. As a result, it is more sensitive to changes in the input.

5.2.3 Loss Function, Learning Rate and Gradient Descent

Before a neural network can be trained, a way has to be found to measure how well the model performs. In order to achieve this, a so-called *loss function* is used which takes the models predictions and compares it with the actual correct answers. The function then returns a numeric value which has to be minimized in order to improve the model.

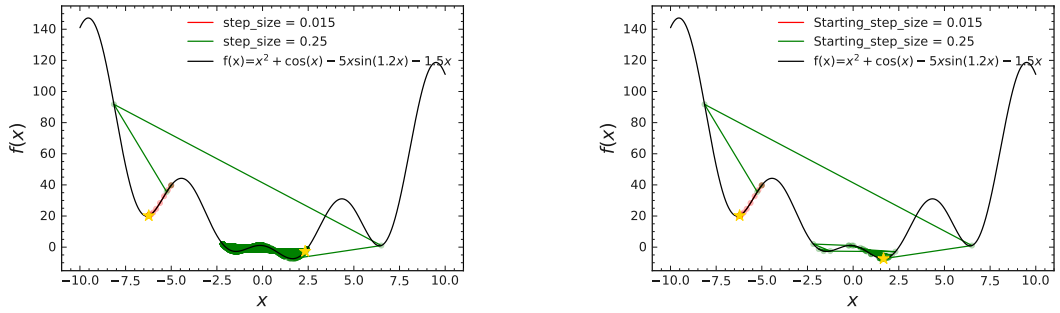
$$l(y) = \frac{1}{2m} \sum_{i=1}^{m} (\hat{y} - y)^2 \tag{5.2}$$

A simple loss function is shown in Eq. 5.2 where $\hat{y}$ is the predicted outcome, y is the actual value and m is the number of samples provided. As already mentioned, the main goal of the training will be to minimize the output of the loss function. This problem can be solved with an algorithm called gradient descent. A simple example of this algorithm is depicted in Fig. 5.3a.

The algorithm starts at a random point. It then takes a step in a direction where the output of the function becomes smaller (it follows the gradient). Each dot represents an iteration of the algorithm. This will be repeated until there is no further improvement. If the step size is constant, two major problems can occur.

- Small step size: The algorithm may get stuck on a local minimum and will not find the global minimum since it is not able to escape the local minimum. Also it will take a lot of computing power since the algorithm only takes small steps.
- Large step size: The algorithm may get closer to the global minimum but it is unlikely that it will get it exactly right. It will *dance* around the minimum.

The best compromise is to start with a large step size and to decrease the step size once there is no further improvement. Fig. 5.3b shows a such procedure. The gold stars mark the found minima by the algorithms. Note that the algorithm with a small starting step size was not able to escape the first local minimum just like in Fig. 5.3a. However, the algorithm with a larger starting step size did manage to



(a) (b)

FIGURE 5.3: Gradient descent with a constant (a) and dynamic (b) step size to find the global minimum. All algorithms start at the same point and the stars indicates the final point of the algorithms. In (a) the algorithms with the small and large step size are colored red and green, respectively. Both fail to find the lowest minimum displayed. In (b) the step size is decreased with time. The algorithm with the large starting step size is now able to find the lowest minimum shown. In both cases the algorithm with the small starting step size is not able to escape the first local minimum.

find the lowest minimum displayed. In the context of neural networks, the step size is called *learning rate*. Another problem that can occur while training is vanishing gradients. This means that the gradient becomes so small that the neural network is effectively stopped from training. This can be prevented by batch normalization (Sec. 5.4.2) and some activation functions (e.g. *ReLU*). In general, since each weight and bias introduces a new dimension, finding the global minimum in this multi-dimensional space is computationally very expensive.

Randomness

A way to escape local minima or saddle points is to use *randomness*. The algorithm with decreasing learning rate will be used multiple times with different random starting parameters and in the end the best result will be taken. However, even with a good starting point and a decreasing learning rate it is rare that the *global minimum* is found by the neural network especially with a high number of variables.

Optimizer

TensorFlow 2.0 provides several so-called optimizers. Each has a different strategy to find the lowest result of the loss function. The optimizer used in this thesis is *adam*. This optimizer adapts the learning rate for each parameter individually based on the past gradients. Also, it uses a method which takes the exponentially decaying average of the past gradients in order to smooth out the updates and hence accelerate convergence towards a minimum [70].

Hyper-Parameters

In addition to the optimizer, the hyper-parameters[5] need to be chosen. They are the starting point before the training begins. Unfortunately, hyper-parameters have to be chosen by *trial-and-error* until a model is found that *bites*[6]. After that, these parameters are *tweaked* to find a better model until there are no further improvement. Yet again, as already stated, there is no way of knowing if the found local minimum is also the global minimum [65]. In Sec. 5.8 some techniques will be introduced in order to find the best possible hyper-parameters.

5.2.4 Training Strategy

A neural network has to be trained in order for it to return meaningful results. For this a special dataset which includes the actual value of the desired outcome is required. The process of learning itself can be separated into four steps.

Forward Pass The output of the network is calculated by passing the input data through.

Error Calculation Using the special dataset and the loss function, the error between the desired and the network output is calculated.

Backward Pass The error is propagated backwards through the network and the gradient of the error on the weights are calculated.

Weight Update The weights are updated in the direction of the gradient. As a result, the weights are changed in a way that reduces the discrepancy between the output of the neural network and the desired output.

This training procedure is repeated multiple times until the desired number of iterations is reached or the error converges to an acceptable small value (Sec. 5.3.2).

Using all of the introduced techniques, it would already be possible to write a functional neural network. However, there are a lot of additional techniques which can be used in order to improve it further. The techniques used in this analysis will be briefly introduced and discussed in the following sections.

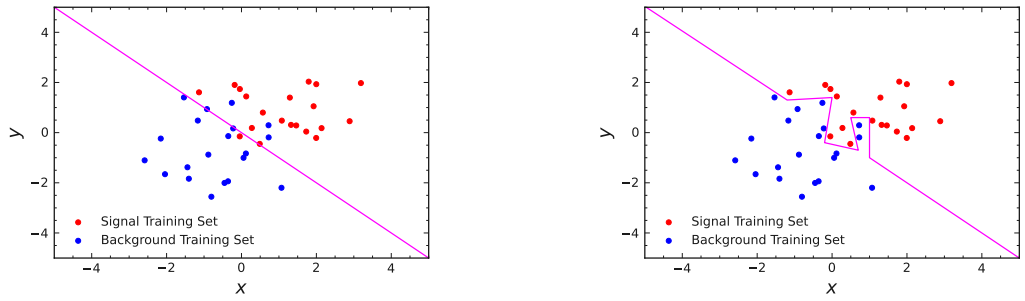
5.3 Preventing Overfitting

Neural networks are prone for overfitting. This means that the network learns features/patterns of the training set and applies them to the weights in a way that the training set will be perfectly fitted, but at the same time, the network prediction fails for a slightly different data set.

In Fig. 5.4 an example for a good and an overfitted model is given. The signal and background points are Gaussian distributed around the points $(1, 1)$ and $(-1, -1)$, respectively. One would assume that a simple straight line with a slope of -1 and an offset of 0 would be the best way to divide the area in signal and background regions. This model is shown in Fig. 5.4a. In Fig. 5.4b, an overfitted model is shown.

[5] This includes (but not exclusively) the number of layers and nodes of each layer, the activation function, the loss function, the learning rate, the factor by which the learning rate decreases and the number of iterations before early stopping aborts the process, ...

[6] *biting* in this context means that the network learns something at all. Most of the time a network with random starting hyper-parameters won't learn.



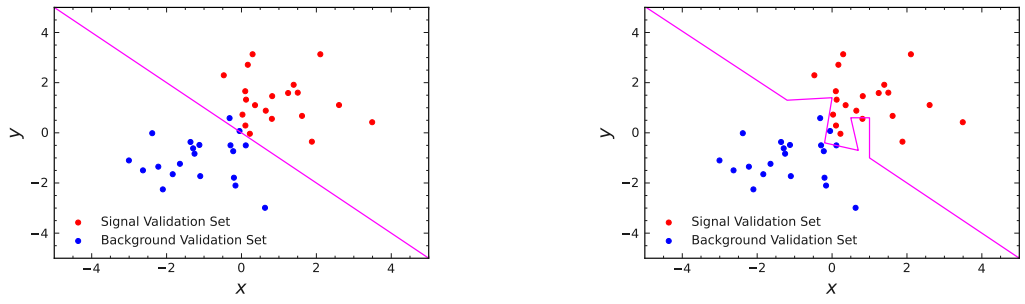
(a) (b)

FIGURE 5.4: On the left the signal region is divided from the non-signal region with a pink line. Some background events creep in the signal region and one signal event is lost in the background region but the line fits well. On the right are the same data points but the model is overfitted. This model appears to be better than the model on the left but this only works well on this training dataset.

The model perfectly divides the signal from the background on the training sample. However, it is obvious that this model will perform far worse on other similar data samples.[7]

5.3.1 Validation Data Set

The simplest way to find overfitting is to take another dataset and apply the neural network model during the training on it as well. This dataset will be called *validation dataset*. If there is no overfitting then the results of both datasets during training are similar. However, if the network performs way better on the training dataset compared to the validation dataset and, even more importantly, the performance of the model on the validation data set starts to decrease, then overfitting occurs, resulting in a network which will not work as expected on real data.



(a) (b)

FIGURE 5.5: On the left the same good model is applied to the validation data set. It still fits well to the data points. On the right the same overfitted model from Fig. 5.4 is applied to the validation dataset. This model is now far worse compared to the good model on the left.

[7] This simple example also shows that it is not always necessary to use a neural network for classification. In this case it would be far more efficient and stable to just use a straight line as a classification model. It is therefore important, to understand underlying structures in the data before starting to write and use a neural network.

An example for a validation set is shown in Fig. 5.5. Since the validation set and the training set should be comparable, the points for signal and background are created in the same way. The good model (Fig. 5.5a) still divides signal very well from background. However, the overfitted model (Fig. 5.5b) now performs way worse. This method of applying a validation dataset and calculating the loss on it is a very simple and effective way to detect overfitting.

5.3.2 Early Stopping

In general, if the network trains for too long, it is likely that overfitting will take place. The network will have too much time to learn features of the training dataset. This can be prevented by the method *early stopping*. To be able to use this, the introduced validation dataset is needed. After each iteration (*epoch*), the loss of the validation dataset is calculated. Then, if this loss has not changed or got larger after a certain number of training iterations (epochs), the training stops, preventing overfitting. The number of epochs waited before stopping is another hyper-parameter and can be changed.

5.3.3 L2-Regularization

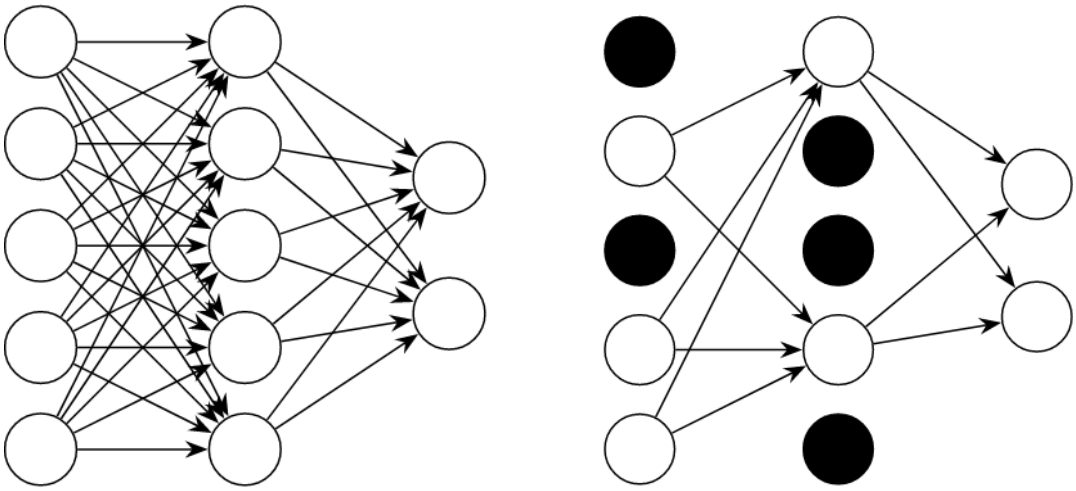
Another method to prevent overfitting is the *L2-Regularization*. To apply this method, another term needs to be added to the loss function (Eq. 5.2) resulting in:

$$l(y) = \frac{1}{2m}\sum_{i=1}^{m}(\hat{y} - y)^2 + \lambda \sum_{j=1}^{n} w_j^2 \tag{5.3}$$

In this equation w_j is the weight of the jth feature with n being the number of features and λ is the so-called regularization parameter which can be tuned. A large value of λ will result into a strong penalization of large weight values. In contrast, the regularization effect will be smaller for lower λ values. This reduction of the weights results in a smaller value of z in Eq. 5.1 which again reduces the impact of the activation function resulting in a simpler function [71] [72]. This will reduce the effect of overfitting.

5.3.4 Dropout

Another way to reduce overfitting is excluding nodes in the training randomly. With each iteration of training, different nodes will be deactivated. The probability of this happening can be changed, resulting in another hyper-parameter. The motivation behind the idea of this method is that the neural network will no longer be able to rely on single nodes with high weights to give a good result since this node could be deactivated by chance. Thus resulting in more distributed weights across the network with overall smaller values. This method is not allowed for the last layer of the network. Otherwise, a possible correct output of the network would be deactivated and the network would not be able to provide the correct answer in the first place [73]. An example for a neural network sketch with a dropout mechanism is shown in Fig. 5.6b. The black colored nodes are deactivated in contrast to Fig. 5.6a where all nodes are fully connected. In general, when a dropout mechanism is used, the loss of the validation data set is lower during training compared to the loss of the training data set. This is because the mechanism is only applied to the network during training and not during validation, resulting in a *worse* network while training.



(a) (b)

FIGURE 5.6: A sketch of the neurons of a deep neural network. (a) No dropout is applied. Every node of each layer is connected to all nodes in the next layer. The network is fully connected. (b) the same network but with dropout is shown. Each node has a certain chance to be excluded. These nodes are not connected with any other nodes. Note that the dropout method is not applied to the output layer (last layer) [73].

5.4 Batches

5.4.1 Batch Size

In order to improve the speed of training it is advantageous to only train over a number n_b (batch size) smaller than the total number of available sample size. The two most important improvements are:

- The memory required to perform the training is drastically decreased since the network is using smaller batches. This procedure is especially important when the dataset is too big for the machine's memory.
- In general, the networks train faster with small batch sizes. After each propagation, the parameters of the network are updated. If there are ten batches instead of one complete sample, the network would be updated ten times instead of one time with the complete sample.

However, this also has disadvantages. The size of the batches affects the accuracy of the estimate of the gradient. An example for this effect can be seen in Fig. 5.7. The blue line represents a complete data set as the batch. Green are mini batches, so their size is smaller than the total number of the sample. Red is stochastic, corresponding to a batch size of one. It is clear that the green line fluctuates more in comparison to the large batch size (blue). However, it still reaches its goal or at least gets fairly close to it while the red line appears to be scattered wildly around the minimum. As a consequence, the batch size itself is a hyper-parameter which needs to be optimized. Often, the size of the batches is a power of two $(4, 8, 16, 32, 64, 128, \dots)$ [74].

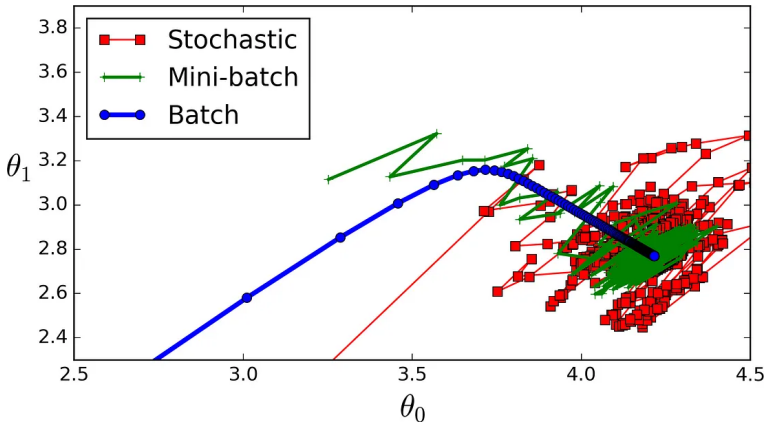


FIGURE 5.7: Behavior of the gradient for different batch sizes [74].

5.4.2 Batch Normalization

It can be advantageous to normalize each layer in a neural network by fixing the mean and variance of each of their inputs within a batch. This will reduce the so-called *internal covariate shift*, which is the effect of sources of randomness on the distribution in the input data to the neural network. While the reason for its effectiveness is still a matter of discussion, it is evident that the inclusion of batch normalization makes the training of neural networks faster and more stable. Other benefits are believed to be robustness to larger learning rates without running the risk of resulting in vanishing or exploding gradients [75].

5.5 Feature Engineering

Using all the methods introduced, a rather well working neural network can already be programmed. However, a model can only be as good as the features it was trained with. In general, one wants to add as many useful features as possible to the training. In the context of a binary classifying network, a good feature separates *signal* from *non-signal*. This quality can be improved by manipulating and combining these variables in some way. This is called feature engineering. It is not straightforward and takes a lot of time but, provided a good new feature is found, the performance of neural network can be drastically improved. An example for a *new* feature is a variable which represents the total sum of the momenta for all final state particles.

5.5.1 Handling Outliers

Another part of feature engineering is the handling of outliers. It can happen that some features contain entries which are very far away from most of the others. Moving them closer to the others can improve the resulting network because now the entries of the feature are in a fixed interval. This method, called *capping*, has to be done with care and for each feature individually if necessary.

An example for this procedure can be seen in Fig. 5.8a. The outliers on the right are moved closer to the other values around 0. It is important to note that they are not removed from the data sample.

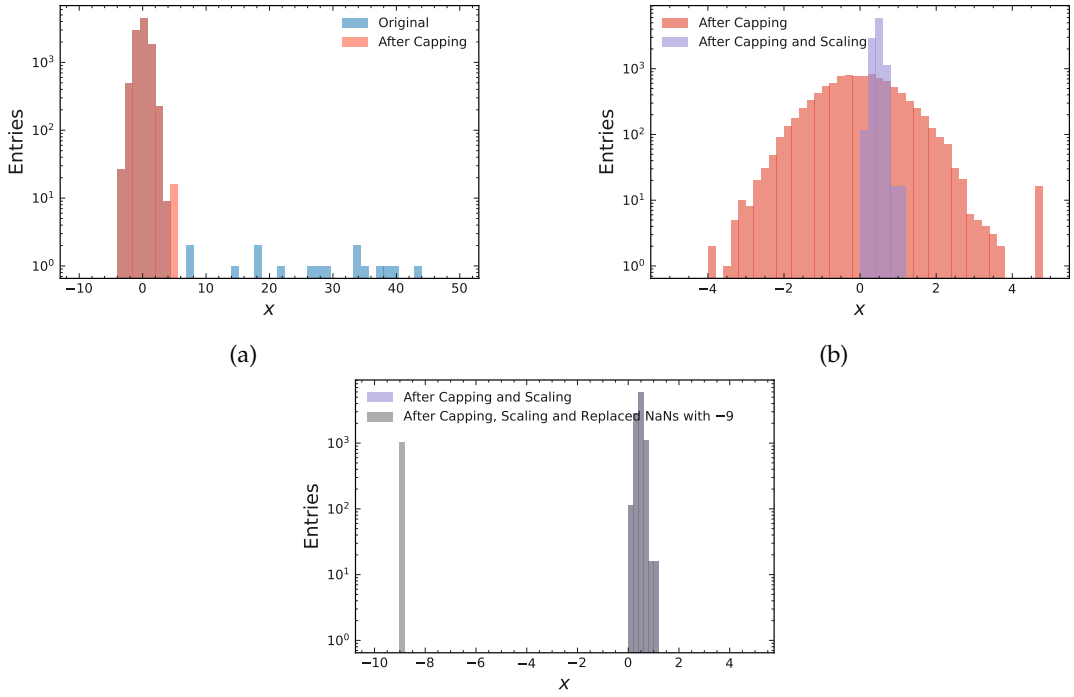


FIGURE 5.8: (a) Impact of the capper. The far away values are moved closer to the others. (b) Scaling the values to an interval of $[0,1]$ in order to improve the quality of the neural network. (c) NaNs are replaced with -9. Note that in the interval between 0 and 1 nothing changed.

5.5.2 Input Normalization

In general, neural networks and their corresponding activation function perform way better when their features are normalized. Therefore, each feature will be scaled to an interval of $[0,1]$. The input normalization takes the complete training sample into account. The result for the previous example can be seen in Fig. 5.8b. When the model is applied to other datasets, the same scaler needs to be loaded. Otherwise, the prediction output for the evaluation could not be interpreted. This is also the biggest difference to the batch normalization described in Sec. 5.4.2, where the normalization changes during training.

5.5.3 Handling Missing Values

It can happen that some features are not always filled with information. Instead they contain some NaNs (not a number). For example, this can happen if the features represent the energy of the rest of event tracks. If there are no tracks in the rest of event then these features will be empty. In general, the network can not work with NaNs and they need to be excluded or replaced. Since exclusion would mean that either the feature could not be used or the entire event would be excluded and, as a result, the amount of available data would be drastically reduced, the only feasible option is replacement. This way, the network is also able to extract some information from this condition. In the case of the example, the network would *know* that there are no additional tracks in the rest of event. However, this needs to be done with care. As a rule of thumb, the NaNs should be replaced with a value at least three standard

deviations away from the mean or median of the other values in the variable. As a consequence, this needs to be done after the scaling of the values, since doing it before would render the scaling process rather useless because the *real* values would then all be close to either zero or one.

5.6 Ensemble Neural Networks

As already mentioned, the quality of the network can depend on the initial weights chosen for the training. In order to reduce this random effect, it is possible to use an ensemble of neural networks with the same structure but different starting weights at once and take the average of the predicted output. This way the network is overall more stable and less effected by fluctuations in input data. In general, the performance of an ensemble neural network tends to be better compared to a single neural network [76].[8]

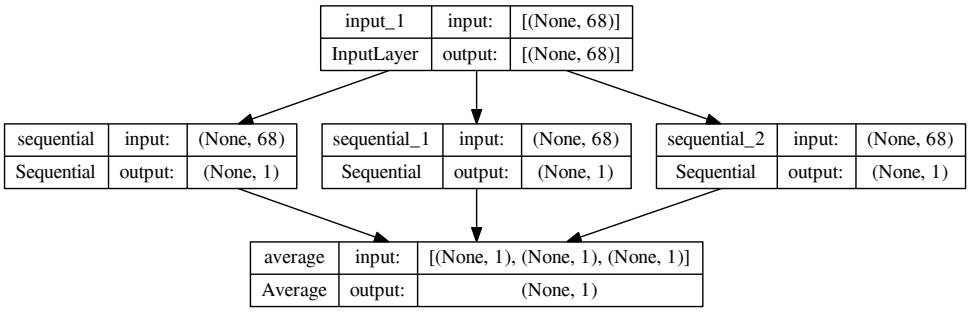


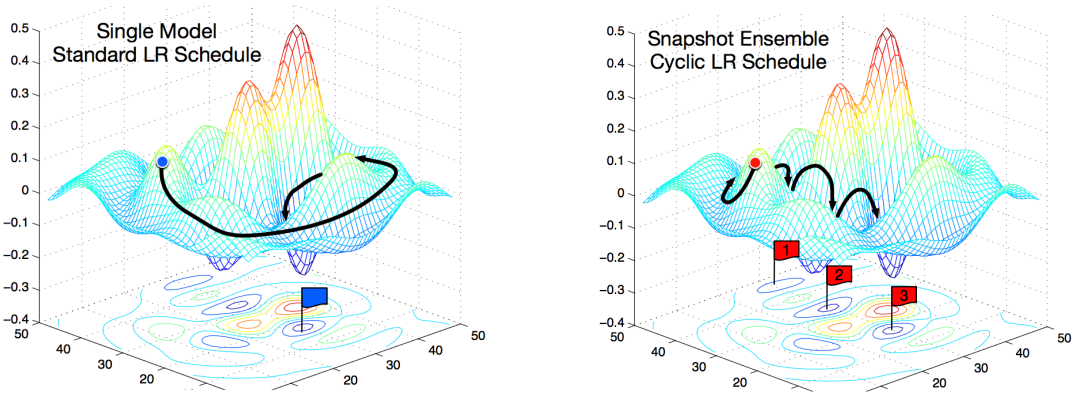
FIGURE 5.9: Model depiction of an ensemble neural network.

An example for a model containing an ensemble of neural networks can be seen in Fig. 5.9. Here, after the input layer, three sequential neural networks are placed in parallel. These are complete networks with multiple layers and activation functions. In principle, they can even have completely different structures. The output of each of these networks is combined in the final output layer where the average will be calculated.

5.6.1 Snapshot Ensemble

It is preferred to let the neural network converge to the lowest local minimum possible. However, each local minimum found along the optimization path also contains information which can be used to improve the quality of the model. This information can be obtained by saving a snapshot of the model after it converged and then let the learning rate increase again to leave the local minimum and continue training. This process of lowering and increasing the learning rate is called *Cyclic Learning Rate Schedule*. These processes are repeated until the training ends. After the training is done, all snapshots are assembled together and their average is taken. It is also possible to assign different weights to the output of the snapshots which is recommended because the last snapshot tends to be the better one of the ensemble.

[8] Even the impact of each neural network in the ensemble to the final prediction can have a different weight.



(a) (b)

FIGURE 5.10: (a) Example the gradient during a training of a normal neural network. The neural network only finds one local minimum. (b) Same network but with Snapshot Ensembling and a Cyclic Learning Rate Schedule. In each found local minimum, the current model is saved and the learning rate is increased again [77].

In Fig. 5.10a, the optimization process for a normal single model with a standard learning rate scheduler is shown. Here, only the final model is saved. In Fig. 5.10b, the optimization path for a snapshot ensembling model with a cyclic learning rate scheduler can be seen. Once the model reaches a local minimum, its weights are saved and the learning rate is increased again so it can escape and is able to reach the next local minimum. This process repeats until the training finishes. Afterwards, all saved weights are loaded and merged to an ensemble neural network. Again, this approach results in more stable models with a lower error rate [77].

It is important to note that if a snapshot ensemble technique is chosen, the early stopping method should be deactivated and the length of the training and the timing when the snapshots happen must be chosen in such a way that overfitting does not take place.

5.7 Function Approximation

So far, artificial neural networks were introduced as a way to classify data samples. In addition to that, according to the *Universal Approximation Theorem*, neural networks are also able to approximate any continuous function. According to theory, a single-hidden-layer neural network is able to approximate any continuous function with arbitrary precision, provided that the activation function fulfills certain conditions and that the number of nodes can get arbitrarily large [78] [79]. In practice, oftentimes a more smooth approximation of the function is desired in contrast to exact replication of the input data. Having this in mind, the number of required nodes can be drastically reduced. According to [80], it is possible to approximate any continuous function with a neural network with exactly three hidden layers.[9] The result is a computationally reasonable approach to function approximation [81] [82].

[9] In general it is preferred to go deep instead of wide. This means computational resources are reduced by building a deeper neural network with less nodes overall compared to a one layer neural network with a lot of nodes.

5.8 Hyper-Parameter Tuning

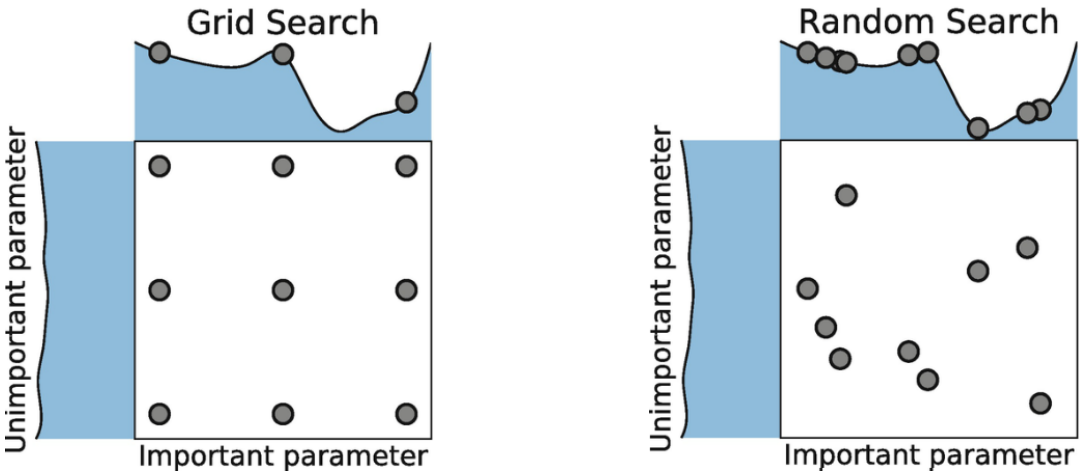
As already described, there are a lot of hyper-parameters which need to be tuned. In the following subsections, three techniques will be described which aim to find the *best* set of hyper-parameters within computationally reasonable time frame.

5.8.1 Grid Search

The most straightforward method is to choose possible settings for each hyper-parameter, train a network for each combination and then choose the best. A sketch of the grid search working principle can be seen in Fig. 5.11a. This strategy has several disadvantages. For one, even if there are only six hyper-parameters ($n_p = 6$) each with five possible settings ($n_c = 5$), then a total of $n_G = n_c^{n_p} = 15\,625$ trainings would be needed to test all of the possible combinations. Obviously, there is a big risk of wasting resources on unimportant parameter choices in addition to the disadvantage that the chance of hitting the best set of parameters is limited by grid granularity, which refers to the spacing between points on the grid of hyper-parameter values.

5.8.2 Random Search

A better way to find a good set of hyper-parameters is to search for them randomly. If possible, replace the fixed grid with continuous ranges and choose n_R random sets of hyper-parameters in order to search for the best. Usually $n_R < n_G$ is true and given that n_R is still sufficiently large, the chances of finding a good set of parameters are higher compared to the grid search. An example sketch for randomized search can be found in Fig. 5.11b.



(a) (b)

FIGURE 5.11: Hyper-parameter optimization using (a) Grid Search and (b) Random Search [83].

5.8.3 Sequential Search

However, the best way is to perform a more informed search. The idea is to start with an initial coarse or random search. These sets of hyper-parameters together

with their corresponding loss are then fed into a Bayesian optimizer. This will return a prediction for the best next hyper-parameters to test and perform a new training. These steps are repeated. However, even with this method, there is no way of knowing that the found set of hyper-parameters is the best set there is. Once the performance of the neural network is satisfying, the best found hyper-parameters are chosen and the search ends [83].

5.9 Boosted Decision Trees

There are a variety of methods to distinguish between signal and non-signal data. Another very popular method are the so-called Boosted Decision Trees (BDTs) which can be used to improve the performance of neural networks even further. BDTs are rather successful classifier in the realms of physics because they combine many weak classifiers with a technique called boosting. The working principle of a BDT will be described briefly in this section.

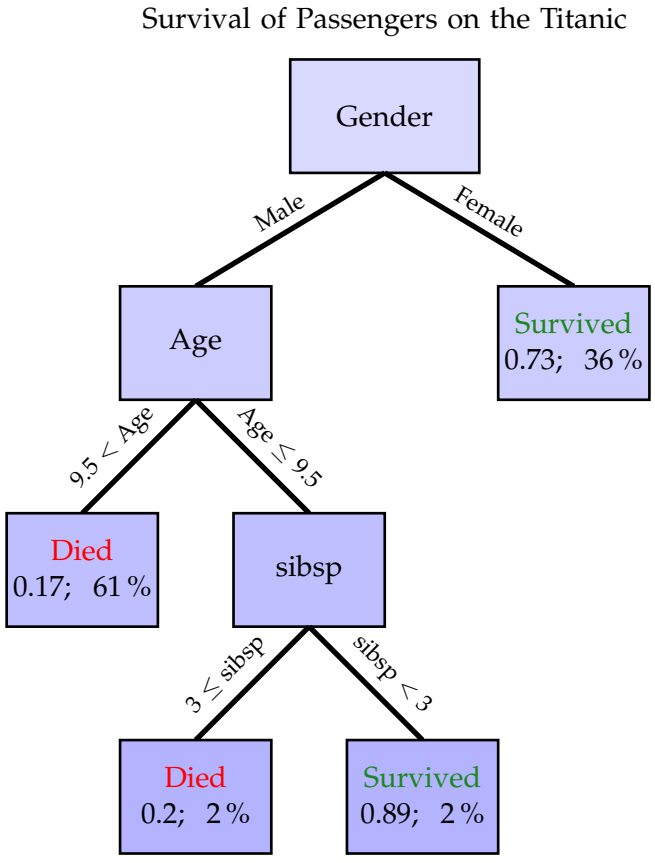


FIGURE 5.12: Example for a structure of a decision tree with a tree depth of three. The probability to survive the Titanic accident will be determined based on the features gender, age and how many siblings and spouses (sibsp) are on board. The numbers in each box (leaf) show the probability of survival (left number) and the percentage of the total data sample (right number). For example, 36% of all people on board were female and each of them had a 73% chance of survival according to the decision tree [84]. If the probability of survival is higher than 50 % the leaf is labeled Survived, otherwise Died.

Before describing BDTs, decision trees in general will be discussed. The working principle behind a decision tree can be represented by a tree like structure, hence the name. An example for this structure can be seen in Fig. 5.12. A decision tree will separate the training data in two regions. In the case of the example the regions are *This person will (not) survive the Titanic*. The decision tree does this by consecutively cutting on one variable at a time. After all the conditions are applied, signal enhanced and suppressed subsets are created. The requirements on the features are chosen in order to maximize the separation gain, which measures the improvement in classification accuracy achieved by splitting the data based on a particular feature. The number of consecutively applied requirements is called the depth of the tree. In the example, the depth is three.

The most simple decision trees are prone for overfitting, especially when they have a large tree depth. As a consequence, they are not powerful classifiers in general. However, they can become so-called weak-learners by limiting their depth. Now, a *Gradient Boosting* algorithm combines multiple weak learners to one single classifier with great separation power. In this algorithm, the weak-learners are trained consecutively. In the first iteration of the training, the first weak-learner returns the mean of the particular feature. In addition, the residual errors of the output of this weak-learner algorithm will be calculated and passed as output target to the next weak-learner which will be trained.

This last part of the procedure is repeated multiple times. After the training stops, the final output is calculated as a weighted sum of the outputs of all weak-learners. In order to prevent overfitting, it is common to train each weak-learner with its own randomly created subset of the whole training data. This will also speed up the training.

The next generation of Gradient Boosting is called *XGBoost* (*eXtreme Gradient Boosting*) which is an improved version. The key difference is that XGBoost also includes regularizations (*L1* and *L2*), which speed up the training further, resulting in even better results.

The most important hyper-parameters for a BDT are the number of weak-learners, including their depth, and the learning rate of the algorithm [85].

5.9.1 Feature Importance

It can be an advantage to understand how much each of the feature contributes to the final prediction of a classifier and rank them. This ranking is called *feature importance*. Knowing this ranking, features which contribute little to nothing to the final prediction can be excluded from the training, resulting in a faster and more resource friendly training [86].

The feature importance will be calculated with the BDT since using the tree-like structure makes it very easy to estimate the impact of each feature. This information will also be applied to the neural network studies because for a convolutional neural network this calculation is not trivial. Since the nodes are so highly connected it is almost impossible to determine the impact of one feature to the final output. However, one can assume that the impact of a feature to the final prediction should be similar for both, BDTs and neural networks.

Part II

$B \to K^* \ell\ell$ Analysis

Chapter 6

Event Reconstruction Using basf2

> But how do you live and have no story to tell.
>
> Fyodor Dostoevsky, *White Nights* (1848)

Before the $B \to K^* \ell\ell$ angular analysis can start, data from both the real detector and simulation is needed. Then the $B \to K^*(\to K\pi)\ell\ell$ decays have to be reconstructed from these data. During each of these steps the *Belle II Analysis Software Framework* (basf2) provided by the Belle II collaboration is used.

6.1 basf2

basf2 is a software used in all areas of the data-processing pipeline at Belle II [87]. This includes generating simulated data (Monte Carlo), unpacking real raw data, tracking, clustering, etc. and high-level analysis reconstruction. In Fig. 6.1 both large colored boxes are done with the help of basf2 and its modules. The first big box shows the procedure of how data is gathered at Belle II, both for real data (Data Taking) and MC. The collaboration provides data files which already has undergone all of these steps. The analysts then have to reconstruct the decay they are interested in. For this, release 06-01-15 of basf2 was used during the creation of this thesis. The final analysis alongside its preparation is done *offline*, meaning that it is not using basf2 but rather a variety of python scripts which are run locally.

6.2 Particle Lists and Event Reconstruction

In order to be able to reconstruct events, some lists with particles in them need to be loaded first. For each final state particle and the K^0_S and π^0 particle, these lists are provided by the collaboration. Both neutral particles decay almost instantly after creation via very distinct decay channels. It was therefore decided to provide particle lists which already contain these particles candidates.

In general, every particle list contains both, particle and anti-particle. Charged particles candidates have to fulfill certain criteria to be included in a list.

Track Quality The track has to be within the CDC acceptance ($17° < \theta_{\text{CDC}} < 150°$) and the number of CDC hits must be higher than 20.

IP Cut The origin of the track has to be close to the interaction region. ($\texttt{dr} < 0.5\,\text{cm}$ and $|\texttt{dz}| < 2.0\,\text{cm}$)

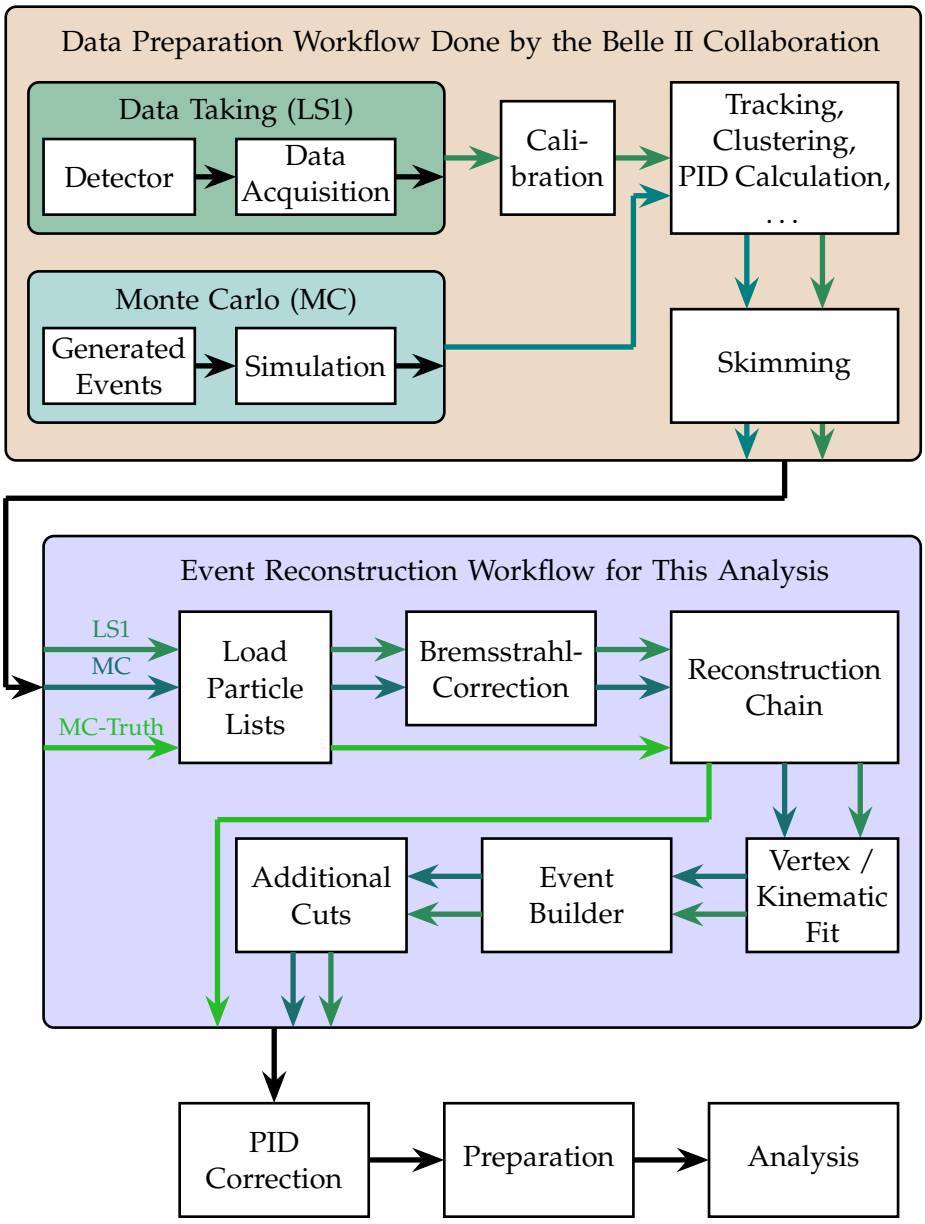


FIGURE 6.1: Workflow for the event reconstruction. The first large box are preparations done by the collaboration, the second one describes the event reconstruction for this analysis.

PID Cut The particle identification for the required mass hypotheses according to Sec. 3.9 must be greater than 0.5 (0.1) for the *good* (*loose*) list category.

Particle candidates are added to the lists based on their properties. Therefore, a particle candidate can be included in multiple lists at the same time if it meets the criteria for each. Furthermore, since electrons can perform bremsstrahlung, electron candidates need to be corrected for that. This is done using a module provided by basf2. Tab. 6.1 shows all the used lists during this analysis.

In order to reconstruct a decay chain, particle candidates from lists are combined, conserving energy and momentum, to create a mother candidate in a new particle list. In the context of this thesis, a particle from the π list is combined with a particle from the K list to create an entry in a newly created K^* list. In the next step these

TABLE 6.1: Used particle lists and their category. Even though K_S^0 and π^0 are not final state particles, Belle II decided to provide particles lists for them. The category `loose` (`good`) refer to a PID cut value of PID $>$ 0.1 (0.5). For the π^0 list, criteria where chosen in such a way, that 50 % of all π^0 are reconstructed. The K_S^0 list is created by reconstructing two charged pions. In addition to that, the vertex fit must not fail (Sec. 6.3) and the reconstructed invariant mass of the K_S^0 must be within $0.450\,\mathrm{GeV}/c^2 < M_{\mathrm{inv},K_S^0} < 0.550\,\mathrm{GeV}/c^2$.

Particle List Loaded	Category
Electron	`loose`
Muon	`good`
K^+	`good`
π^+	`good`
K_S^0	`merged`
π^0	`50eff`

particle candidates are combined with two oppositely charged leptons from the corresponding lepton list ($\ell = e$ or $\ell = \mu$) to create a particle candidate which is filled in the B candidate list.

6.3 Vertex and Kinematic Fit

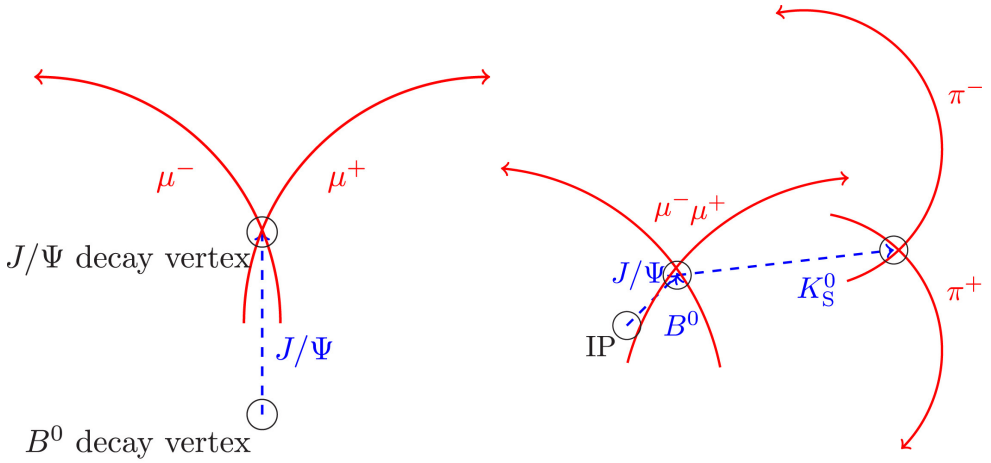
The quality of the reconstructed candidates can be improved with modules provided by basf2. One of these modules is the so-called *vertex treefit*. In the case of the K_S^0 list, its decay vertex position, flight length, and four vector and their uncertainties can be obtained by combining the measurements of the two pion daughters assuming they come from the same vertex. Performing a vertex fit will result into an improvement of the pion track momenta and the χ^2 of the fit can be used to suppress background [88].

Using the vertex treefit, all particles within the decay chain are updated starting with the final state particles working its way up, updating track positions and momenta information. This fitter also allows to constrain the mass of the particle in the given list. This is done for the π^0. As a result, the mass of the particle candidate will be set to the PDG value. The effect of this constrain can be seen in Fig. 6.3a.

The effect of the vertex treefit on the variable dz can be seen in Fig. 6.3b. Before the fit, dz was always at one fixed value which is not physical. After the fit the distribution looks way more realistic. Also the momenta of the particles are shifted. Of course this will also affect the resolution of the angles describing the decay. This is more subtle but an improvement can be observed as depicted in Fig. 6.4 for the variable $\cos\theta_\ell$.

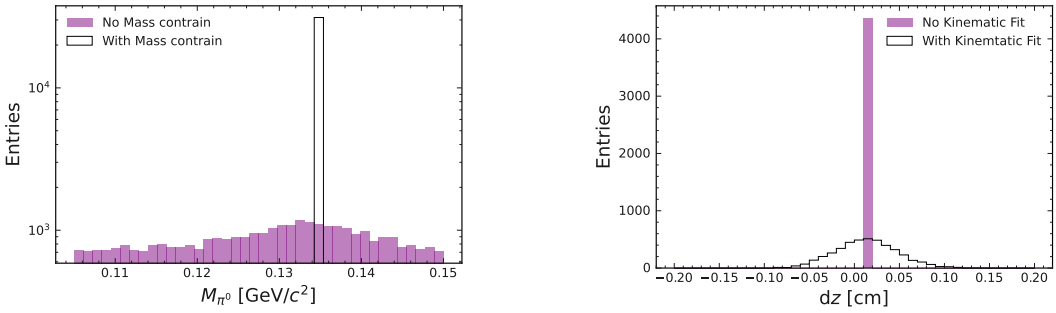
6.4 Continuum Suppression, Event Kinematics and Rest of Event

More additional information about the event in general can be gained by calling the following methods. This information can be used later on to improve the quality (higher purity without losing too much efficiency) of the particle lists.



(a) (b)

FIGURE 6.2: The red lines show the track helix approximations determined by the tracking detector, the blue dashed lines show the decaying particle momentum vectors obtained by the fit. (a): Illustration of the $J/\psi \to \mu\mu$ decay. Since the detector is unable to see the decay length of the J/ψ, its vertex is taken to be the one of the B^0. (b): Illustration of the $B^0 \to K^0_S(\to \pi^+\pi^-)J/\psi(\to \mu^+\mu^-)$ decay. The shown lengths are not up to scale [88].



(a) (b)

FIGURE 6.3: (a): Effect of the mass constrain for the π^0 in red compared to no kinematic fit in blue. (b): Variable dz before (purple) and after (black) the kinematic fit.

buildRestOfEvent This method introduces new variables like the *Rest of Event Energy* and *Rest of Event Mass*.

buildContinuumSuppression This method introduces access to new variables like the *Super-FoxWolfram* variables and *Cleo Cone Continuum Suppression* variables, etc..

buildEventShape Using this method, event-level quantities like *thrust*, *sphericity* and *FoxWolfram Moments* will be calculated and provided.

buildEventKinematics This method calculates global event kinematics like visible energy in an event, missing momentum and missing mass.

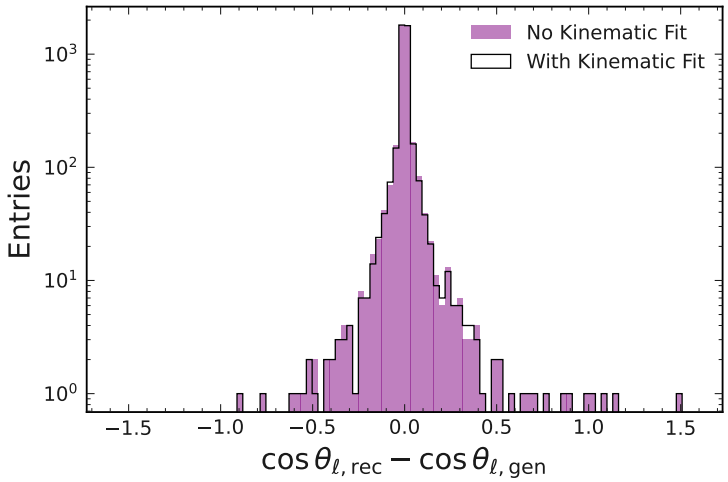
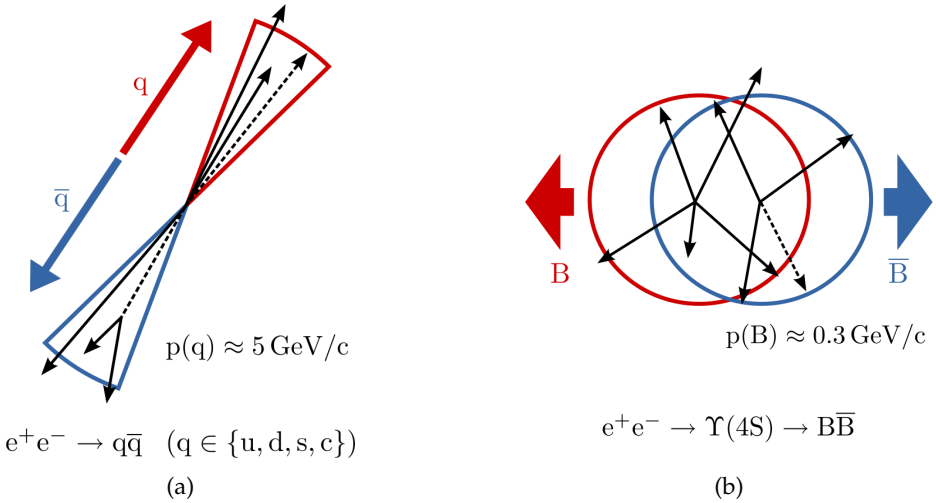


FIGURE 6.4: Effect of the vertex fit on the variables $\cos\theta_\ell$.

6.4.1 Continuum Suppression

A significant amount of background is contributed by continuum events ($e^+e^- \to u\bar{u}, d\bar{d}, c\bar{c}, s\bar{s}$). For this reason, a few variables which can help to suppress these kind of events will be introduced briefly.

All of these continuum suppression variables use the fact, that events coming from continuum have different kinematic shapes in contrast to $\Upsilon(4S)$ events. This is due to the fact, that the beam energy is close to the mass-threshold for the production of two B mesons. As a consequence, the B mesons are produced almost at rest in the center-of-mass system and these events tend to be almost isotropic in the detector. Continuum events, on the other hand, undergo large energy releases. The spatial distribution is therefore a more back to back jet-like structure. An illustration of the difference between continuum and B meson events can be seen in Fig. 6.5.



(a) (b)

FIGURE 6.5: Illustration of the difference between continuum (a) and B meson events (b) in the center-of-mass system [89].

Fox-Wolfram Moments

In order to calculate the Fox-Wolfram moments [90], all tracks in the events are considered and the angular distribution between them is described using Legendre polynomials. The moments are calculated by

$$H_k = \sum_{i,j}^{N} \frac{|\vec{p}_i||\vec{p}_j| P_k(\theta_{ij})}{E_{\text{vis}}^2} \tag{6.1}$$

with N being the number of reconstructed charged particles in the event, $|\vec{p}_l|$ the momentum of the charged particle l, P_k the kth Legendre polynomial, θ_{ij} the angle between the momentum of the ith and the jth particle and E_{vis} the total visible energy in the event. It was found that it can be advantageous to form the ratio

$$R_k = \frac{H_k}{H_0} \tag{6.2}$$

between the kth Fox-Wolfram moment and the 0th. Especially, R_2 has proven to be effective in separating $B\bar{B}$ from continuum events.

Super Fox-Wolfram Moments

Even more information about the event can be extracted when the summation over all particles is split into groups of the combination of the final state particles belonging to the B meson candidate (s) and the remaining tracks (o). Using this technique, three types of variables can be created: R_k^{ss}, R_k^{so} and R_k^{oo}. These new variables are called Super Fox-Wolfram Moments [91]. A total of 18 of them will be used in this thesis.

6.5 Selections Before and During Event Reconstruction

In addition to the conditions introduced by the provided lists, there are some more applied during the reconstruction. A very important requirement is the so-called *skim cut*. In fact, this cut is even applied before the event reconstruction by the analyst starts as depicted in Fig. 6.1. The Belle II collaboration decided to provide pre-treated files filled with events which all passed some minor conditions in order to remove obvious non-signal events. As a result, the number of events that are even considered for event reconstruction is reduced. This saves computation time and storage resources. Details about the used skim cut are listed in Tab. 6.2. As mentioned, these requirements are rather loose since the main goal of them is to keep the efficiency as high as possible while rejecting as much obvious background as possible.

In addition to the skim, there are three additional conditions applied during event reconstruction in order to remove obvious background events. These requirements are shown in Tab. 6.3. The cuts on the K^* mass is chosen to be about two times the decay width of the particle in each direction ($\Gamma \approx 50\,\text{MeV}/c^2$). The other two conditions are chosen by considering the physical properties of the signal decay. The $\Upsilon(4S)$ decays into two B mesons, each carrying half of the center-of-mass frame beam energy E_{Beam}. Using this knowledge, it is possible to construct two independent and therefore uncorrelated variables ΔE and M_{bc}, defined as:

$$M_{\text{bc}} = \sqrt{E_{\text{Beam}}^2 - |\vec{p}_B|^2} \qquad \text{and} \qquad \Delta E = E_B - E_{\text{Beam}} \tag{6.3}$$

TABLE 6.2: Requirements of the $B \to X\ell\ell$ (no *lepton flavor violation* (LFV)) skim. The skim code is 12160200.

Level	Selection	Cut Value
Event-Level	foxWolframR2	< 0.5
	p_t of Particles	$> 0.1\,\mathrm{GeV}/c$
	Cluster Energy	$> 0.1\,\mathrm{GeV}$
	Number of Tracks	≥ 3
Electrons	electronID	> 0.1
Muons	muonID	> 0.5
Dilepton System	Dilepton Energy $E_{\ell\ell}$	$> 1.5\,\mathrm{GeV}$
Leptons	p_{lab}	$> 0.395\,\mathrm{GeV}/c$
	dr	$< 0.5\,\mathrm{cm}$
	$\lvert \mathrm{d}z \rvert$	$< 2.0\,\mathrm{cm}$

with E_B and $\vec{p}_B$ being the energy and momentum of the reconstructed B meson, respectively.

TABLE 6.3: Additional requirements during reconstruction.

Variable	Requirement
M_{K^*}	$0.796\,\mathrm{GeV}/c^2 < M_{K^*} < 0.996\,\mathrm{GeV}/c^2$
$\lvert\Delta E\rvert$	$< 0.30\,\mathrm{GeV}$
M_{bc}	$> 5.20\,\mathrm{GeV}/c^2$

The conditions shown in Tab. 6.3 are rather loose considering that the signal regions are at $M_{\mathrm{bc}} > 5.27\,\mathrm{GeV}/c^2$ and $|\Delta E| < 0.05\,\mathrm{GeV}$. This can be seen in Fig. 6.6 in the left plots. This results in so-called signal and non-signal regions (or *sidebands*) within these variables. On the right side of the same figure, background events are shown. Later on, the variables ΔE and M_{bc} will be used to perform a simultaneous discriminating fit and providing a signal and non-signal region is advantageous for the quality (stability) of the fit.

6.6 Best-Candidate Selection

There can only be up to one true signal candidate in an event.[1] Despite this fact, there will be more than one candidate reconstructed in a lot of the events after all of the requirements and methods are applied. This is depicted in Fig. 6.7 where the number of reconstructed candidates per event is shown.

There are a lot of events with more than one reconstructed candidate in a single event. At the end of the reconstruction and preparation, only one candidate is allowed to survive in an event. This best candidate selection will be done by the classifying neural network later on in this analysis.

[1] In principle it is possible that both B mesons of an event decay through the desired decay chain. However, in the case of the electroweak decay $B \to K^*\ell\ell$, the branching fractions are at the order of 10^{-6}. Therefore, it is extremely unlikely ($\approx 1:1\,000\,000$) to have two true signal candidates in a single event and a lot of background can be removed with the assumption that there can only be one.

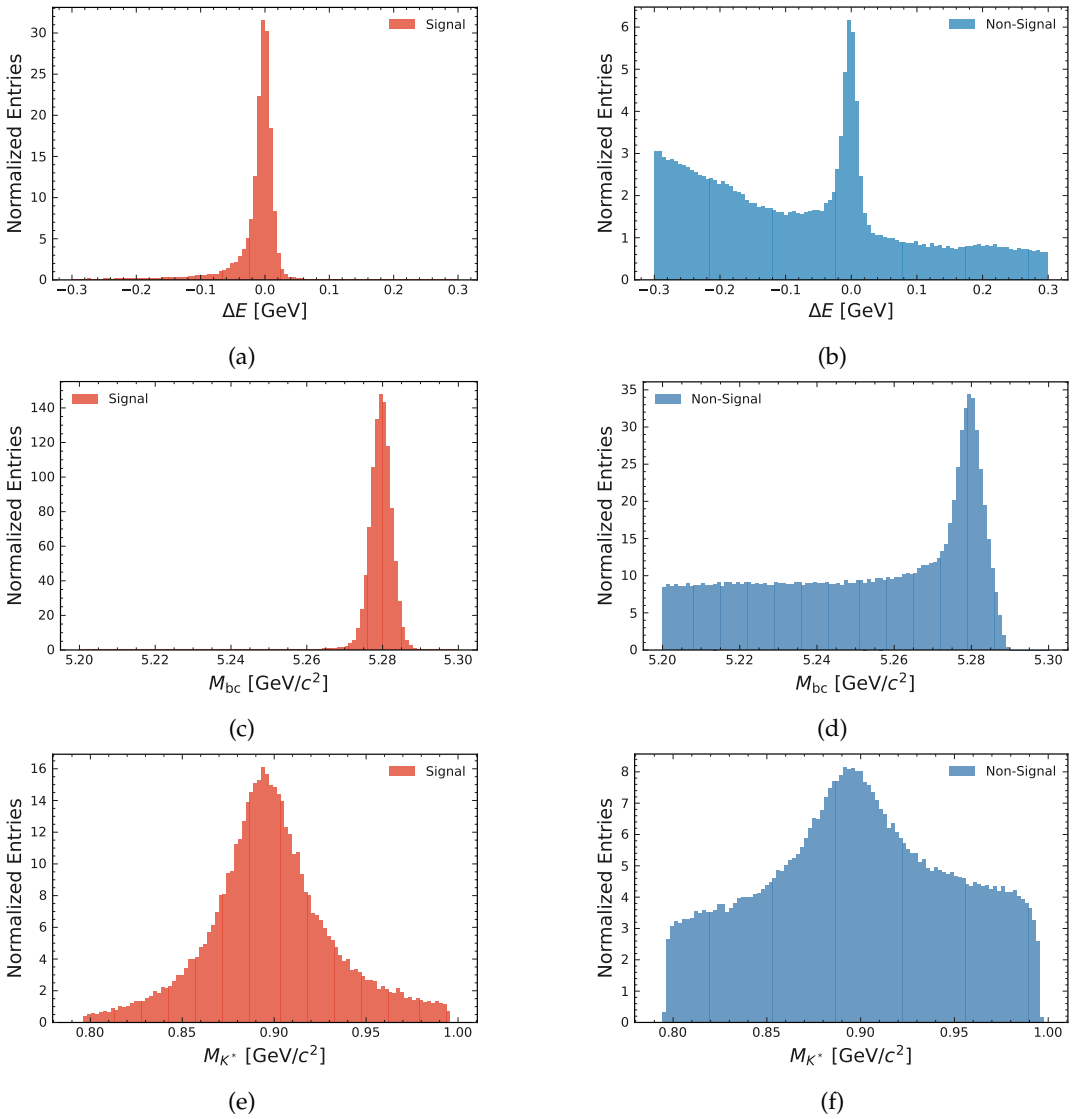


FIGURE 6.6: Variables that have to fulfill requirements during the reconstruction. On the left, signal events with the condition `isSignal` == 1 only are shown (The `isSignal` variable will be introduced in the next section). On the right, the same variable but for non-signal events (background; `isSignal` == 0) are shown. Here, the signal shape also appears. This is due to the J/ψ and $\psi(2S)$ decays (charmonium resonances; pseudo-signal events). In Sec. 7.4 a veto will be introduced in order to suppress these kind of decays. The plots for the variables ΔE and M_{bc} for background events after the veto is applied can be found in Fig. 7.7.

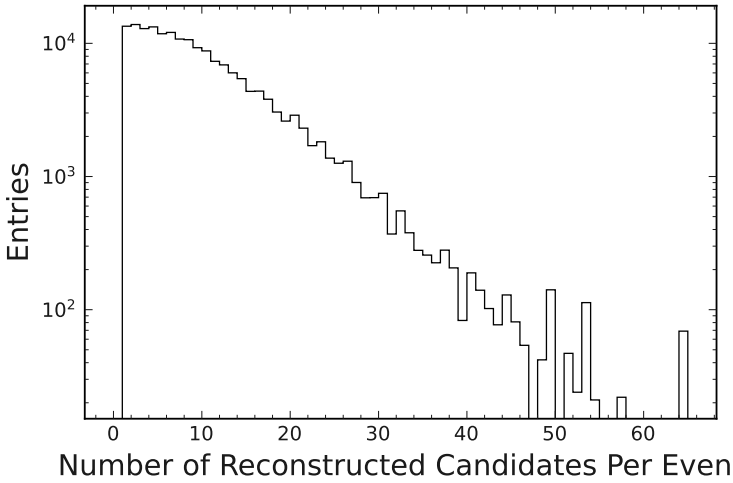


FIGURE 6.7: Number of reconstructed candidates per event for the decay chain $B^+ \to K^{*+}(K^+\pi^0)\ell\ell$. Despite the fact that a maximum of one signal candidate is present in each event, significantly more than one are reconstructed most of the times.

6.7 MC-Truth

With all the introduced methods it is already possible to have a complete working event reconstruction and this is all that can be done for LS1 data. However, it is possible to extract even more information from Monte Carlo. For this, each reconstructed particle has to be matched with its generated particle. This can easily be done with the *Truth Matching* module of basf2.

Once the module is loaded, generator information are accessible. The most important Monte Carlo variable is the so-called *isSignal* variable. This variable is created by comparing the assigned PDG codes at reconstruction and generation. Only if all of these codes within a reconstruction chain match, the variable will be one, otherwise a zero will be assigned. If some particle in the reconstruction chain has no matching generated particle, *isSignal* will also be assigned with a zero. This variable is very important while learning about the properties of the $B \to K^*\ell\ell$ decays.

Furthermore, it is also possible to reconstruct the generated events directly. This way it is possible to gain the true information of all the generated decays within a file, including the ones which were not detected by the detector and are therefore not reconstructed.

Using this generator-level information it is possible to count the number of generated events (n_{gen}) together with the number of reconstructed events (n_{rec}) and an efficiency (ϵ) according to

$$\epsilon = \frac{n_{\text{rec}}}{n_{\text{gen}}} \tag{6.4}$$

can be calculated. Tab. 6.4 shows the event reconstruction efficiency for this analysis for several q^2-regions and each decay channel. These q^2-regions will be introduced with more detail in Sec. 7.4.

TABLE 6.4: Event Reconstruction efficiency for several q^2-regions. These regions will be introduced with more detail in Sec. 7.4.

Channel	Efficiency [%]				
	q^2_1	q^2_2	q^2_3	q^2_4	q^2_{comb}
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	18.0 ± 0.3	20.4 ± 0.3	23.0 ± 0.5	24.2 ± 0.5	20.6 ± 0.2
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	18.2 ± 0.3	19.7 ± 0.3	22.9 ± 0.3	24.8 ± 0.3	20.4 ± 0.1
$B^0 \to K^{*0}(\to K^0_S\pi^0)ee$	8.9 ± 0.6	9.3 ± 0.7	8.3 ± 0.8	10.3 ± 0.7	9.2 ± 0.3
$B^0 \to K^{*0}(\to K^0_S\pi^0)\mu\mu$	7.7 ± 0.5	9.9 ± 0.6	9.5 ± 0.6	10.4 ± 0.6	8.9 ± 0.3
$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	12.0 ± 0.3	12.7 ± 0.4	13.1 ± 0.5	13.6 ± 0.4	12.6 ± 0.2
$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	11.8 ± 0.2	11.2 ± 0.3	12.1 ± 0.3	12.9 ± 0.4	13.7 ± 0.3
$B^+ \to K^{*+}(\to K^+\pi^0)ee$	14.6 ± 0.4	16.8 ± 0.4	19.2 ± 0.6	18.5 ± 0.5	16.6 ± 0.2
$B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$	14.5 ± 0.3	17.1 ± 0.4	18.3 ± 0.4	18.3 ± 0.4	16.2 ± 0.2

Chapter 7

Preparation and Preselection

Better to wait actively than passively.

Stephen L. Carter

This chapter will provide an overview over the techniques that are used in the workflow of this analysis. In order to reduce the differences between real data and simulation, corrections are applied. Then a veto will be introduced to exclude the charmonium resonances which have the same final state particles as the signal decay. In the next step, a neural network is trained and applied to the data in order to reduce the non-signal contributions, which dominate the events. With another neural network the detector-response function will be approximated. In the last section, some correlations between variables are studied. This is an important step since the sWeight method (Sec. 4.3.3), which will be used in the analysis to obtain the true shape of the kinematic variables, requires no correlation between discriminating and target variables. The general workflow of this chapter can be seen in Fig. 7.1.

7.1 Particle-ID Corrections

Simulated data is playing a crucial role throughout this thesis. For this reason, it is important that these simulated events represent the LS1 data as well as possible. Unfortunately, no simulation is perfect but some discrepancies can be reduced. This will be done with the so-called *Particle-ID* (PID) correction. This method tries to correct the tracking part of the simulation by calculating weights for each charged particle leaving a track within the detector. These weights are dependent of the polar angle and the momentum of the charged particle in the lab frame and of the chosen PID selection value. In order to save computing resources, the weights are pre-calculated and stored in lookup tables which can be loaded according to the workflow presented in Fig. 7.1. Here, the first large color box depicts this PID correction workflow. The PID correction is only applied to MC15_ri, MC15_rd and Large MC Signal Files I/II.

In Fig. 7.2 a visual example for the grid of the μ (Fig. 7.2a) and the charged π (Fig. 7.2b) lookup tables is shown. The size of the cells depends on the polar angle θ and the momentum p. These tables are calculated by counting well understood processes in MC and LS1 data and then correcting these numbers accordingly. For example, the electron lookup table is created by studying the Bhabha process ($ee \to ee(\gamma)$) [92] [93].

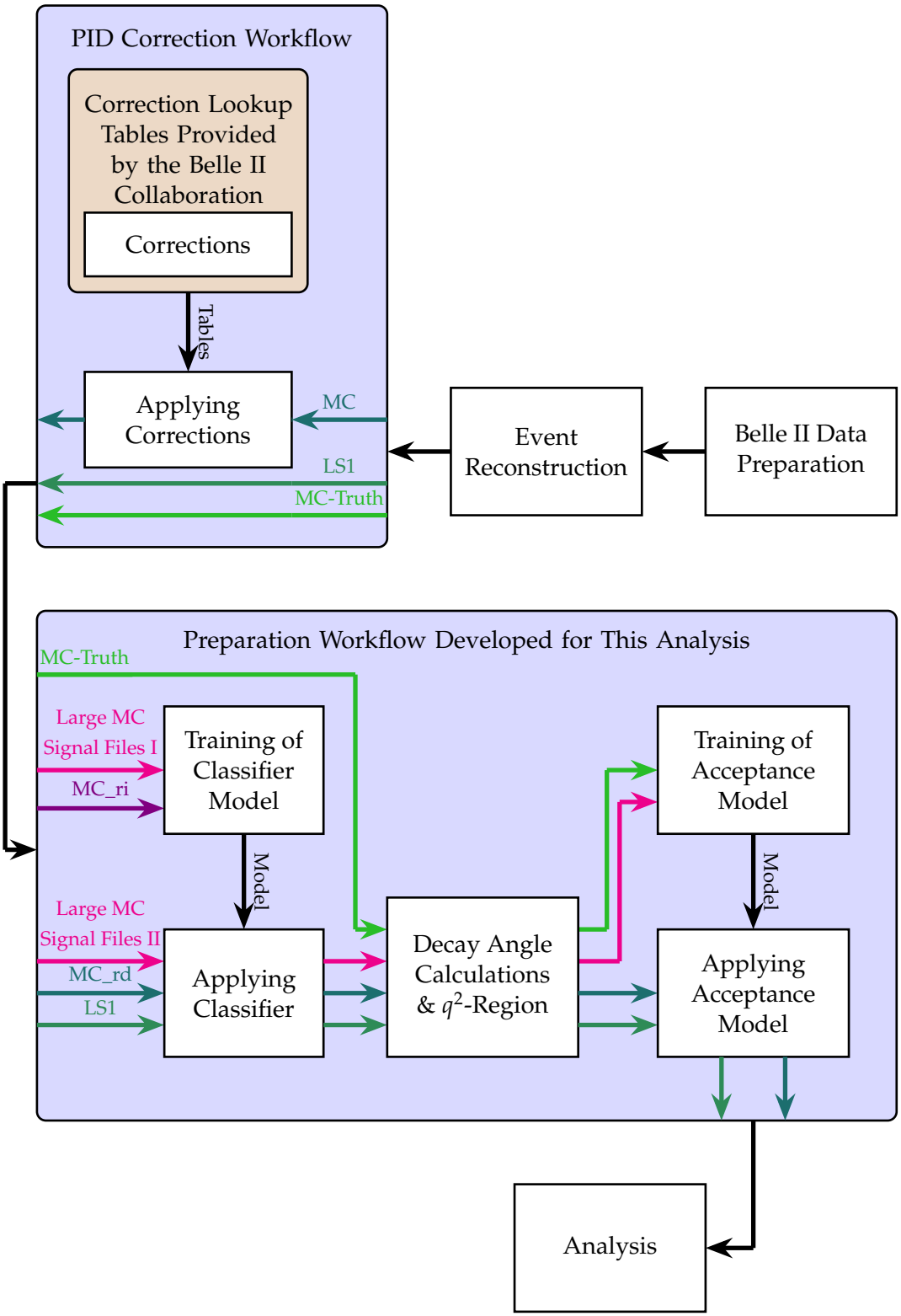
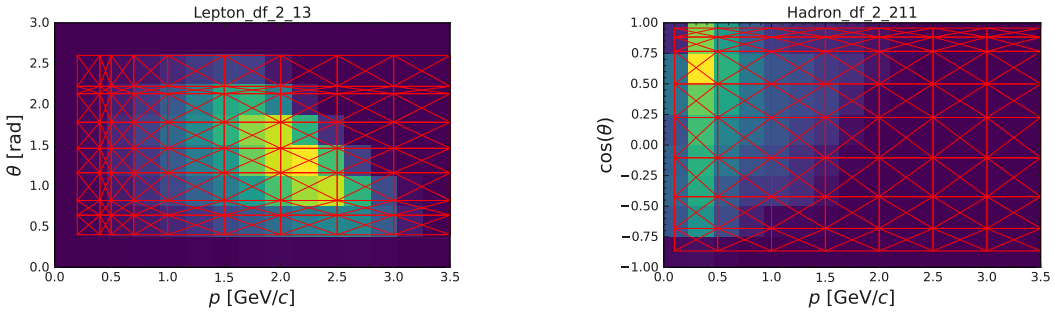


FIGURE 7.1: Workflow for the preparation of the analysis.



(a) (b)

FIGURE 7.2: Example for the granularity of PID-correction lookup tables. The correction table for a μ (a) and charged π (b) is shown.

7.2 Comparing LS1 Data and Monte-Carlo 15

Even after this correction, there will be differences between simulated MC15_rd and LS1 data. For this reason, the variables ΔE and M_{bc} will be shown for both LS1 data and MC15_rd. The pull for this comparison is be calculated by:

$$\text{Pull} = \frac{n_{\text{LS1}} - n_{\text{MC15\_rd}}}{\sqrt{\sigma^2_{\text{LS1}} - \sigma^2_{\text{MC15\_rd}}}} \tag{7.1}$$

where n_{LS1} is the counts and σ_{LS1} is the square root of the counts in each bin of the LS1 histogram and

$$n_{\text{MC15\_rd}} = \sum_{i=1}^{n} w_i \qquad \text{and} \qquad \sigma_{\text{MC15\_rd}} = \sqrt{\sum_{i=i}^{n} w_i^2}. \tag{7.2}$$

Here, w_i is the PID weight of the ith entry and n is the total number of entries in the bin of the MC15_rd histogram.

In Fig. 7.3 the shapes for the discriminating variables ΔE (Fig. 7.3a) and M_{bc} (Fig. 7.3b) are shown alongside the six most important variables for the classifier training. More information about these variables will be presented in Sec. 7.5.3. Overall, the shapes of the LS1 and MC15_rd dataset is very close. This is important because the classifier is trained with simulated MC and it has to be applied on LS1 data as well. This only makes sense if both datasets are comparable. The plots for the other decay channels can be found in the App. D.

7.3 Calculation of the Angular Variables

In order to use the variables which fully describe the decay $B \to K^* \ell\ell$, introduced in Sec. 2.2.1, they have to be calculated first. This will be done after the reconstruction. In order to verify the calculations of the angular variables, a test is performed. For this, events are generated with a flat phase space distribution[1]. This flat distribution is also reconstructed and can be seen in Fig. 7.4.

[1] This can easily be done with EvtGen by adding the argument *PHSP* at the end of the decay in the *decay.dec* file.

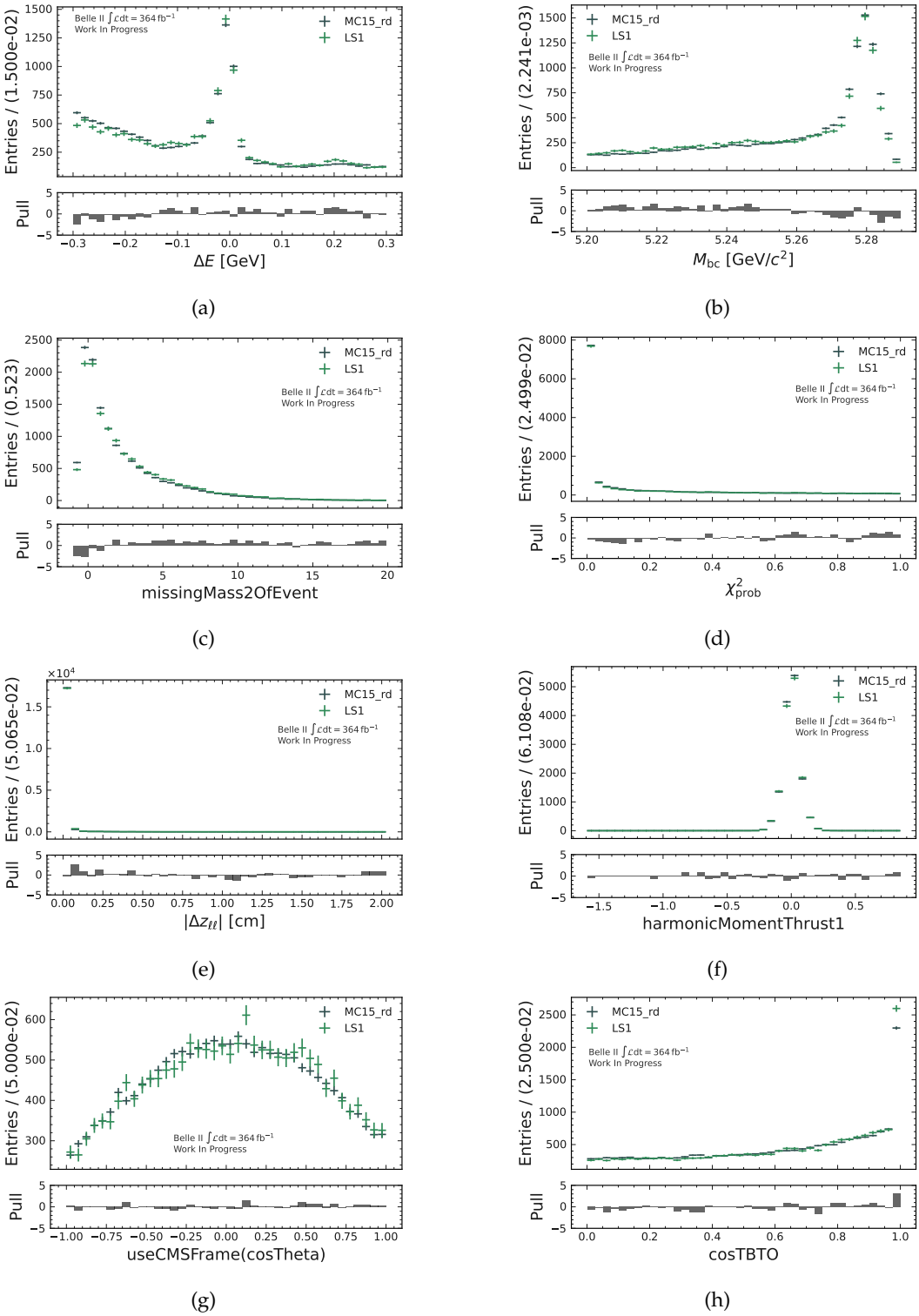
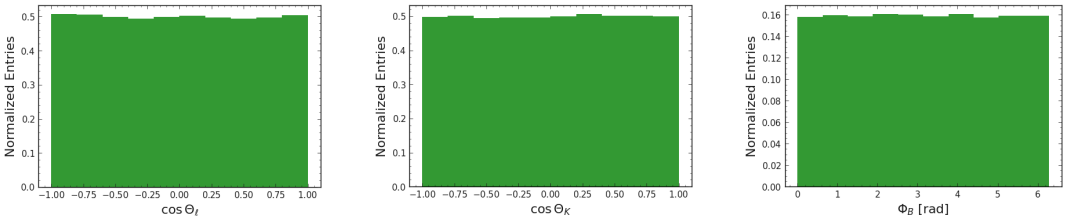


FIGURE 7.3: Comparison between LS1 data and MC15_rd for the decay channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.



(a) (b) (c)

FIGURE 7.4: Distribution for the angles for events generated with a flat phase-space. (a) shows $\cos\theta_\ell$, (b) $\cos\theta_K$ and (c) ϕ_B, respectively.

7.4 J/ψ and $\psi(2S)$ Veto and q^2-Regions

In addition to the $B \to K^*\ell\ell$ decay, there is also the tree level decay $B \to K^* J/\psi(\to \ell\ell)$ which produces the same final state particles. The corresponding Feynman diagram of this decay can be seen in Fig. 7.5.

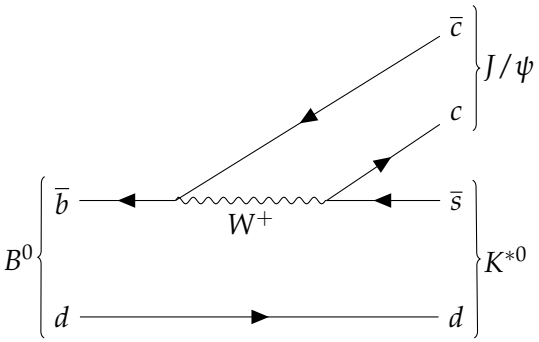


FIGURE 7.5: Tree level Feynman diagram for the decay $B^0 \to K^{*0} J/\psi$. The J/ψ can decay into two leptons. Therefore, the final state particles will be the same as for the FCNC $B^0 \to K^{*0}\ell\ell$ making this a so-called pseudo signal and a control channel.

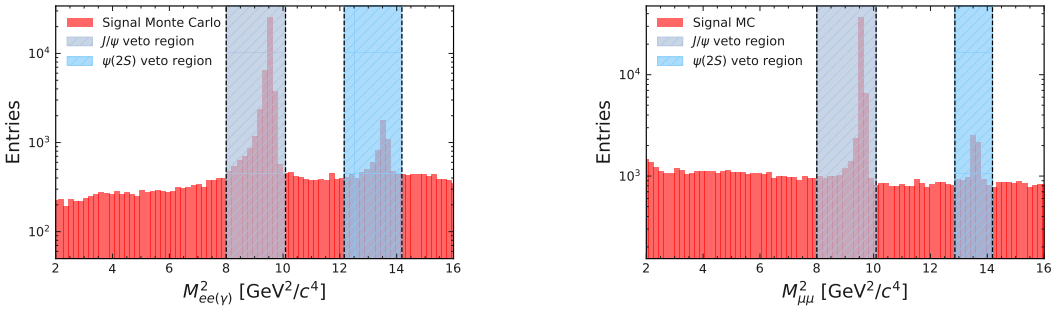
This pseudosignal has to be removed for the analysis since it is impossible to distinguish it from $B \to K^*\ell\ell$ events. Fortunately, events containing a charmonium resonance can be removed with veto regions on the invariant mass of the lepton system corresponding to the mass of the J/ψ and $\psi(2S)$. As a result, the veto regions are:

$$8.00\,\mathrm{GeV}^2/c^4 < M^2_{\ell\ell} < (M_{J/\psi} + 0.08)^2\mathrm{GeV}^2/c^4 \tag{7.3}$$

$$-0.20\,\mathrm{GeV}/c^2 < M_{ee(\gamma)} - M_{\psi(2S)} < 0.08\,\mathrm{GeV}/c^2 \tag{7.4}$$

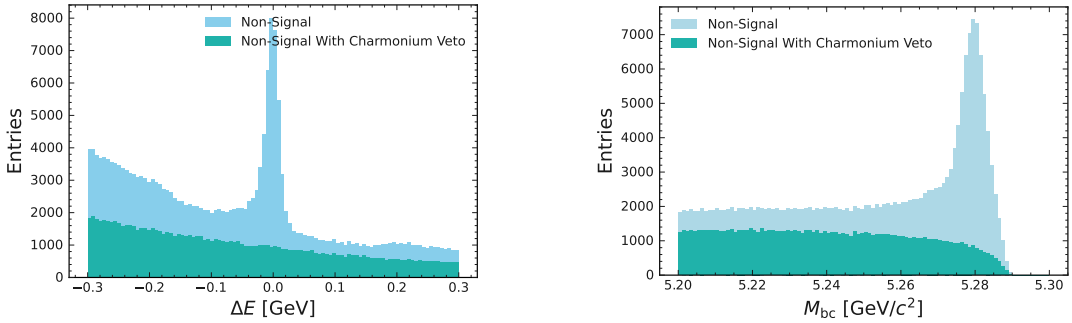
$$-0.10\,\mathrm{GeV}/c^2 < M_{\mu\mu} - M_{\psi(2S)} < 0.08\,\mathrm{GeV}/c^2 \tag{7.5}$$

Only the range of the veto region for the $\psi(2S)$ differ between electrons and muons. This is due to the q^2-regions which will be introduced in the next section (Sec. 7.4) and the fact that electrons are able to perform bremsstrahlung which results in a worse energy resolution compared to muons. These veto regions can be seen in Fig. 7.6a for electrons and Fig. 7.6b for muons, respectively. Their effect on the variables ΔE and M_{bc} can be found in Fig. 7.7. The peaks caused by the pseudo signal and which were visible in Fig. 6.6 disappear.



(a) (b)

FIGURE 7.6: Depiction of the veto regions for the J/ψ and $\psi(2S)$ in (a) for $\ell\ell \hat{=} ee$ and (b) for $\ell\ell \hat{=} \mu\mu$. The $\psi(2S)$ veto region is larger for the electrons due to their poorer energy resolution in the detector caused by bremsstrahlungs photons.



(a) (b)

FIGURE 7.7: The shape for the variables ΔE (a) and M_{bc} (b) are shown for non-signal events before (light blue) and after (teal) the charmonium veto. As a result of the veto, the peak in the signal region, appears to vanish completely. The events shown are from the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ decay channel.

Due to the low expected statistics of this analysis, it is not possible to use q^2 as a fit variable in the final fit. Instead the events will be divided into four analysis regions plus the two already introduced veto regions. The ranges for the former four regions follow [26] and are listed in Tab. 7.1. A plot of the q^2 distribution for generated MC signal can be seen in Fig. 7.8.

7.5 Machine Learning - Classification

In Fig. 7.9 the variables ΔE and M_{bc} of the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ for the MC15_rd before the signal/non-signal classification are shown. These figures contain three different contributions:

Background These are reconstructed candidates with no signal candidate present in the event. These events include continuum ($e^+e^- \to q\bar{q}$; $q \in u, d, c, s$), B pairs without a signal event and τ pairs ($e^+e^- \to \tau^+\tau^-$).

TABLE 7.1: Chosen q^2-ranges. The J/ψ- and $\psi(2S)$-veto regions follow Eqs. 7.3 to 7.5.

q^2-region	Range [GeV$^2/c^4$]
q^2_1	[0.10, 4.00]
q^2_2	[4.00, 8.00]
q^2_3	[Upper J/ψ-Veto Region, Lower $\psi(2S)$-Veto Region]
q^2_4	[Upper $\psi(2S)$-Veto Region, 19.00]
$q^2_{J/\psi}$	[Lower J/ψ-Veto Region, Upper J/ψ-Veto Region]
$q^2_{\psi(2S)}$	[Lower $\psi(2S)$-Veto Region, Upper $\psi(2S)$-Veto Region]
q^2_{comb}	$q^2_1 + q^2_2 + q^2_3 + q^2_4$

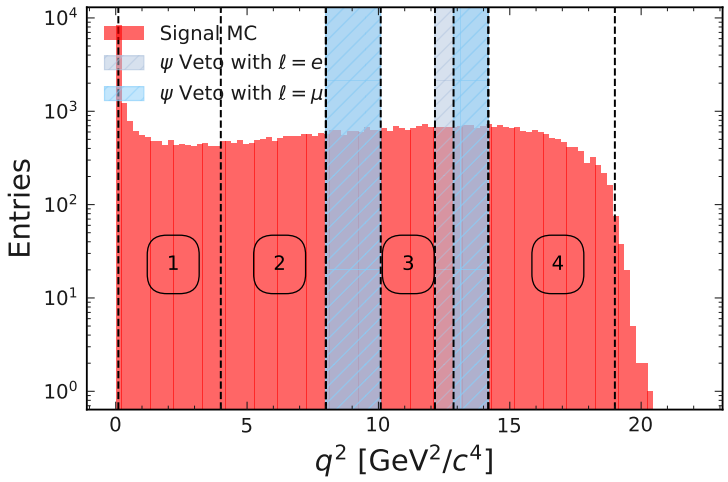
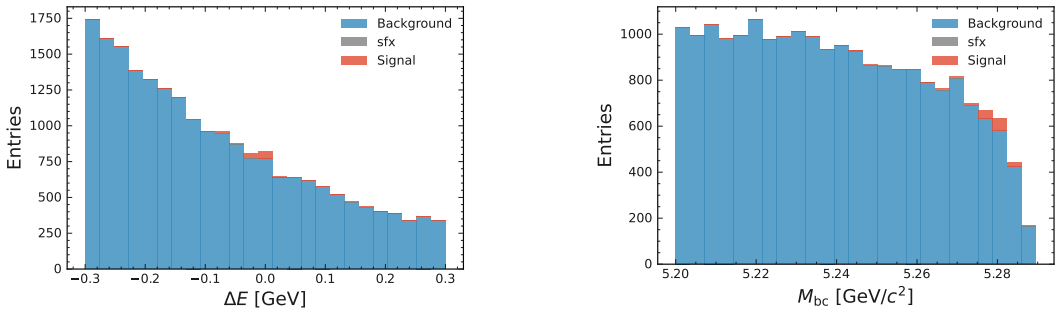


FIGURE 7.8: Generated signal q^2 distribution and the regions used for analysis.



(a) (b)

FIGURE 7.9: Shapes of the ΔE and M_{bc} variables before the classifier for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ decay channel in MC15_rd data. Blue are reconstructed candidates with no signal candidate present in the event. Gray are badly reconstructed candidates with a signal candidate present in the event. Red are correctly reconstructed signal events. In Fig. 7.10 the variables after the classifier can be seen.

sfx These are candidates with a signal candidate present in the event. However, something went wrong during reconstruction and the candidate is considered non-signal. These candidates are also called self-cross feed.

Signal These are correctly reconstructed signal events.

In both variables the signal peak is barely visible. As a result, a way has to be found to increase the signal to non-signal ratio. This will be done using a classifier based on machine learning.

This section will provide a brief overview over the neural network that was used to classify each reconstructed event. It will start with the features used, continue with the structure of the final models and end with a brief discussion about the performance of the chosen neural networks.

7.5.1 Variables

Using the feature importance method of BDTs (Sec. 5.9.1), the variables used for classification are determined.[2] Here, a complete description of these features will be provided. If the feature was not used in all classifiers, it will be stated at the end of the description and additional information will be provided.

sphericity Sphericity of an event, defined as the linear combination of the sphericity eigenvalues λ_i: $S = {}^3/_2(\lambda_2 + \lambda_3)$.

visibleEnergyOfEventCMS Visible energy in the detector in the center-of-mass frame.

$|\Delta z_{\ell\ell}|$ Difference between the dz position of the leptons.

$|\Delta z_{K\pi}|$ Difference between the dz position of the K and the π.

$M(K^*)$ Mass of the reconstructed K^*.

cos **TBTO** Cosine of the angle between the thrust axis of the B meson and the thrust axis of the Rest-Of-Event.

RestOfEventExtraE Energy of the Rest-Of-Event.

missingMass2OfEvent Missing mass squared of the event.

$\cos\theta_{\mathbf{CMS}}$ Cosine of the θ angle of the B meson in the center-of-mass system.

foxWolframR(*i*) Four FoxWolfram moments according to Sec. 6.4.1.

KSFW A total of 16 Super FoxWolfram moments according to Sec. 6.4.1.

CleoConeCS(*i*) ith Cleo cone from the continuum suppression. Cleo cones are the shapes of the cones created by charged particles.

harmonicMomentThrust(*i*) Squared ratio between the momentum of the ith particle and the total momentum of all particles.

$\chi_{\mathbf{prob}}$ χ_{prob} value of the vertex fit for the B meson.

max($\chi_{\mathbf{prob}}$) Maximum χ_{prob} for all particles in the reconstruction chain.

[2] This is done by writing a fully functional BDT classifier. The performance of the classifier will be compared to the performance of the neural network classifier in Sec. 7.6.

min(χ_{prob}) Minimum χ_{prob} for all particles in the reconstruction chain.

p_t(**final state**) Transverse momentum of the final state particles of the decay (K, π and $\ell\ell$).

$\sum p_t$(**final state**) Sum of the transverse momentum of the final state particles of the decay (K, π and $\ell\ell$).

ArmenterosDaughter1Qt(K^*) Transverse momentum of the first daughter with respect to the K^*.

daughterAngle(π) Angle between the both daughters of the π. Only used in the classifying networks for the channels containing a π^0.

DifferenceOfPhiDaughterAngle(π) Difference between in the ϕ angles of the daughters of the π. Only used in the classifying networks for the channels containing a π^0.

$\log \sqrt{\Delta x_{\ell\ell}^2 + \Delta y_{\ell\ell}^2}$ Logarithm of the square root of the sum of the Δx and Δy position squared of the B meson (Fig. E.2).

$\cos p \angle v$ Cosine of the angle between the momentum p and the vertex vector v. The vertex vector is the vector connecting the IP and fitted vertex position of the particle.

$\ell_\Omega^+ - \ell_\Omega^-$ Difference between the curvatures of the lepton tracks (Fig. E.4a).

$\ell_\Omega^+ + \ell_\Omega^-$ Sum of the curvatures of the lepton tracks (Fig. E.4b).

7.5.2 Setup

In general, the underlying structure of each of the classifying neural networks is very similar. Solely, the number of nodes in each layer differs. Each network consists of three fully connected hidden layers. The number of nodes is determined with a gradient randomized search. The result of this search can be seen in Tab. 7.2.

TABLE 7.2: Chosen number of nodes for each neural network.

Channel	1st Layer	2nd Layer	3rd Layer
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	53	34	91
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	51	37	66
$B^0 \to K^{*0}(\to K_S^0\pi^0)ee$	56	16	49
$B^0 \to K^{*0}(\to K_S^0\pi^0)\mu\mu$	62	48	56
$B^+ \to K^{*+}(\to K_S^0\pi^+)ee$	37	136	104
$B^+ \to K^{*+}(\to K_S^0\pi^+)\mu\mu$	136	65	43
$B^+ \to K^{*+}(\to K^+\pi^0)ee$	77	34	56
$B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$	62	46	56

The activation function of the input layer is *Linear*. For this reason, all variables are scaled to an interval of 0 and 1 before the network. All of the hidden layers, including the output layer, have a *sigmoid* activation function and as a result, after each layer a batch normalization takes place. This brings the weights to values around zero with a standard deviation of one. Finally, dropout with a value of 0.35 is

added to each of the hidden layers. The loss function of these networks is the *binary cross-entropy* loss function

$$l(y) = -\frac{1}{m}\sum_{i=1}^{m} y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log (1 - \hat{y}_i) \tag{7.6}$$

and the *adam* optimizer is used. The starting learning rate is $lr = 0.001$ which gets reduced constantly at the end of each epoch. After 100, 300, 700 and 1500 epochs a snapshot of the whole network is taken and the learning rate is increased again to escape the local minimum. In order to increase the stability of the neural network, a total of 100 of these networks are trained in parallel.

In the end, each snapshot contributes to the final output of the mayor neural network. However, the weight of the output depends on the snapshot. For example, the first snapshot has a lower weight compared to the last, since this snapshot should in principle be better overall[3]. All networks within the same snapshot stage are assigned the same weight.

7.5.3 Training

The classifying networks are trained with the MC15_ri datasets. For this the complete *run independent* dataset, as introduced in Sec. 3.13, was used with an exception of the mixed signal and charged signal files (*Large MC Signal File I*). Here, only half of all the available data was used during training. This was done to make sure, that there are enough signal data that did not see the classifying training left for the training of the detector acceptance approximation neural network (Sec. 7.7). The dataset is split into a training and a validation dataset with a ratio of $4:1$. In addition to these two datasets, a *Testing* dataset is added, which corresponds to MC15_rd. This dataset is not involved during the training of the network. However, after the training is done, the binary classifying output and the ROC curve is calculated for this more realistic MC as well.

Since there can only be one signal candidate in each event, the candidate with the highest neural network prediction is considered the best candidate and every other candidate in the event is treated as non-signal.

For all datasets it was made sure that they do not include J/ψ and $\psi(2S)$ events. This was done using MC truth information. This can not be done in reality, however removing charmonium events makes sure that the network will not want to start to calculate q^2 by its own as for some q^2-region the ratio between signal and background drops significantly. Also since the $B \to K^*\psi$ decays has the same final state particles but is considered background, the neural network could get *confused* and the overall performance could drop.

In Fig. 7.10a the output for signal/non-signal events used as training, validation and testing datasets for the classifying model of the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel is shown. For each dataset signal is nicely separated from most of the non-signal. The pink line indicates the threshold value determined by the ROC curve which is presented in Fig. 7.10b. This threshold is in the rather flat area of Fig. 7.10a indicating a stable classification. In Tab. 7.3 information about the classifying networks for each decay channel can be found. In Fig. 7.10c and 7.10d the variables ΔE and M_{bc} after the classifier for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel can be seen, respectively. In both variables a clear signal peak at $\Delta E \approx 0$ and $M_{\mathrm{bc}} \approx 5.28$ can be seen.

[3] It was trained for a longer time and the learning rate became smaller after escaping several local minima.

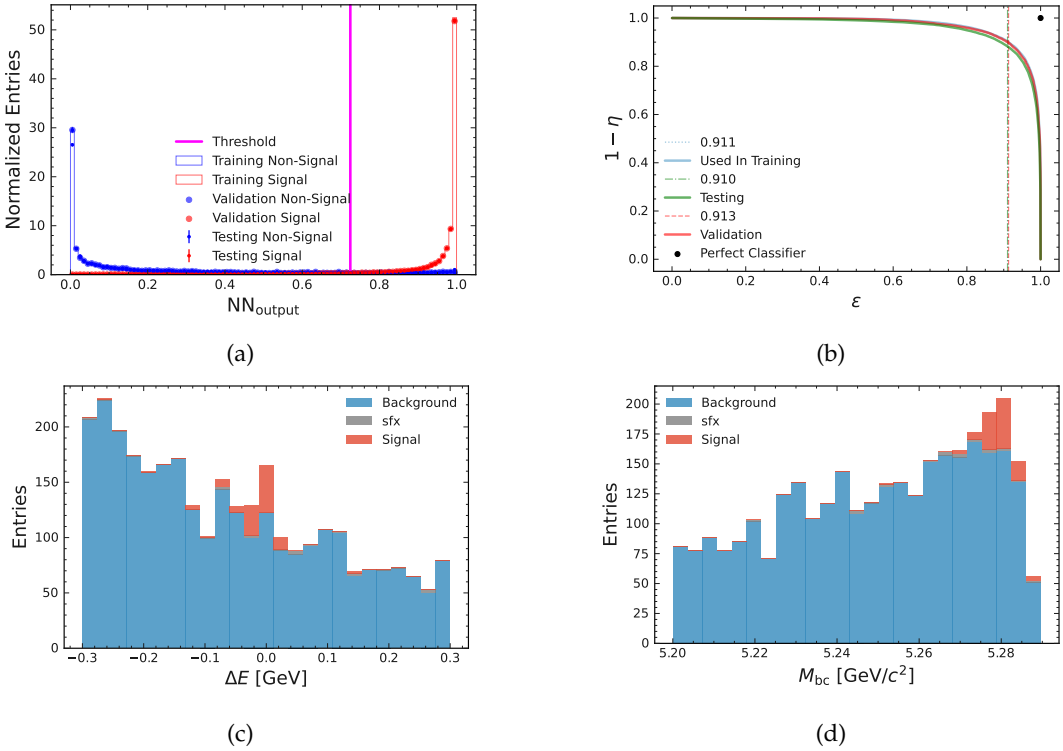


FIGURE 7.10: Classifying model for the channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend. In (c) and (d) the variables ΔE and M_{bc} can be seen after the classifier. Now, a clear signal peak (red) can be observed.

TABLE 7.3: AUC-Values and classifier thresholds for the classifying neural networks.

Channel	AUC	Threshold
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	0.971	0.735
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	0.964	0.718
$B^0 \to K^{*0}(\to K^0_S\pi^0)ee$	0.975	0.104
$B^0 \to K^{*0}(\to K^0_S\pi^0)\mu\mu$	0.970	0.063
$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	0.956	0.715
$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	0.950	0.689
$B^+ \to K^{*+}(\to K^+\pi^0)ee$	0.980	0.428
$B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$	0.973	0.262

7.5.4 Performance

The performance for each decay channel is shown in Tab. 7.4. In this table, the purity (ξ) and the efficiency (ϵ) in each q^2-region is shown. In order to get these number, the datasets are restricted on the signal region ($M_{\text{bc}} > 5.27\,\text{GeV}/c^2$ and $|\Delta E| < 0.05\,\text{GeV}$).

TABLE 7.4: Total efficiencies and purities in percent in the signal region ($M_{\mathrm{bc}} > 5.27\,\mathrm{GeV}/c^2$ and $|\Delta E| < 0.05\,\mathrm{GeV}$) for all channels after the network selection is applied.

Channel		q^2_1	q^2_2	q^2_3	q^2_4	q^2_{comb}
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	ϵ	14.2 ± 0.3	15.6 ± 0.3	17.4 ± 0.4	18.7 ± 0.3	15.9 ± 0.2
	ξ	50 ± 13	43 ± 10	46 ± 15	57 ± 13	51 ± 6
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	ϵ	14.9 ± 0.2	17.0 ± 0.3	20.0 ± 0.3	21.4 ± 0.3	17.4 ± 0.1
	ξ	30 ± 5	38 ± 7	45 ± 9	45 ± 7	38 ± 3
$B^0 \to K^{*0}(\to K^0_S\pi^0)ee$	ϵ	5.8 ± 0.5	5.6 ± 0.5	5.3 ± 0.5	6.5 ± 0.6	5.9 ± 0.3
	ξ	8 ± 5	2 ± 2	9 ± 5	10 ± 7	6 ± 2
$B^0 \to K^{*0}(\to K^0_S\pi^0)\mu\mu$	ϵ	5.9 ± 0.4	7.9 ± 0.5	7.1 ± 0.5	8.0 ± 0.5	6.8 ± 0.2
	ξ	2 ± 1	3 ± 2	1 ± 1	2 ± 2	2 ± 1
$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	ϵ	9.4 ± 0.3	9.8 ± 0.3	10.1 ± 0.4	10.8 ± 0.4	9.9 ± 0.2
	ξ	58 ± 18	41 ± 14	31 ± 13	72 ± 20	53 ± 8
$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	ϵ	9.1 ± 0.3	10.3 ± 0.3	11.2 ± 0.3	11.9 ± 0.3	10.1 ± 0.2
	ξ	17 ± 5	22 ± 8	24 ± 9	24 ± 9	24 ± 4
$B^+ \to K^{*+}(\to K^+\pi^0)ee$	ϵ	10.0 ± 0.3	11.4 ± 0.4	13.1 ± 0.5	12.5 ± 0.4	11.4 ± 0.2
	ξ	16 ± 4	7 ± 2	15 ± 4	16 ± 4	13 ± 2
$B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$	ϵ	10.6 ± 0.3	13.1 ± 0.3	14.5 ± 0.4	13.7 ± 0.3	12.3 ± 0.2
	ξ	4 ± 1	6 ± 2	7 ± 2	5 ± 1	5 ± 1

Tab. 7.5 shows a direct comparison between Belle and this analysis. The Belle values are from [26]. While the studied channels are the same at Belle and this analysis at hand, the classifier at Belle had some different requirements. The main difference is that ΔE was a part of the features used in training and as a result, it can not be used as a discriminating variable. Also, the signal region for channels with $\ell = e$ was wider at Belle in ΔE ($-0.10\,\mathrm{GeV} < \Delta E < 0.05\,\mathrm{GeV}$). Overall, the performance of the classifier introduced in this analysis seems to outperform the ones used at Belle.

In Tab. 7.6, the expected amount of signal events for an integrated luminosity of $\int \mathcal{L}\,\mathrm{d}t = 364\,\mathrm{fb}^{-1}$ in every q^2-region is shown. Especially the low number for the $B^0 \to K^{*0}(\to K^0_S\pi^0)\ell\ell$ channels is interesting. However, this is expected once the decay chain for this channel is considered. Due to isospin, only 1/3 of all K^{*0} decay into $K^0\pi^0$. Half of all K^0 decay further into K^0_S and the other half decay into K^0_L which

TABLE 7.5: Comparison between the performance of the classifier of Belle ([26], p. 67) and this analysis. In contrast to this analysis at hand the signal region for the channels with $\ell = e$ at Belle was $-0.10\,\text{GeV} < \Delta E < 0.05\,\text{GeV}$. The ΔE variable was also part of the features within the classifier training, as a result it can not be used as a discriminating variable at Belle. The M_{bc} signal region is the same and neither analyses use it in the training.

	Channel	Efficiency [%]	Purity[%]
Belle	$B^0 \to K^{*0}(\to K^+\pi^-)ee$	5.37	47 ± 5
	$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	14.28	37 ± 3
	$B^+ \to K^{*+}ee$	2.94	17 ± 3
	$B^+ \to K^{*+}\mu\mu$	3.54	18 ± 3
This Analysis	$B^0 \to K^{*0}(\to K^+\pi^-)ee$	15.9 ± 0.2	51 ± 6
	$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	17.4 ± 0.1	38 ± 3
	$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	9.9 ± 0.2	53 ± 8
	$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	10.1 ± 0.2	24 ± 4
	$B^+ \to K^{*+}(\to K^+\pi^0)ee$	11.4 ± 0.2	13 ± 2
	$B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$	12.3 ± 0.2	5 ± 1

TABLE 7.6: Expected number of signal events for a luminosity of $364\,\text{fb}^{-1}$ after reconstruction and the neural network classifier. These numbers are determined with the MC15_rd dataset with a luminosity of ($\int \mathcal{L}_{\text{LS1}}\,\text{d}t = \int \mathcal{L}_{\text{MC15\_rd}}/4\,\text{d}t = 4 \cdot 364\,\text{fb}^{-1}/4 = 364\,\text{fb}^{-1}$).

Channel	q_1^2	q_2^2	q_3^2	q_4^2
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	8.7 ± 0.6	8.0 ± 0.7	5.7 ± 0.6	8.8 ± 0.7
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	10.7 ± 0.7	10.8 ± 0.7	11.3 ± 0.8	13.3 ± 0.9
$B^0 \to K^{*0}(\to K^0_S\pi^0)ee$	0.6 ± 0.2	0.2 ± 0.2	0.2 ± 0.2	0.5 ± 0.2
$B^0 \to K^{*0}(\to K^0_S\pi^0)\mu\mu$	0.8 ± 0.2	0.7 ± 0.2	0.7 ± 0.2	0.9 ± 0.2
$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	4.4 ± 0.4	3.6 ± 0.4	2.7 ± 0.3	4.5 ± 0.4
$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	3.6 ± 0.3	2.6 ± 0.3	2.9 ± 0.3	3.5 ± 0.4
$B^+ \to K^{*+}(\to K^+\pi^0)ee$	4.9 ± 0.4	5.0 ± 0.5	3.3 ± 0.4	6.2 ± 0.6
$B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$	4.0 ± 0.4	4.0 ± 0.4	4.2 ± 0.4	5.3 ± 0.5

are very difficult to detect, even at Belle II. For this reason, they are neglected in this analysis. The K^0_S decays about $1/3$ of the time to $\pi^0\pi^0$ which decay into a total of four γ. It would be very difficult to reconstruct two π^0 in an event and the amount of expected background would be huge. As a result, this channel is also neglected. The other $2/3$ of the K^0_S decay channel goes into $\pi^+\pi^-$. Therefore, only a small fraction ($1/3 \cdot 1/2 \cdot 2/3 = 1/9$) is even considered for reconstruction. In addition to this low branching fraction comes the 50% reconstruction efficiency of the π^0 making it a very difficult decay channel to reconstruct. More details about the branching fractions within the $B \to K^*\ell\ell$ decays can be found in Fig. 7.11. Considering the very low expected yields for the channels $B^0 \to K^{*0}(\to K^0_S\pi^0)ee$ and $B^0 \to K^{*0}(\to K^0_S\pi^0)\mu\mu$, these channels will not be studied further in the context of this thesis.

7.6 Neural Network vs. Boosted Decision Tree

In Sec. 5.9, Boosted Decision Trees were introduced as an alternative classifying method. Since they are also able to perform a signal/non-signal classification, for

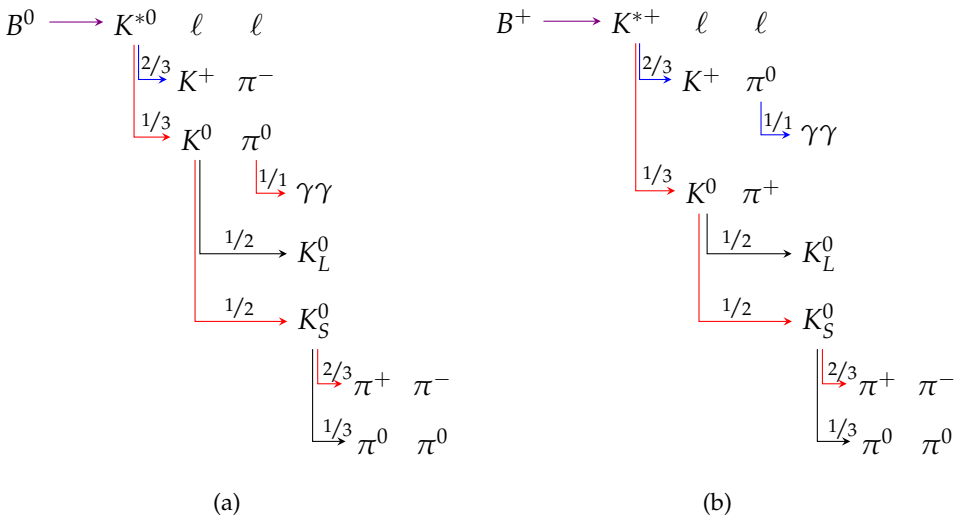


FIGURE 7.11: Dominant decay chains with approximated probabilities for the $B^0 \to K^{*0}\ell\ell$ (a) and $B^+ \to K^{*+}\ell\ell$ decays (b). The red (blue) arrows represents the decay chains containing a K^0 (K^+). The π^0 reconstruction efficiency at Belle II is $\sim 50\%$. Decays with a black arrow are not reconstructed for this analysis.

both tools (BDT and NN), the performance can be calculated, compared and the better method will be chosen. For this brief study, only the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel is investigated and the efficiency and purity after both classifying methods as well as the AUC of the ROC curves are compared. The ROC and signal/non-signal separation plots for the BDT can be found in the appendix in Fig. F.8. The results can be found in Tab. 7.7.

TABLE 7.7: Performance comparison between the BDT and NN for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_{comb}-region.

	Total Efficiency [%]	Purity [%]	AUC
BDT	16.4 ± 0.2	34 ± 6	0.966
NN	15.9 ± 0.2	51 ± 6	0.971

As presented in this table, the dataset after the BDT has a higher total efficiency. However, the purity of this dataset is worse compared to the NN. In order to compare both classifiers, the threshold of the BDT prediction is chosen in such a way, that the relative efficiency of both classifiers is the same. This results in a purity of $43 \pm 6\%$ for the BDT. As a result, the neural network classifier is considered to perform better than the BDT. This can also be seen in the AUC value which is a little bit higher for the NN compared to the BDT. Therefore, and for the sake of consistency, the neural network was chosen as a classifier and for function approximation[4] (Sec. 5.7).

[4]It is not possible to use a BDT as a function approximation since the output of the BDT is not smooth but rather steps. This is not desired in an acceptance function.

7.7 Detector Acceptance Corrections

In Ch. 3 the Belle II detector with its subdetectors was introduced. It is clear that this machine does not work perfectly and the reconstruction efficiency for the particles depends on their angles and momenta (*acceptance*).[5] These inconsistencies need to be corrected in order to perform an analysis which will depend on the exact number, position and momenta of the reconstructed particles. To perform these acceptance corrections, an approximation with a neural network according to Sec. 5.7 is developed. (Function Approximation)

7.7.1 Variables to Correct

According to theory (Sec. 2.2.1) the decay $B \to K^* \ell\ell$ can be fully described with only five variables. Due to the limitations in the amount of available data, the $M_{K\pi}$ variable will be considered a fixed value. In addition to that, the q^2 variable will only be divided in the six already introduced regions, leaving only three angular variables: ϕ_B, $\cos\theta_\ell$ and $\cos\theta_K$.

ϕ_B Acceptance Corrections

Studying the properties of ϕ_B, it is found that this variable is uncorrelated to $\cos\theta_\ell$ and $\cos\theta_K$ and the reconstruction efficiency is rather flat, this can be seen in Fig. 7.12 and 7.13, respectively. As a consequence, this variable will not be used in the training for the correction since the corrections done to $\cos\theta_\ell$ and $\cos\theta_K$ will also bring ϕ_B back to the generated form. This will be described in the next section. The correlation and efficiency plots for the other decay channels can be found in the App. G.1 and App. G.2, respectively.

$\cos\theta_\ell$ and $\cos\theta_K$ Acceptance Corrections

In order to perform the detector acceptance corrections, a two dimensional function approximation with keras-TensorFlow will be done. For this, the efficiencies as functions of the two angular variables, $\cos\theta_\ell$ and $\cos\theta_K$, are used in the training. The final model will be able to return the expected efficiency as a function of both angular variables. In Fig. 7.13 the efficiencies as a projection for the three angles are depicted. These shapes of the efficiencies also depend on the q^2-region. For this reason, each q^2-region will get an individually trained neural network.

7.7.2 Structure of the Neural Network and Training - Acceptance

For each q^2-region of the remaining $B \to K^* \ell\ell$ decays, a separate neural network is trained. However, the structure of each network is be the same. Solely the input 2D efficiency histogram changes. For a given number of bins, the binning of this two dimensional histogram is determined using generated events and the method introduced in Sec. 4.2.2. This method ensures, that each bin in the 2-dimensional plot has roughly the same amount of generated events and, as a consequence, the calculated efficiencies are more stable. Otherwise, a bin with zero generated events could be created and it would be impossible to calculate the efficiency. This method is applied to several different amounts of total bins in order to increase the total

[5] This becomes obvious considering that even though Belle II covers a large spatial area, there are no detectors at very small and large polar angles (the polar angle acceptance is $17° < \theta < 150°$). Hence, no particle can be reconstructed in these areas.

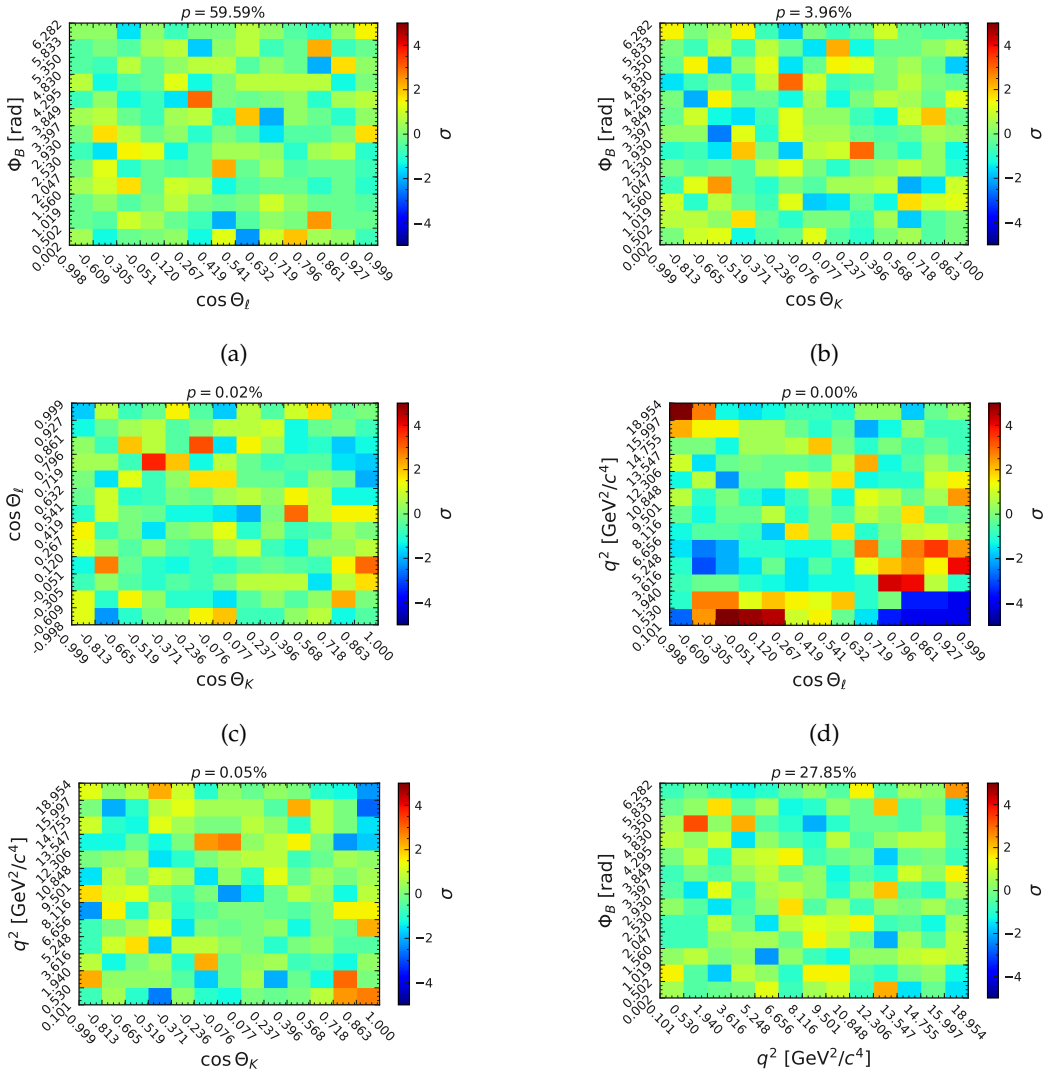


FIGURE 7.12: Correlations between four decay variables for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$. There is a correlation between $\cos\theta_\ell$ and q^2. However, this correlation is expected since both of these variables depend on both leptons in the decay.

number of entries of efficiency points which can be provided to the training of the network.

For these neural networks, feature engineering is done as well in order to give it more structural information and to help during the training. The features used in the training are:

1. $f_1(\theta_\ell) = \cos\theta_\ell$
2. $f_2(\theta_K) = \cos\theta_K$
3. $f_3(\theta_\ell) = \cos^2\theta_\ell$
4. $f_4(\theta_K) = \cos^3\theta_K$
5. $f_5(\theta_\ell, \theta_K) = \cos\theta_\ell \cdot \cos\theta_K$

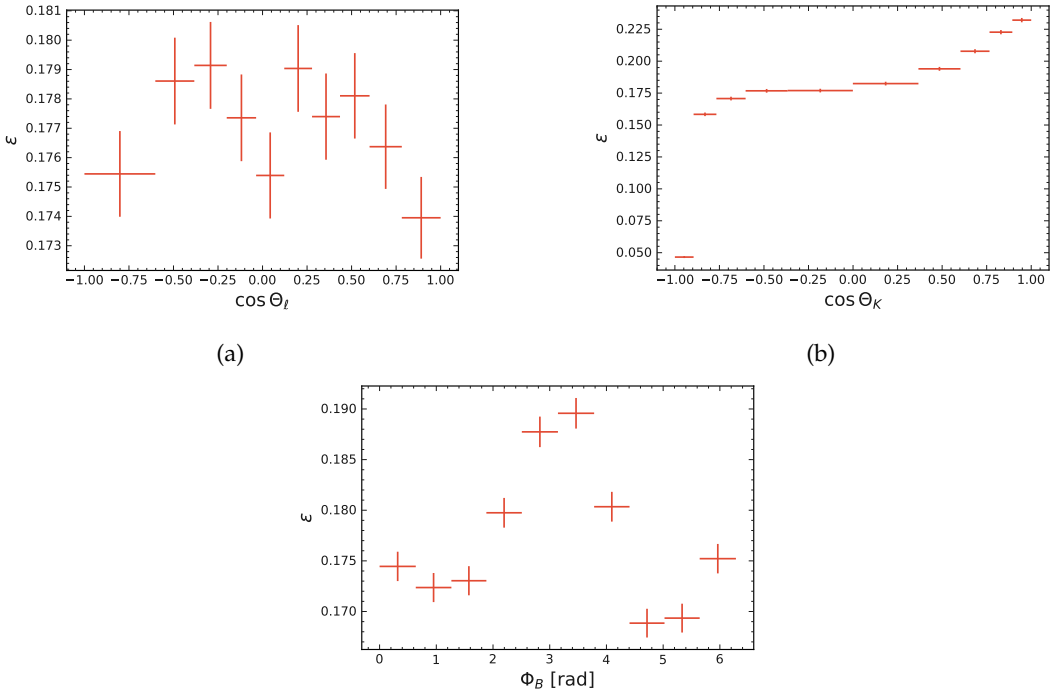


FIGURE 7.13: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q_2^2-region.

6. $f_6(\theta_\ell, \theta_K) = \theta_\ell \cdot \theta_K$
7. $f_7(\theta_\ell, \theta_K) = \sqrt{\cos^2\theta_\ell + \cos^2\theta_K}$

This function approximation neural network uses the optimizer *adam*. The network itself consists of two hidden layers. The first hidden layer has 16 nodes and a *ReLU* activation function. The activation function of the following layer is *sigmoid* with 16 nodes as well. Between each layer, a batch normalization takes place. The training lasts 350 epochs while the following custom loss function is chosen:

$$l(y) = \frac{1}{m}\sum_{i=1}^{m} |\log \hat{y}_i - \log y_i| \tag{7.7}$$

This custom loss was chosen over a mean-squared-error (*mse*) loss function because in order to calculate the *acceptance weights*, 1 over the efficiency has to be calculated. However, the model is trained directly with the efficiency. It's easy to understand the resulting problem with an example. Assuming the target efficiency is 50 %, this results in an acceptance weight of 2. The *mse* loss function would punish a model with a predicted efficiency of 40 % the same as a model with a predicted efficiency of 60 % (Both are an absolute of 10 % off to the real value). However, the acceptance weight for the former model is $1/0.4 = 2.5$ and the later $1/0.6 = 1.\bar{6}$. Even though both models have the same *mse*, the calculated weights of the last model is closer to the true value. For this reason, the custom function was written. The log was chosen since for small efficiencies, $1/\epsilon$ can get large very quickly. This new loss function has a very similar problem, however, while testing the resulting models appeared to perform better.

For training, the starting learning rate is $lr = 0.02$ and it gets multiplied by a factor of 0.6 after a plateau which lasts 50 epochs is reached. The dataset used for training is the second half of the MC signal data (*Large MC Signal Files II*; Sec. 7.5.3). Only candidates passing the neural network classifier and the best candidate in each event are considered.

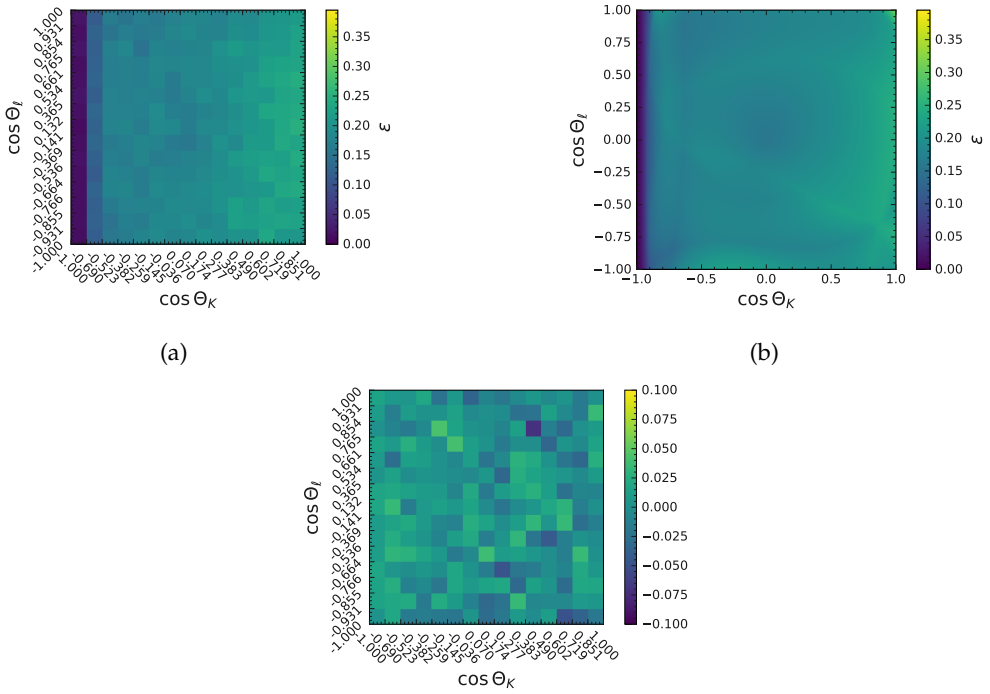


FIGURE 7.14: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_2^2-region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.

The result is a model that depends on $\cos\theta_\ell$ and $\cos\theta_K$ and returns the detector acceptance efficiency at this point. For the q_2^2-region and the channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$, the resulting model can be seen in Fig. 7.14b. In Fig. 7.14c the difference between this model and the real efficiency values is shown as a two dimensional histogram. The model plots for the other channels can be found in App. G.3.

7.7.3 Corrected Variables

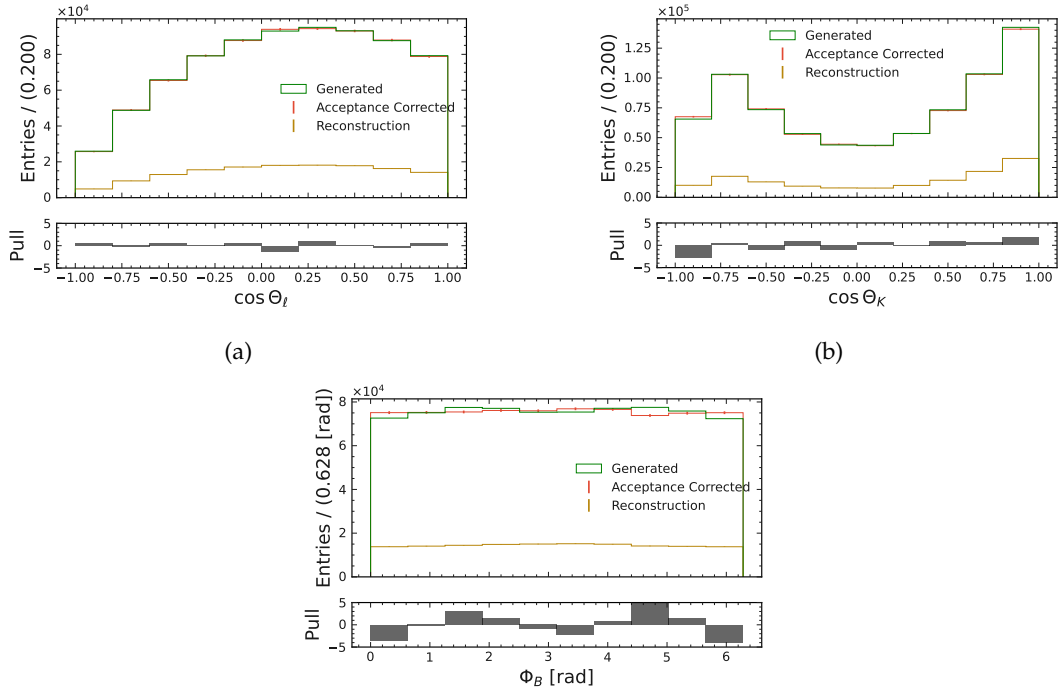
After analyzing the training data for the acceptance models and the resulting models, it is found that there are some areas which have very low efficiency. This can be seen as dark blue areas in Figs. 7.14a and 7.14b. In these low efficiency areas, small errors in the modeled efficiency lead to large errors for the calculated weights, as it was already mentioned in Sec. 7.7.2. For this reason, the low efficiency areas will be cut away. This will make the further analysis more stable and since there are only a few reconstructed signal events in these areas in the first place, the total efficiency will be reduced only slightly. A detailed table with all the angular requirements can be found in Tab. 7.8. To determine the areas to cut off, it is important to keep in mind that $\cos\theta_\ell$ has to be treated symmetrically because there will be a transformation of

this variable according to Eq. 2.57. On the other hand, $\cos\theta_K$ will not be transformed, therefore there are no special limitations for cutting on this variable. The procedure of cutting off low efficiency regions in $\cos\theta_\ell$ and $\cos\theta_K$ will be referred to as trimming.

TABLE 7.8: Chosen requirements for the trimming. Regions in $\cos\theta_\ell$ and $\cos\theta_K$ with a very low efficiency are removed in order to improve the fit stability.

q^2-Region	$\cos\theta_\ell$	$\cos\theta_K$
q^2_1	$\lvert\cos\theta_\ell\rvert < 0.75$	$\cos\theta_K > -0.90$
$q^2_{2-\psi(2S)}$	/	$\cos\theta_K > -0.90$
q^2_4	$\lvert\cos\theta_\ell\rvert < 0.95$	$\lvert\cos\theta_K\rvert < 0.95$

After applying the angular variable trimmer, the corrected angular variables for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_2-region can be seen in Fig. 7.15.



(a) (b)

(c)

FIGURE 7.15: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q^2_2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.

Here, the pull is calculated by

$$\text{Pull} = \frac{n_{\text{gen}} - n_{\text{PA}}}{\sigma_{\text{PA}}} \tag{7.8}$$

where n_{gen} is the number of generated events within the bin, and

$$n_{\text{PA}} = \sum_{i=1}^{n} w_i \qquad \text{and} \qquad \sigma_{\text{PA}} = \sqrt{\sum_{i=i}^{n} w_i^2} \tag{7.9}$$

where w_i is the acceptance weight multiplied with the PID weight of the ith entry and n is the total number of entries of the bin. This trimming is only applied right before the angular analysis. The correction plots for the other decay channels can be found in the App. G.4.

7.8 Correlations

In the next analysis step, the sWeights method (Sec. 4.3.3) will be used in order to reduce the influence of non-signal on the angular (*target*) variables. The signal/non-signal separation will be done with a 2D fit on ΔE and M_{bc} (*discriminating*). However, the sWeights method only works if the discriminating variables used in the signal/non-signal fits are uncorrelated to the target variables where only the signal component will be projected out. For the channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$, the correlation plots between $\Delta E/M_{\text{bc}}$ and $\cos\theta_\ell$, $\cos\theta_K$ and ϕ_B can be seen in Fig. 7.16. No correlation is found. The correlation plots between the target and discriminating variables in the other channels for background (signal) events can be found in the App. H.1.1 (H.2.1). Another requirement of sWeights is that the method can only be applied to unweighted datasets. Therefore, all the weights calculated earlier, like PID-correction weights (for MC datasets) and acceptance weights, have to be applied after the sWeights are calculated. This can only be done, if there is no correlation between ΔE and M_{bc} and both of these aforementioned weights. This is investigated for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in Fig. 7.17. For the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels, no correlation is observed and the sWeights method can be applied. Unfortunately, some correlations are found in the $B^+ \to K^{*+}(\to K^+\pi^0)\ell\ell$ channels. This will be discussed with more detail in the next section. The correlation plots between the discriminating variables and weights in background (signal) for the other channels can be found in the App. H.1.2 (H.2.2).

7.8.1 Correlations in the $B^+ \to K^{*+}(\to K^+\pi^0)\ell\ell$ Channels

It was found that the channels $B^+ \to K^{*+}(\to K^+\pi^0)ee$ and $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$ have some strong correlations in variables which should be uncorrelated in order for this analysis strategy to work. The problematic correlation arises in the variables M_{bc} and $\cos\theta_\ell$ and $\cos\theta_K$ for background events. This can be seen in Figs. 7.18 and 7.19 for the $B^+ \to K^{*+}(\to K^+\pi^0)ee$ and $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$ channel, respectively. M_{bc} is used in the fit to determine the signal and background PDF (discriminating variables), therefore it has to be uncorrelated to the variables that are used later in the angular fit (target variables), otherwise the sWeights, calculated with the sPlot method, will not sum up to the true shape of these angles [58].

However, since the sWeight method enables an analysis, which does not rely on fixing a background shape determined by simulation it was decided that this advantage is larger then the loss of the decay channels containing a π^0. For this reason, the analysis will continue with the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ and $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channels and the $B^+ \to K^{*+}(\to K^+\pi^0)\ell\ell$ channel will be dropped.

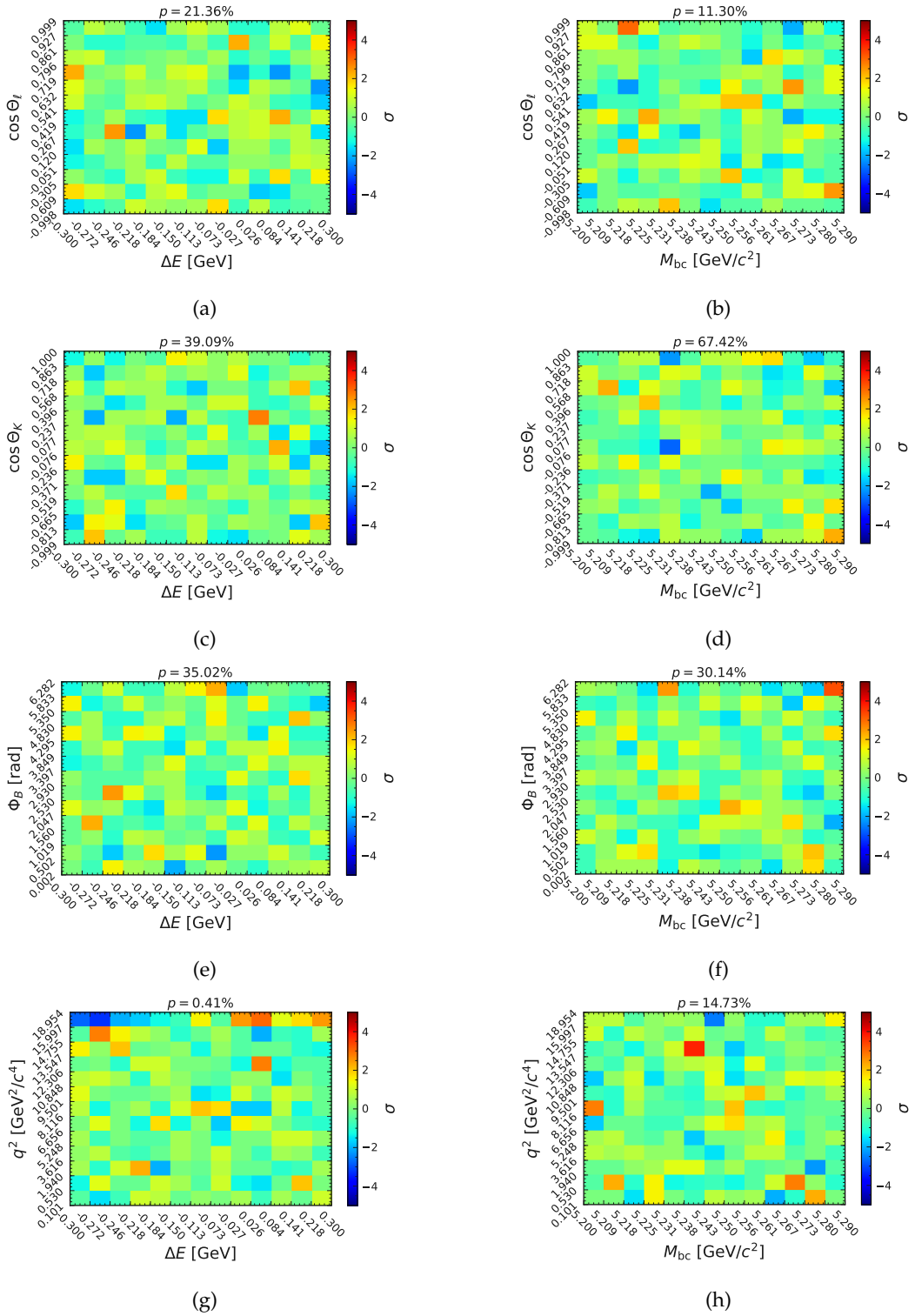


FIGURE 7.16: Correlations between Angles and ΔE & M_{bc} in the channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$ for background events.

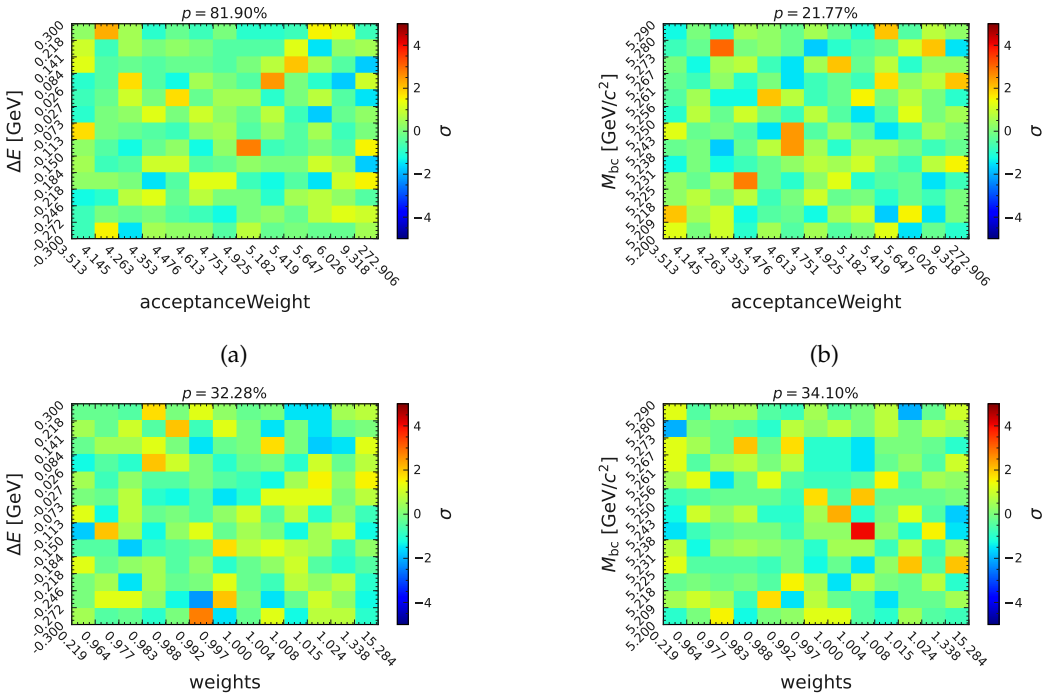


FIGURE 7.17: Correlations between the $\Delta E / M_{\text{bc}}$ and acceptance and PID weights in the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel for background events.

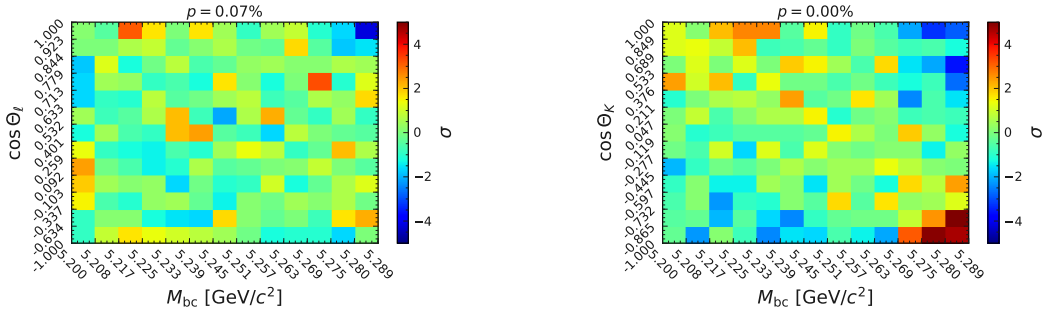
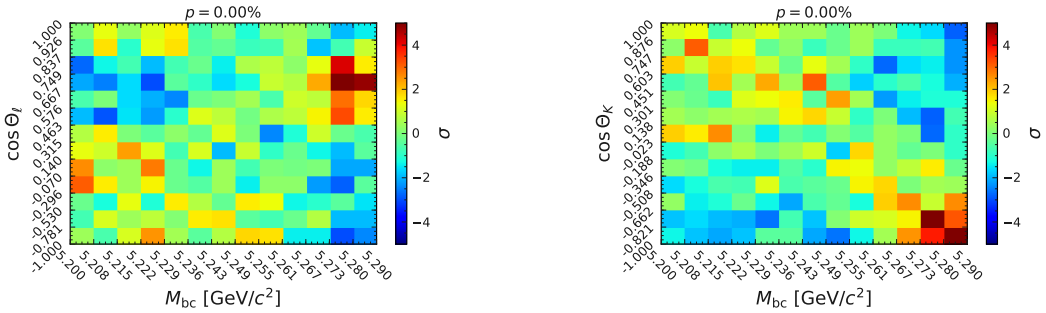


FIGURE 7.18: Correlation between M_{bc} and $\cos\theta_\ell$ and $\cos\theta_K$ in the $B^+ \to K^{*+}(\to K^+\pi^0)ee$ channel for background events. A rather strong correlation can be observed between M_{bc} and $\cos\theta_\ell$. Due to this correlation, this channel can no longer be analyzed with the sWeights method.



(a) (b)

FIGURE 7.19: Correlation between M_{bc} and $\cos\theta_\ell$ and $\cos\theta_K$ in the $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$ channel for background events. A rather strong correlation can be observed between M_{bc} and $\cos\theta_\ell$. In the case of this μ-channels, there also appears to be a correlation between M_{bc} and $\cos\theta_\ell$. Due to these correlations, this channel can no longer be analyzed with the sWeights method.

Chapter 8

Analysis

At the moment of commitment the
entire universe conspires to assist you.

Johann Wolfgang von Goethe

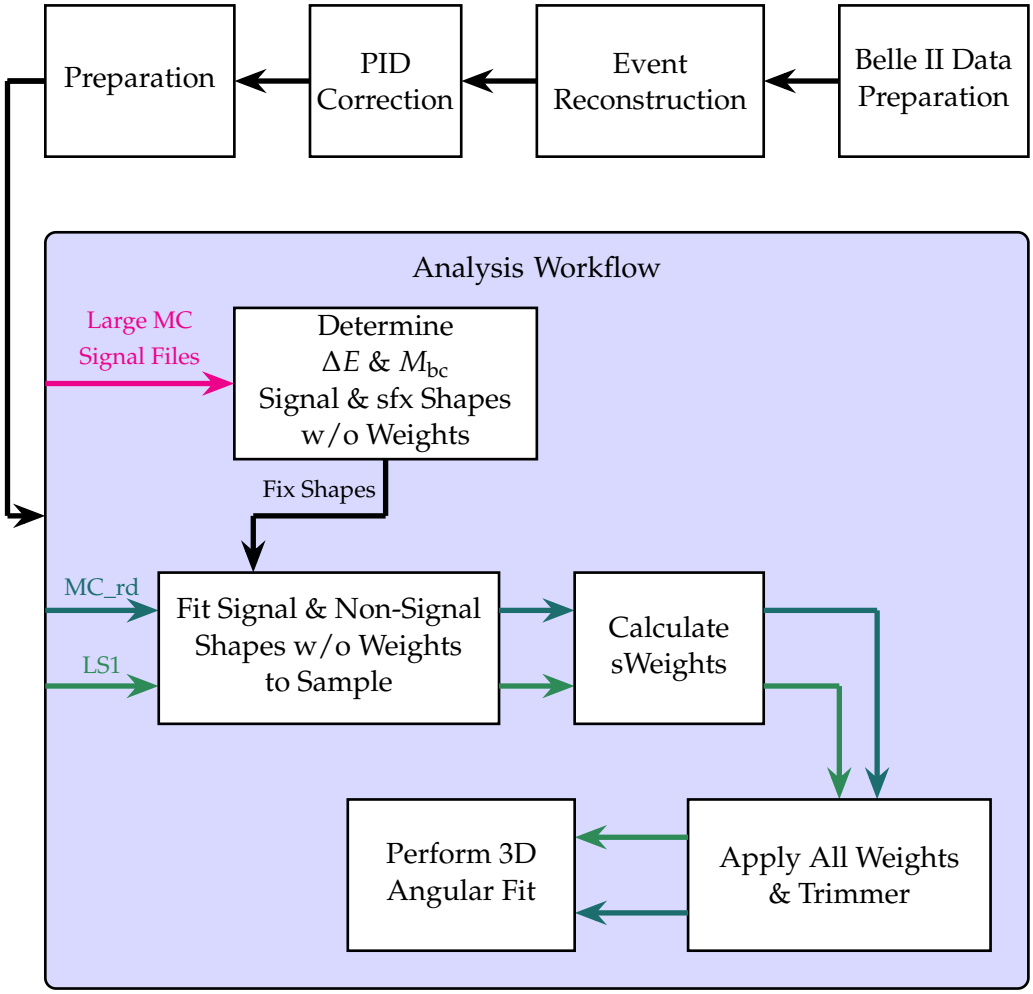


FIGURE 8.1: Workflow for the angular analysis.

With all the event reconstruction and preparation done, the fitting procedure can start. This procedure can be split up in several steps. First, the PDFs in the discriminating variables, M_{bc} and ΔE, are determined for signal and sfx events using MC files containing only signal events. Then the fit parameters, as well as the ratio, obtained from this fit will be fixed in the following procedure while determining

the parameters and ratios for the background PDFs of the discrimination variables. Using these combined PDFs it is now possible to calculate the sWeights according to Sec. 4.3.3 in order to obtain the true shapes of the angular variables (*target variables*). After computing the weights, the trimmer, as introduced in Sec. 7.7.3 and Tab. 7.8, is applied. Finally, a three dimensional angular fit, using Eq. 2.57 as the PDF, is performed and the values for F_L, S_3 and P_5' are determined. During the fit, these parameters are restricted to $F_L \in [0,1]$ and S_3, $P_5' \in [-2,2]$. The fitting procedure is repeated for each channel and each q^2-region independently.

Before the analysis procedure can start, for each discriminating variable, the PDF models for each of the three contributions in the data set have to be determined. These contributions are *signal*, *sfx* and for the background the contributions of *continuum*, *B meson* and τ *pair* events are combined to a single model. The chosen PDF compositions are listed in Tab. 8.1.

TABLE 8.1: Chosen PDFs for the discriminating variables and each contribution in the data sets.

Variable	Signal	sfx	Background
ΔE	Gaussian + Crystal-Ball	Polynomial	Polynomial
M_{bc}	Gaussian	Argus	Argus

8.1 Error Estimation

Once the final parameters are determined in the fit, an error to these values has to be assigned. This error estimation will be done with a MC toy study. For this, the obtained signal and non-signal PDFs are used in order to generated a toy MC sample. The $\cos\theta_\ell$, $\cos\theta_K$ and ϕ_B shapes are approximated with a Legendre-Polynomial for the non-signal model and with the PDF according to Eq. 2.57 for the signal models. Now, since the shape of these variables after reconstruction is required, the acceptance value will be calculated of each event according to Sec. 7.7 alongside a random number between 0 and 1. Then, if the random number is larger than the calculated acceptance value, the event will be dismissed, otherwise it will be retained. After this procedure, the shapes of this toy MC dataset will correspond to the MC15_rd dataset after reconstruction. Finally, the sWeights for this new toy MC dataset are calculated and using these weights, the originally generated shapes for $\cos\theta_\ell$, $\cos\theta_K$ and ϕ_B can be obtained and the 3D angular fit can be performed, saving the covariant matrix of the fit. An example for these three steps (1st: generating; 2nd keep only events where the random number is smaller than the calculated acceptance; 3rd applying acceptance weights) can be seen in Fig. 8.2.

This process will be repeated 50 times for each decay channel and q^2-region with the corresponding integrated luminosity. Finally, the standard deviation of the obtained parameters is calculated. This value is then considered the error of the original parameter value.

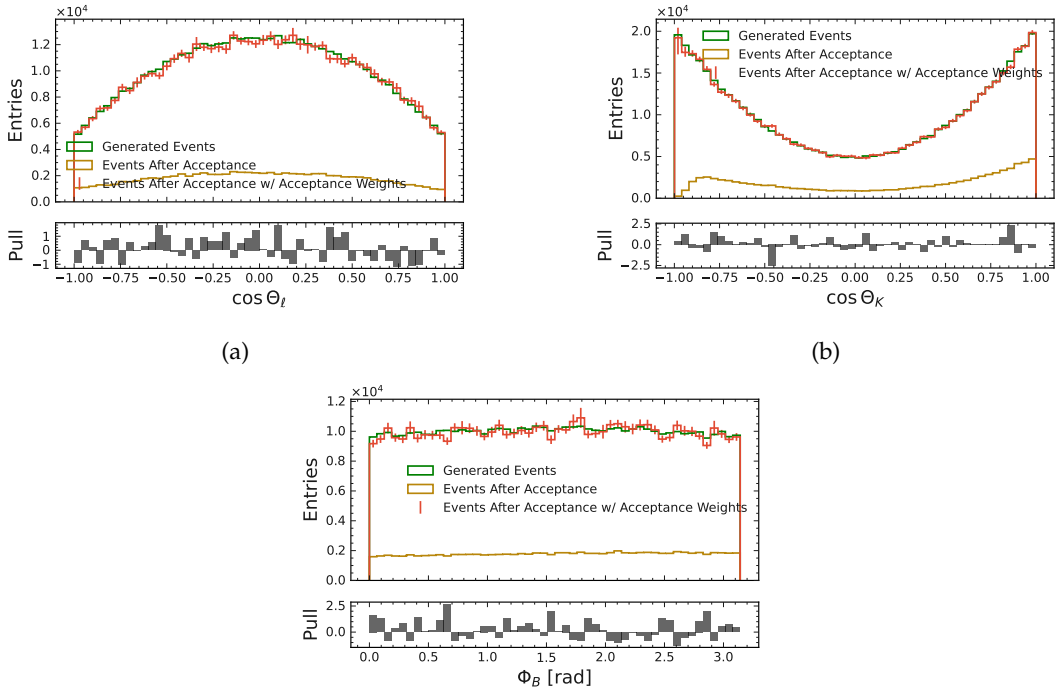


FIGURE 8.2: Example for pure toy study events. The shapes after acceptance are created by creating a random number between zero and one for each event. If this number is smaller than the acceptance value of this event, the event is kept otherwise discarded. Applying the acceptance weight to the events which survive the selection brings the shape back to the generated one.

8.2 MC15_rd

As a proof of concept, the whole analysis will be done with MC15_rd data first to ensure that the procedure works. This dataset was generated to be very similar to the LS1 dataset. The integrated luminosity of this dataset corresponds to $\int \mathcal{L}_{\text{MC15}}\,\mathrm{d}t = 4 \cdot 364\,\text{fb}^{-1}$. Examples for these fits are shown in Figs. 8.3 and 8.4 for the discriminating variable and angular fits, respectively. The dataset for both examples are from the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q_2^2-region with $\ell = e$ ($\ell = \mu$) on the left (right). The discriminating and angular fits for the other decay channels can be found in App. I.1.1 and I.1.2, respectively.

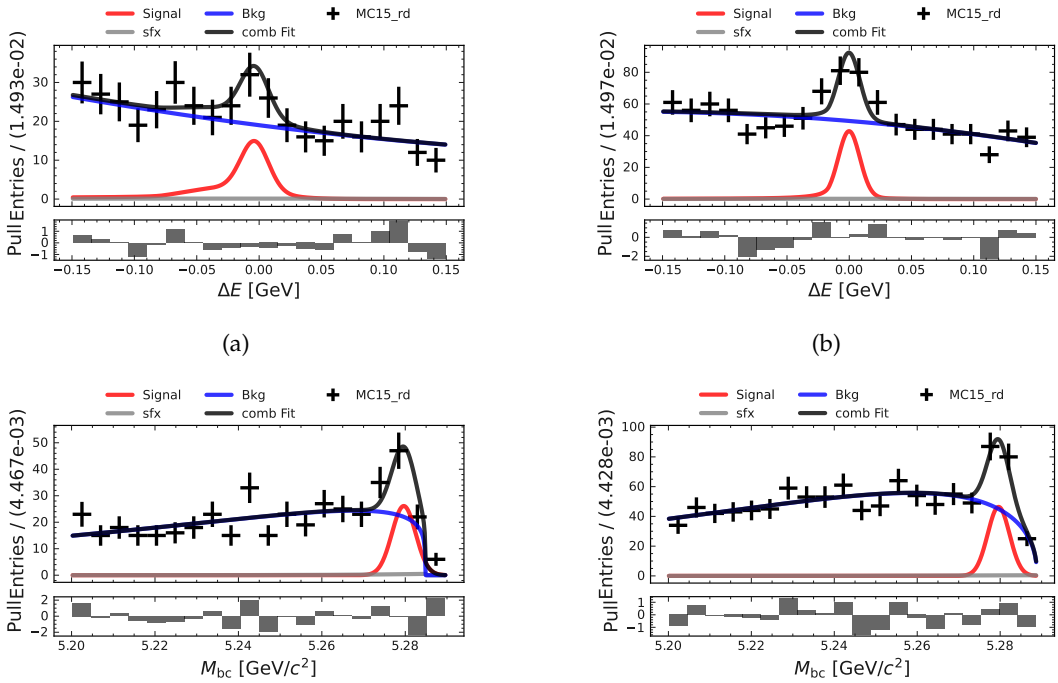


FIGURE 8.3: M_{bc} and ΔE fits in the q_2^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the MC15_rd dataset. The fits for $\ell = e$ ($\ell = \mu$) are shown on the left (right).

The resulting parameters of the three dimensional angular fit on the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ are presented in Fig. 8.5. In this figure the values for the electron (orange cross) and muon (brown cross) channels are shown side by side. Keep in mind, that both channels are generated with the same set of Wilson coefficients. As a result, the obtained parameters should be the same. The fit values after combining the electron and muon dataset are shown as a dark slate gray cross. The green line shows the value obtained by a fit on a MC generated truth data set.

For the F_L and S_3 parameters, the results appear to be consistent considering the rather large error bars. Especially S_3 seems to be a very stable parameter with the exception of the q_2^2-region, where the value for the electron and muon channel have rather large discrepancies. Unfortunately, P_5' has the largest deviations between the channels. Also the error bars are very large. Especially, the fit on the electron channel returns a rather high value in the third q^2-region. Overall, the values from the combined measurement lie between the values obtained by the electron and muon channel separately, as expected.

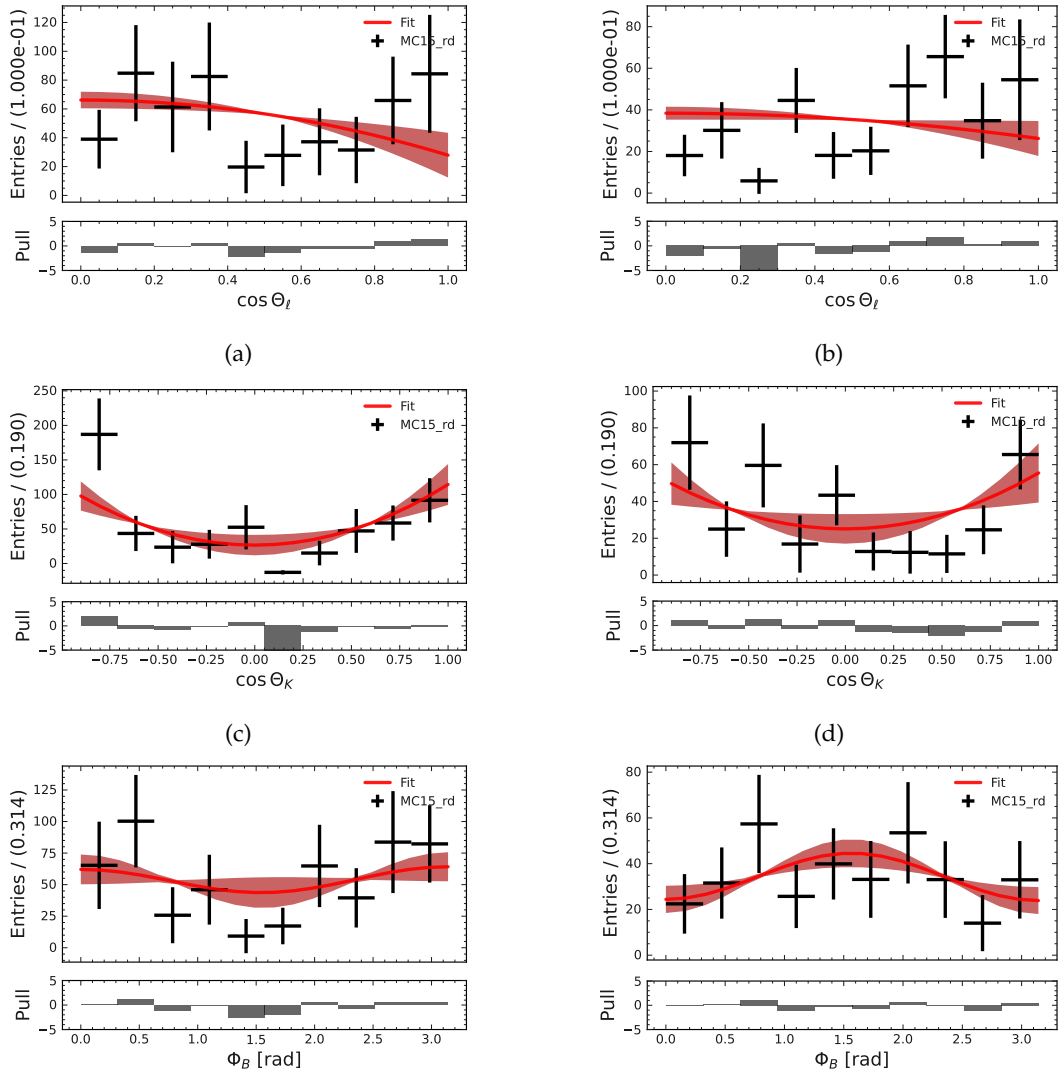
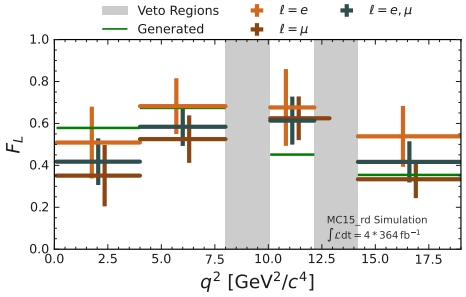
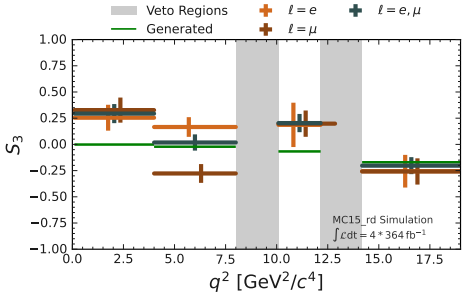


FIGURE 8.4: Angular fits for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q^2_2-region containing the MC15_rd dataset. The fits for $\ell = e$ ($\ell = \mu$) are shown on the left (right).

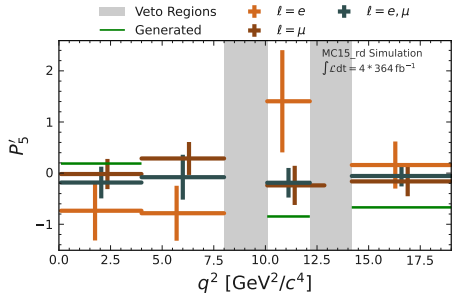
Since the expected amount of signal events is much smaller for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$, compared to the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$, it is also expected that the obtained errors are larger for these channels. This can be seen in Fig. 8.6. Again the parameter with the smallest (largest) error bars is S_3 (P'_5).



(a)

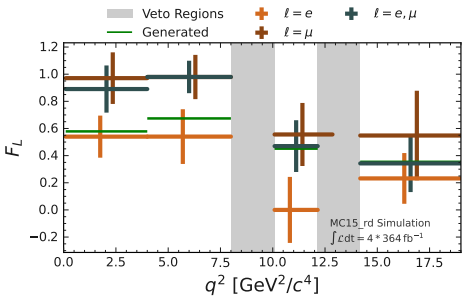


(b)

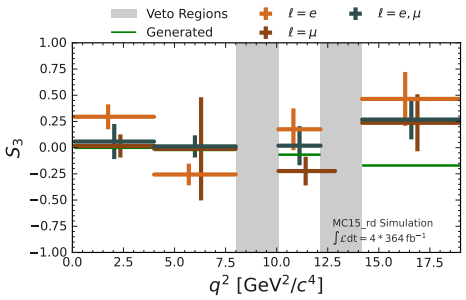


(c)

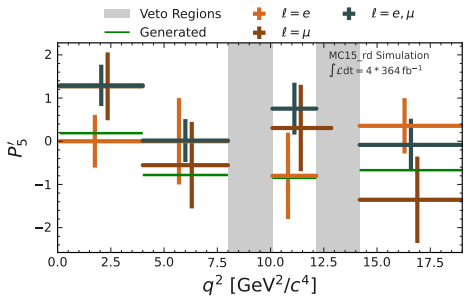
FIGURE 8.5: Angular parameters obtained by the three dimensional fit as a function of q^2 for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels in the MC15_rd dataset. (a) shows F_L, (b) S_3 and (c) P_5'.



(a)



(b)



(c)

FIGURE 8.6: Angular parameters obtained by the three dimensional fit as a function of q^2 for the $B^+ \to K^{*+}(\to K^0\pi^+)\ell\ell$ channels in the MC15_rd dataset. (a) shows F_L, (b) S_3 and (c) P_5'.

8.3 Long Shutdown 1 Data

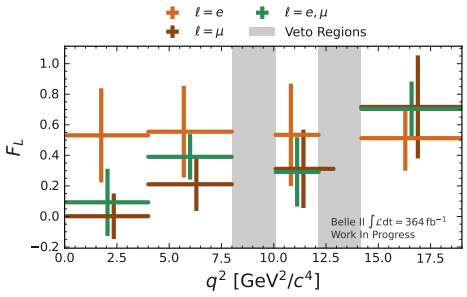
The complete analysis pipeline shall now be done for the Long Shutdown 1 data set of Belle II. This dataset has a total integrated luminosity of $\int \mathcal{L}_{\text{LS1}}\,\mathrm{d}t = 364\,\text{fb}^{-1}$. Unfortunately, according to Tab. 7.6, the number of expected reconstructed signal candidates is too low to perform a three dimensional angular fit in the channels $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$. For this reason, the analysis will only be done with LS1 data for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels. However, even for these channels, the expected amount of signal candidates is around ten (Tab. 7.6), which makes a three dimensional angular fit still very challenging. The discriminating and angular fits for all channels can be found in the App. I.2.1 and I.2.2, respectively.

8.3.1 LS1 Results

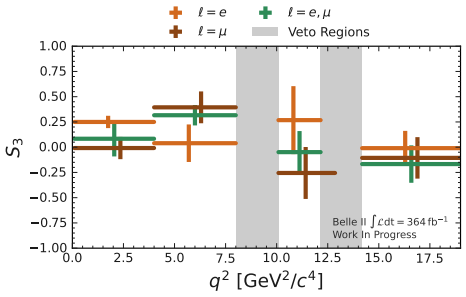
The final fit parameters of the LS1 dataset for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ are presented in Fig. 8.7. The orange (brown) crosses represent the obtained values from the $\ell = e$ ($\ell = \mu$) channel. The results from the combination are represented by the dark green cross. The only parameter which seems to produce good results throughout the q^2-regions is S_3. However, even for this parameter, there is a large discrepancy between values obtained from the electrons and muons. The errorbars for F_L are quite large and the fits in the first q^2-region seem to have failed for the muon channel since F_L is determined to be exactly zero. For the parameter P'_5 in the first q^2-region the muon channel returns a very high value. The same is true for all channels in the second q^2-region. Here, the value is almost the same for all three channels, but it is unexpectedly high. In the fourth q^2-region the electron channel returned a very high value, as well.

In Fig. 8.7d the parameter P'_5 determined in this thesis is presented alongside the values calculated by other experiments. With the exception of the second q^2-region, the results obtained from this analysis seems to be in agreement with the results of Belle.

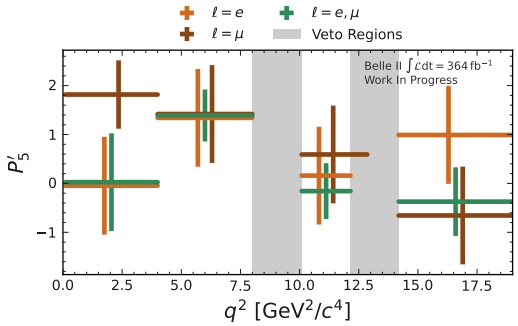
With these values and error bars, it is not possible to claim meaningful results from this analysis on the LS1 dataset. Despite this fact, it was shown, that the analysis workflow works in principle and it is only a matter of time until the total integrated luminosity of Belle II is large enough to produce meaningful and competitive results. Further improvements of this workflow will be discussed in Ch. 10.



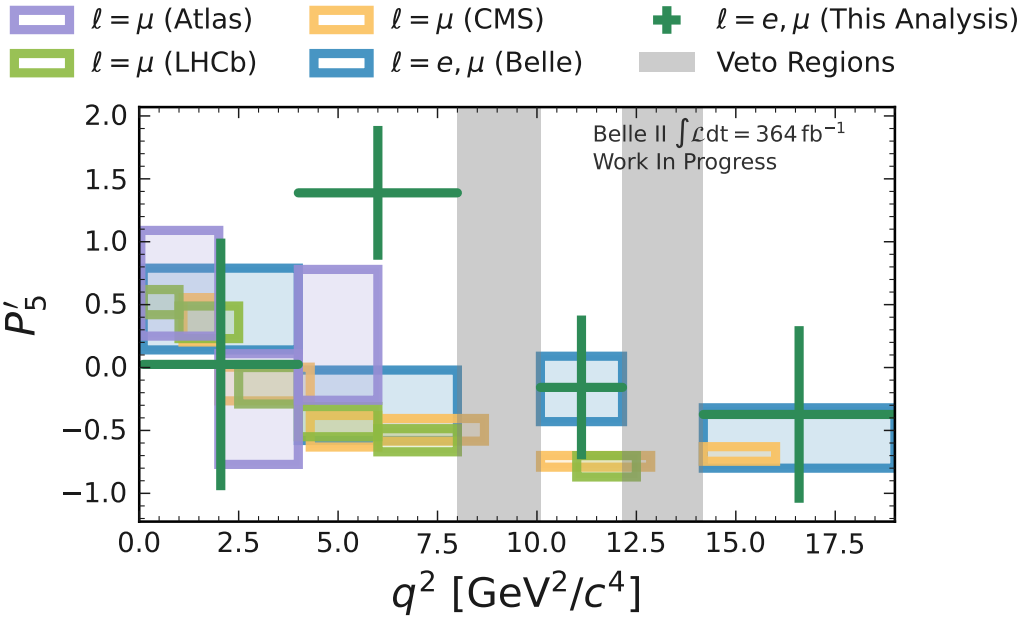
(a)



(b)



(c)



(d)

FIGURE 8.7: Angular parameters obtained by the three dimensional fit as a function of q^2 for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels in the LS1 dataset. (a) shows F_L, (b) S_3 and (c) P_5'. In addition to the resulting values of this thesis (dark green), the values obtained by Belle (blue) [94], LHCb (light green) [95], CMS (yellow) [23] and Atlas (purple) [24] are shown in the P_5' (d) plot as well.

TABLE 8.2: Angular parameters obtained by the three dimensional fit for all the q^2 region in the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels using the LS1 dataset.

Channel	Parameter	q^2_1	q^2_2	q^2_3	q^2_4
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	F_L	0.53 ± 0.31	0.55 ± 0.30	0.53 ± 0.33	0.51 ± 0.21
	S_3	0.25 ± 0.06	0.04 ± 0.19	0.26 ± 0.33	-0.01 ± 0.17
	P'_5	-0.05 ± 1.00	1.34 ± 1.00	0.15 ± 1.00	0.99 ± 1.00
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	F_L	0.00 ± 0.15	0.21 ± 0.17	0.31 ± 0.26	0.72 ± 0.34
	S_3	-0.02 ± 0.11	0.39 ± 0.16	-0.25 ± 0.26	-0.11 ± 0.21
	P'_5	1.81 ± 0.70	1.41 ± 1.00	0.59 ± 1.00	-0.66 ± 1.00
$B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$	F_L	0.09 ± 0.22	0.39 ± 0.15	0.29 ± 0.23	0.70 ± 0.18
	S_3	0.08 ± 0.18	0.32 ± 0.10	-0.05 ± 0.21	-0.17 ± 0.19
	P'_5	0.02 ± 1.00	1.39 ± 0.53	-0.16 ± 0.57	-0.37 ± 0.70

Chapter 9

Systematic Studies

Paths are made by walking.

Franz Kafka

9.1 Branching Fractions of $B \to K^* J/\psi$ and $B \to K^* \psi(2S)$ Decays

A very common way to validate the calculated acceptance correction is to determine the well known branching fractions of the $B \to K^* J/\psi$ and $B \to K^* \psi(2S)$ channels. These transitions happen on tree level and therefore occur around ≈ 1000 times as often compared to the flavor changing neutral currents. In general, the branching fraction can be calculated by

$$\mathcal{B}(B \to K^* X_{c\bar{c}}) = \frac{N_{\text{obs}}}{2 \cdot \epsilon \cdot f_{\ell\ell} \cdot f_{K\pi} \cdot N_{B\bar{B}}} \tag{9.1}$$

where N_{obs} is the observed number of $B \to K^* X_{c\bar{c}}$ events, ϵ is the efficiency after reconstruction and classifier[1] and $f_{\ell\ell}$ is an additional factor from PDG [10] which takes the branching fraction of the $c\bar{c}$ state decaying into lepton pairs into account. This factor is $5.94 \pm 0.06 \cdot 10^{-2}$ ($5.93 \pm 0.06 \cdot 10^{-2}$) for $J/\psi \to ee$ ($J/\psi \to \mu\mu$) and $7.73 \pm 0.17 \cdot 10^{-3}$ ($7.7 \pm 0.8 \cdot 10^{-3}$) for $\psi(2S) \to ee$ ($\psi(2S) \to \mu\mu$). Similar, $f_{K\pi}$ is a correction factor that takes the branching fraction of the $K^* \to K\pi$ into account. Finally, $N_{B\bar{B}}$ is the number of produced B meson pairs. The denominator has to be multiplied by a factor of two since there are two B mesons in an event and each of them can decay through the channel of interest.

Tab. 9.1 shows the obtained branching fractions of the $B \to K^* X_{c\bar{c}}$ channels. These values are shown for each channel separately since each one of them has a separate classifying and acceptance model. These values are calculated using the MC15_rd dataset[2] and the errors are determined by the yield fit on the discriminating variables ΔE and M_{bc}. Looking at the errors for the decays including a J/ψ it becomes obvious that this error estimation has to be done with more detail in the future. Tab. 9.1 also contains the branching fraction value used while generating this dataset and the official PDG value.

Most noticeable is that the branching fraction determined in the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels are consistently lower than the values, that are used to generate these data samples. The branching fractions calculated from charged B mesons are

[1] The acceptance corrections are calculated with this efficiency

[2] The box was not opened for these studies are, and, as a result, the branching fractions within the LS1 dataset were not determined.

TABLE 9.1: Measured branching fractions for each channel in the MC15_rd dataset. The PDG values are also shown in these tables [10].

	$\mathcal{B}(B \to K^* J/\psi)[10^{-3}]$		
Used Workflow Channel	Measured	Gen.	PDG-Value
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	1.17 ± 0.01	1.33	1.27 ± 0.05
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	1.22 ± 0.01		
$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	1.38 ± 0.01	1.43	1.43 ± 0.08
$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	1.47 ± 0.01		

	$\mathcal{B}(B \to K^* \psi(2S))[10^{-4}]$		
Used Workflow Channel	Measured	Gen.	PDG-Value
$B^0 \to K^{*0}(\to K^+\pi^-)ee$	5.7 ± 0.2	6.1	5.9 ± 0.4
$B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$	5.5 ± 0.4		
$B^+ \to K^{*+}(\to K^0_S\pi^+)ee$	6.2 ± 0.3	6.2	6.7 ± 1.4
$B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$	7.9 ± 0.6		

overall closer to their generated values. There is one exception. The determined branching fraction of the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel is roughly three standard deviations above the generated values. This large discrepancy has to be studied with more detail in the future.

More details about these obtained values alongside all the corresponding figures can be found in the master thesis of Jigarkumar Patel [96]. These studies did not include the sWeights method. Instead the yield was determined using a two dimensional fit on the variables ΔE and M_{bc} with weighted events. These weights include PID and acceptance correction weights.

9.2 Angular Fit Stability

A first iteration of systematic studies concerning the angular fit stability is presented in the master thesis of Andre Klotzbücher [19]. His thesis investigated the influence of the transformation introduced in Sec. 2.2.5 transforming Eq. 2.51 with a total of eight independent parameters to Eq. 2.57 with the three parameters F_L, S_3 and P'_5. In addition to that, the performance of the angular fit depending on several integrated luminosities is studied and the expected fit biases of the final three fit parameters are determined depending on the available integrated luminosity.

These studies were done with datasets prepared using the workflow presented with the thesis at hand. These studies were using pure MC15_ri signal files and, once the PDF shapes were determined, toy MC files, as well. They did not include non-signal events and the sWeights method.

9.3 Demonstrating sWeights

Since the sWeights method is rather uncommon, its effectiveness will be studied with a more sophisticated example compared to the one presented in Sec. 4.3.3. For this demonstration, the obtained signal and non-signal PDFs are used in order to generated a toy MC sample similar to Sec. 8.1.

The fit on the discriminating variables is shown in Figs. 9.1a and 9.1b, the resulting figures of the target variables in Figs. 9.1c, 9.1d and 9.1e. These last three

plots show the true generated signal event shape in green, the events after they are shaped to reconstruction level and with acceptance weights in dark orange and the same events but with also the sWeights in red. After applying all weights (acceptance + sWeights) the shape of the *reconstructed* events with all weights applied (red) is very close to the shape of the generated signal events (green). Using this generator level-like shape it is possible to directly fit the three angular parameter F_L, S_3 and P_5' without fixing the background shape determined by simulation. The results for these data can be seen in Sec. 10.1. There, this method is applied for both, $\ell = e$ and $\ell = \mu$ to the second q^2-region for several integrated luminosities.

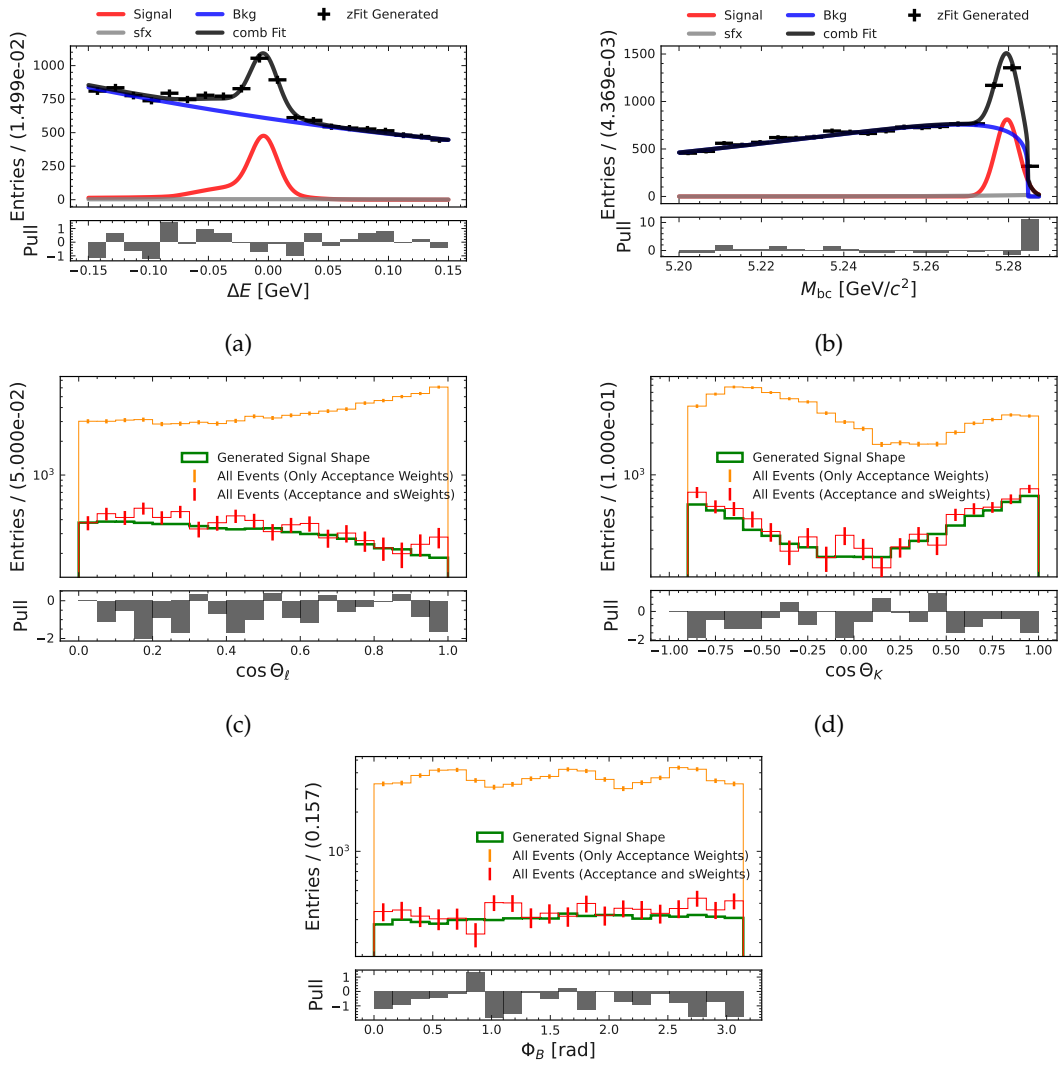


FIGURE 9.1: Showing the effect of the sWeights using toy MC. (a) and (b) show the discriminating variables, (c), (d) and (e) the target variables. The generated signal events are shown in green, the dataset containing signal and non-signal events in dark orange (only acceptance weights are applied) and red (acceptance and sWeights are applied). The amount of events corresponds to an integrated luminosity of $\int \mathcal{L}\,\mathrm{d}t = 50\,\mathrm{ab}^{-1}$.

9.4 Loading More Signal Events

Another test that can be done in order to study the stability of this analysis is to load way more signal events than expected while keeping the background the same. In this study, the complete MC15_rd dataset ($4 \cdot 364\,\mathrm{fb}^{-1}$) is taken for the background. For signal, ten times the amount of expected events are loaded. The complete analysis workflow is applied to this new data sample. This way, the analysis is done with larger datasets which are not a toy study since both background and signal are generated as MC15. An example for the discriminating and angular fit can be found in Fig. 9.2. The fits for the other q^2-regions can be found in the App. J.

The result from the angular fits can be found in Fig. 9.3. As expected, the error bars are smaller compared to Fig. 8.5. For the P_5' variable, there could be a bias since for the q^2-regions 2, 3 and 4, the fit values appear to be consistently higher then expected.

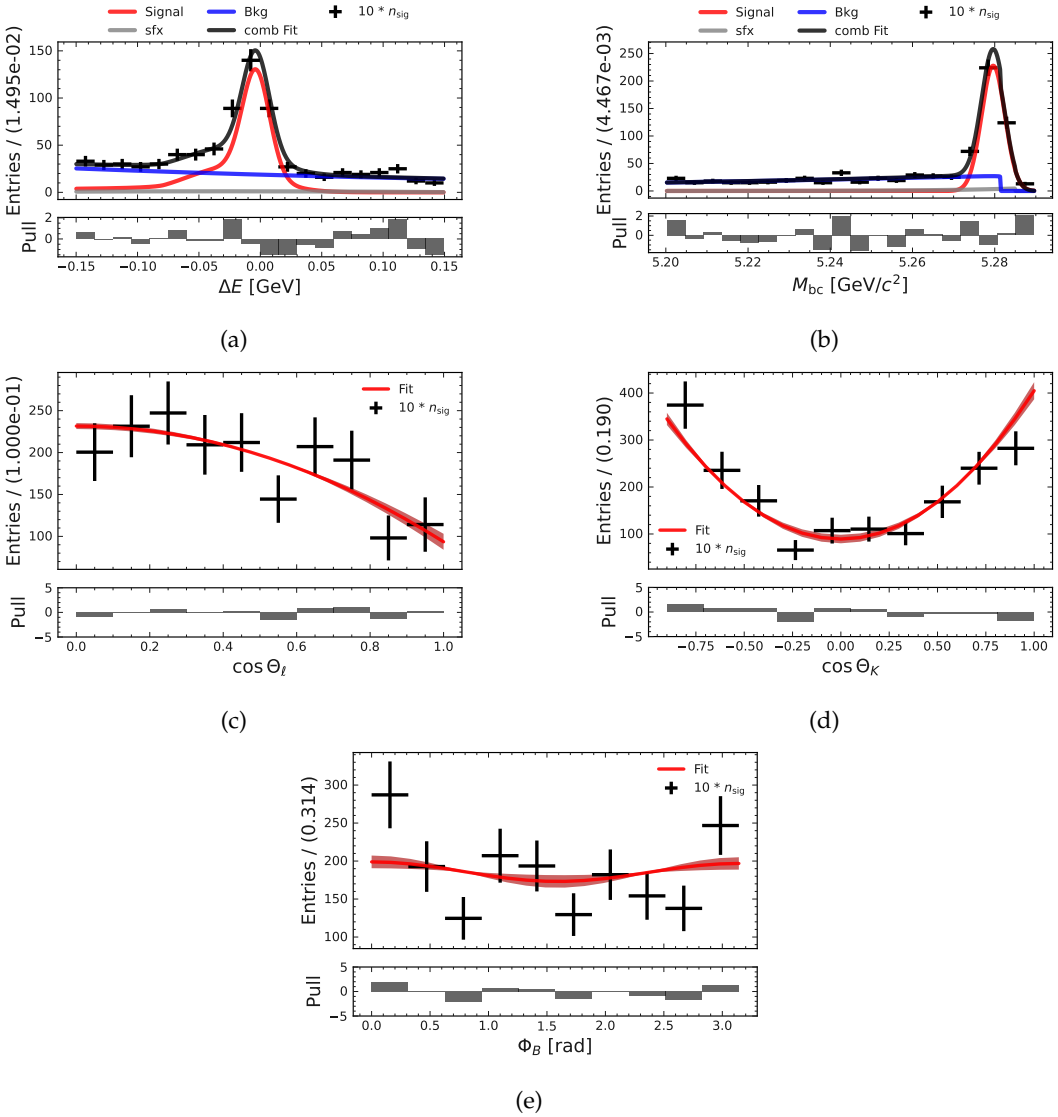
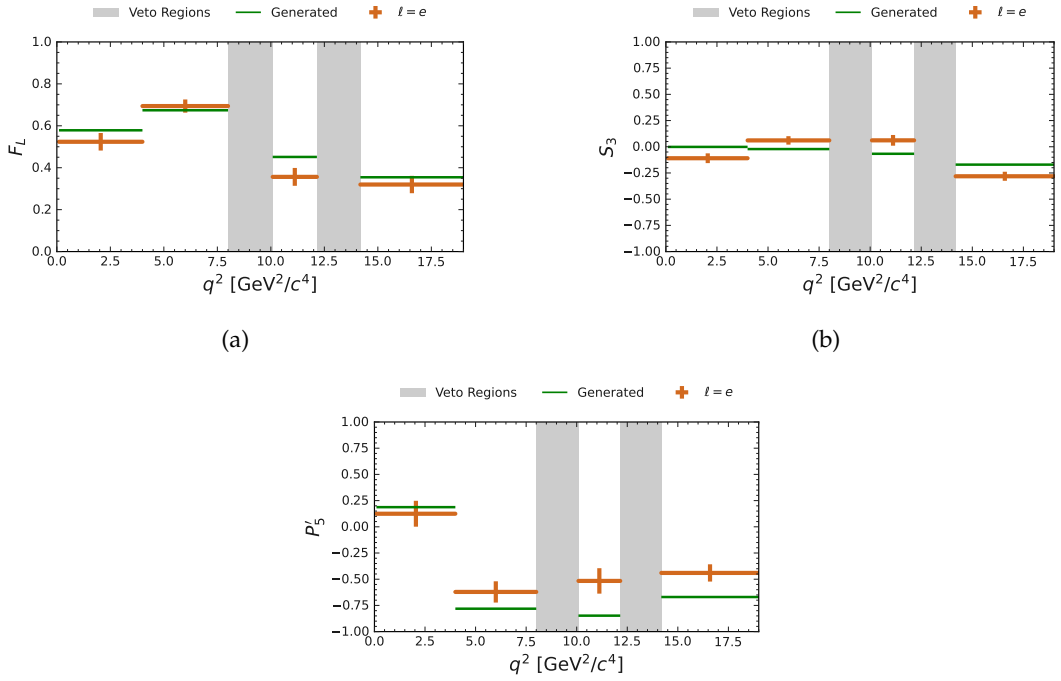


FIGURE 9.2: Figures related to the fit study with $n_{\mathrm{sig}} = 10 \cdot n_{\mathrm{expected}}$ for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_2^2-region. (a) and (b) show the fit on the discriminating variables while (c), (d), and (e) show the fits on the angular variables.



(a) (b)

(c)

FIGURE 9.3: Angular parameters obtained by the three dimensional fit as a function of q^2 for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channels in the MC15_rd dataset with $n_{\text{sig}} = 10 \cdot n_{\text{expected}}$. The green line indicates the value obtained from an angular fit on the MC generated dataset. (a) shows F_L, (b) S_3 and (c) P'_5.

9.5 Fixing the Background Shapes Instead of Using sWeights

sWeights are used in order to get rid of the non-signal contribution in the angular variables. But, the more obvious way to handle these contributions is by determining the background shapes with MC and keep them fixed. The signal contribution can then be determined. Therefore, this analysis will also be done with this *simpler* method and the results will be compared to the sWeights strategy. For this, the background for the angular variables as determined on the MC15_rd background sample using higher degrees Chebyshev polynomials.

In Fig. 9.4 this method of fixing the background contribution is shown for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the second q^2-region. The plots with the fits for the other q^2-regions can be found in App. J.1.

The results for the angular parameter of these fits can be found in Fig. 9.5. Compared to Fig. 8.5 (where the sWeight method was used), the obtained values from the fits for F_L and S_3 are more scattered around the generated value. In the third q^2-region all channels return $P'_5 = -2$, which is the smallest value possible in the fit. For the electron channel also the first and second q^2-region seems to have failed. Overall, only the fourth q^2-region seems to be stable.

This method of fixing the background shape has the big disadvantage, that these shapes can only be directly determined for MC data. Therefore, for real data, these MC determined shapes have to be used as well and, as a result, the quality of this strategy depends heavily on the similarity between MC and real data. Therefore,

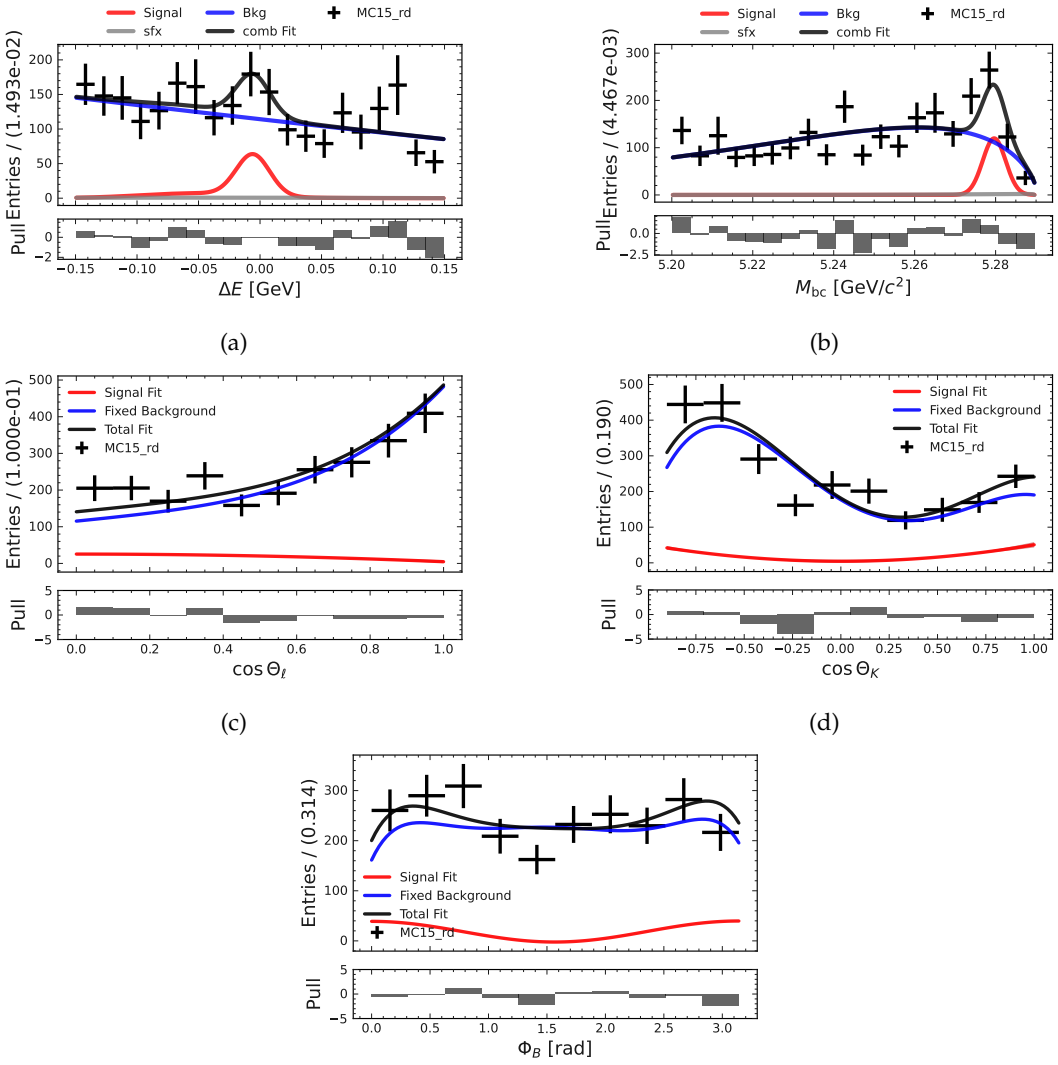
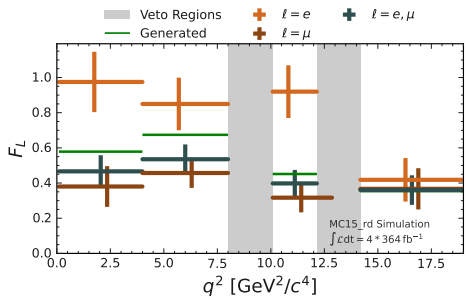
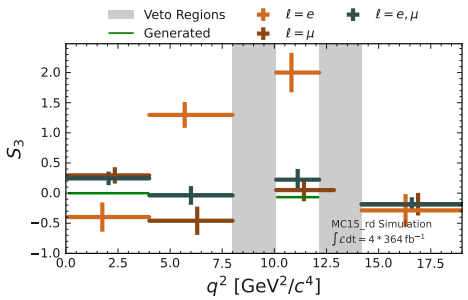


FIGURE 9.4: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_2-region.

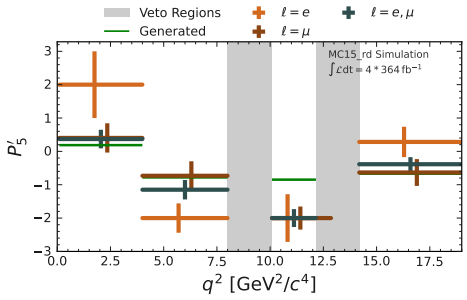
small differences could lead to *poor* background modeling. A good example is the measurement of R_{K^*} with the $B \to K^*\ell\ell$ decay. In 2022 a violation of the Lepton Universality with a significance of $3.1\,\sigma$ was reported by LHCb [97]. In 2023 an improved background model was used which brought all measurements to be in agreement with the SM at $0.2\,\sigma$ level [98].



(a)



(b)



(c)

FIGURE 9.5: Angular parameters obtained by the three dimensional fit as a function of q^2 for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels in the MC15_rd dataset without the sWeights method. Instead the background shape is fixed. (a) shows F_L, (b) S_3 and (c) P_5'.

Chapter 10

Outlook

> There is as yet insufficient data for a meaningful answer.
>
> Isaac Asimov, *The Last Question* (1956)

It was shown that the presented workflow for this analysis works. However, there are several aspects which can be improved further. This chapter will provide a brief overview of some of these improvements.

10.1 The Future of the $B \to K^* \ell\ell$ Angular Fits

Belle II is still at the beginning of data taking. The LS1 dataset was collected between 2019 and 2022 and contains an integrated luminosity of $364\,\text{fb}^{-1}$ which is about half the integrated luminosity of Belle ($\int \mathcal{L}_{\text{Belle}}\,\mathrm{d}t = 710\,\text{fb}^{-1}$). The most obvious way to improve this analysis is by increasing the amount of data used. This will make all the fits more stable and will reduce the overall statistical error. Since Belle II is expected to continue collecting more data with an goal of a total integrated luminosity of $\int \mathcal{L}\,\mathrm{d}t = 50\,\text{ab}^{-1}$, it will only be a matter of time until the amount of collected data is large enough to perform a more competitive analysis.

For now, the performance of this analysis can be estimated by making projections for several integrated luminosities. The workflow for these projections can be seen in Fig. 10.1. To make a projection, the PDFs obtained with the MC15_rd dataset are loaded and events are generated with zFit according to the expected amount of events. Then the complete analysis pipeline is started. This means that after the weights are computed and applied, the three dimensional angular fit is performed. This will be done for several total integrated luminosities. The values for the angular parameters are the same for electrons and muons. Fig. 10.2 shows the discriminating and angular fit for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the second q^2-region for an integrated luminosity of $50\,\text{ab}^{-1}$. The fits for the other integrated luminosities and muons can be found in the App. K.

The result of the fits on theses generated datasets can be seen in Fig. 10.3. Here, the obtained values for the second q^2-regions for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels are shown for several integrated luminosities. As expected, the size of the error is getting smaller for higher integrated luminosities. In addition to that, the values seem to converge against the value used for generation. This shows that the fits will become more reliable for higher integrated luminosities starting with about $10\,\text{ab}^{-1}$.

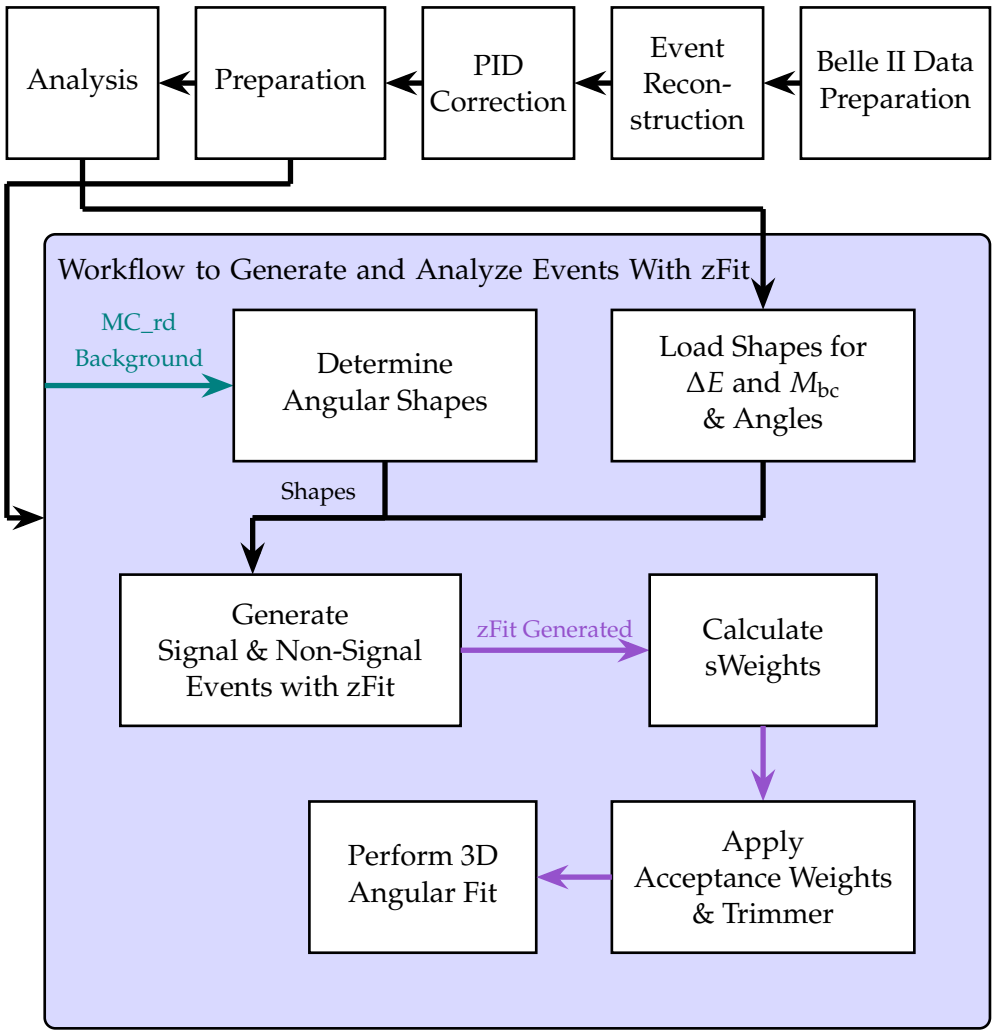


FIGURE 10.1: Workflow to generate data with zFit in order to estimate the error and make projections for several integrated luminosities. This workflow is also used for the error estimation of the final angular fits.

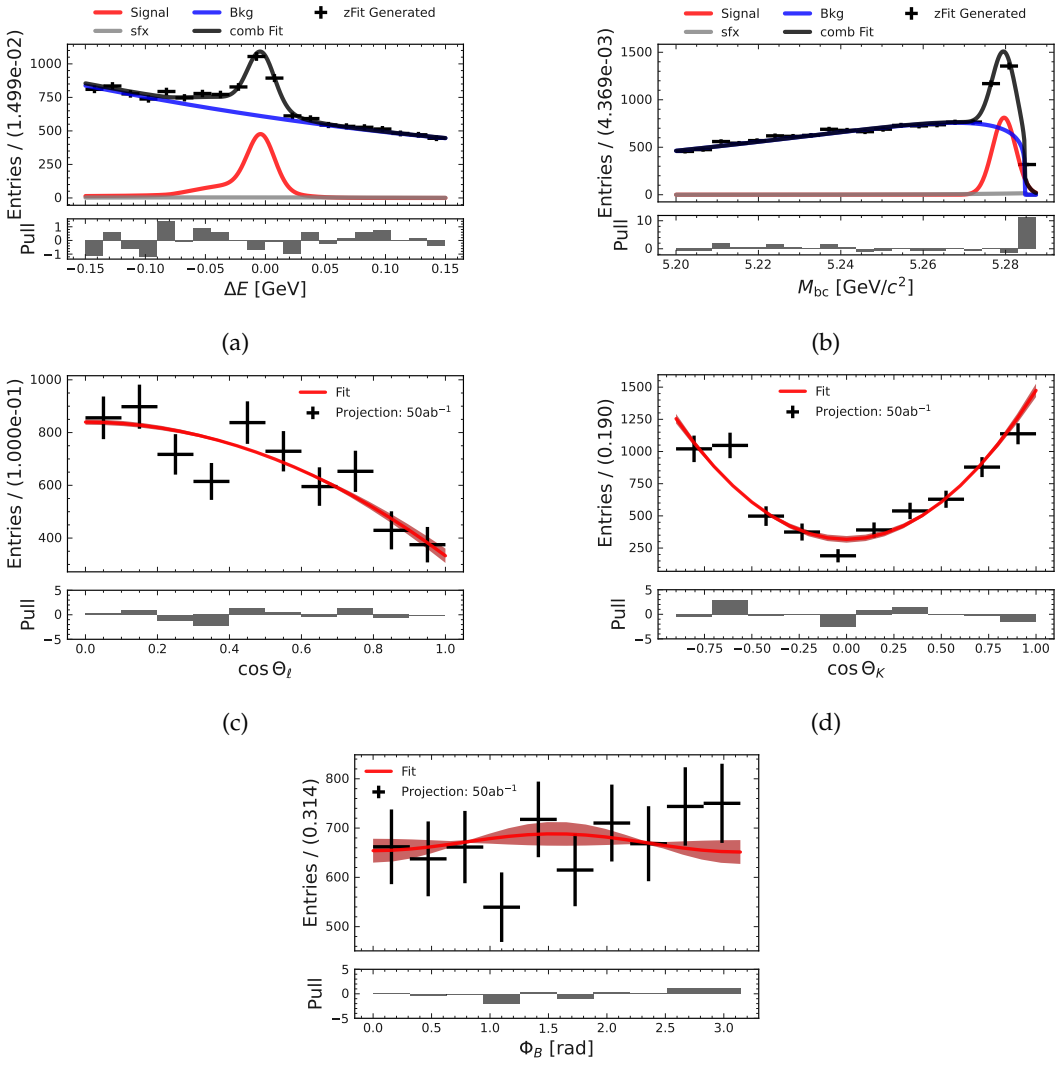


FIGURE 10.2: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 50\,\text{ab}^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$.

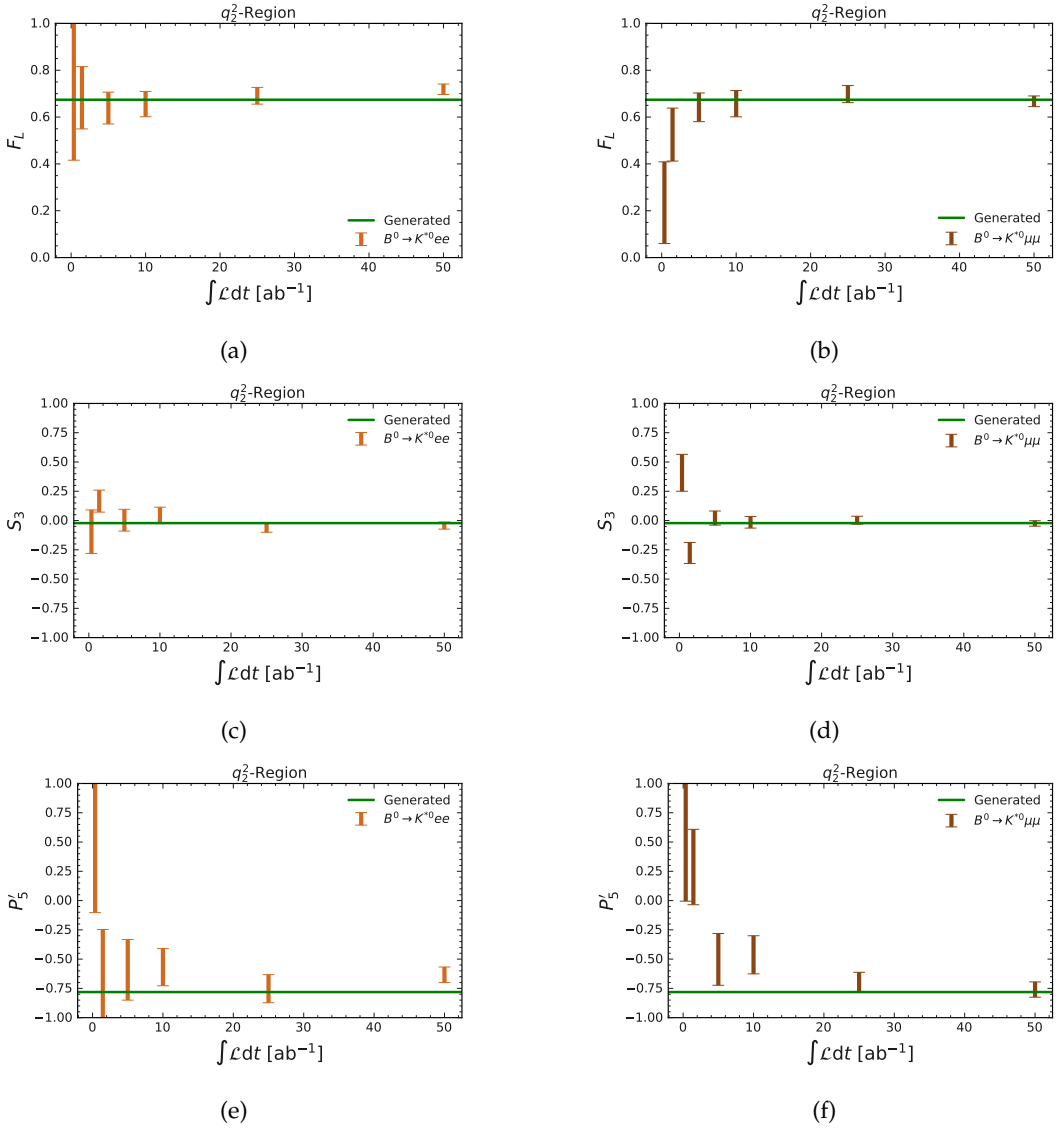
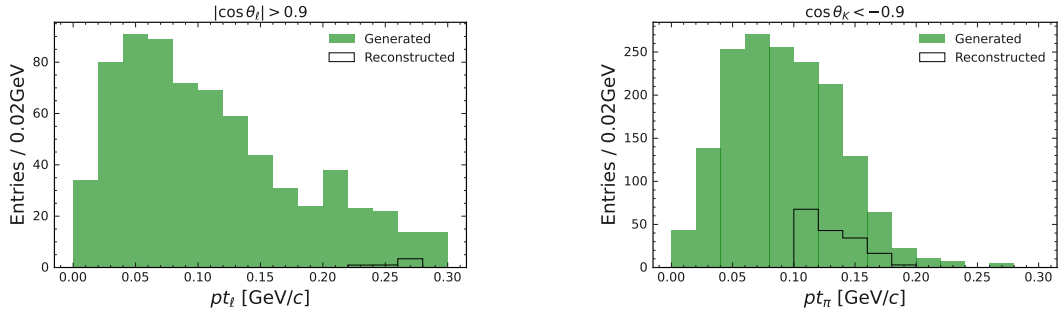


FIGURE 10.3: Computed projection of the angular variables in the second q^2-bin for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channels with $\ell = e$ ($\ell = \mu$) on the left (right). The green line corresponds to the value used to generate these events.

10.2 Recover Low Transverse Momenta Particles

After studying the efficiency as a function of the variables $\cos\theta_\ell$ and $\cos\theta_K$, it appears that there are regions in these variables with a very low efficiency. This was the reason the trimming (Tab. 7.8) was introduced. After studying these regions, the reason for these low efficiencies appears to be the bad reconstruction efficiency of particles with low transverse momentum. This is true for both the lepton (Fig. 10.4a) as well as the π (Fig. 10.4b).



(a) (b)

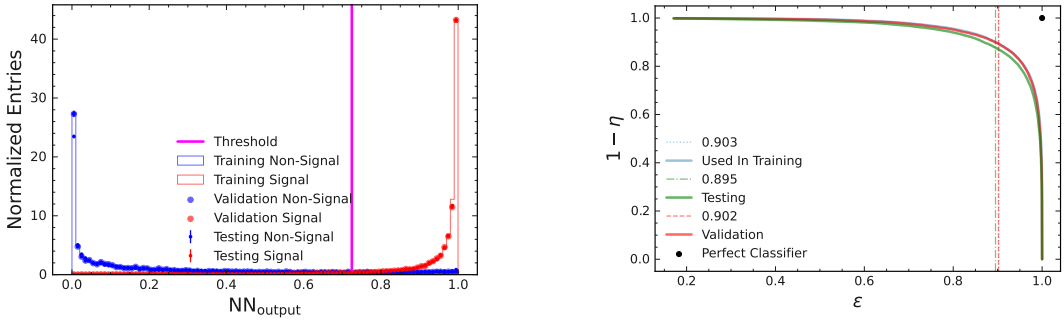
FIGURE 10.4: Number of generated and reconstructed tracks as a function of pt for leptons (a) and pions (b). In (a), only events with $|\cos\theta_\ell| > 0.9$ and in (b), only events with $\cos\theta_K < -0.9$ are taken into account.

Using the innermost detectors of the experiment, (PXD + SVD), the low momentum leptons could be reconstructed. As of now, these events are thrown away by the nCDCHits > 20 condition which is required for PID corrections. Using tracks with low momentum would therefore also mean that new PID correction tables are needed. The transverse momentum of the π on the other hand loses a lot of entries due to the skim, which requires each track to have a transverse momentum of at least 0.1 GeV/c. Using a different skim with a lower requirement on the transverse momentum would therefore improve the reconstruction efficiency of the pions in all channels with a charged π. Using both of these strategies could therefore make the trimming unnecessary and as a result increase the total efficiency and lead to an overall improvement of the three dimensional angular fit.

10.3 Adding the K^*-Mass as a Kinematic Variable

With a larger dataset (Sec. 10.1) it will be possible to also determine the contribution of the S-wave component of the $K^*(892)$ decays. This contribution is not included in the parametrization (Eq. 2.57) and therefore needs to be treated. For this, the window on the invariant mass of the K^* needs to be opened further and a classifying neural network has to be trained without this feature. As a result, it can be added as an additional target variable to the existing kinematic variables of the decay. The training, validation and testing is done with the same dataset as before in Sec. 7.5.3. The performance of this network for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel can be seen in Fig. 10.5 and Tab. 10.1 provides a direct comparison between the classifying model with and without the K^* mass in the training.

Obviously, the performance of the classifier without the M_{K^*} variable in the training performs a bit worse. The efficiency as well as the purity are lower and as a result



(a) (b)

FIGURE 10.5: Classifying model for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ without the mass of the K^* in the training. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.

TABLE 10.1: Comparison between classifying models with and without the K^* mass variable in the training for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel.

Model	Efficiency [%]	Purity [%]	AUC	Threshold
w/ M_{K^*}	15.9 ± 0.2	51 ± 6	0.971	0.735
w/o M_{K^*}	15.7 ± 0.2	47 ± 6	0.960	0.734

the AUC value is lower. However, this is still a good classifier since both values are still comparable.

Besides the performance of the classifying network, it is important that there is no correlation between M_{K^*} and some important variables. The flat correlation plots can be found in Fig. 10.6. In order to use the sWeights method, there must be no correlation between M_{K^*} and both the PID- and acceptance-correction weights since these weights will be applied after the discrimination fit (Figs. 10.6a, 10.6b). Also, since M_{K^*} is a part of the target variables, it has to be uncorrelated to the discriminating variables M_{bc} and ΔE (Figs. 10.6c and 10.6d). Due to the fact, that the K^* mass will be used a target variable alongside $\cos\theta_\ell$, $\cos\theta_K$ and ϕ_B it would be nice, but not crucial, that these variables are uncorrelated among themselves. This can be seen in Figs. 10.6e, 10.6f and 10.6g.

10.4 J/ψ-Leakage

Even though a charmonium veto is applied it is still possible that some J/ψ events appear in the q^2-region due to the width of the J/ψ resonance and the detector resolution. This is a problem for the electron channels especially due to bremsstrahlung. The remaining J/ψ events in the $B \to K^{*0}(\to K^+\pi^-)ee$ channel can be seen in Fig. 10.7a. Here, it is clear that some J/ψ events end up in the signal regions and since the J/ψ decay channel is very similar to the signal channel, it is difficult to exclude it further without losing more signal events.

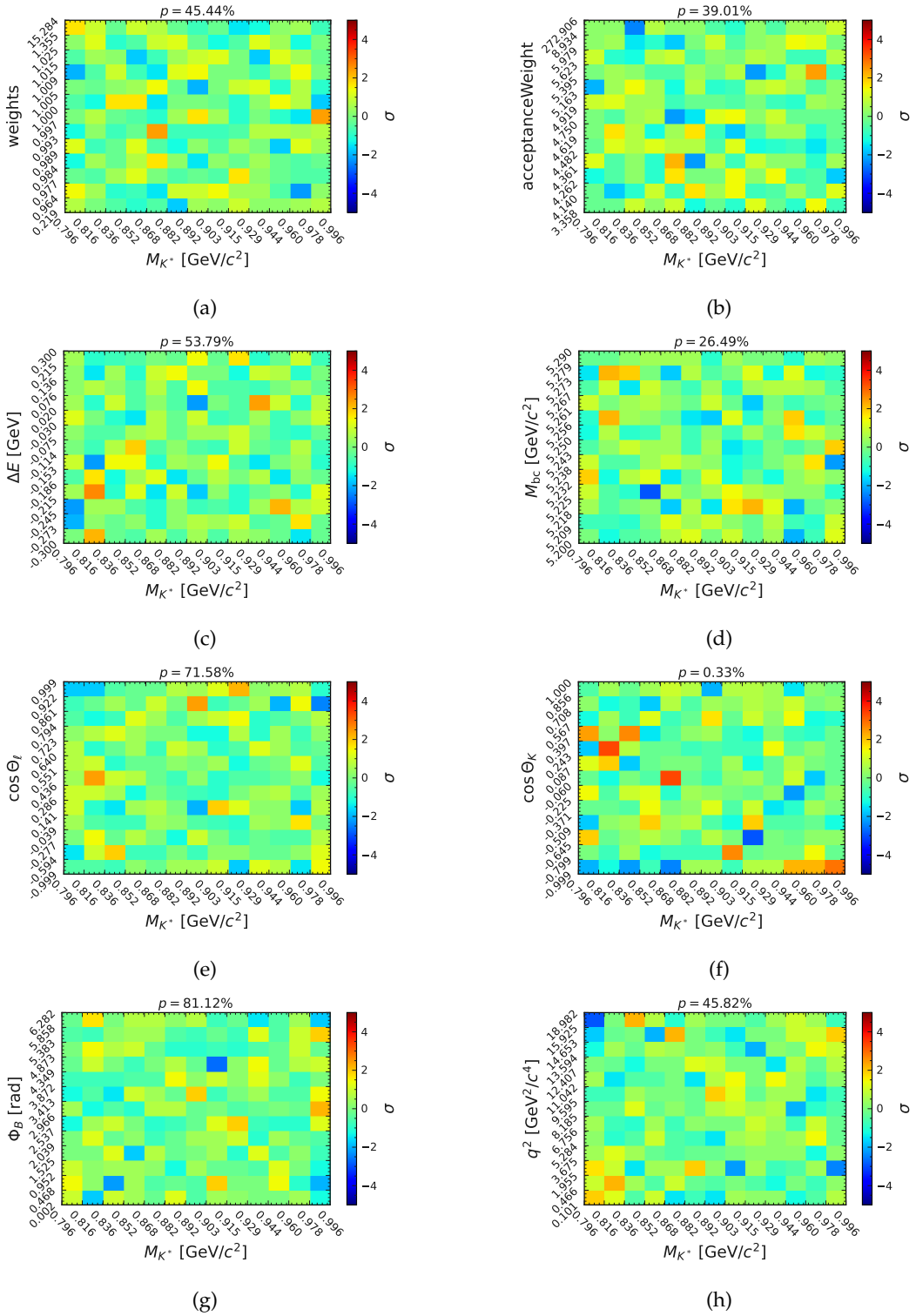


FIGURE 10.6: Correlation between the K^* mass and q^2 (f), weights ((a) and (b)) and variables used for discrimination ((c) and (d)) and angular fitting ((e) - (h)).

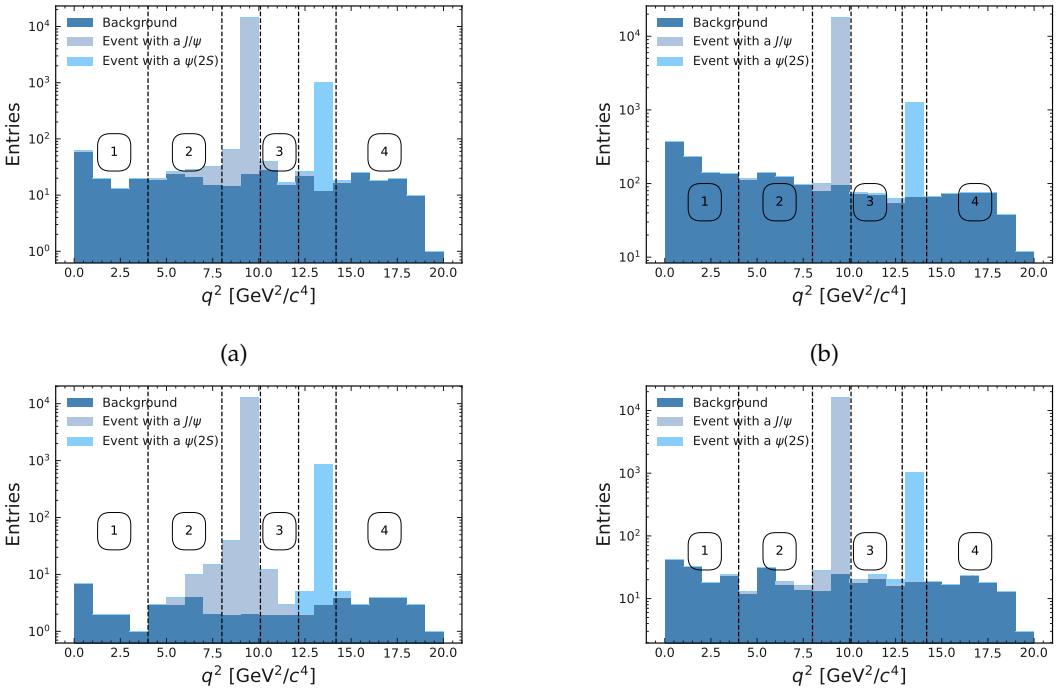


FIGURE 10.7: Pseudo signal leak in signal regions for the channel $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$. (a) ($\ell = e$) and (b) ($\ell = \mu$) show the datasets without passing the neural network. (c) ($\ell = e$) and (d) ($\ell = \mu$) is the same dataset but only candidates which pass the neural network classifier are shown. All figures show pure background. Signal and sfx candidates have been removed.

In Fig. 10.7b the same decay channel but with μ as leptons is shown. Since the energy resolution of the detector performs better for μ compared to electrons, the width of the J/ψ peak is much narrower and there is almost no leak in the signal regions. All figures presented here contain background events only. Signal and sfx events have been removed to make the J/ψ and $\psi(2S)$ contribution to the background more clear.

In Figs. 10.7c and 10.7d, the same dataset is shown for electrons and muons, respectively. However, these figures only contain events which pass the neural network classifier. It appears that a large portion of the background events in the second q^2_2-region still have a J/ψ in them. In order to handle this contribution, a clean $B \to K^* J/\psi(\to \ell\ell)$ sample should be created, preferably in the $\ell = \mu$ channel. Then the angular distribution of these charmonium events should be determined and their amount in the second q^2-region should be estimated. Afterwards this contribution should be fixed. This approach should also be possible on real data without relying on simulation since the J/ψ events dominate and it should be possible to create a rather clean sample. However, studies concerning this approach still have to be performed.

10.5 Recover $B^+ \to K^{*+}(\to K^+\pi^0)\ell\ell$ Channels

In Sec. 7.8.1 a correlation between $\cos\theta_K$ and M_{bc} was found for the $B^+ \to K^{*+}(\to K^+\pi^0)\ell\ell$ channels which caused the channels to be dropped.[1] This correlation has to be studied in greater detail and the question has to be answered if there is a way to recover these channels.

10.6 Error Estimation

For this analysis the errors are estimated calculating the mean of the covariant matrices obtained by fitting over zFit generated toy MC 50 times (Sec. 8.1). This had to be done since the angular fit did not always converge for very low integrated luminosities, especially for the LS1 data set. In general, the estimation of the error on the final values has to be more sophisticated. With in increase in the available integrated luminosity the stability of the angular fits will increase as well and as a result, the covariance matrix can be obtained directly from the fit reliably. In addition to doing a better estimation of the errors, biases originating from the fits have to be studied further and the corrections need to be applied. A first approach of these studies was done in [19].

10.7 Fixing the Background Shape

In Sec. 9.5 the analysis was done without the sWeight method. Instead the background shape is determined and fixed. For future analyses, it would be nice to keep this methods as a sanity check. In Sec. 10.1 it was shown, that the sWeight method will converge against the generated value without relying on a background contribution determined by simulation. It will return values close to the generated ones once the amount of available data is large enough. On the other hand, fixing the background shape is a rather common and established strategy in particle physics. However, this approach has several disadvantages. For it to work, simulation and real data have to be very similar and a good model for the background has to be found. Both of these requirements are vulnerable to errors.

[1] The discriminating and the target variables have to be uncorrelated in order to use the sWeights method.

Chapter 11

Conclusion

> It's the job that's never started as takes longest to finish.
>
> J.R.R. Tolkien, *The Lord of the Rings* (1954)

This thesis shows that an angular analysis of the $B \to K^* \ell\ell$ decay is feasible at Belle II. The most considerable part of this work focuses on creating a workflow with new techniques to execute this kind of analysis.

It starts with the development of snapshot ensemble neural networks which are able to reliably differentiate between signal and non-signal events (Sec. 7.5) while keeping the background side-band of the ΔE and M_{bc} variables *fittable*[1]. It is also shown that the performances of the classifiers developed in this analysis are better compared to the ones used at Belle (Tab. 7.5).

The next major step in the analysis pipeline is the novel approach to approximate the response function of the detector with another neural network (Sec. 7.7). Finally, a rather new technique which is able to get rid of the non-signal events contribution in the angular variables is introduced and demonstrated (Sec. 9.3). This sWeights method returns weights that allow for a direct likelihood fit on *sweighted* events in order to obtain the true signal shape. Using this approach the background contribution is no longer determined by simulation eliminating possible sources of error. In general, this method has to be studied with greater effort in the future and the resulting systematic and statistical errors have to be understood in more detail. However, the advantages of the sWeight method can not be denied and I think it is the way to go for future analyses.

After applying this sWeight method, a data transformation is applied which reduces the dimensions of the angular decay rate from eight down to three dimensions. As a result, the fit is only sensitive to the observables F_L, S_3 and P_5'. In order to validate the developed workflow, these parameters are determined with a multi-dimensional fit on the angular variables for the MC15_rd (Sec. 8.2) and LS1 (Sec. 8.3) dataset.

As a next step of the analysis, the overall errors have to be studied with far more detail in general. This also includes the impact of events coming from the charmonium resonances, which migrate into lower q^2-regions and the S-wave component of the $K^*(892)$ decays.

Belle II only started taking data in 2019 and so far has collected a total integrated luminosity of $\int \mathcal{L}_{\text{LS1}}\, \mathrm{d}t = 364\,\text{fb}^{-1}$ at the $\Upsilon(4S)$ resonance which is about half of its

[1] The output of the classifier does not correlate strongly with these variables leaving a background and a signal side-band.

predecessor Belle. Policies aside, the goal of this thesis is not the exact determination of the parameter F_L, S_3 and P_5' with data which is currently available, but rather the development of a workflow to obtain these parameters in the future at Belle II. Therefore, this rather small dataset of Belle II is the most obvious bottleneck of this analysis at the moment. As of right now, the amount of expected signal events is in the single to low double digits making it very difficult to perform a three dimensional angular fit. However, Belle II continues to collect data and is expected to do so until 2035 when a total integrated luminosity of $\int \mathcal{L}\,\mathrm{d}t = 50\,\mathrm{ab}^{-1}$ will be reached, increasing the number of expected signal events to over one thousand. With this amount of events, the analysis will become more stable and the size of the errors will decrease drastically.

Overall, this analysis approach is promising. Neural networks can be used reliably as classifiers and for function approximations and with the sWeight method it is possible to obtain the true shape of the kinematic variables without relying on simulation for background modeling. Once the integrated luminosity of Belle II reaches about $\int \mathcal{L}\,\mathrm{d}t = 10\,\mathrm{ab}^{-1}$ I would recommend to redo this analysis with the presented workflow.

List of Figures

List of Tables

Appendix A

Theoretical Foundations

A.1 Prediction of the *c* Quark Through the GIM Mechanism

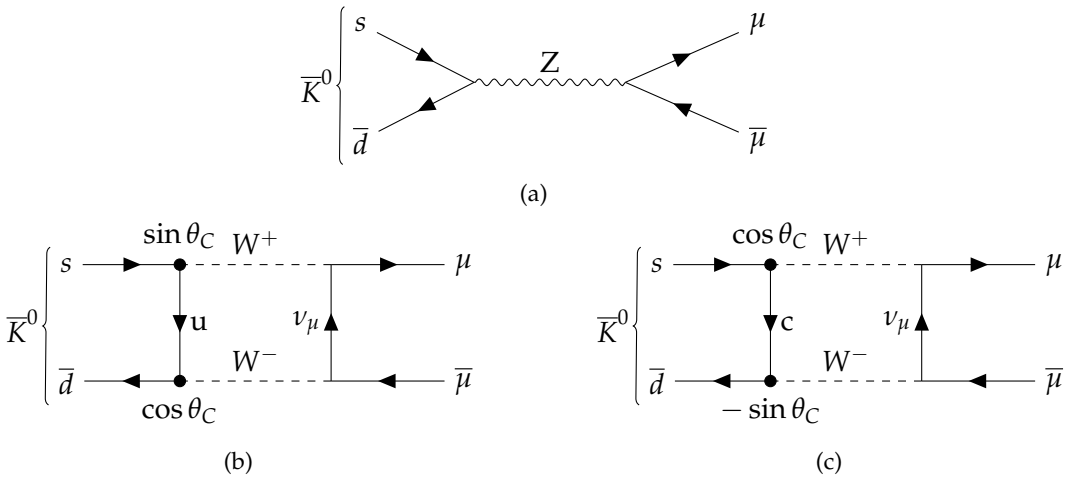


FIGURE A.1: The $K^0 \to \mu^+\mu^-$ decay is extremely suppressed. This lead to the introduction of the GIM mechanism forbidding (a). The transition depicted in (b) is possible according to the SM (θ_C is the Cabibbo angle) and assuming there are only u, d and s quarks this decays should have been observed. Introducing a c quark (4th quark) also enables the transition shown in (c). Adding up the amplitudes of both decays nullifies the total amplitude (caused by the minus sign at the vertex in (c)) and, as a result, this decay is highly suppressed.

Appendix B

Experimental Setup

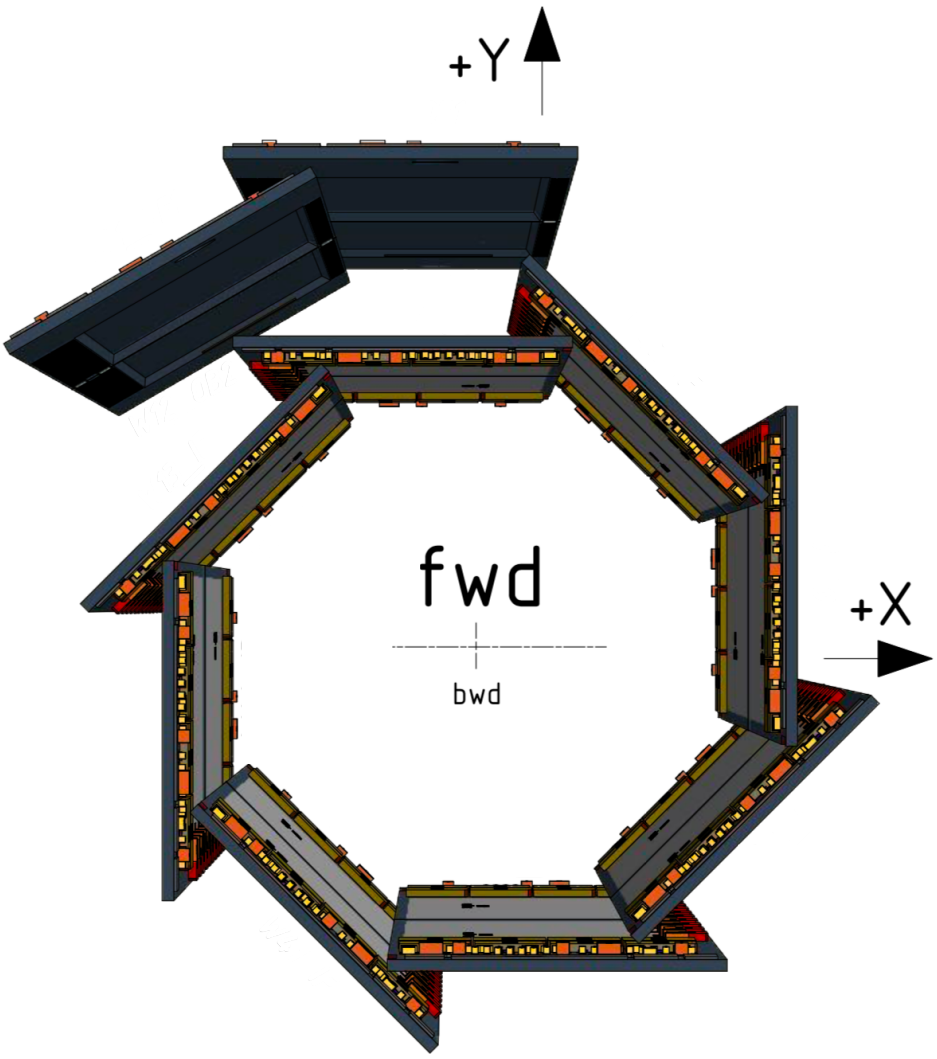


FIGURE B.1: Installed PXD before the first long shutdown. Only two out of the 12 outer ladders were installed [99].

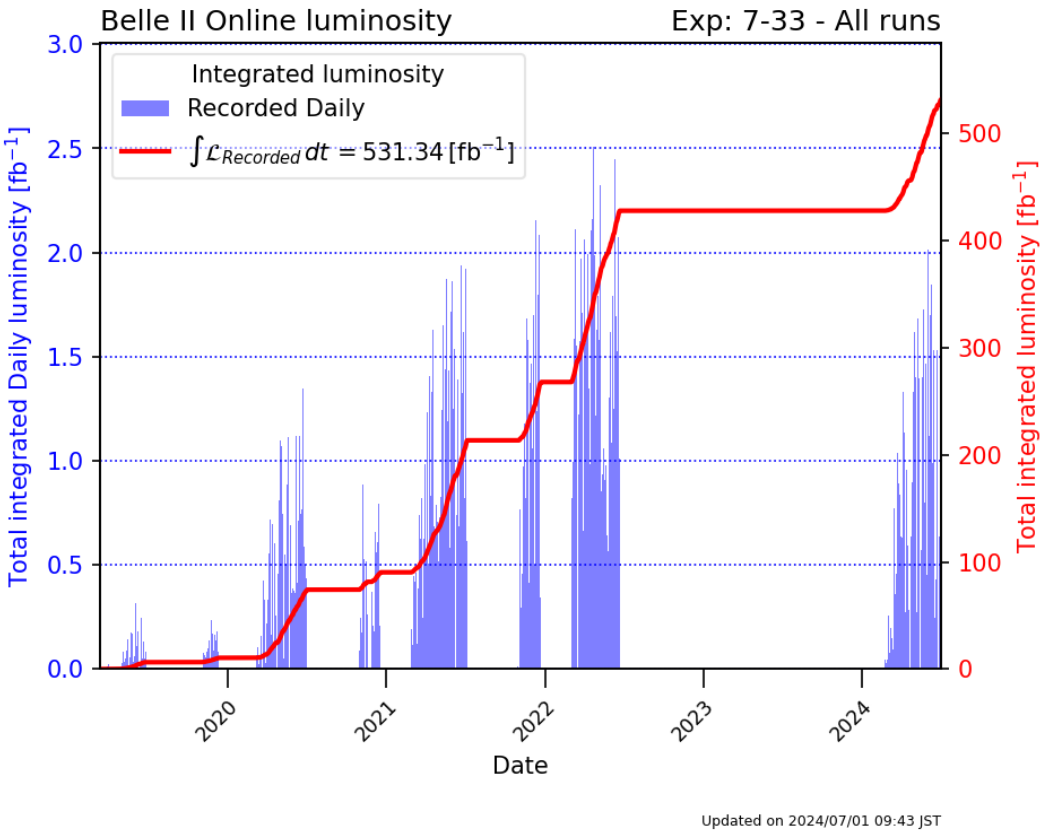
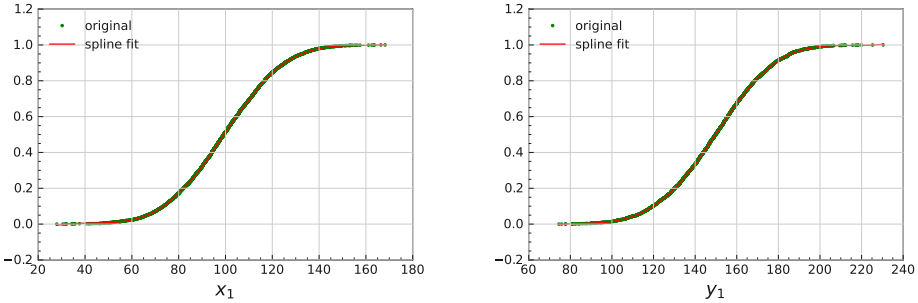


FIGURE B.2: Most recent (integrated) luminosities as of summer 2024. A total integrated luminosity of $\int \mathcal{L}\, \mathrm{d}t = 531.34\,\mathrm{fb}^{-1}$ has been recorded at Belle II. The data taking was paused in July of 2024 in order to perform some more upgrades of the detector. In October 2024 data taking will continue [100].

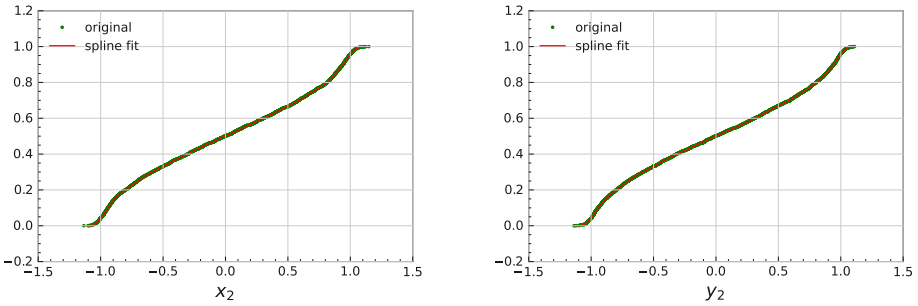
Appendix C

Analysis Tools



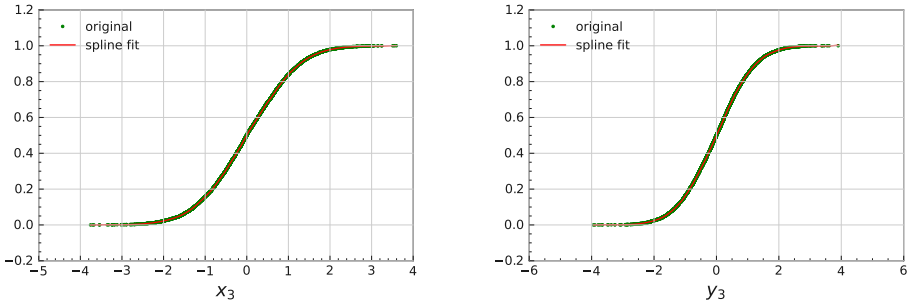
(a) (b)

FIGURE C.1: Spline fit to the line correlation example in Fig. 4.2.



(a) (b)

FIGURE C.2: Spline fit to the circle correlation example in Fig. 4.3.



(a) (b)

FIGURE C.3: Spline fit to the two dimensional gaussian correlation example in Fig. 4.4.

Appendix D

MC-Data Comparison

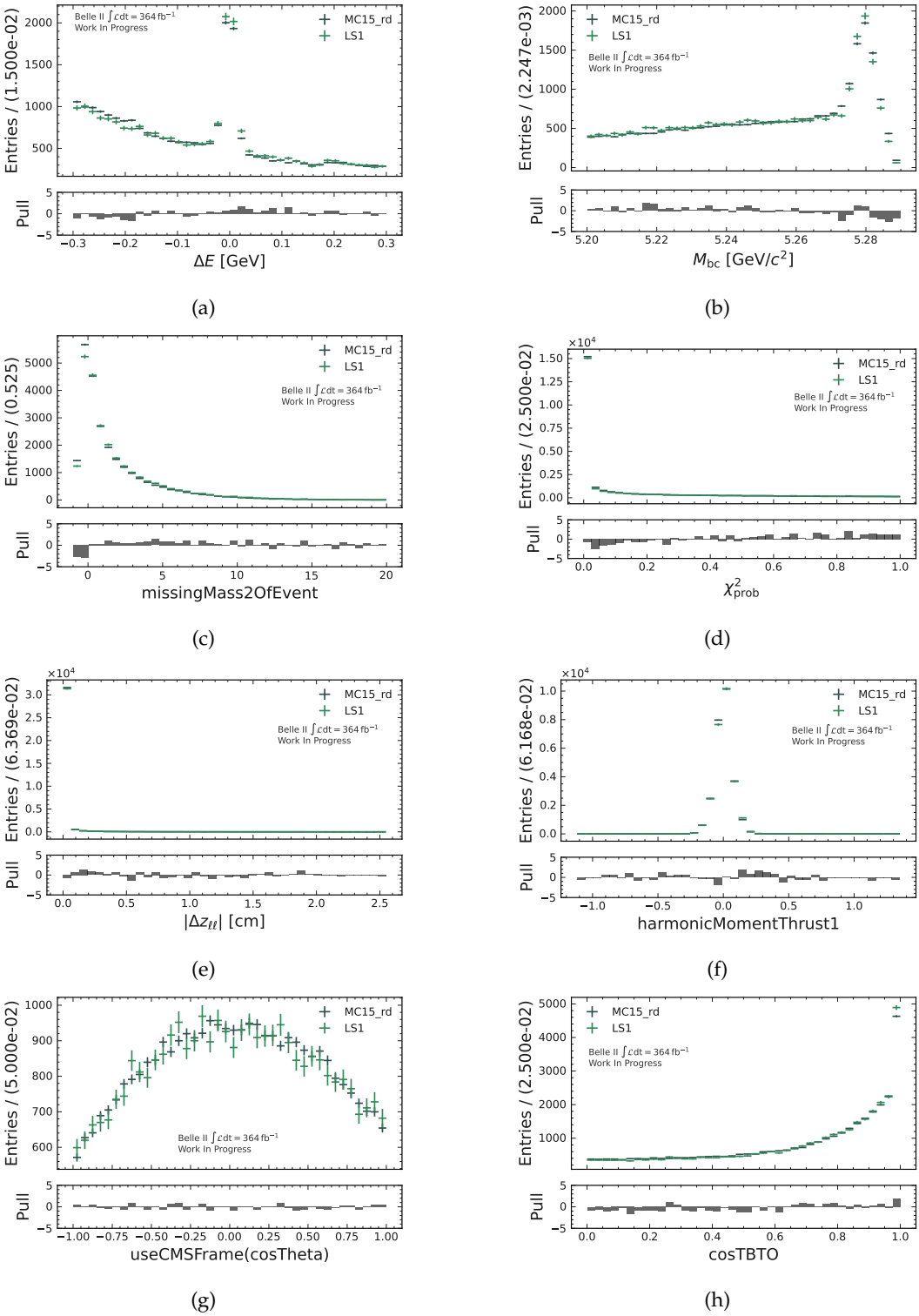


FIGURE D.1: Comparison between LS1 data and MC15_rd for the decay channel $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.

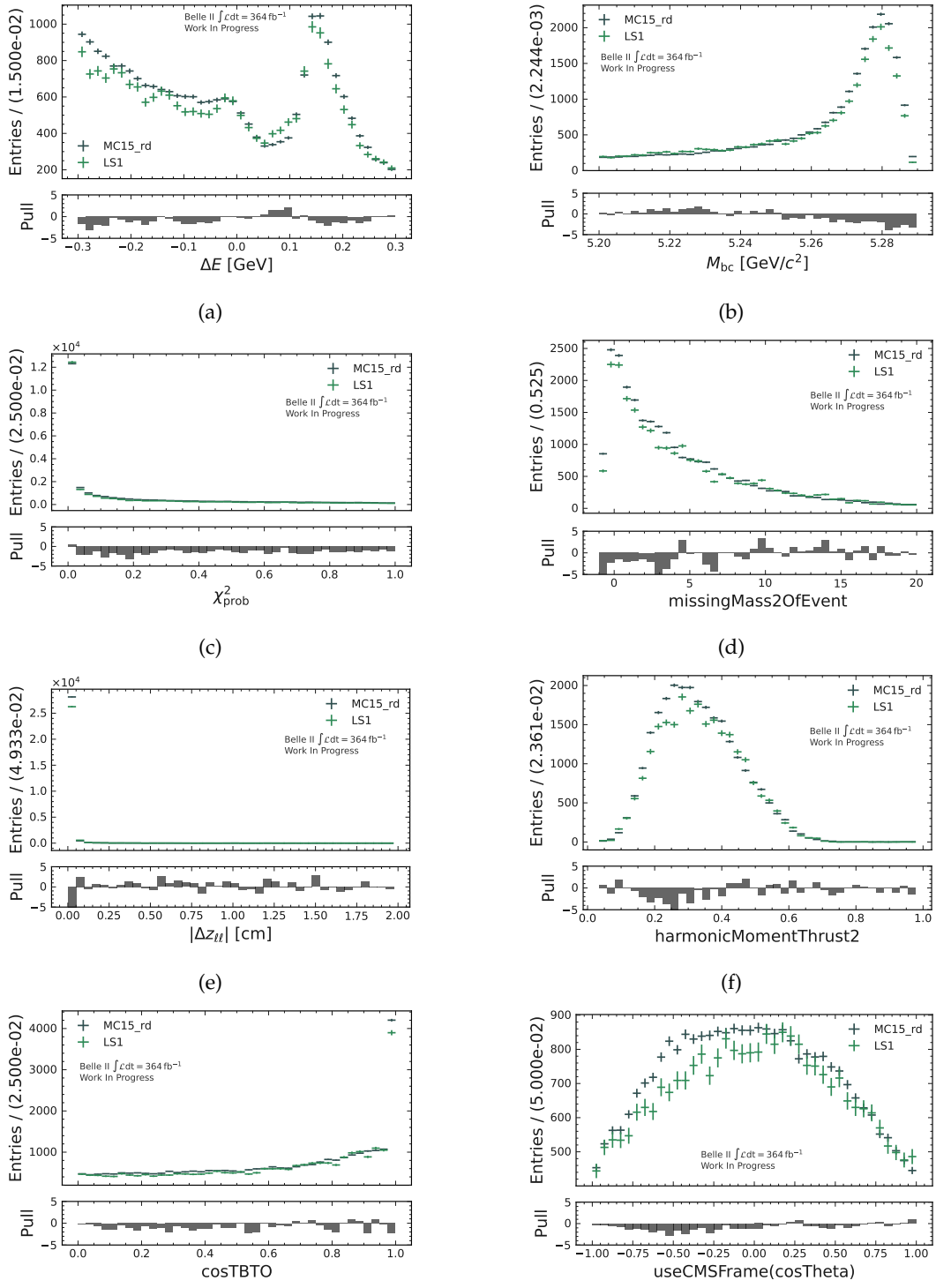


FIGURE D.2: Comparison between LS1 data and MC15_rd for the decay channel $B^0 \to K^{*0}(\to K^0_S \pi^0) ee$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.

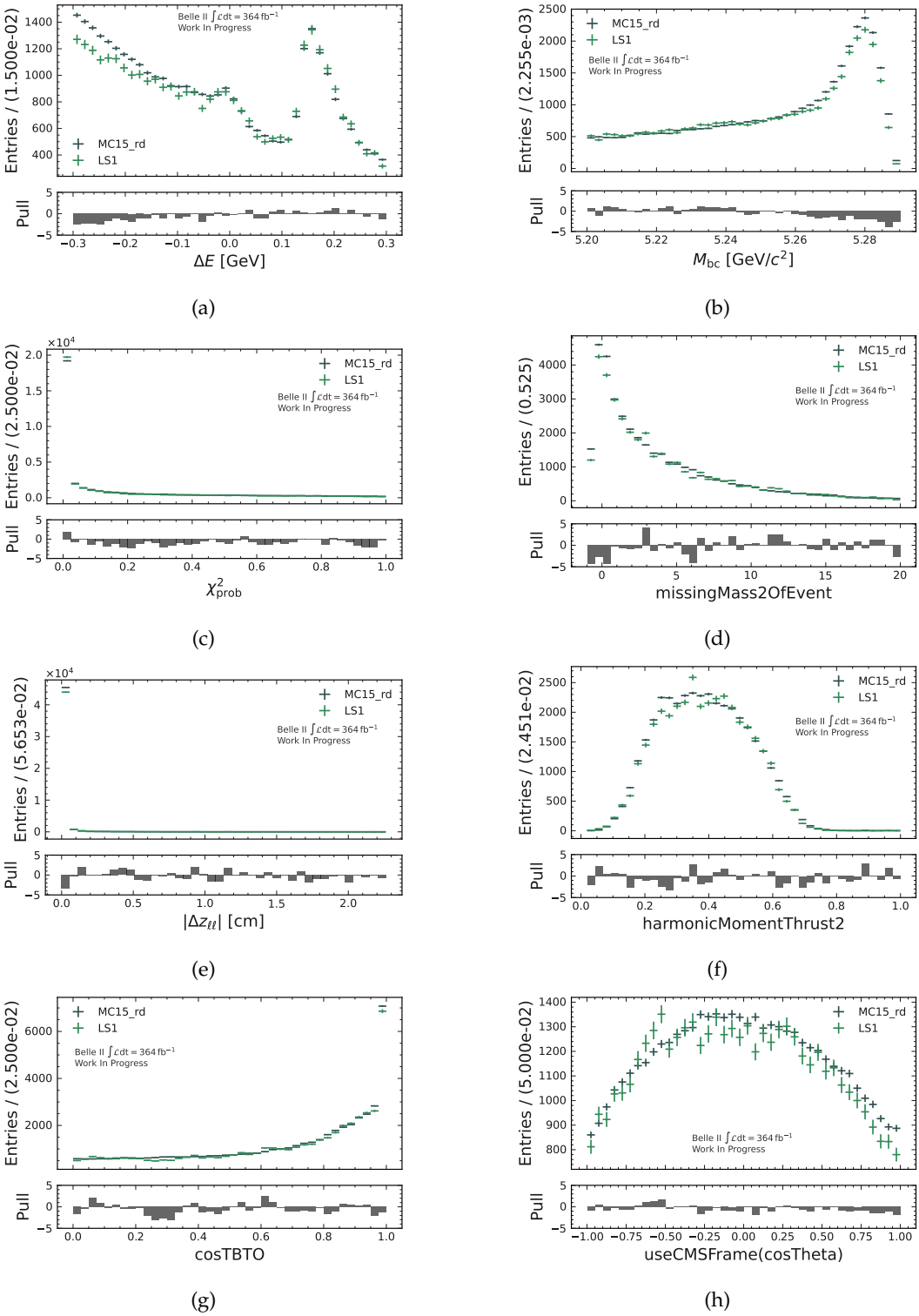


FIGURE D.3: Comparison between LS1 data and MC15_rd for the decay channel $B^0 \to K^{*0}(\to K^0_S \pi^0)\mu\mu$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.

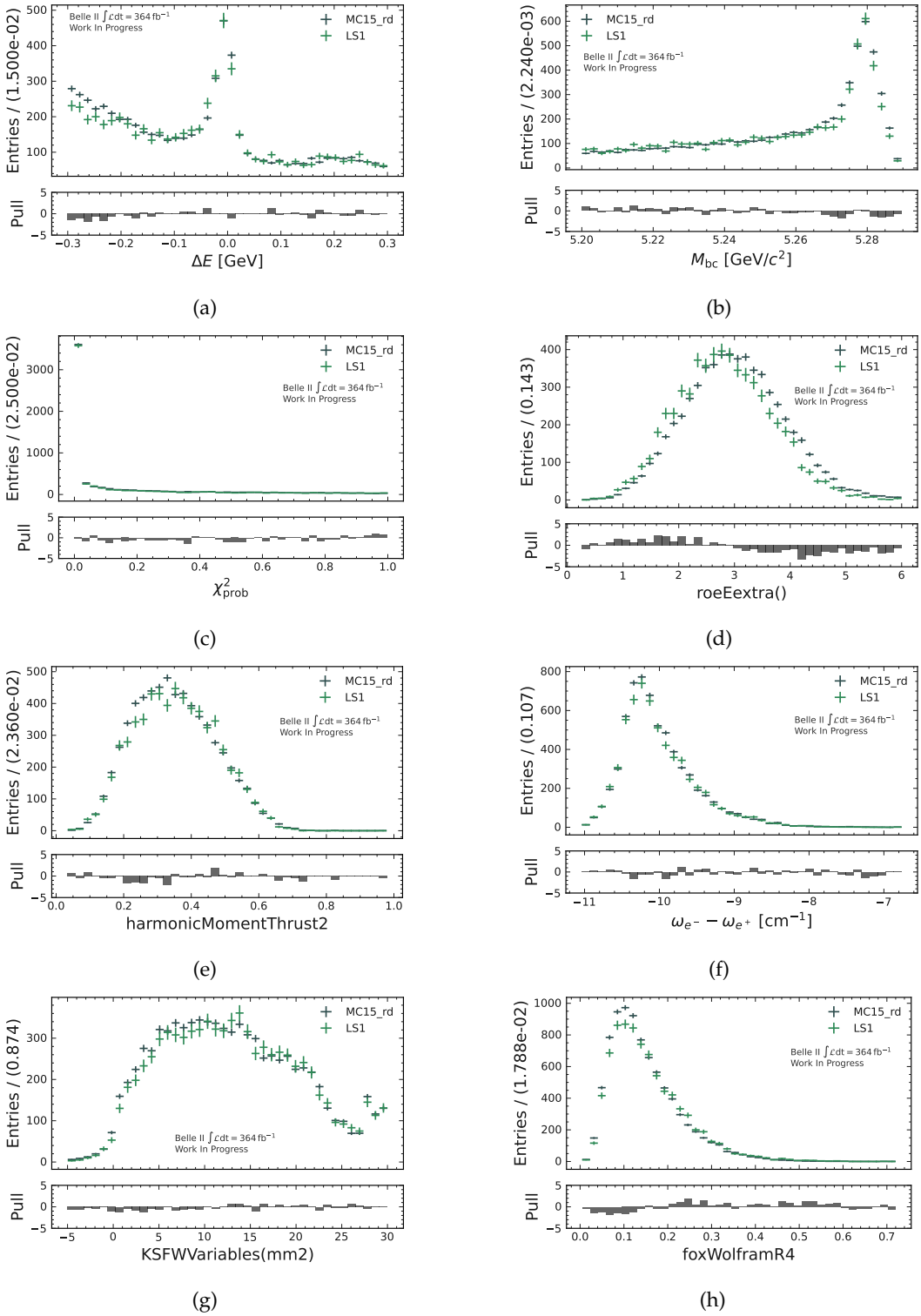
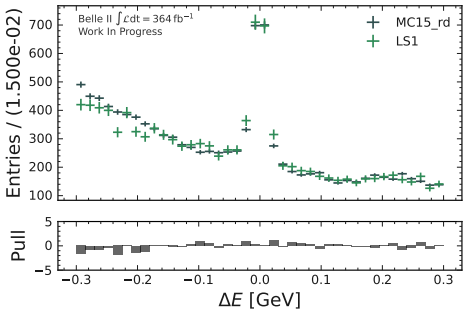
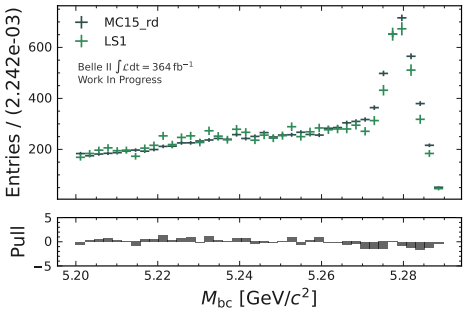


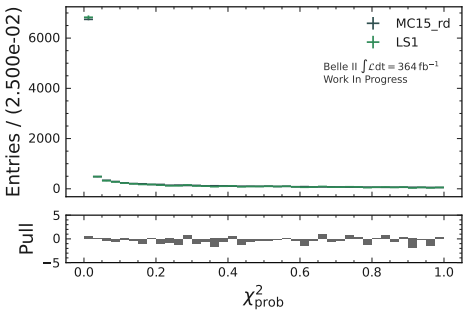
FIGURE D.4: Comparison between LS1 data and MC15_rd for the decay channel $B^+ \to K^{*+}(\to K^0_S \pi^+)ee$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.



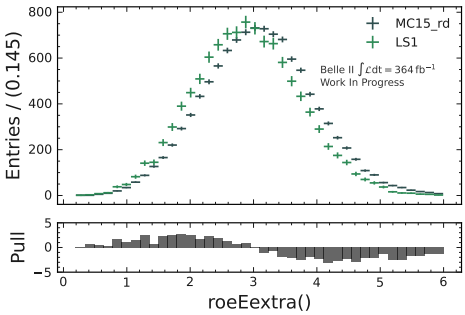
(a)



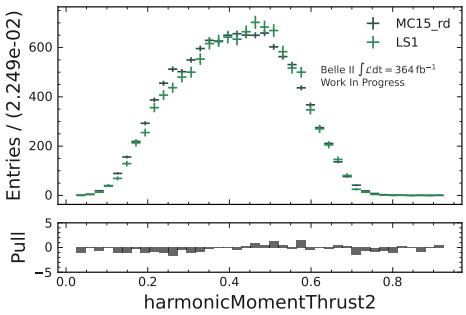
(b)



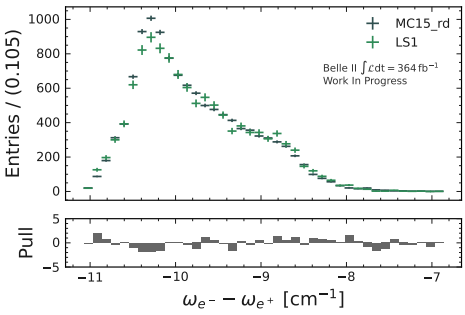
(c)



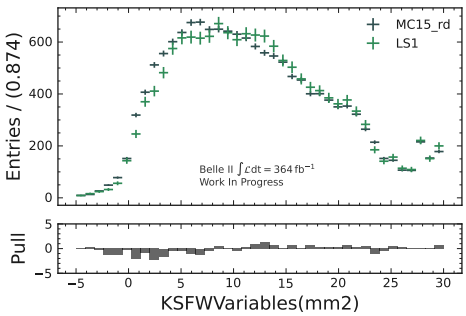
(d)



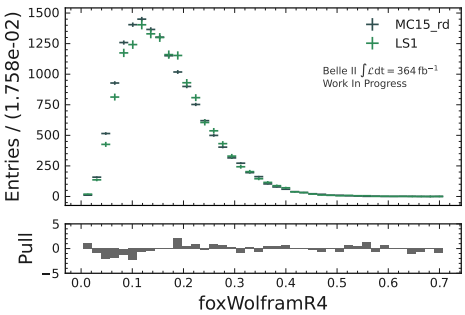
(e)



(f)



(g)



(h)

FIGURE D.5: Comparison between LS1 data and MC15_rd for the decay channel $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.

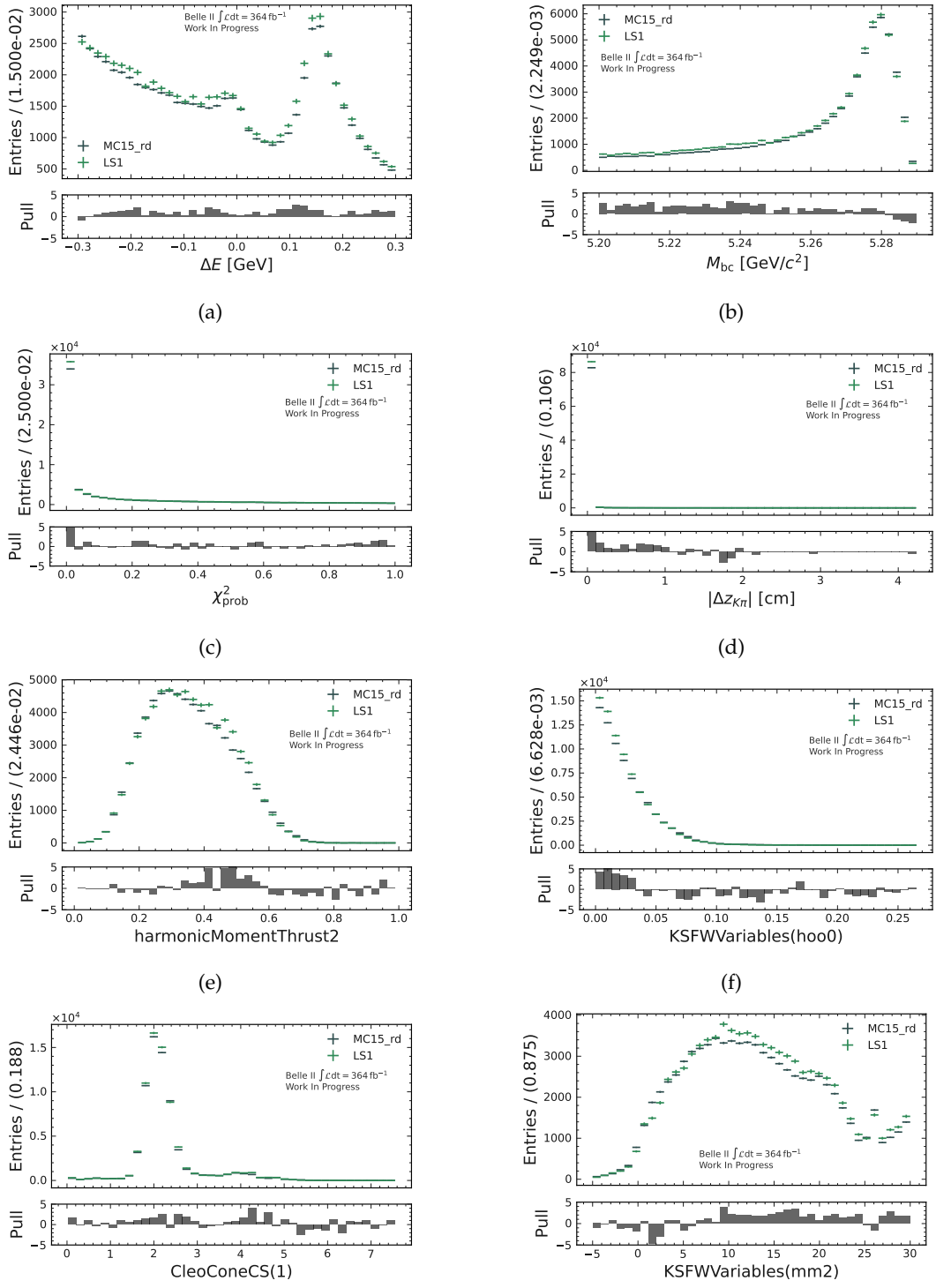


FIGURE D.6: Comparison between LS1 data and MC15_rd for the decay channel $B^+ \to K^{*+}(\to K^+\pi^0)ee$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.

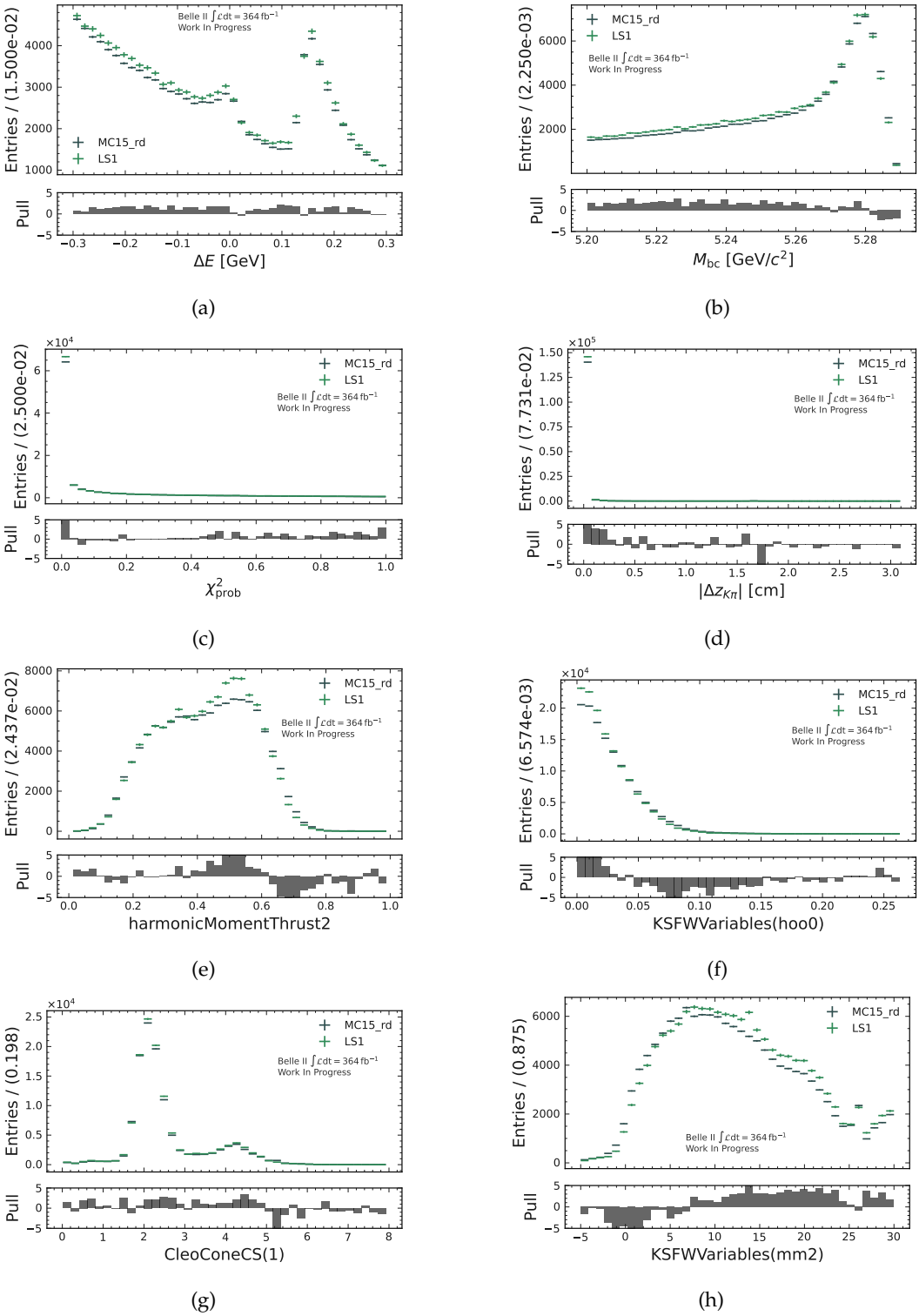


FIGURE D.7: Comparison between LS1 data and MC15_rd for the decay channel $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$. (a) and (b) show the discriminating variables ΔE and M_{bc}, respectively. (c) - (h) show the comparison for the six most important features for classification of this decay channel.

Appendix E

Classifying Network Preparation

E.1 Handling Outliers / Scaling

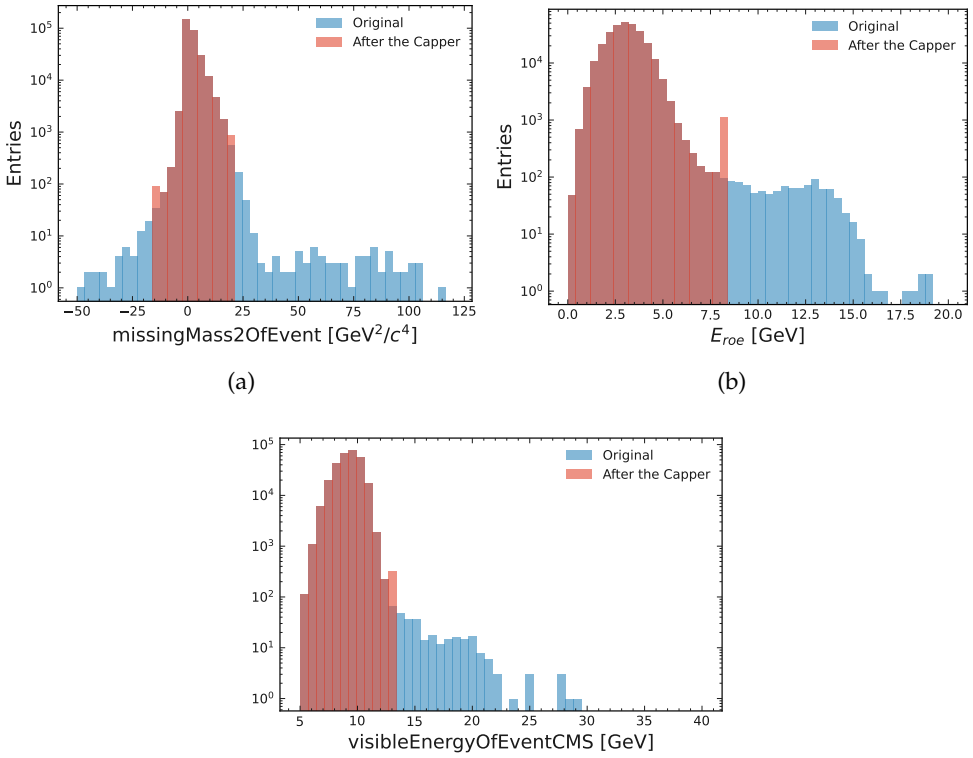


FIGURE E.1: The result of using the capper on the features *missingMass2OfEvent* (a), energy of the rest of event (b) and *visibleEnergyOfEventCMS* (c). The few outliers are moved closer to the rest of the values.

E.2 Feature Engineering

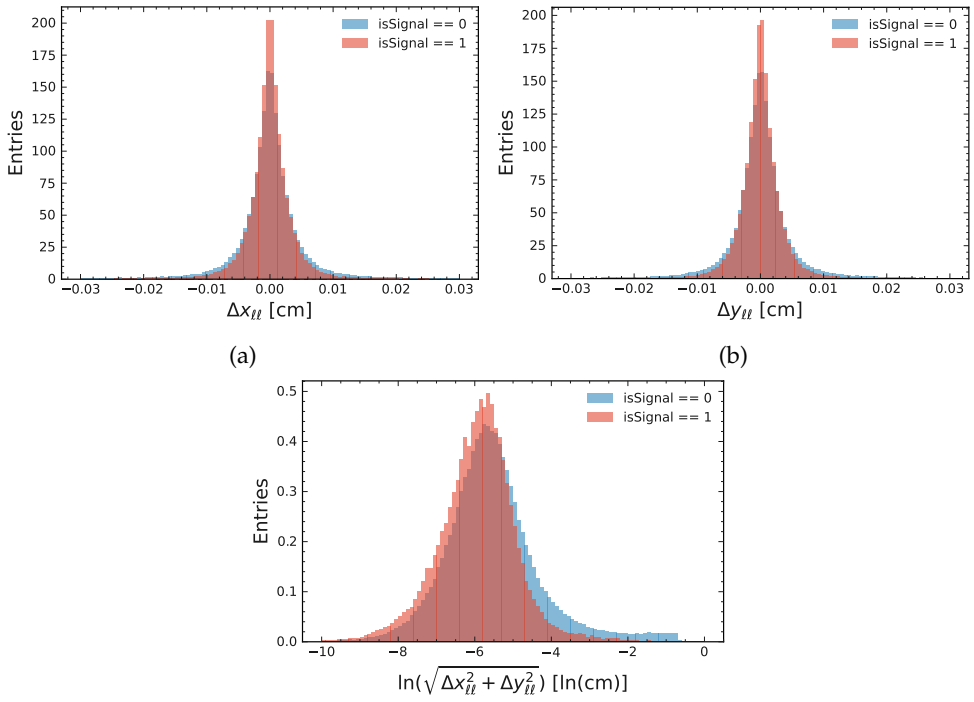
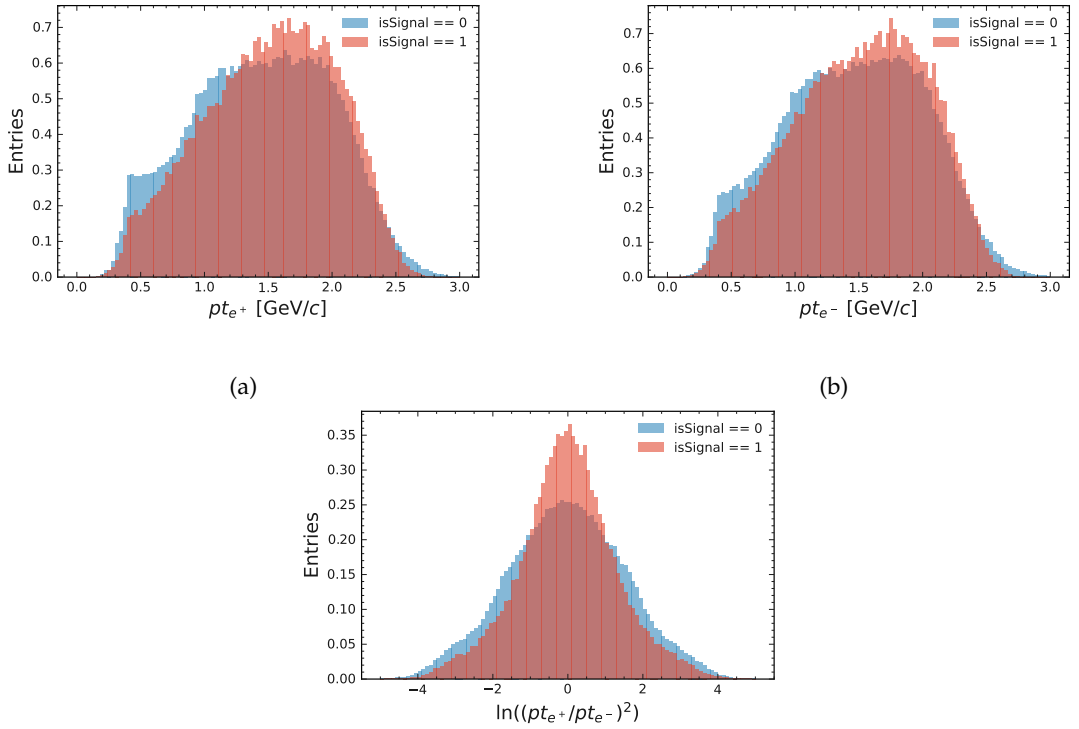


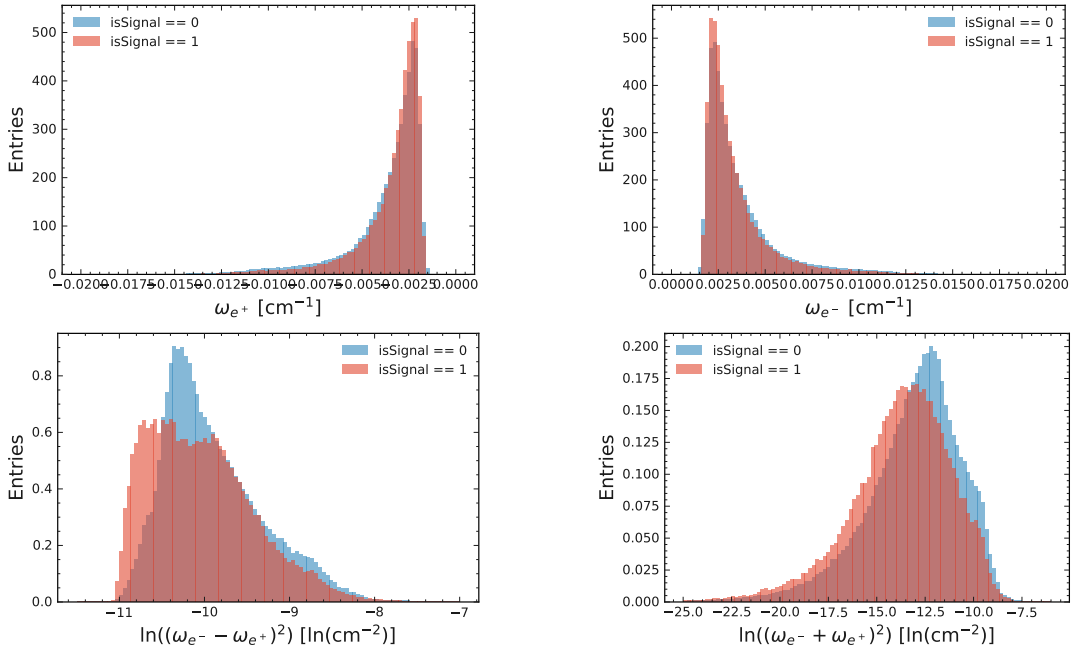
FIGURE E.2: Example for the creation of a new feature. Now signal and non-signal can be discriminated better than before.



(a) (b)

(c)

FIGURE E.3: Transverse momenta of the leptons are shown in (a) and (b). Ratio of the transverse momenta of both reconstructed leptons is shown in (c).



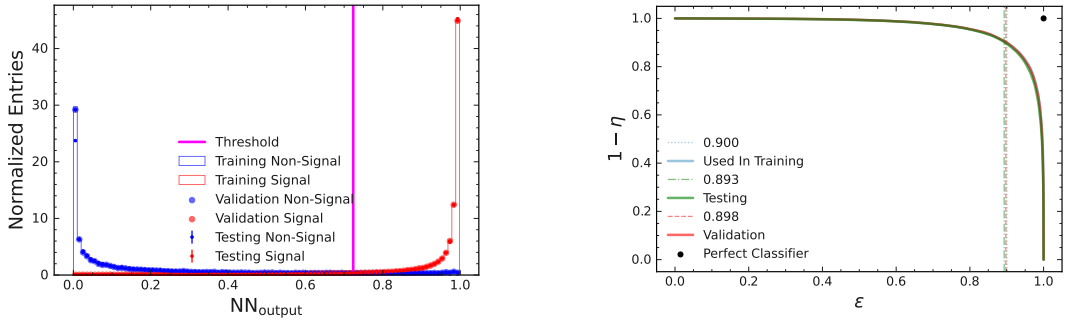
(a) (b)

FIGURE E.4: Feature engineering with the curvature variable of the lepton tracks.

Appendix F

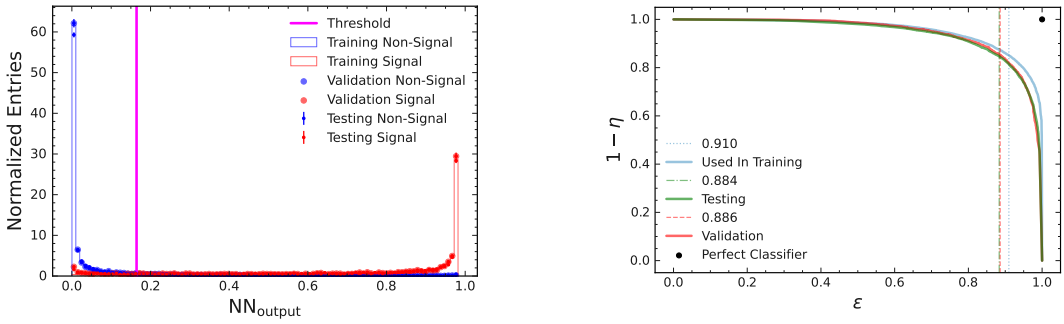
Training Of The Neural Network - Classifier

F.1 Neural Network Classifier



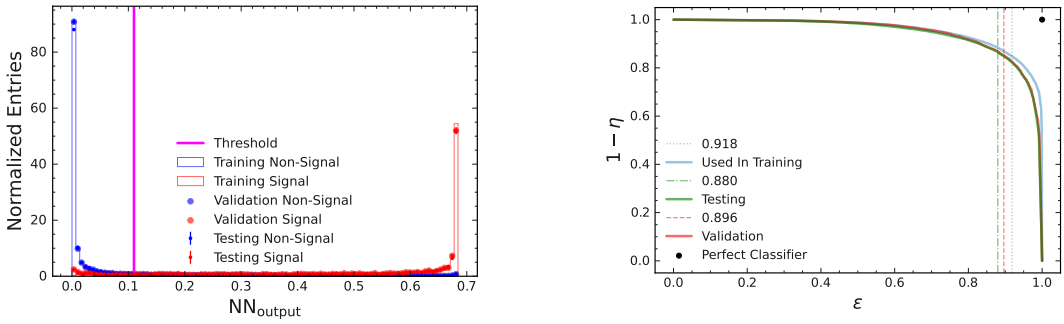
(a) (b)

FIGURE F.1: Classifying model for the channel $B^0 \rightarrow K^{*0}(\rightarrow K^+\pi^-)\mu\mu$. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.



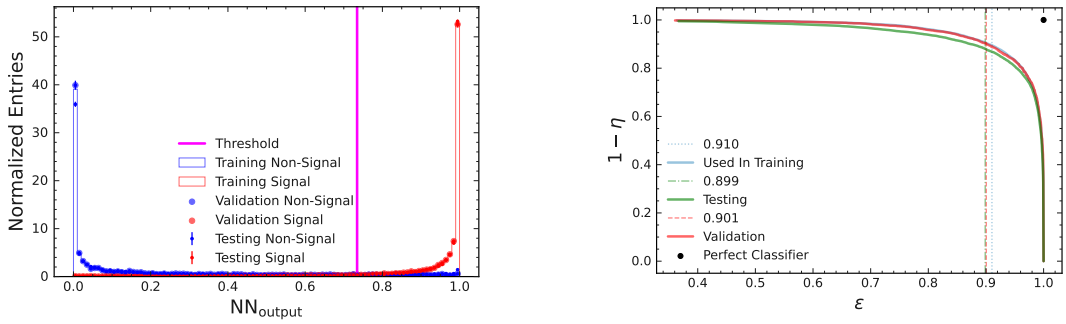
(a) (b)

FIGURE F.2: Classifying model for the channel $B^0 \to K^{*0}(\to K^0_S \pi^0)ee$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.



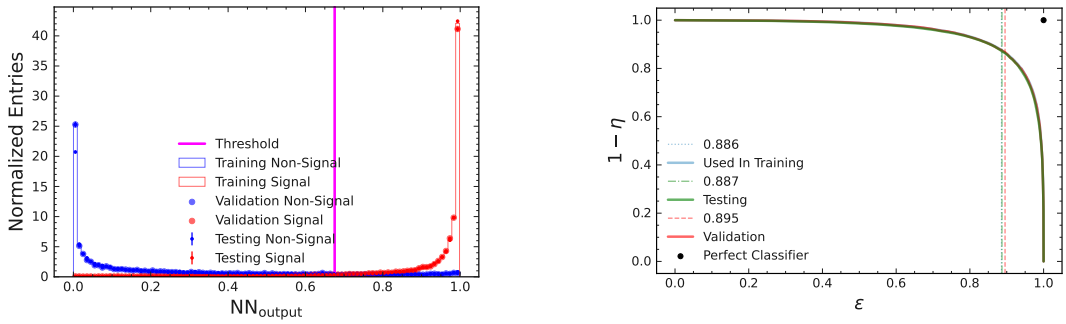
(a) (b)

FIGURE F.3: Classifying model for the channel $B^0 \to K^{*0}(\to K^0_S \pi^0)\mu\mu$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.



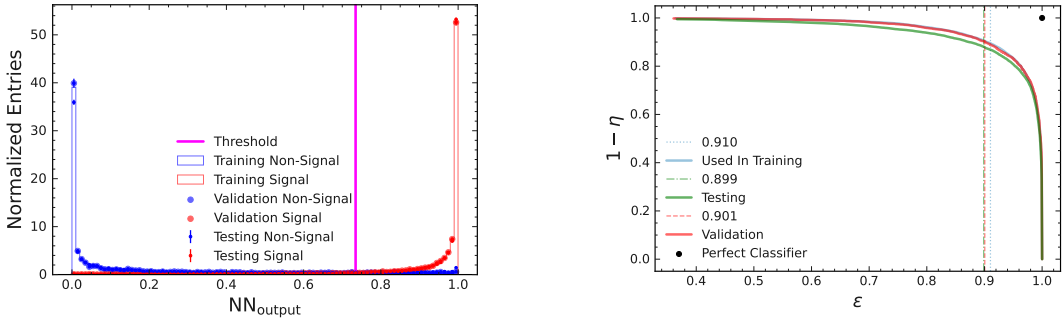
(a) (b)

FIGURE F.4: Classifying model for the channel $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.



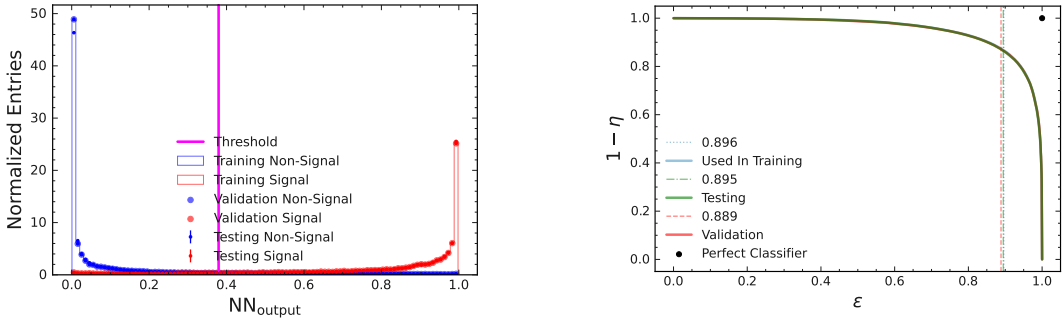
(a) (b)

FIGURE F.5: Classifying model for the channel $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.



(a) (b)

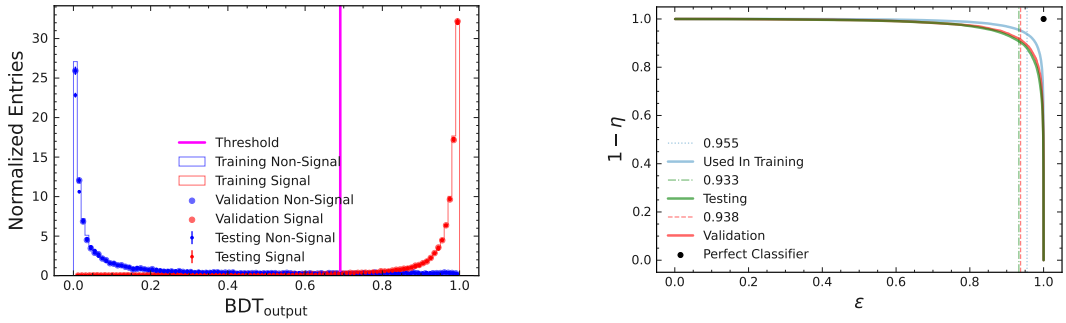
FIGURE F.6: Classifying model for the channel $B^+ \to K^{*+}(\to K^+\pi^0)ee$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.



(a) (b)

FIGURE F.7: Classifying model for the channel $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend.

F.2 BDT Classifier



(a) (b)

FIGURE F.8: Classifying BDT model for the channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$. Both plots contain the three datasets that are used during training, validation and testing. In (a) the binary classifying output of the model and in (b) the ROC is shown. In (b) the vertical lines indicate the relative efficiency of the network which produces the closest point in the ROC curve to the perfect classifier at (1,1). The position of these lines, which also represents the relative efficiency of the classifier, is stated in the legend. There is a rather large difference between the training curve and the test/validation curve. This shows the need for validation data sets.

Appendix G

Acceptance Corrections

G.1 Angular Correlations

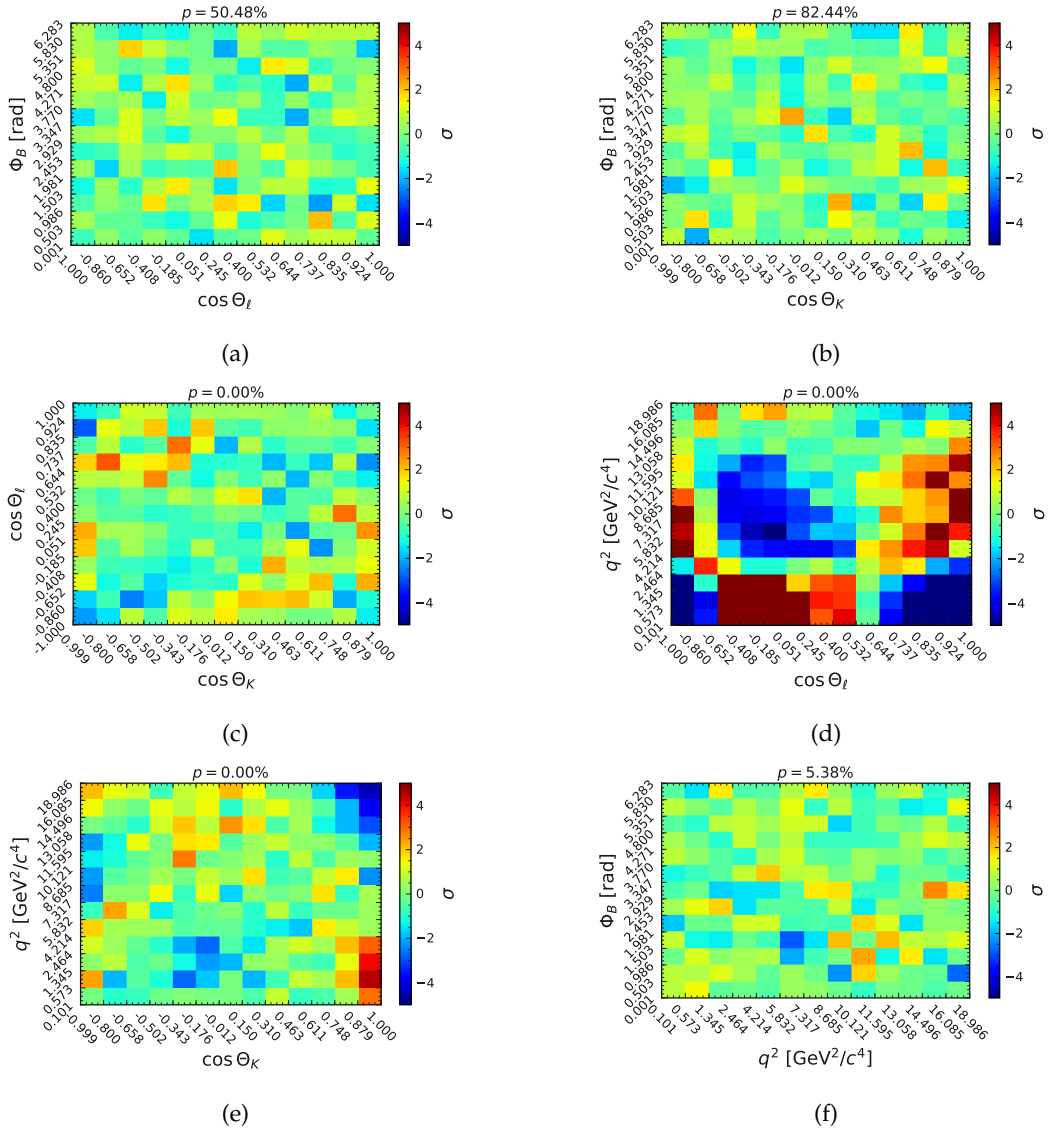


FIGURE G.1: Correlations Between the Four Decay Variables for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$. There is a correlation between $\cos\theta_\ell$ and q^2. However, this correlation is expected since both of these variables depend on both leptons in the decay.

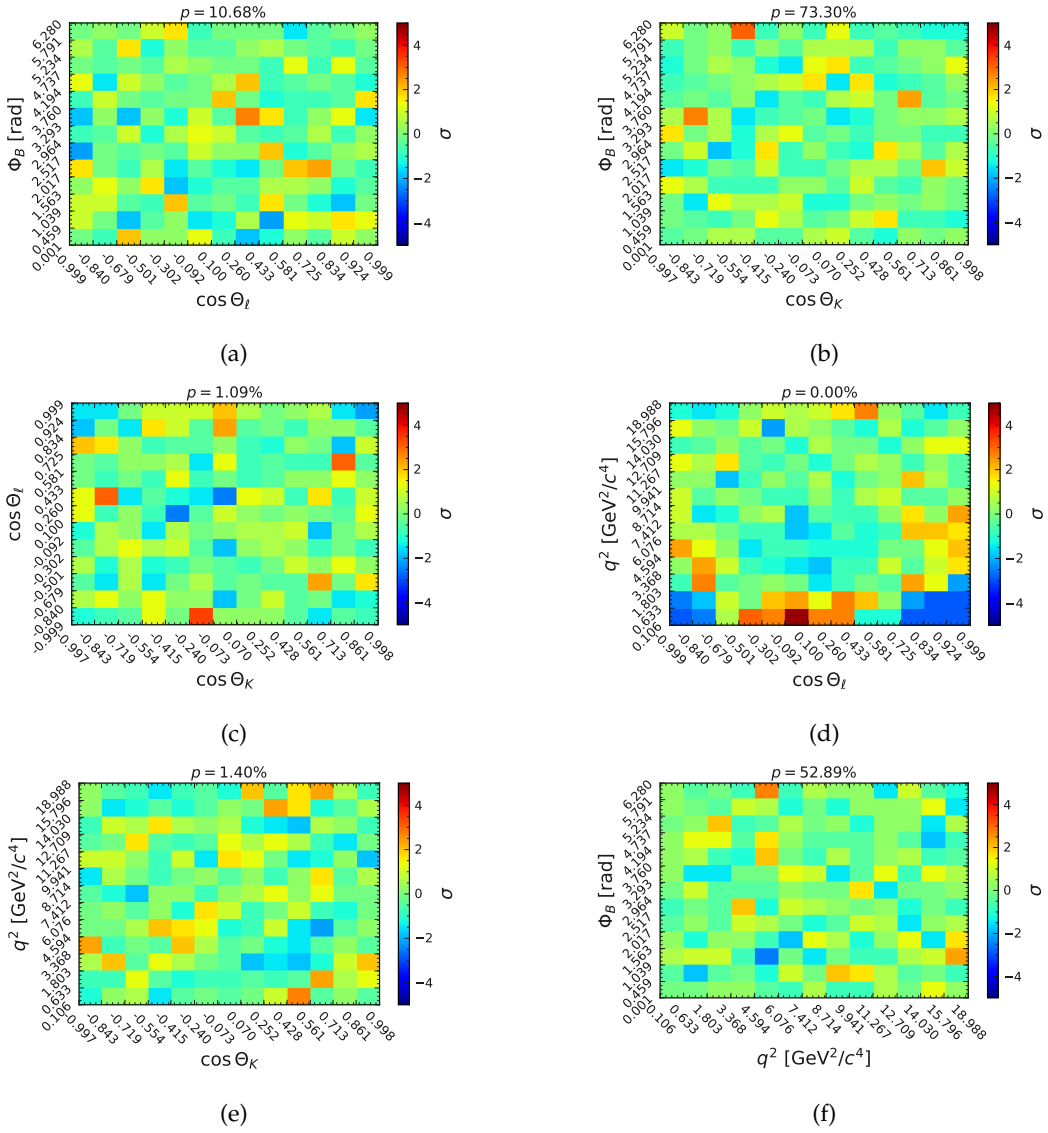


FIGURE G.2: Correlations Between the Four Decay Variables for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$. There is a correlation between $\cos\theta_\ell$ and q^2. However, this correlation is expected since both of these variables depend on both leptons in the decay.

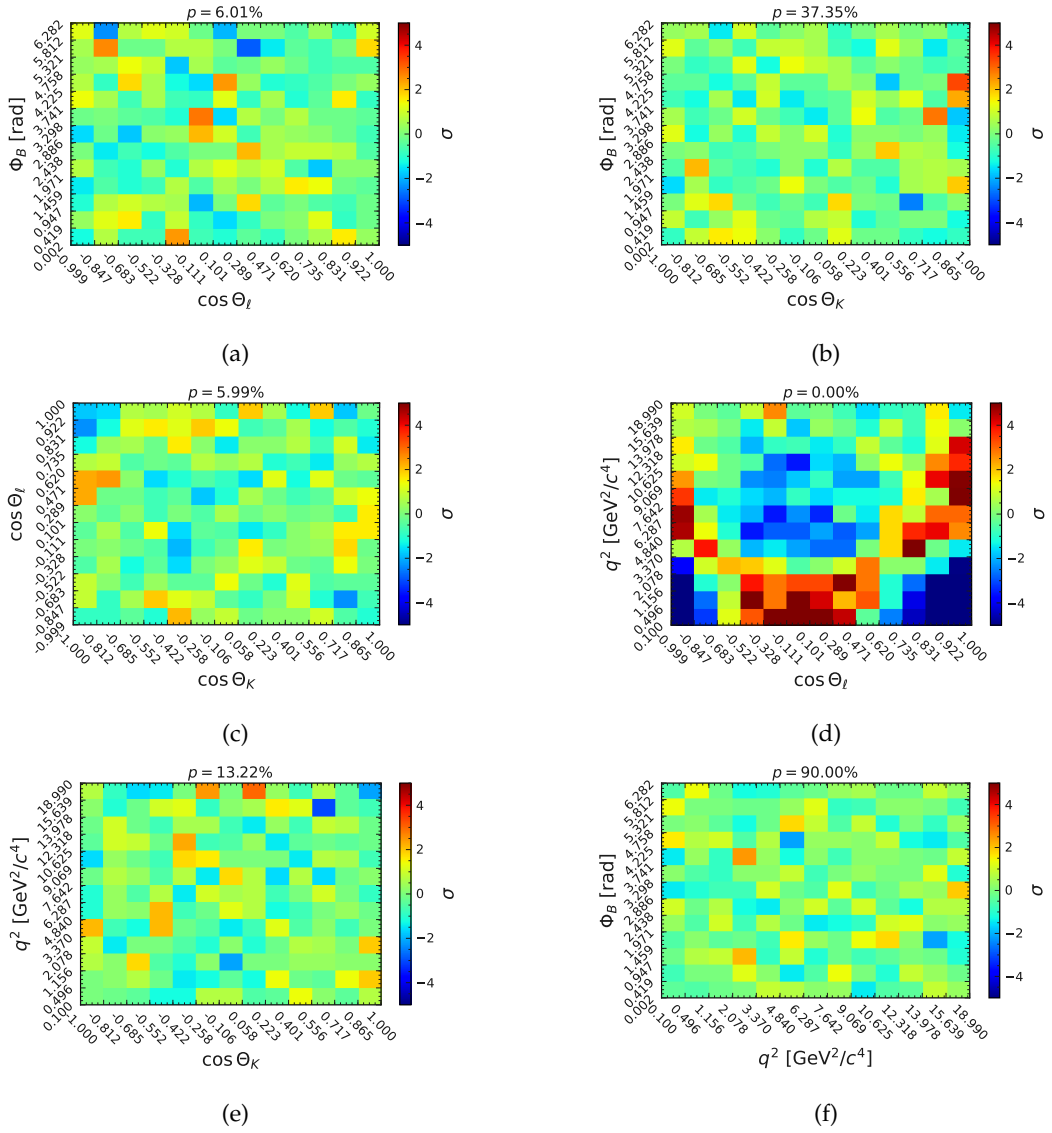


FIGURE G.3: Correlations Between the Four Decay Variables for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$. There is a correlation between $\cos\theta_\ell$ and q^2. However, this correlation is expected since both of these variables depend on both leptons in the decay.

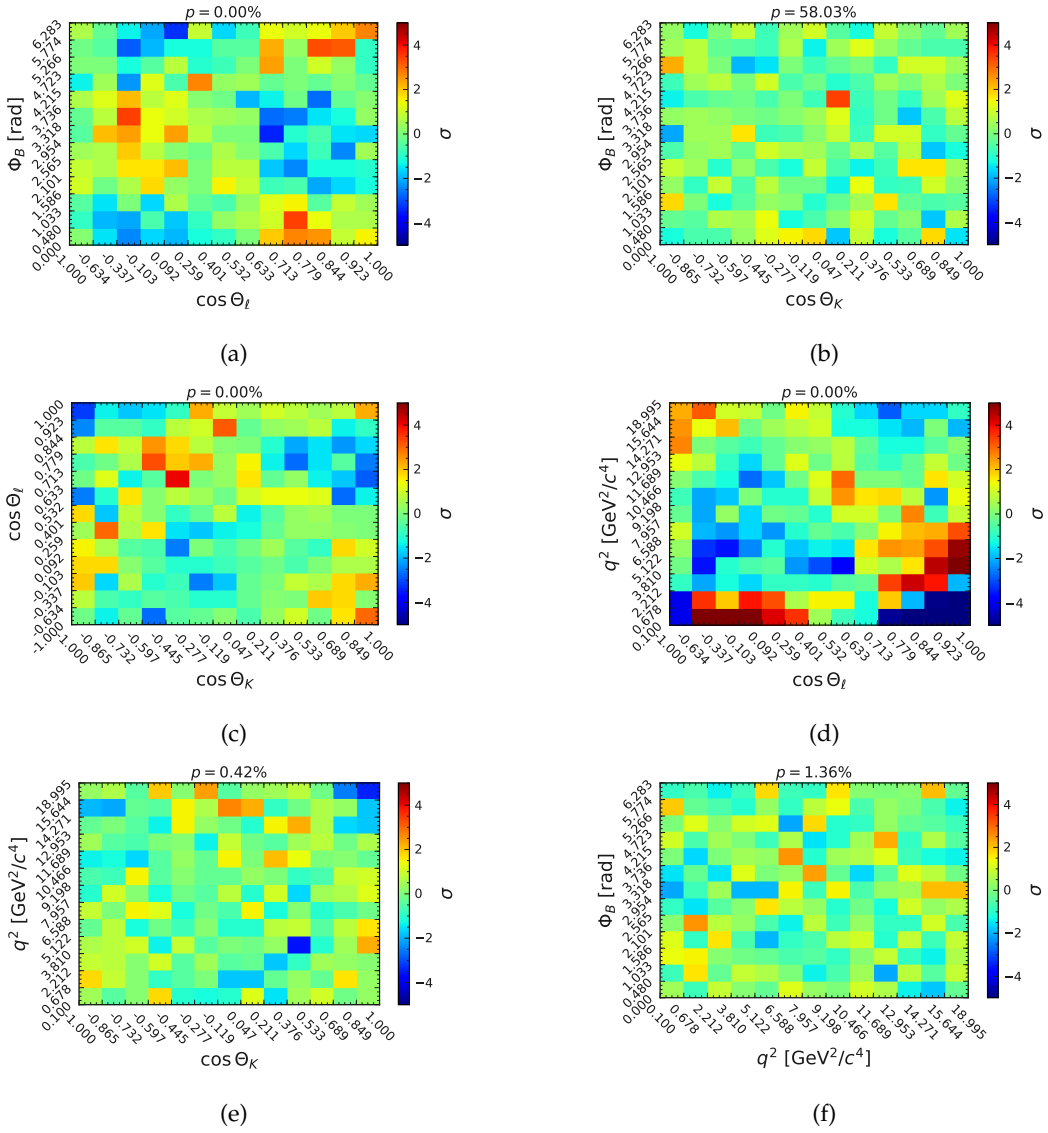


FIGURE G.4: Correlations Between the Four Decay Variables for the $B^+ \to K^{*+}(\to K^+\pi^0)ee$. There is a correlation between $\cos\theta_\ell$ and q^2. However, this correlation is expected since both of these variables depend on both leptons in the decay.

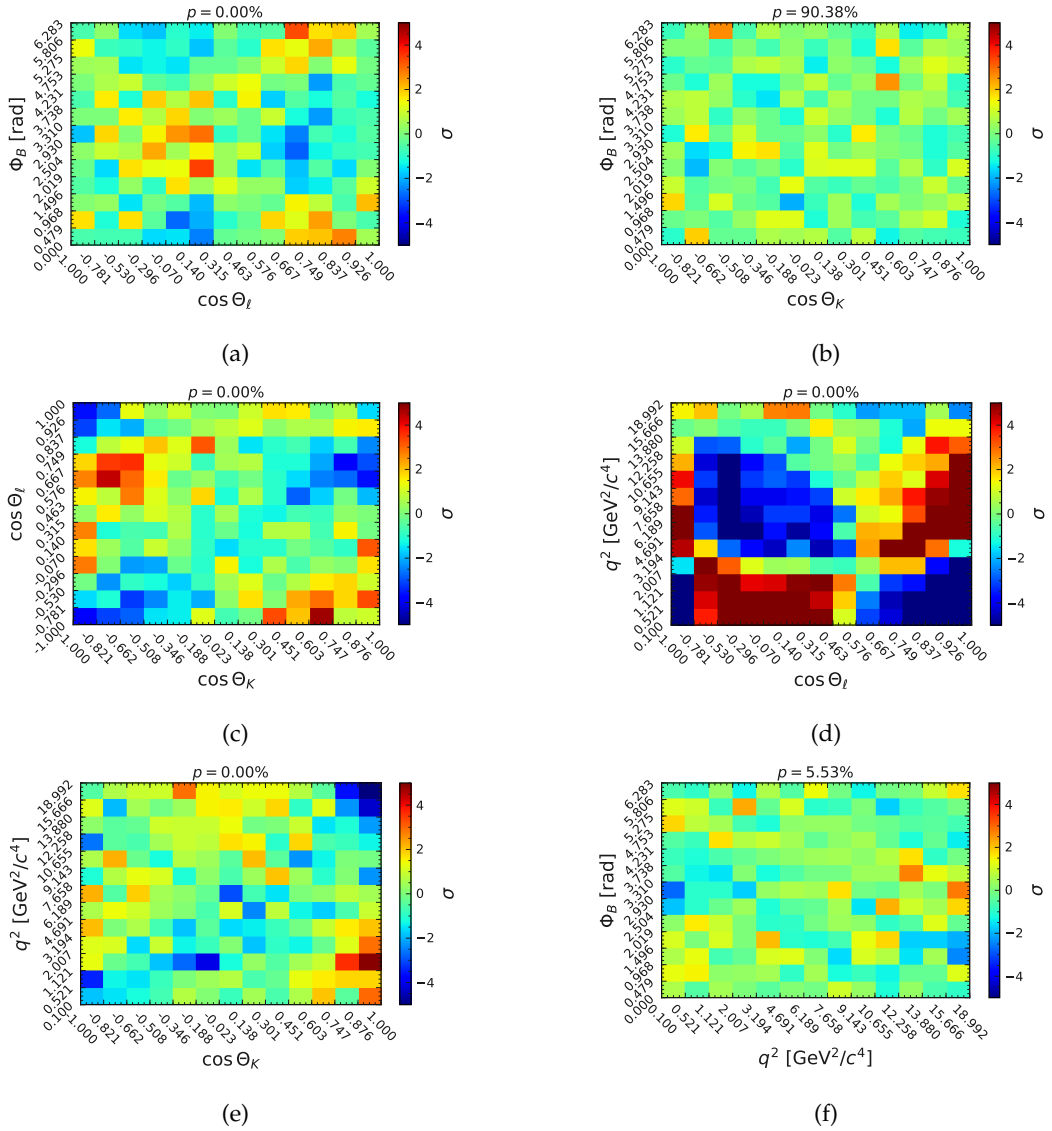
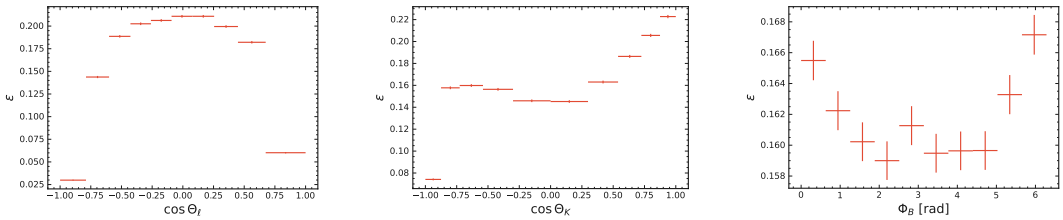


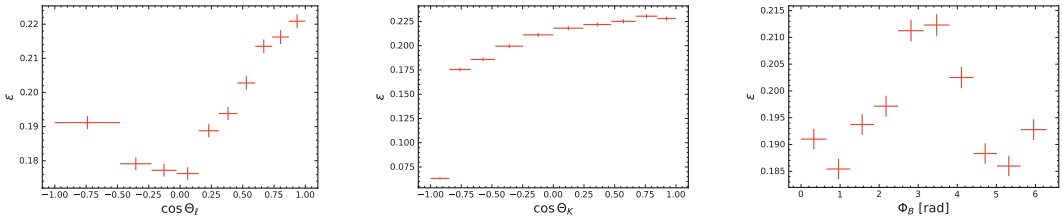
FIGURE G.5: Correlations Between the Four Decay Variables for the $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$. There is a correlation between $\cos\theta_\ell$ and q^2. However, this correlation is expected since both of these variables depend on both leptons in the decay.

G.2 Efficiencies



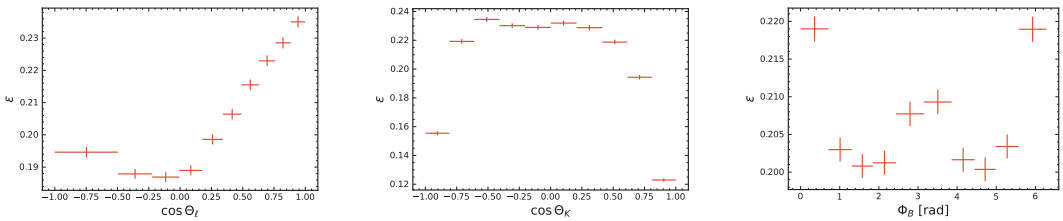
(a) (b) (c)

FIGURE G.6: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_1 region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel.



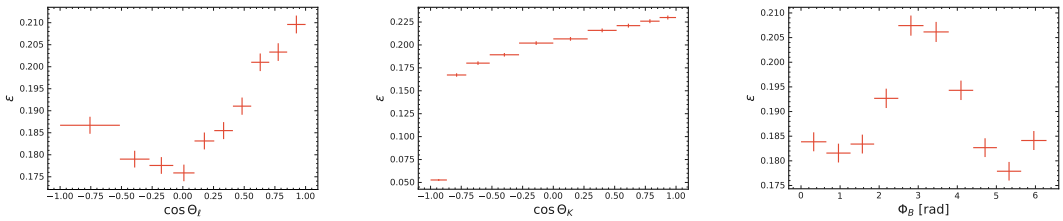
(a) (b) (c)

FIGURE G.7: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_3 region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel.



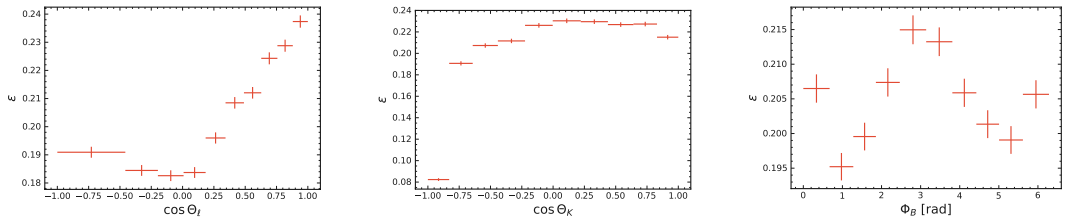
(a) (b) (c)

FIGURE G.8: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_4 region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel.



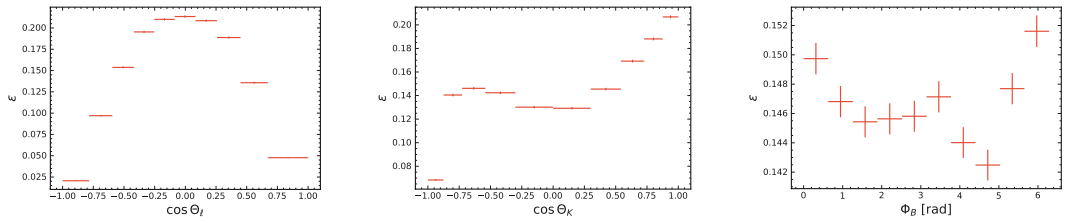
(a) (b) (c)

FIGURE G.9: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{J/\psi}$ region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel.



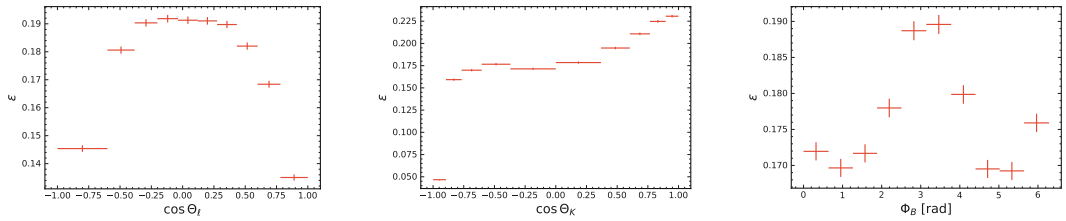
(a) (b) (c)

FIGURE G.10: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{\psi(2S)}$ region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel.



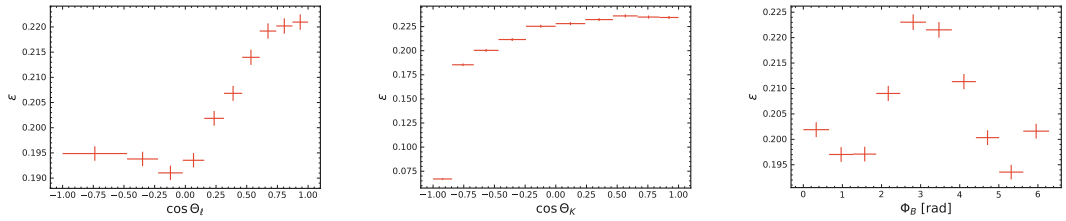
(a) (b) (c)

FIGURE G.11: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_1 region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel.



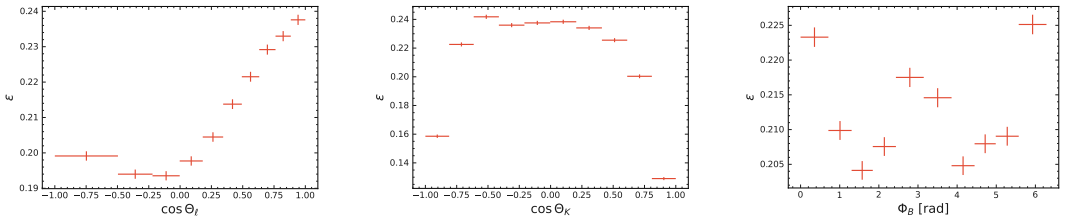
(a) (b) (c)

FIGURE G.12: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_2 region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel.



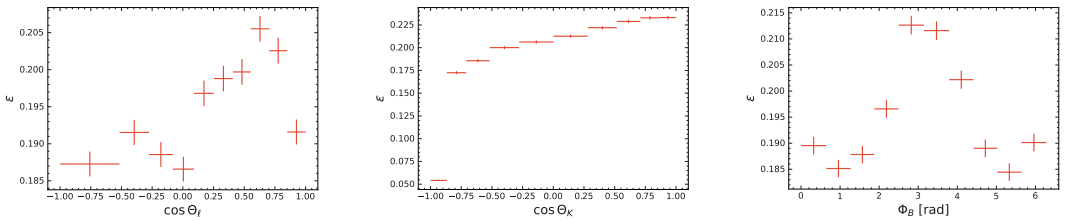
(a) (b) (c)

FIGURE G.13: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_3 region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel.



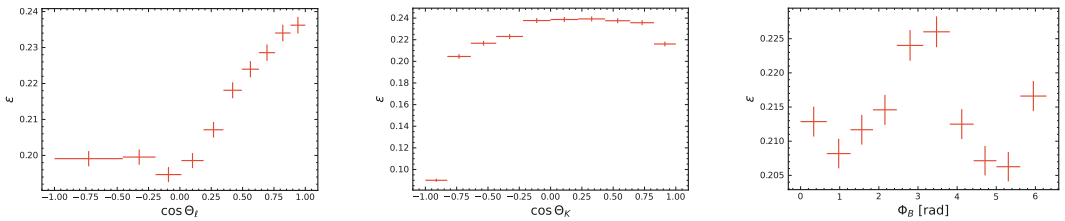
(a) (b) (c)

FIGURE G.14: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_4 region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel.



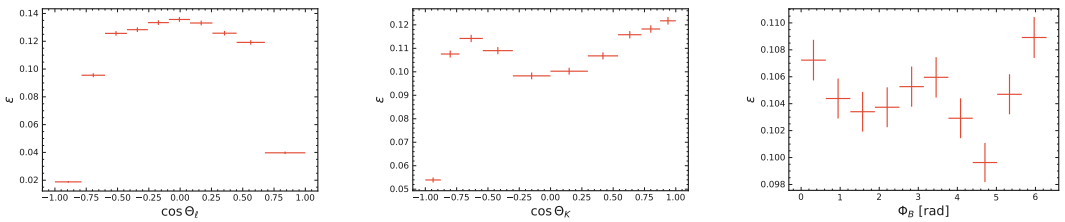
(a) (b) (c)

FIGURE G.15: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{J/\psi}$ region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel.



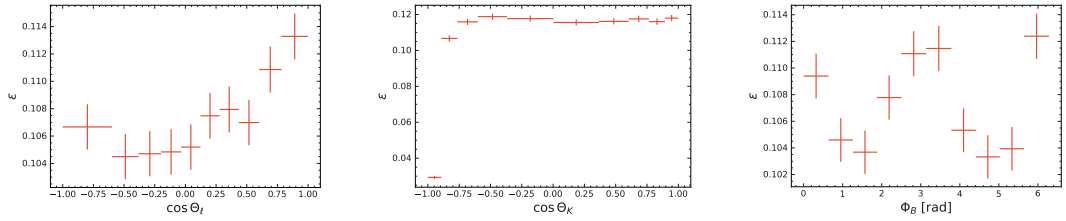
(a) (b) (c)

FIGURE G.16: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{\psi(2S)}$ region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel.



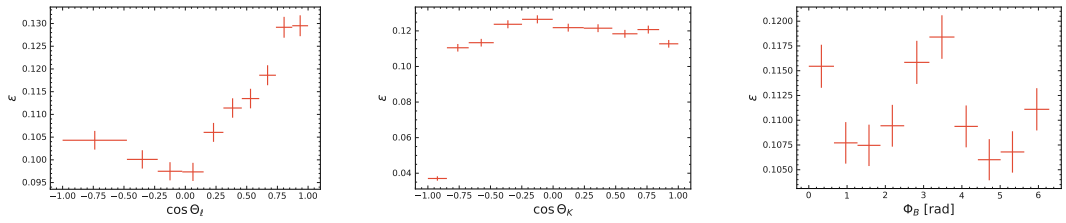
(a) (b) (c)

FIGURE G.17: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_1 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel.



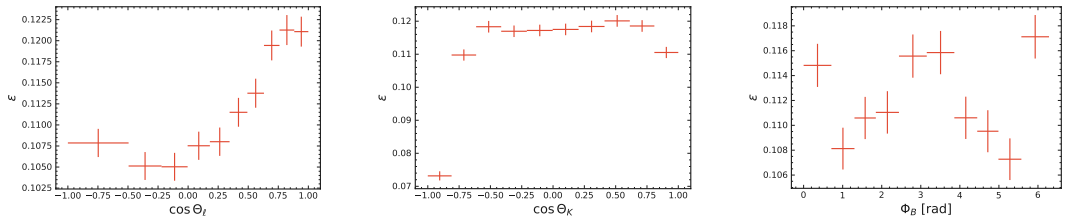
(a) (b) (c)

FIGURE G.18: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_2 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel.



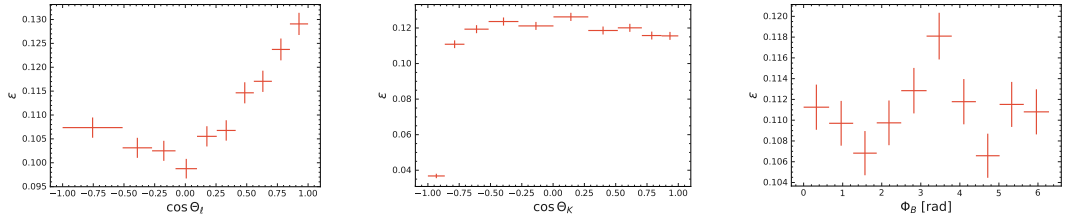
(a) (b) (c)

FIGURE G.19: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_3 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel.



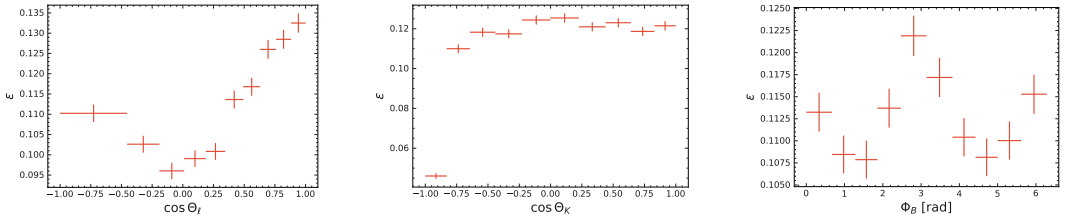
(a) (b) (c)

FIGURE G.20: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_4 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel.



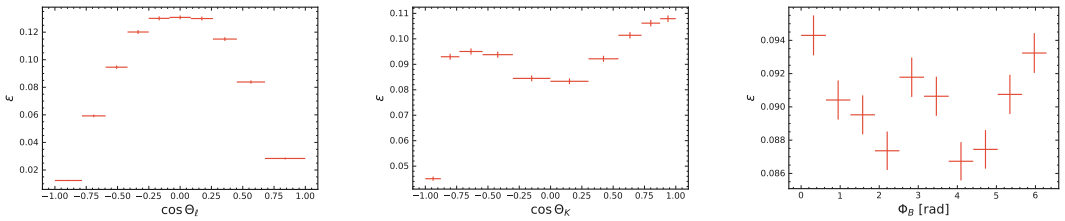
(a) (b) (c)

FIGURE G.21: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{J/\psi}$ region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel.



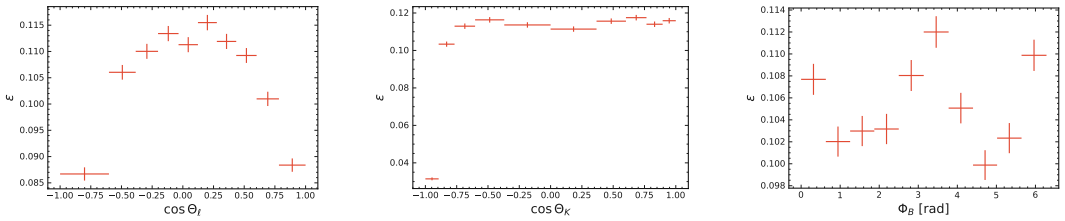
(a) (b) (c)

FIGURE G.22: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{\psi(2S)}$ region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel.



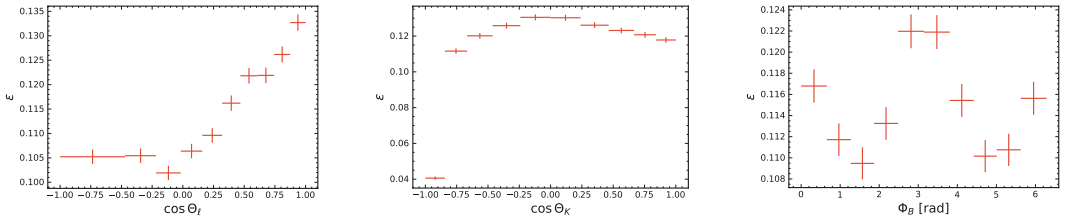
(a) (b) (c)

FIGURE G.23: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_1 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel.



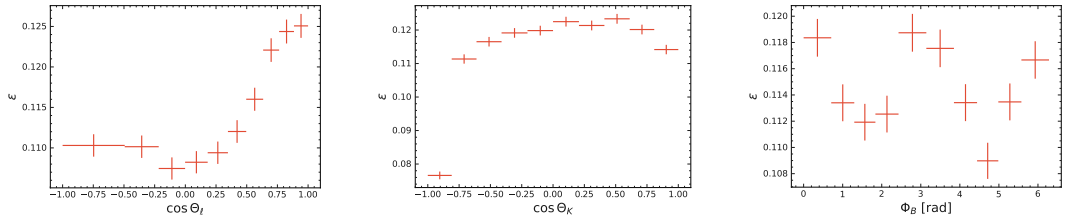
(a) (b) (c)

FIGURE G.24: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_2 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel.



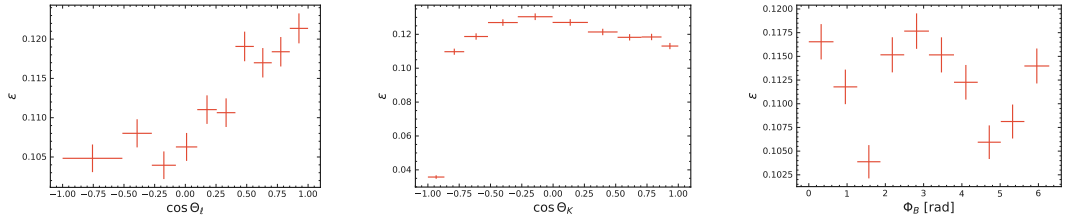
(a) (b) (c)

FIGURE G.25: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_3 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel.



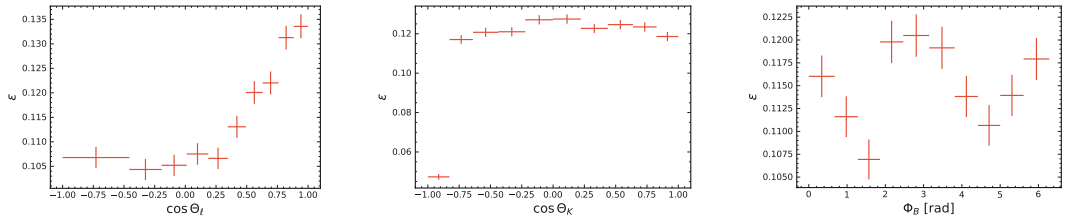
(a) (b) (c)

FIGURE G.26: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the q^2_4 region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel.



(a) (b) (c)

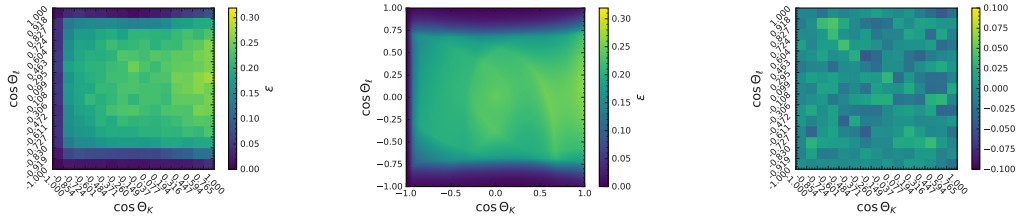
FIGURE G.27: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{J/\psi}$ region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel.



(a) (b) (c)

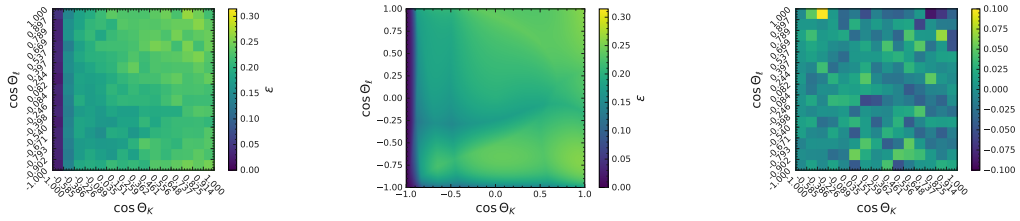
FIGURE G.28: Efficiencies for the angles $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) in the $q^2_{\psi(2S)}$ region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel.

G.3 Models



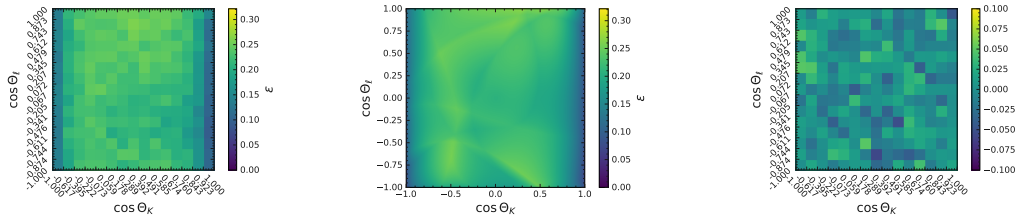
(a) (b) (c)

FIGURE G.29: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_1^2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



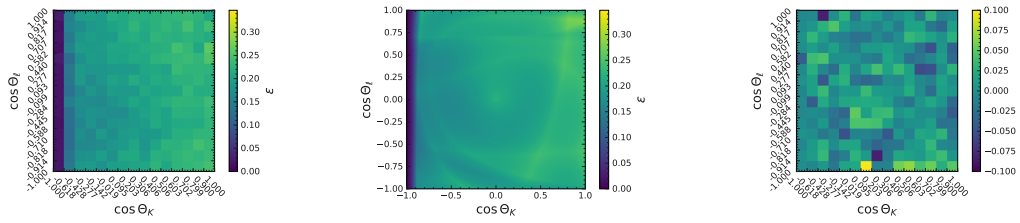
(a) (b) (c)

FIGURE G.30: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_3^2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



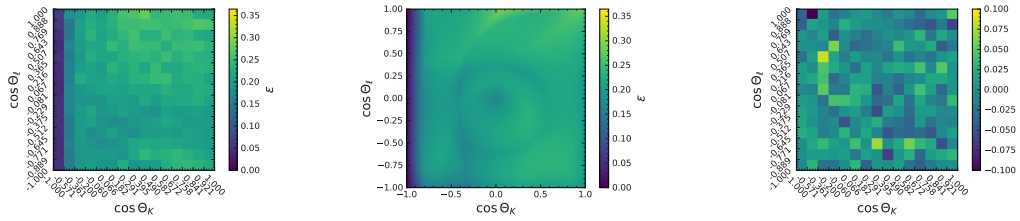
(a) (b) (c)

FIGURE G.31: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_4^2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



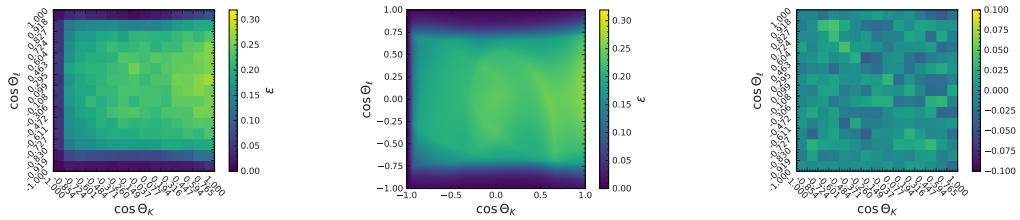
(a) (b) (c)

FIGURE G.32: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the $q^2_{J/\psi}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



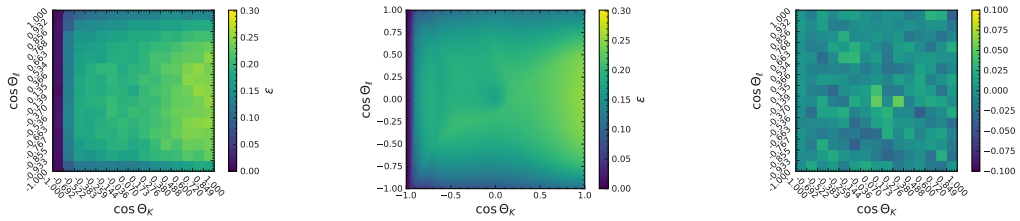
(a) (b) (c)

FIGURE G.33: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the $q^2_{\psi(2S)}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



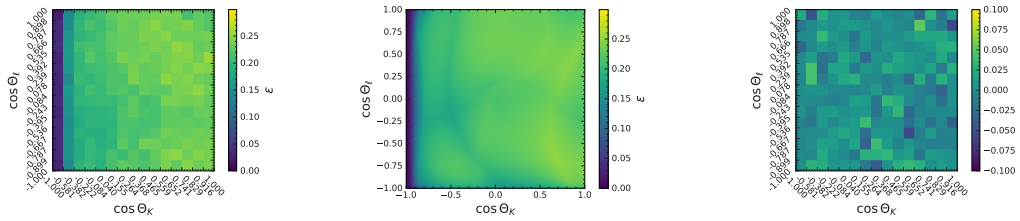
(a) (b) (c)

FIGURE G.34: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q^2_1 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



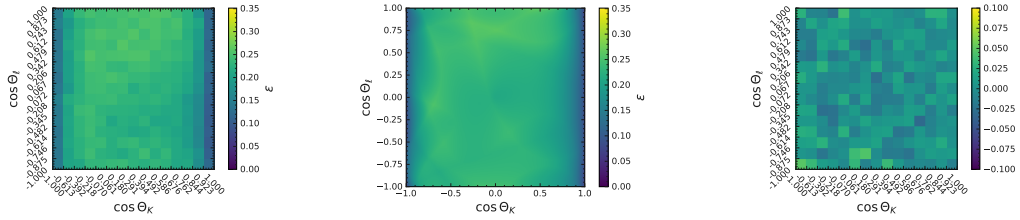
(a) (b) (c)

FIGURE G.35: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q_2^2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



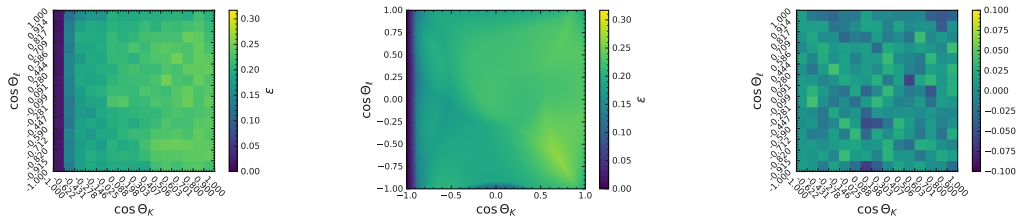
(a) (b) (c)

FIGURE G.36: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q_3^2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



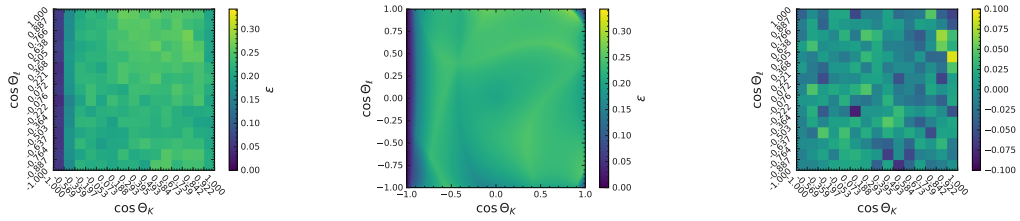
(a) (b) (c)

FIGURE G.37: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q_4^2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



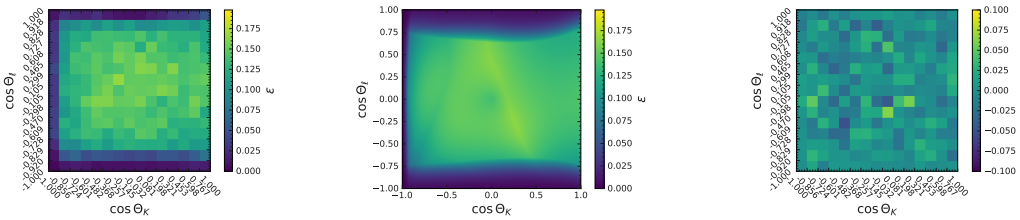
(a) (b) (c)

FIGURE G.38: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the $q^2_{J/\psi}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



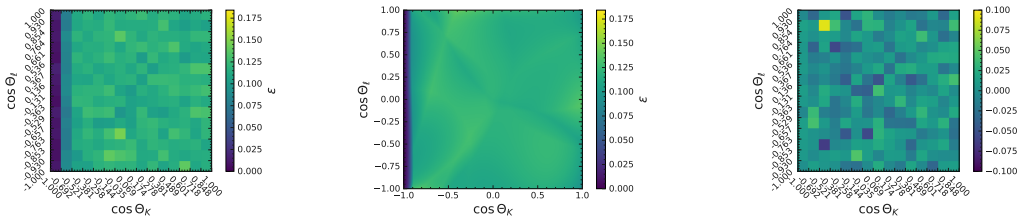
(a) (b) (c)

FIGURE G.39: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the $q^2_{\psi(2S)}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



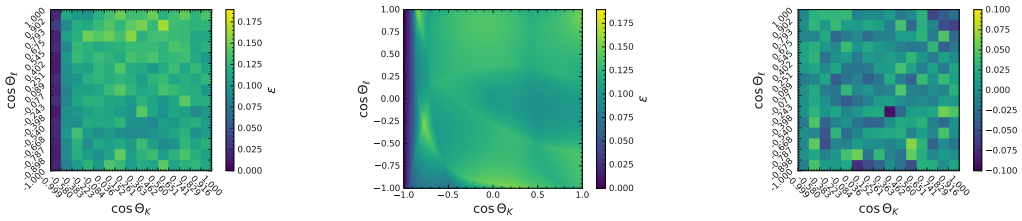
(a) (b) (c)

FIGURE G.40: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ in the q^2_1 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



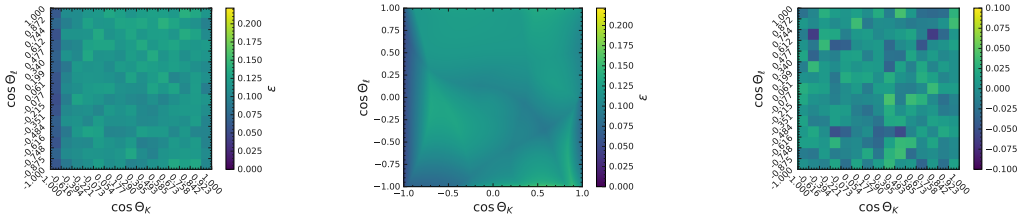
(a) (b) (c)

FIGURE G.41: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ in the q^2_2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



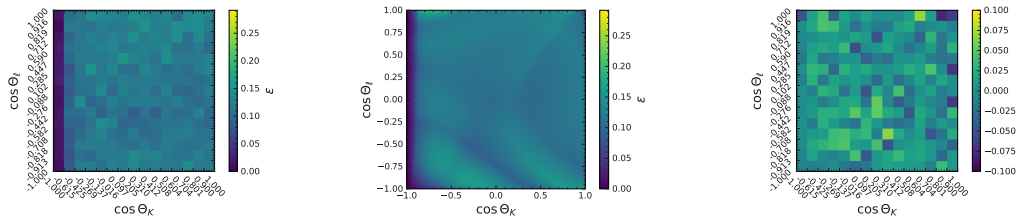
(a) (b) (c)

FIGURE G.42: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ in the q^2_3 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



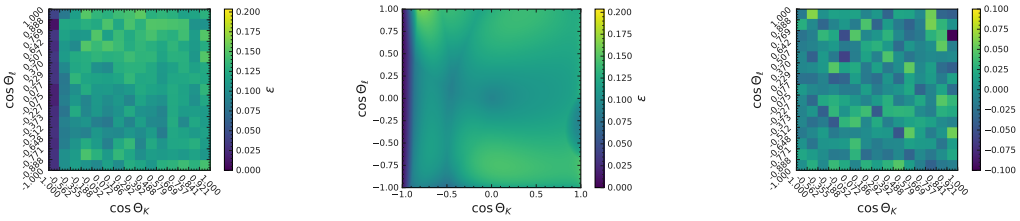
(a) (b) (c)

FIGURE G.43: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ in the q^2_4 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



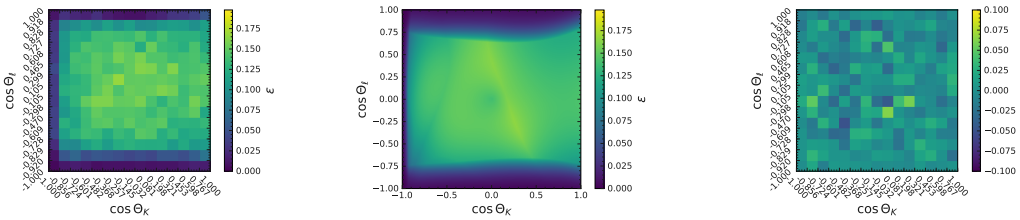
(a) (b) (c)

FIGURE G.44: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ in the $q^2_{J/\psi}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



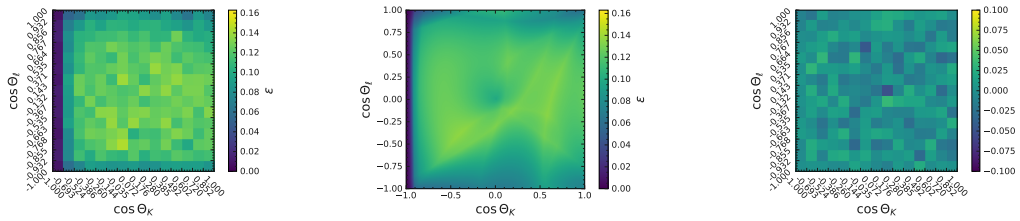
(a) (b) (c)

FIGURE G.45: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ in the $q^2_{\psi(2S)}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



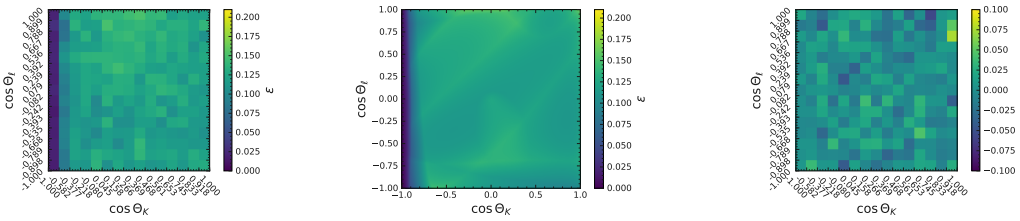
(a) (b) (c)

FIGURE G.46: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ in the q^2_1 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



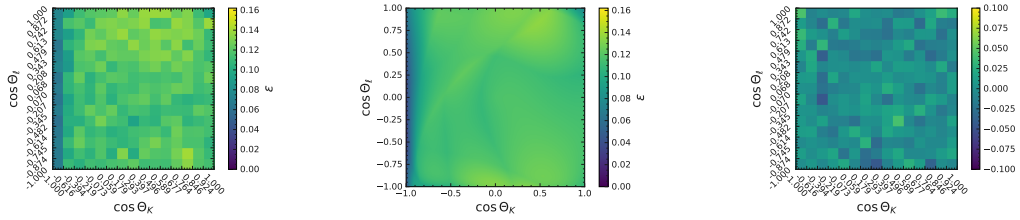
(a) (b) (c)

FIGURE G.47: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ in the q^2_2 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



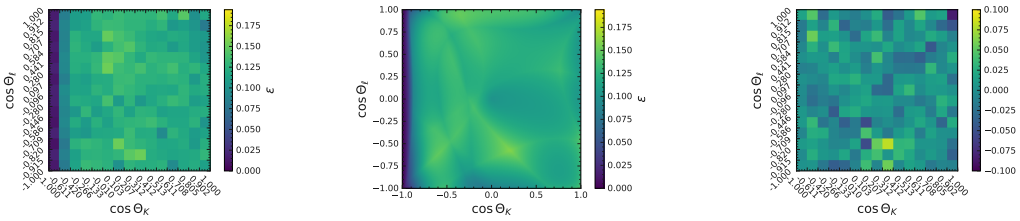
(a) (b) (c)

FIGURE G.48: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ in the q^2_3 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



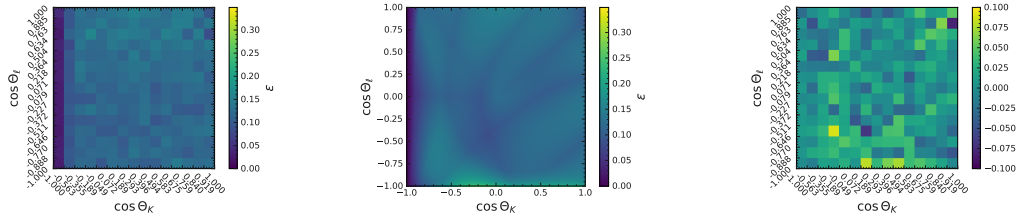
(a) (b) (c)

FIGURE G.49: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ in the q^2_4 region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



(a) (b) (c)

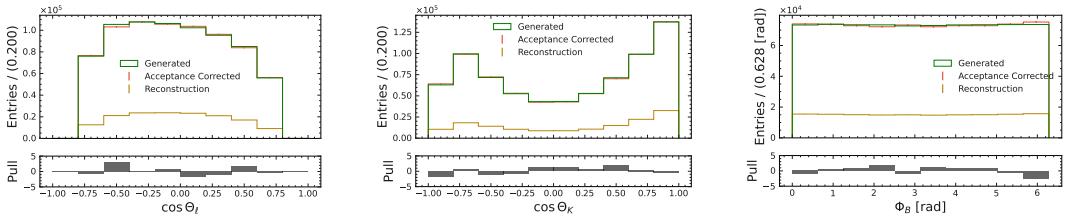
FIGURE G.50: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ in the $q^2_{J/\psi}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.



(a) (b) (c)

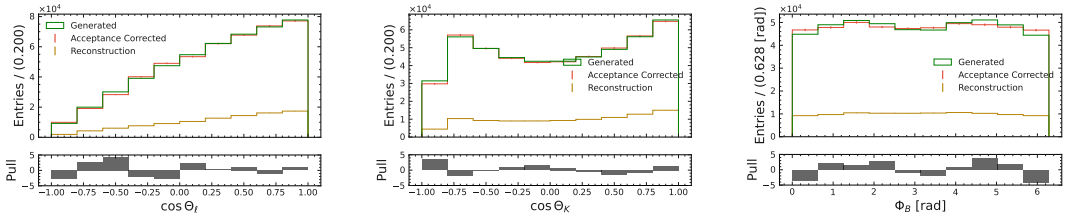
FIGURE G.51: (a) Input efficiencies for the acceptance model training. Resulting acceptance correction model for the decay $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ in the $q^2_{\psi(2S)}$ region. In (b) the resulting model function is shown. In (c) the difference between the predicted and true efficiency is shown.

G.4 Corrections



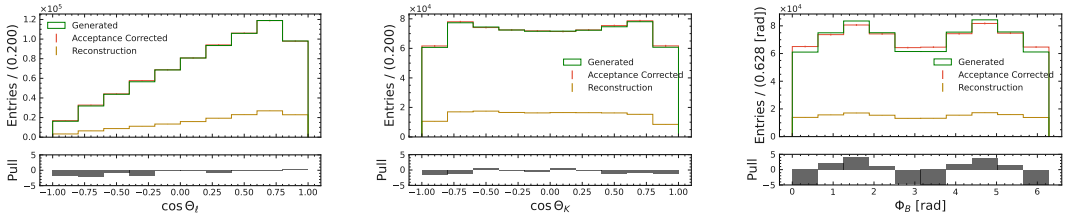
(a) (b) (c)

FIGURE G.52: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_1^2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



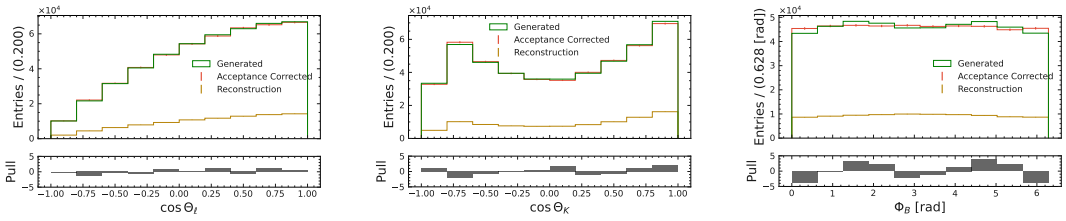
(a) (b) (c)

FIGURE G.53: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_3^2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



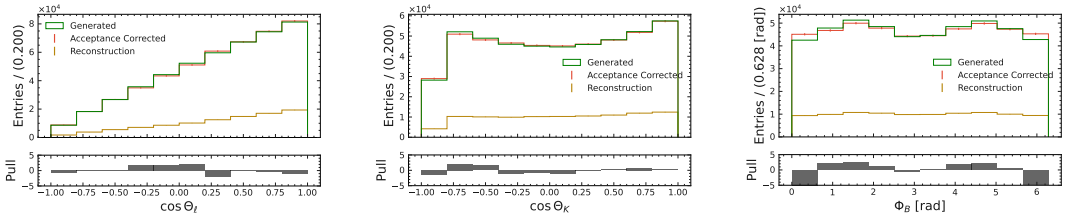
(a) (b) (c)

FIGURE G.54: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_4^2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



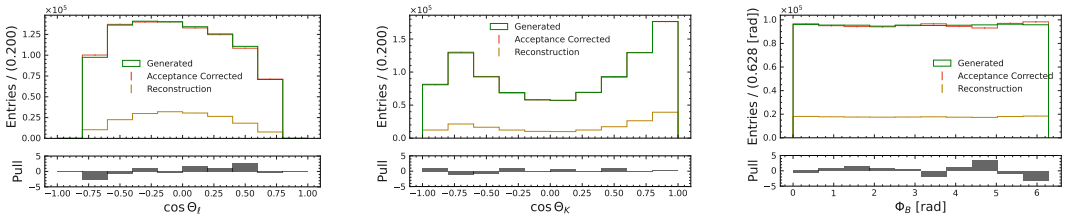
(a) (b) (c)

FIGURE G.55: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the $q^2_{J/\psi}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



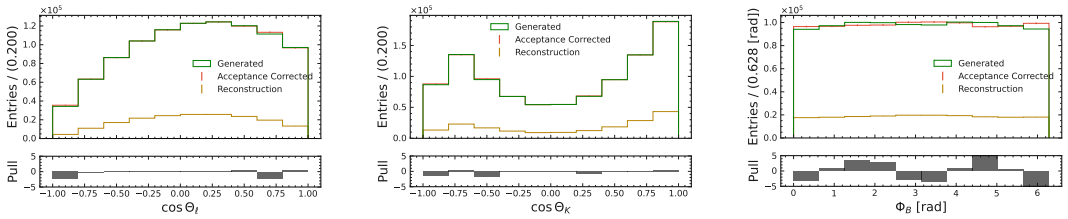
(a) (b) (c)

FIGURE G.56: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the $q^2_{\psi(2S)}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



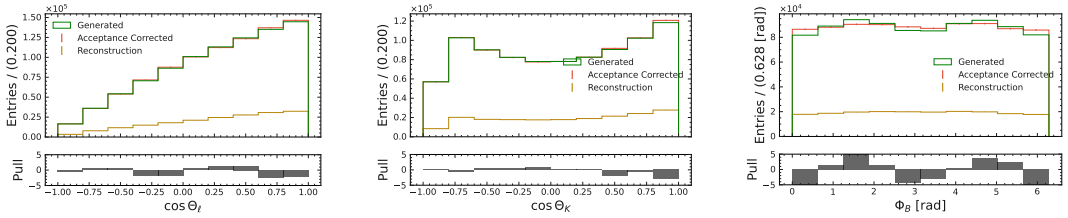
(a) (b) (c)

FIGURE G.57: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q^2_1-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



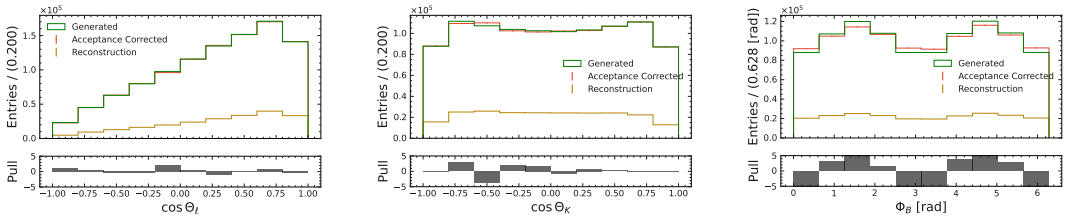
(a) (b) (c)

FIGURE G.58: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q^2_2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



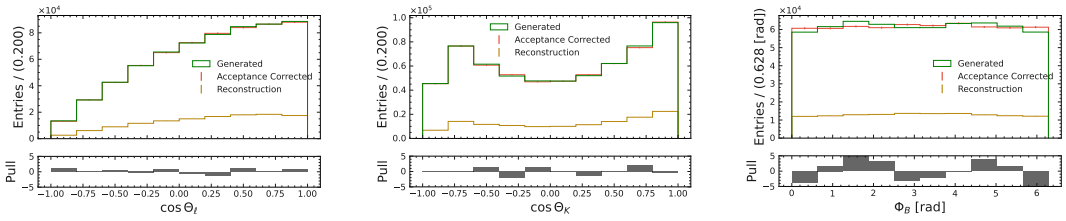
(a) (b) (c)

FIGURE G.59: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q^2_3-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



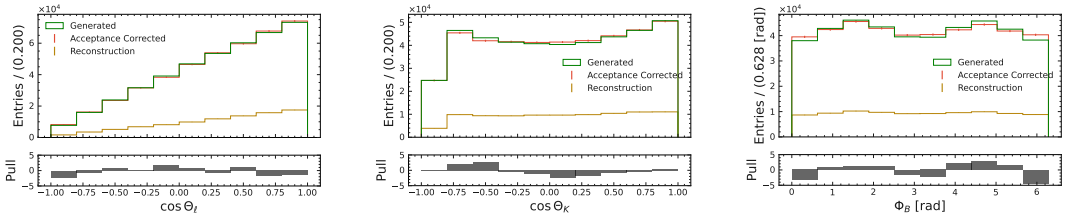
(a) (b) (c)

FIGURE G.60: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q^2_4-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



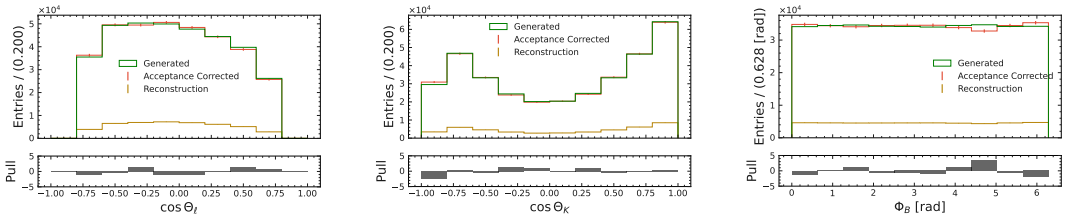
(a) (b) (c)

FIGURE G.61: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the $q^2_{J/\psi}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



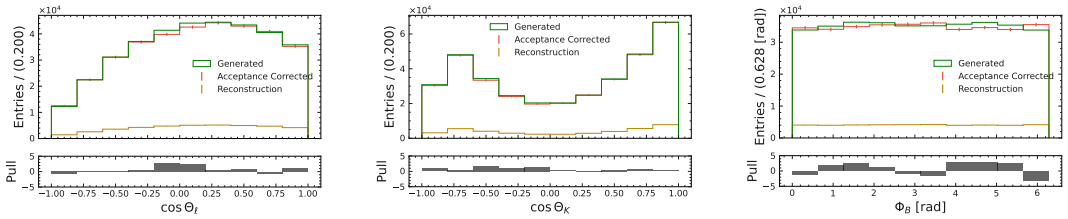
(a) (b) (c)

FIGURE G.62: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the $q^2_{\psi(2S)}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



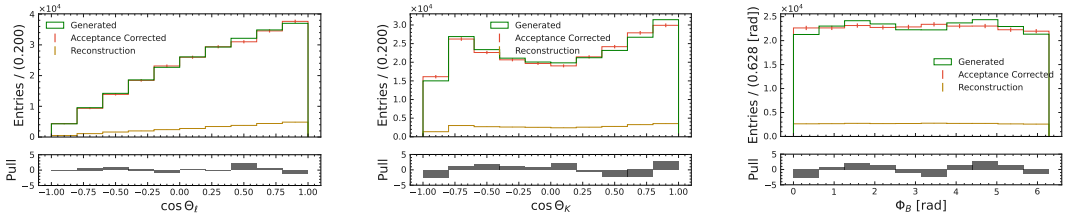
(a) (b) (c)

FIGURE G.63: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_1-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



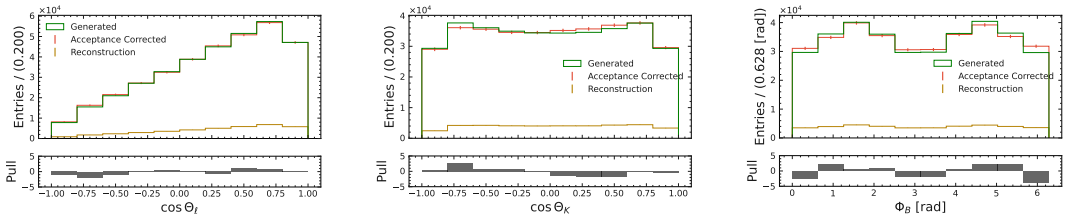
(a) (b) (c)

FIGURE G.64: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



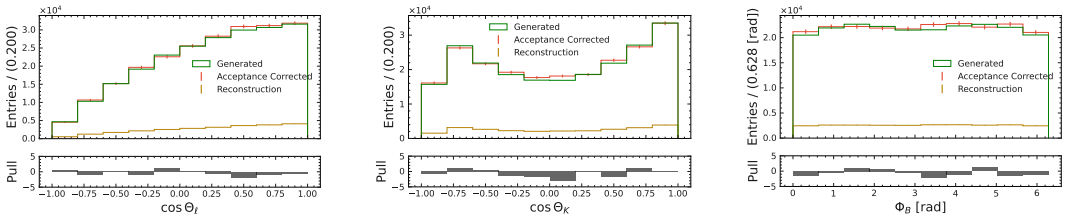
(a) (b) (c)

FIGURE G.65: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_3-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



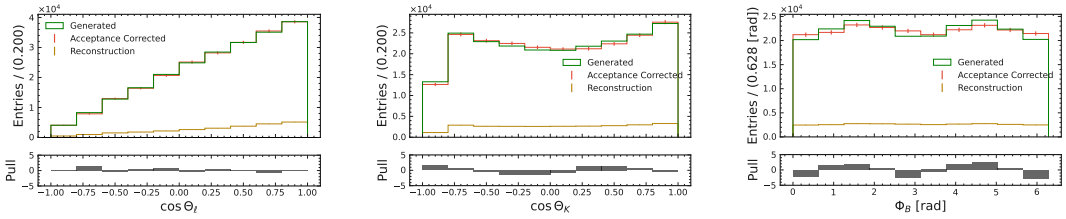
(a) (b) (c)

FIGURE G.66: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_4-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



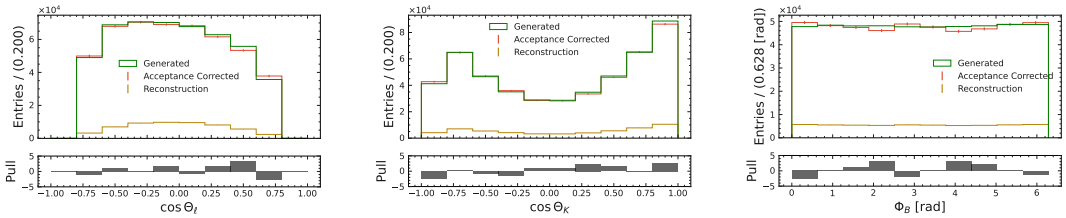
(a) (b) (c)

FIGURE G.67: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the $q^2_{J/\psi}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



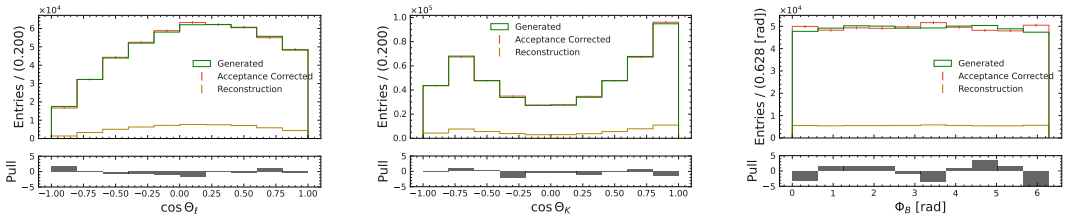
(a) (b) (c)

FIGURE G.68: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the $q^2_{\psi(2S)}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



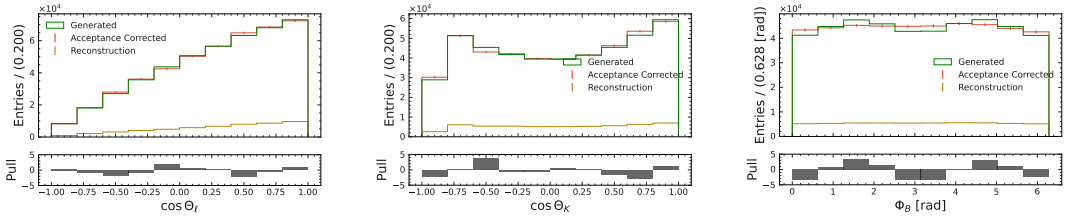
(a) (b) (c)

FIGURE G.69: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_1-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



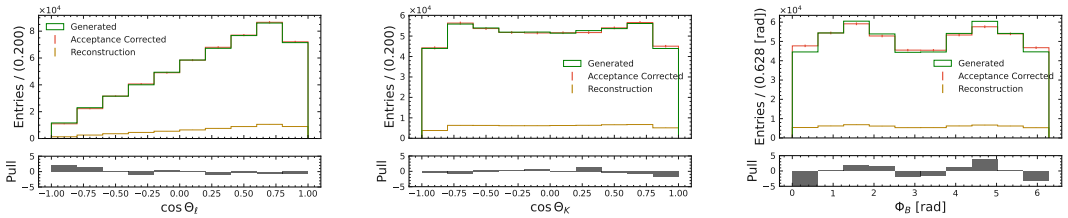
(a) (b) (c)

FIGURE G.70: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_2-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



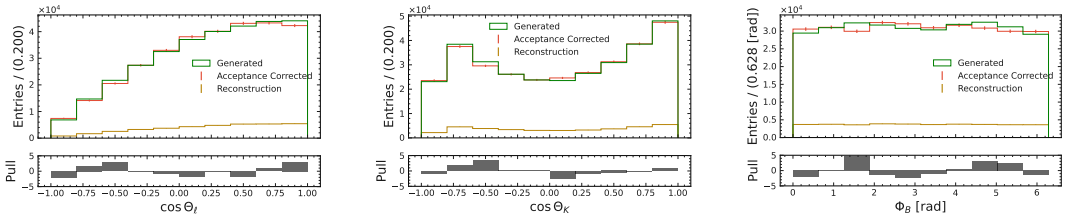
(a) (b) (c)

FIGURE G.71: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_3-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



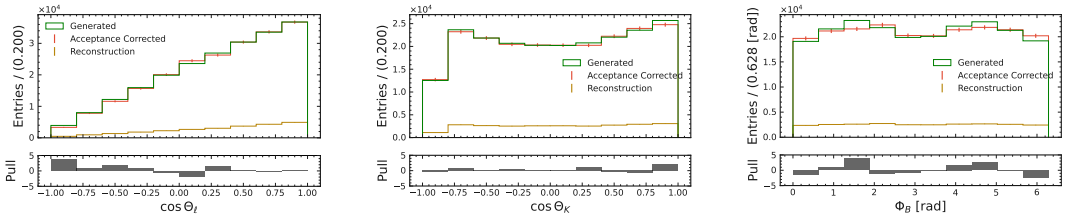
(a) (b) (c)

FIGURE G.72: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_4-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



(a) (b) (c)

FIGURE G.73: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the $q^2_{J/\psi}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.



(a) (b) (c)

FIGURE G.74: Decay angle acceptance correction for $\cos\theta_\ell$ (a), $\cos\theta_K$ (b) and ϕ_B (c) for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the $q^2_{\psi(2S)}$-region. Green shows the distribution on generator level. The reconstructed candidates are shown in dark orange. They are corrected with the acceptance correction and the result is shown in red.

Appendix H

Correlations

H.1 Background

H.1.1 Angular Variables

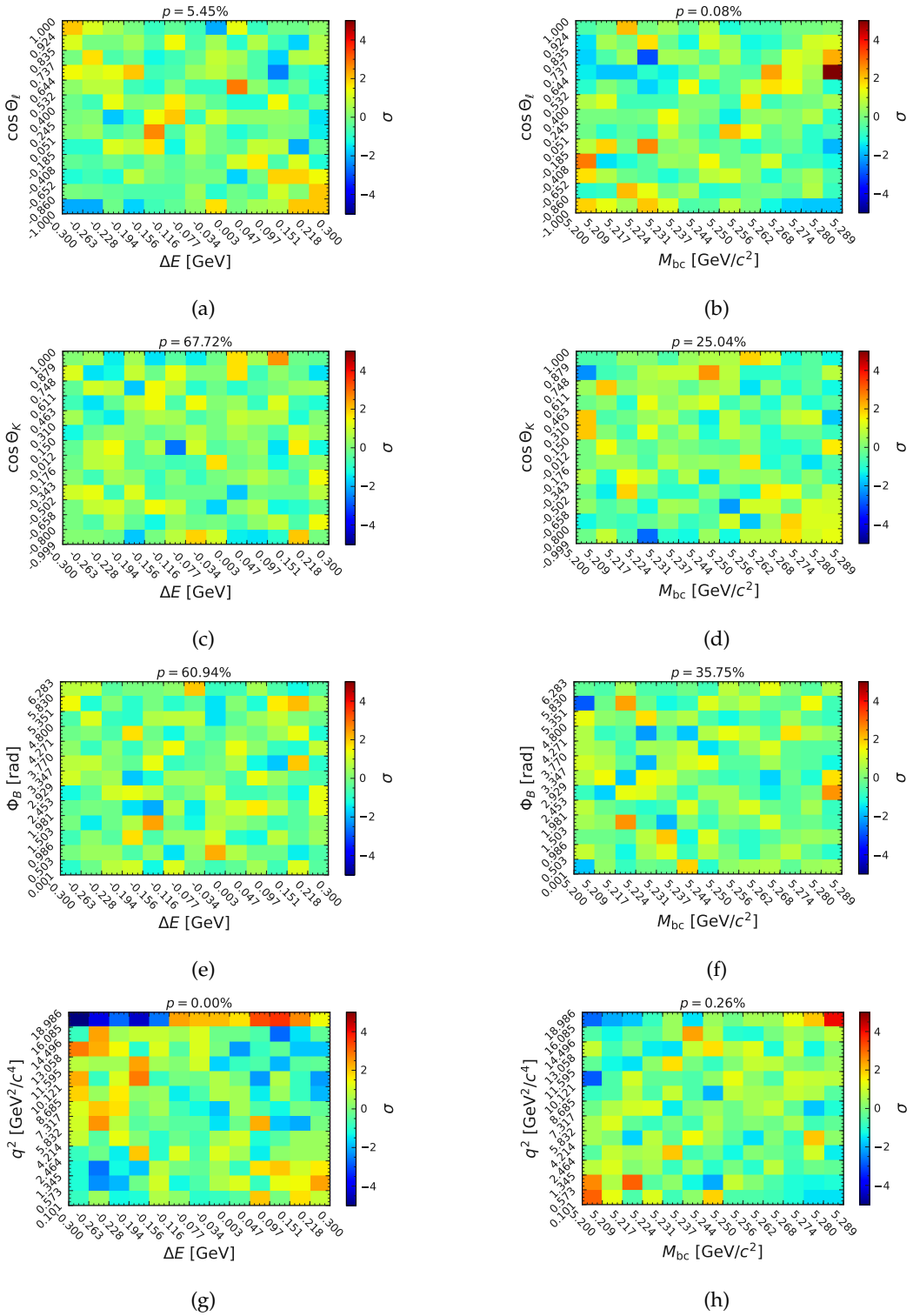


FIGURE H.1: Correlations between angles and ΔE & M_{bc} in the channel $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ for background events.

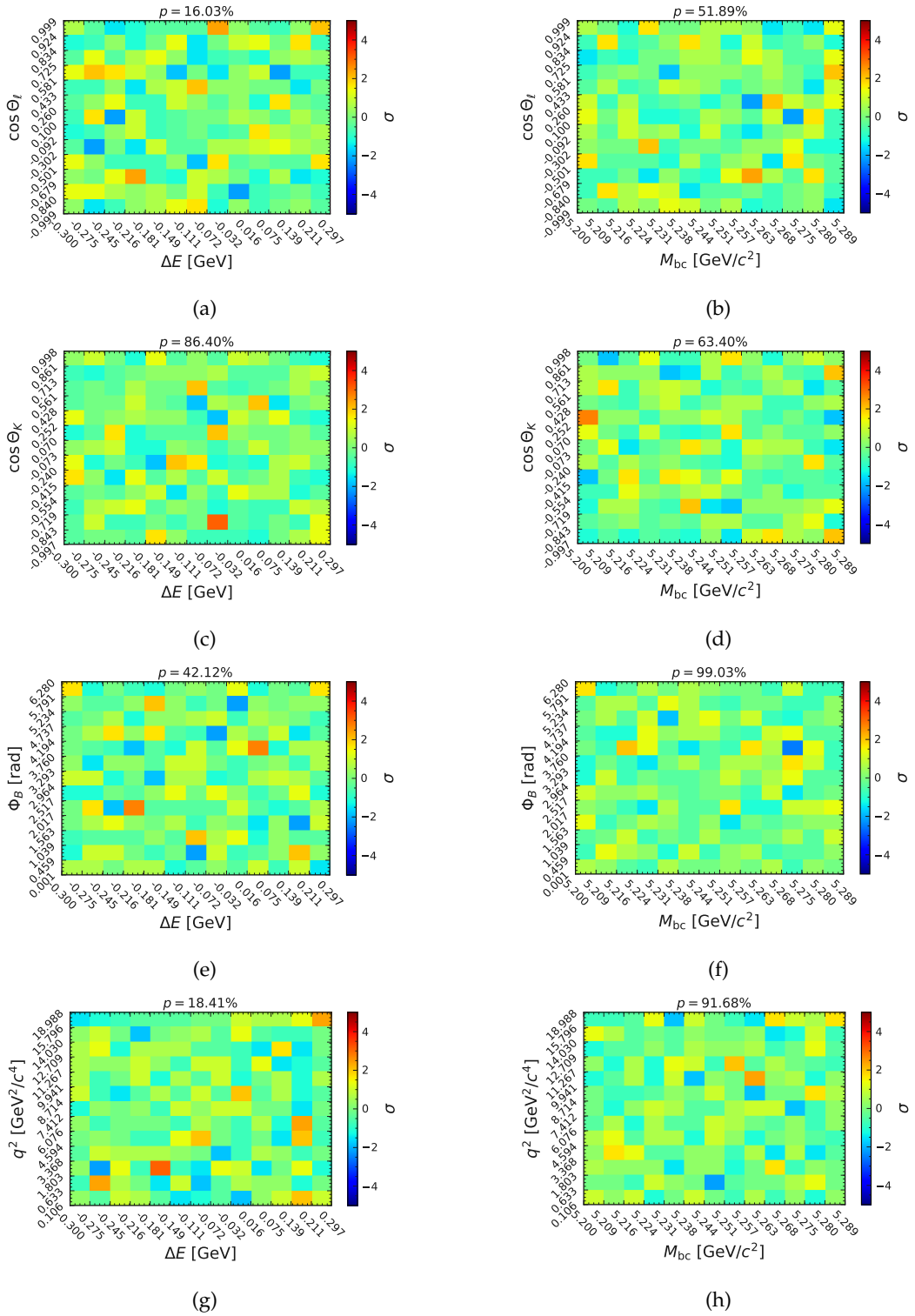


FIGURE H.2: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^0_S \pi^+) ee$ for background events.

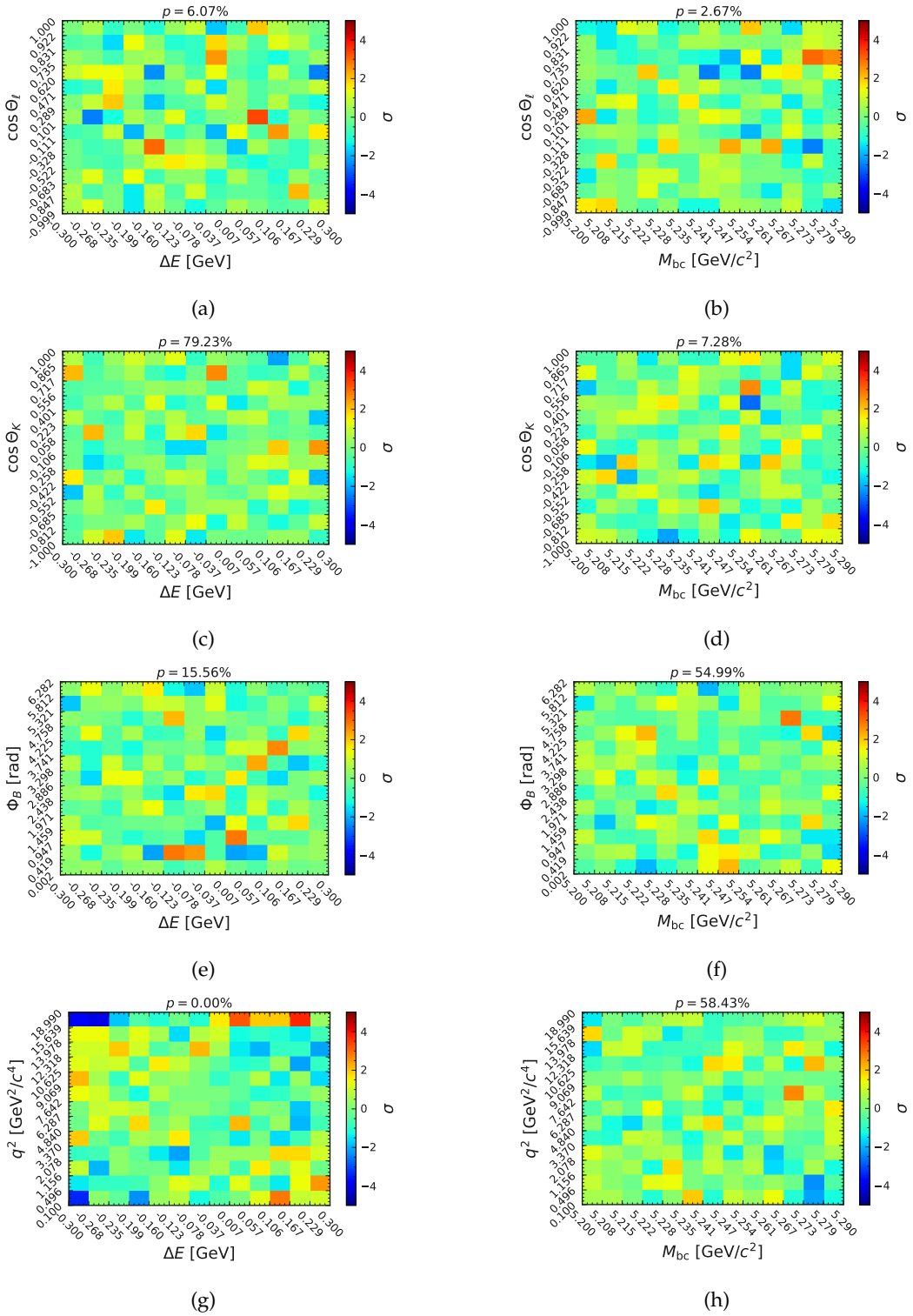


FIGURE H.3: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^0_S \pi^+)\mu\mu$ for background events.

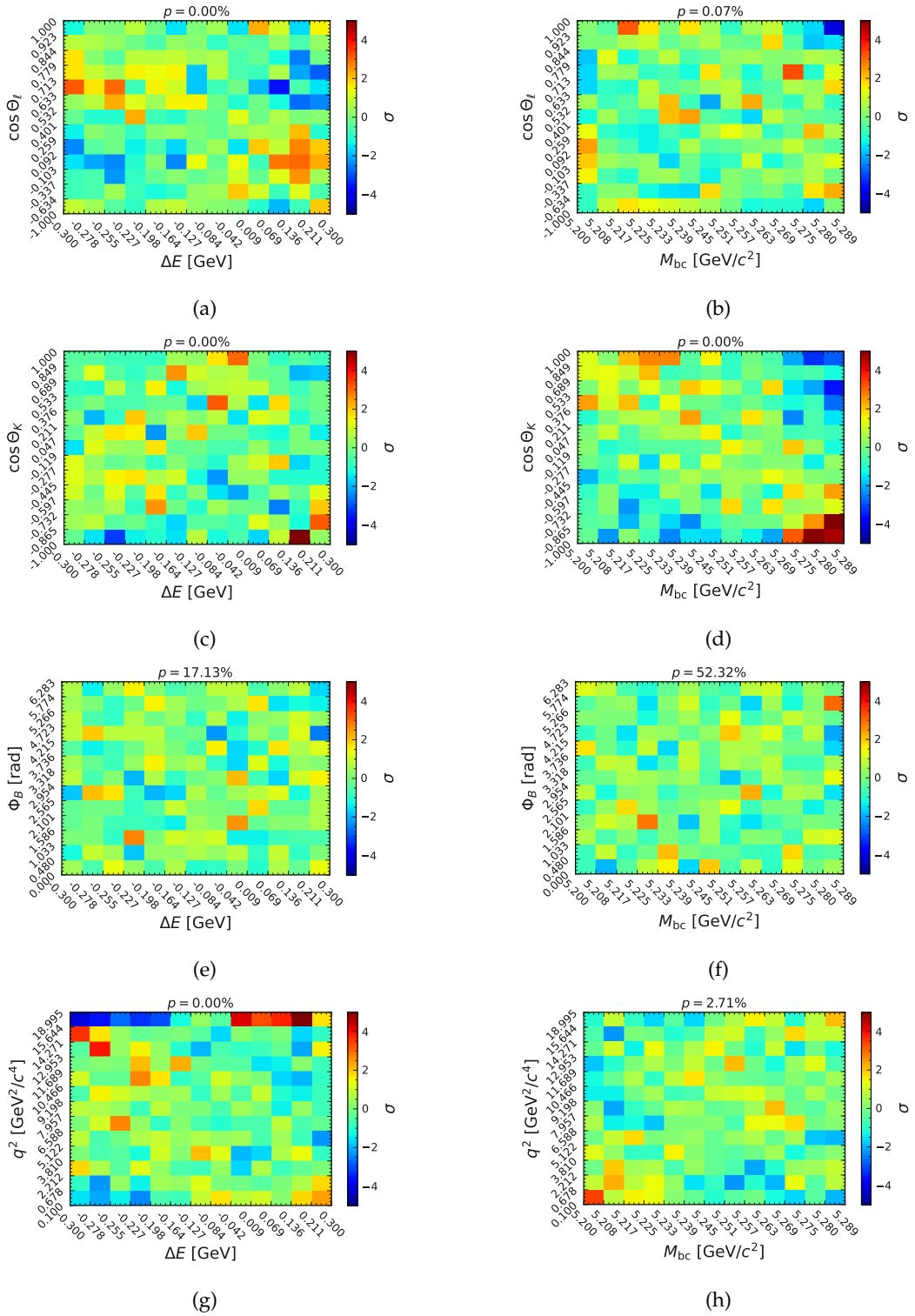
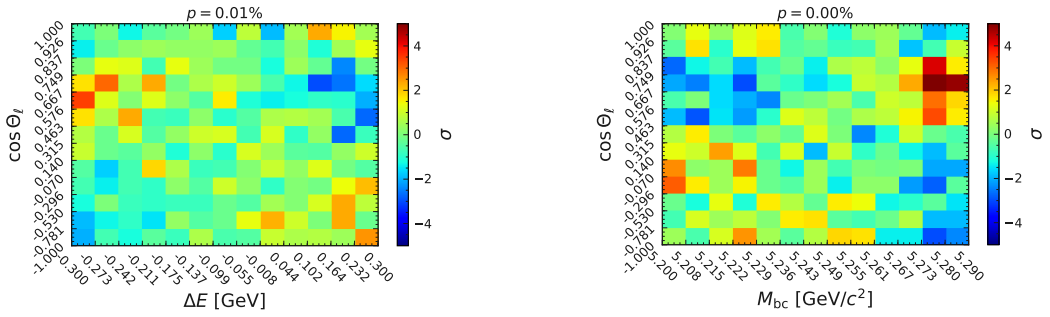
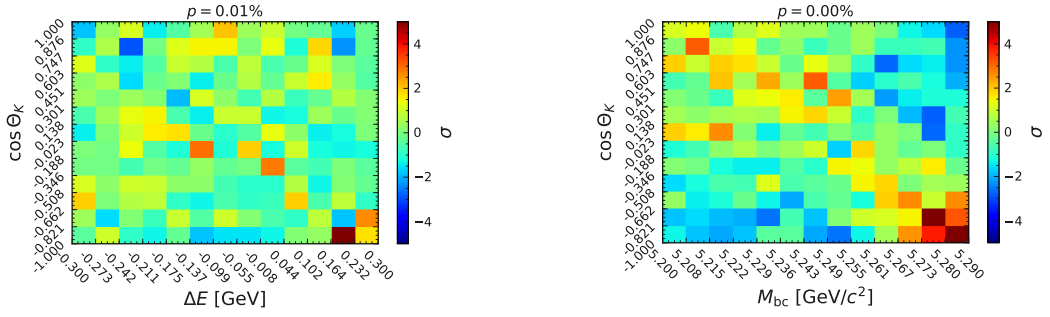


FIGURE H.4: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^+\pi^0)ee$ for background events.



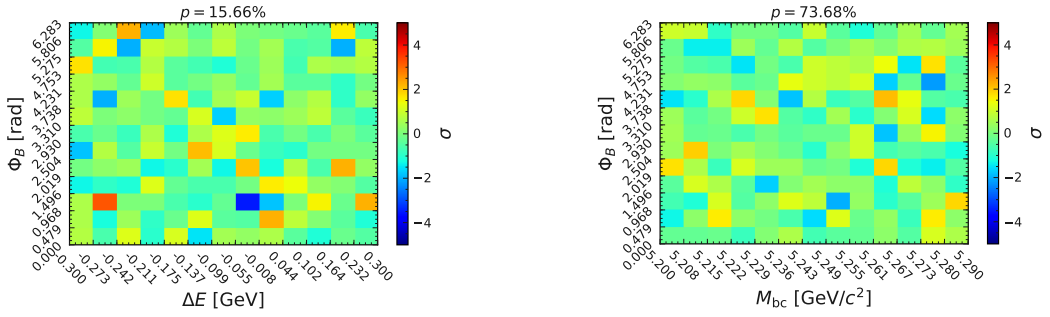
(a)

(b)



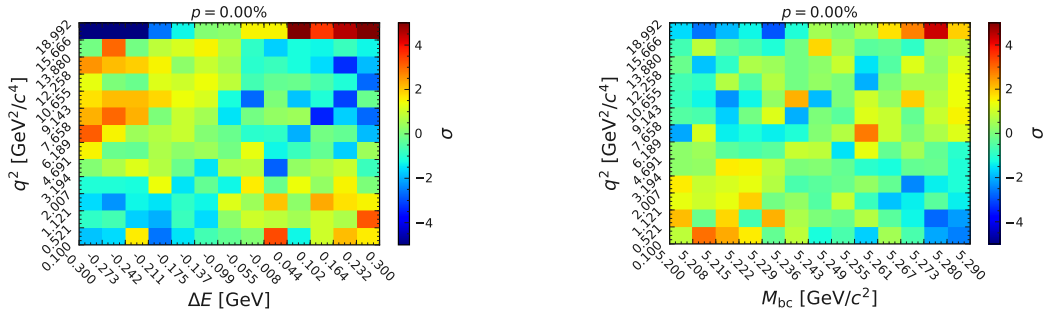
(c)

(d)



(e)

(f)



(g)

(h)

FIGURE H.5: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$ for background events.

H.1.2 Weights

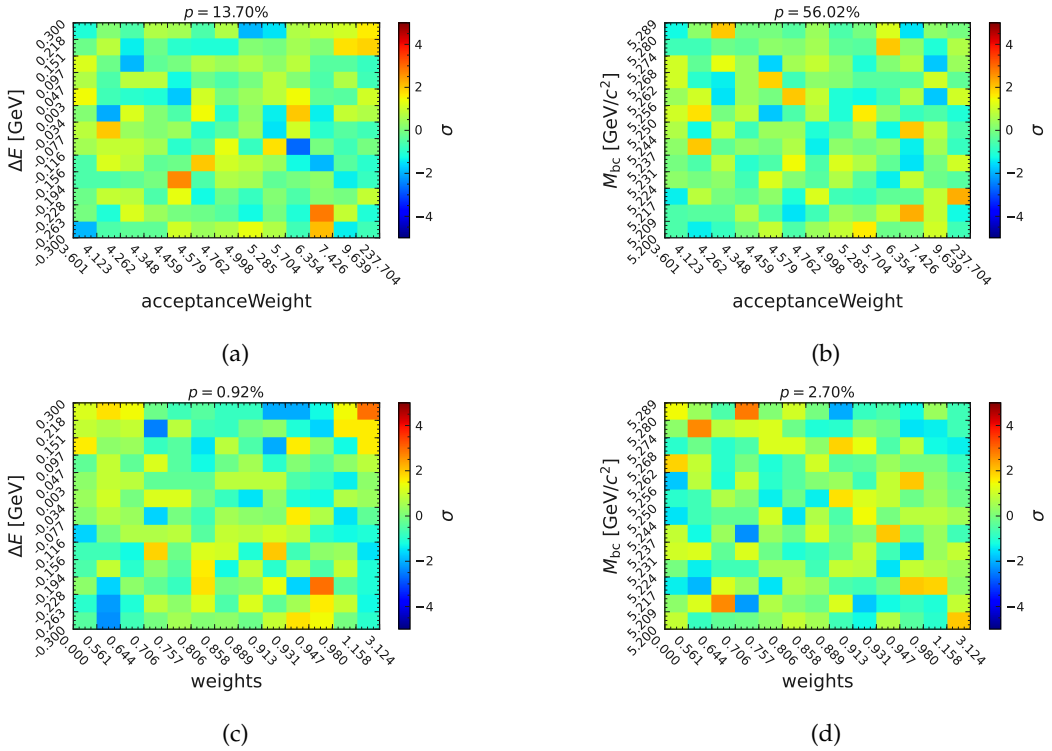


FIGURE H.6: Correlations between the $\Delta E/M_{\mathrm{bc}}$ and acceptance and PID weights in the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ for background events.

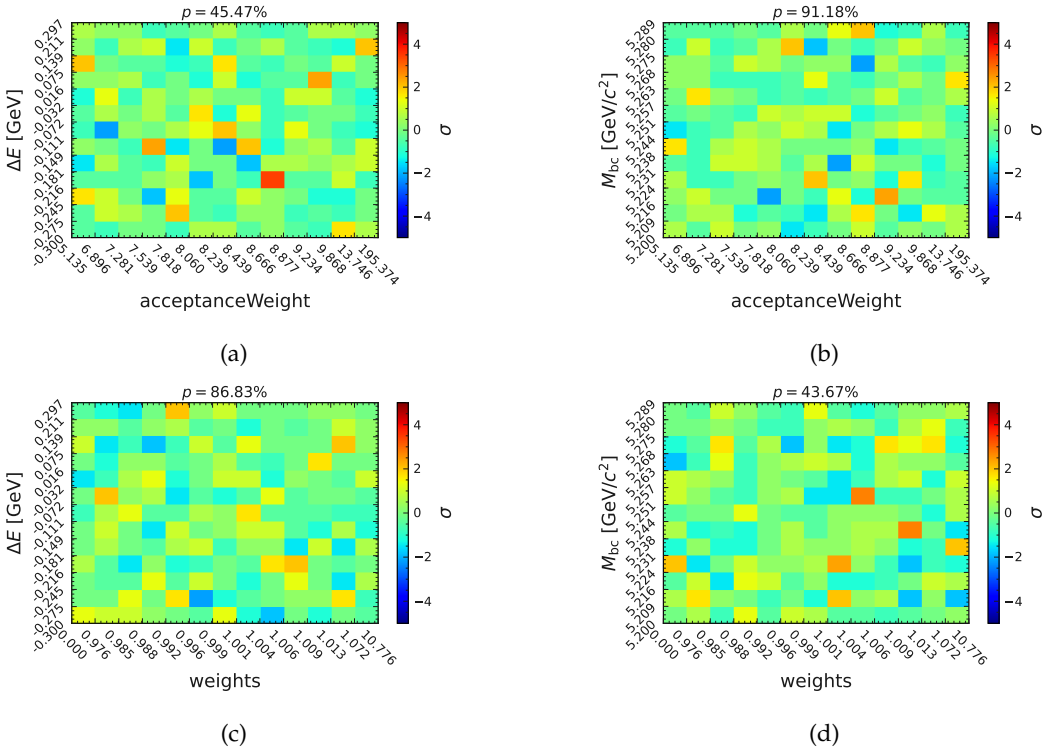


FIGURE H.7: Correlations between the $\Delta E/M_{\text{bc}}$ and acceptance and PID weights in the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ for background events.

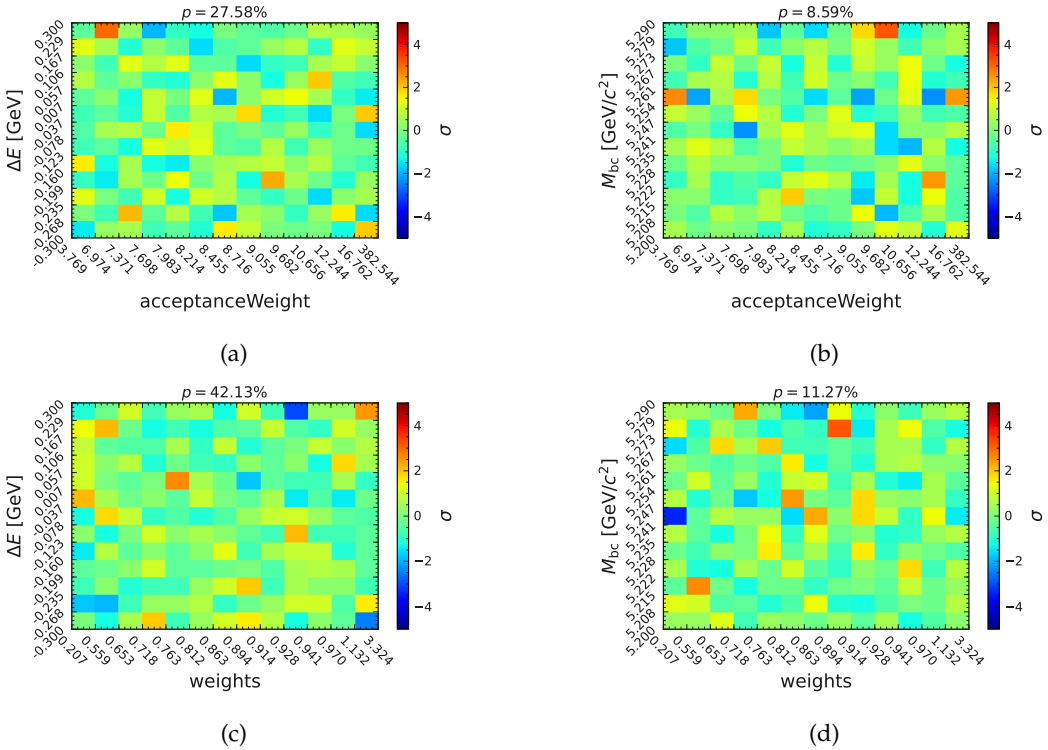


FIGURE H.8: Correlations between the $\Delta E/M_{\text{bc}}$ and acceptance and PID weights in the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ for background events.

H.2 Signal

H.2.1 Angular Variables

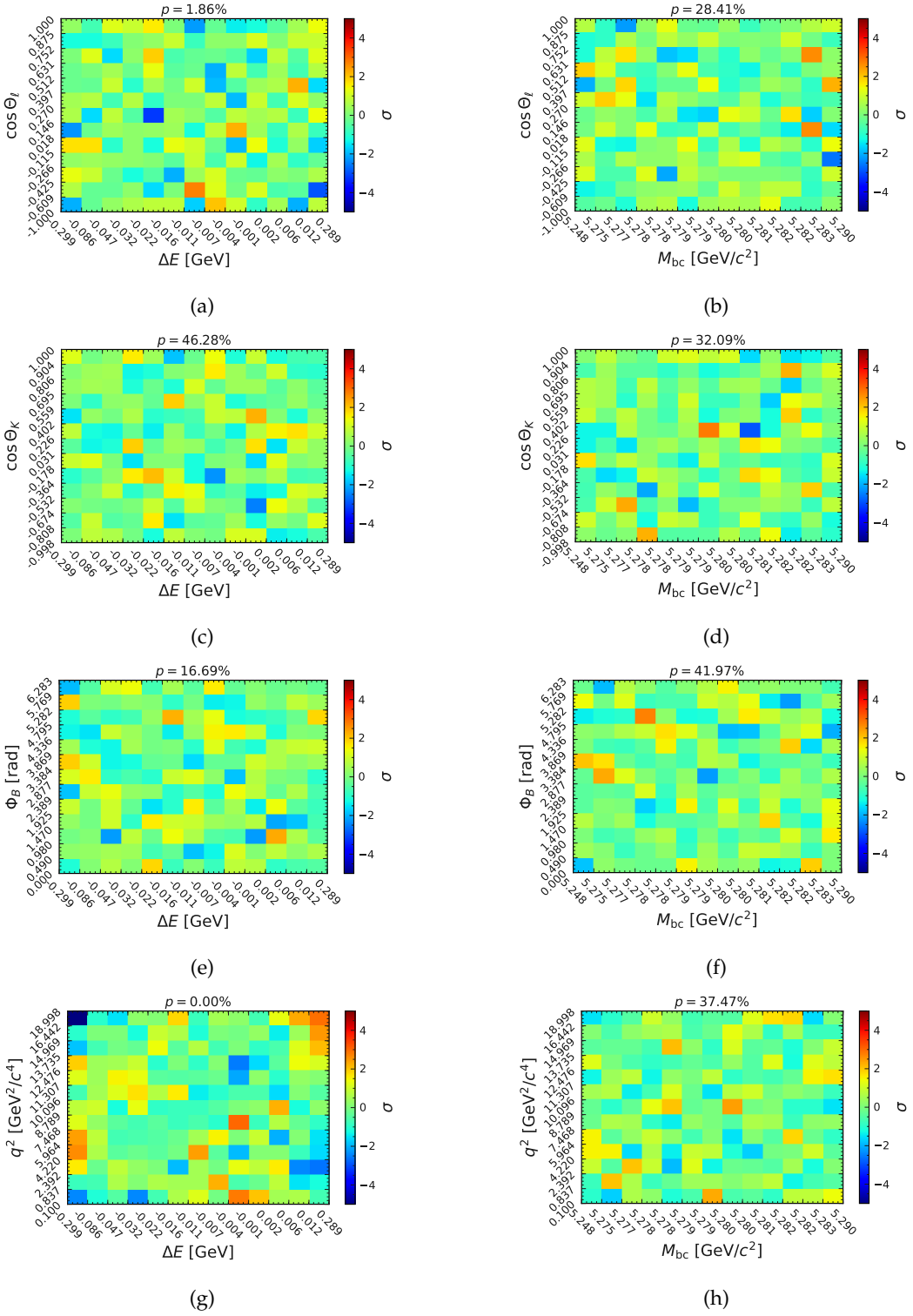
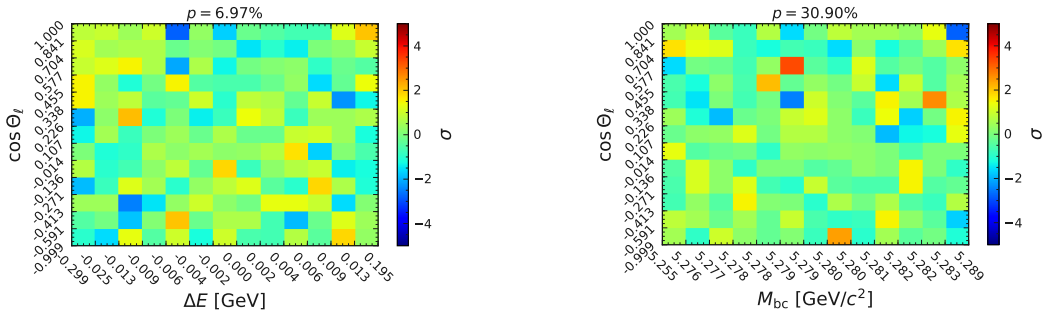
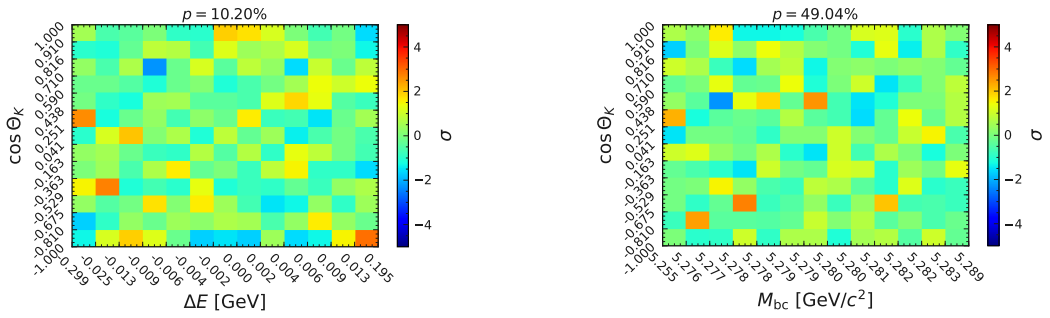


FIGURE H.9: Correlations between angles and ΔE & M_{bc} in the channel $B^0 \to K^{*0}(\to K^+\pi^-)ee$ for signal events.



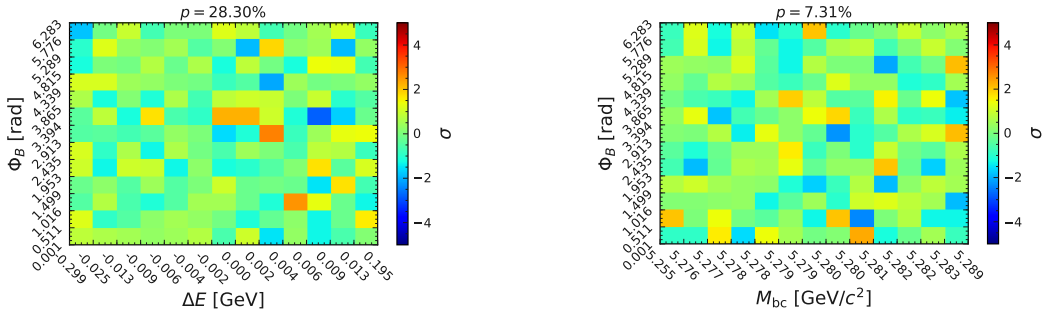
(a)

(b)



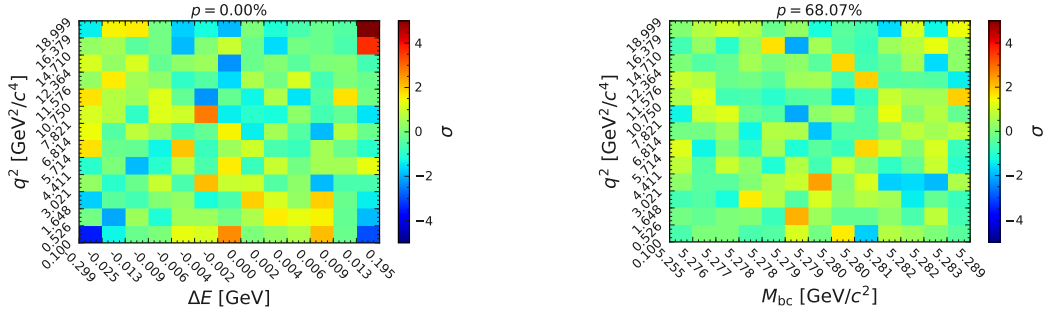
(c)

(d)



(e)

(f)



(g)

(h)

FIGURE H.10: Correlations between angles and ΔE & M_{bc} in the channel $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ for signal events.

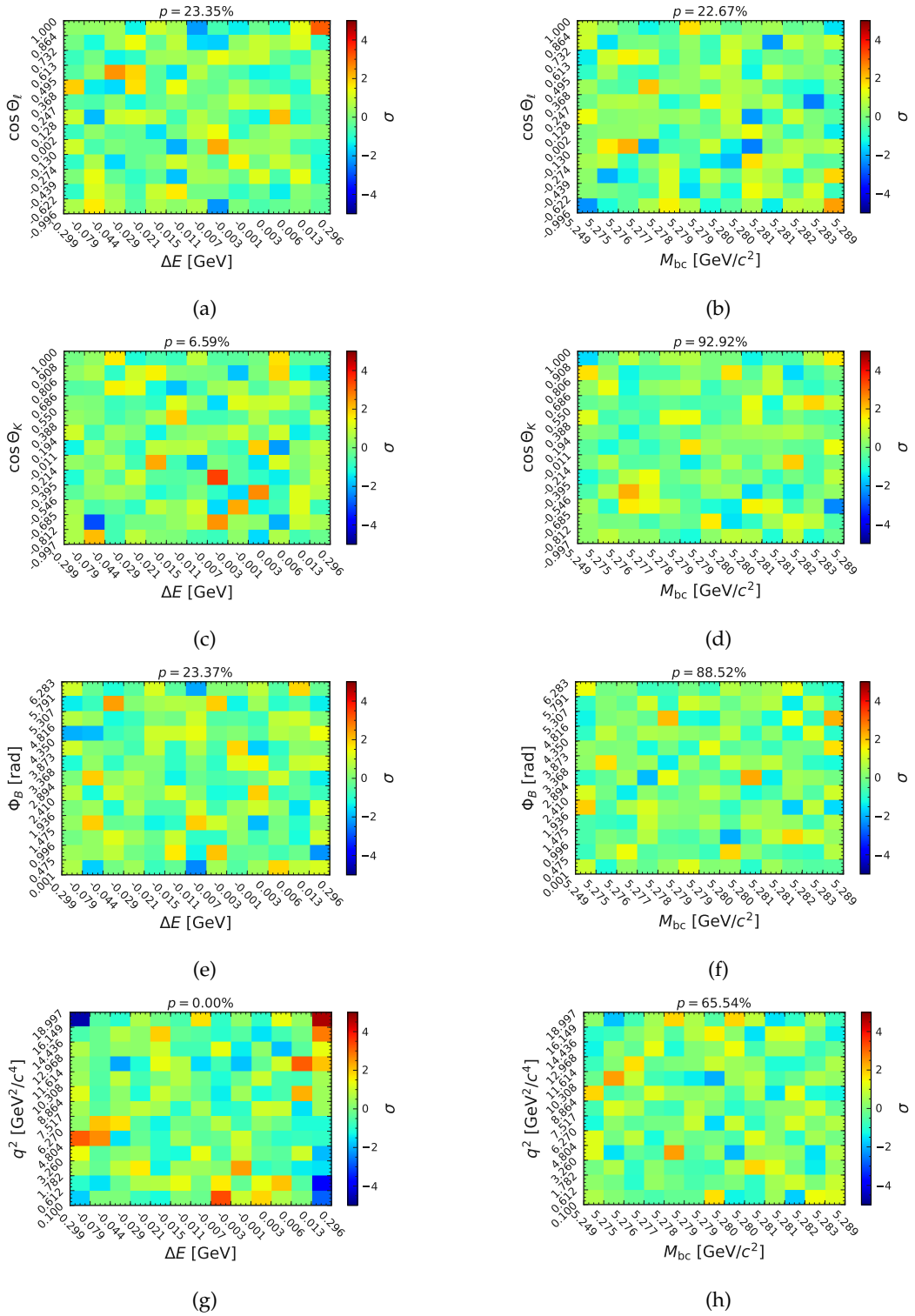


FIGURE H.11: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^0_S \pi^+)ee$ for signal events.

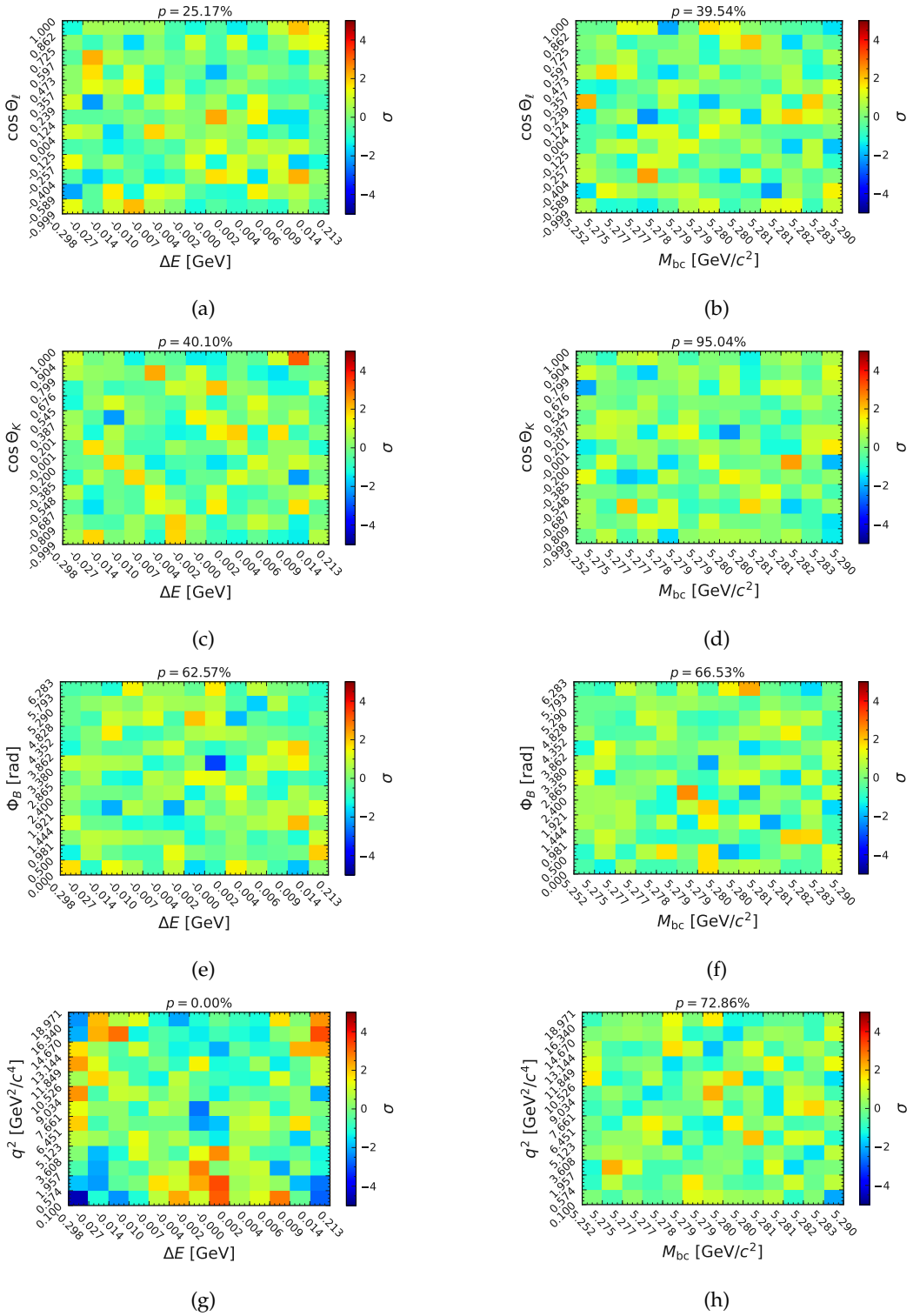


FIGURE H.12: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^0_S \pi^+)\mu\mu$ for signal events.

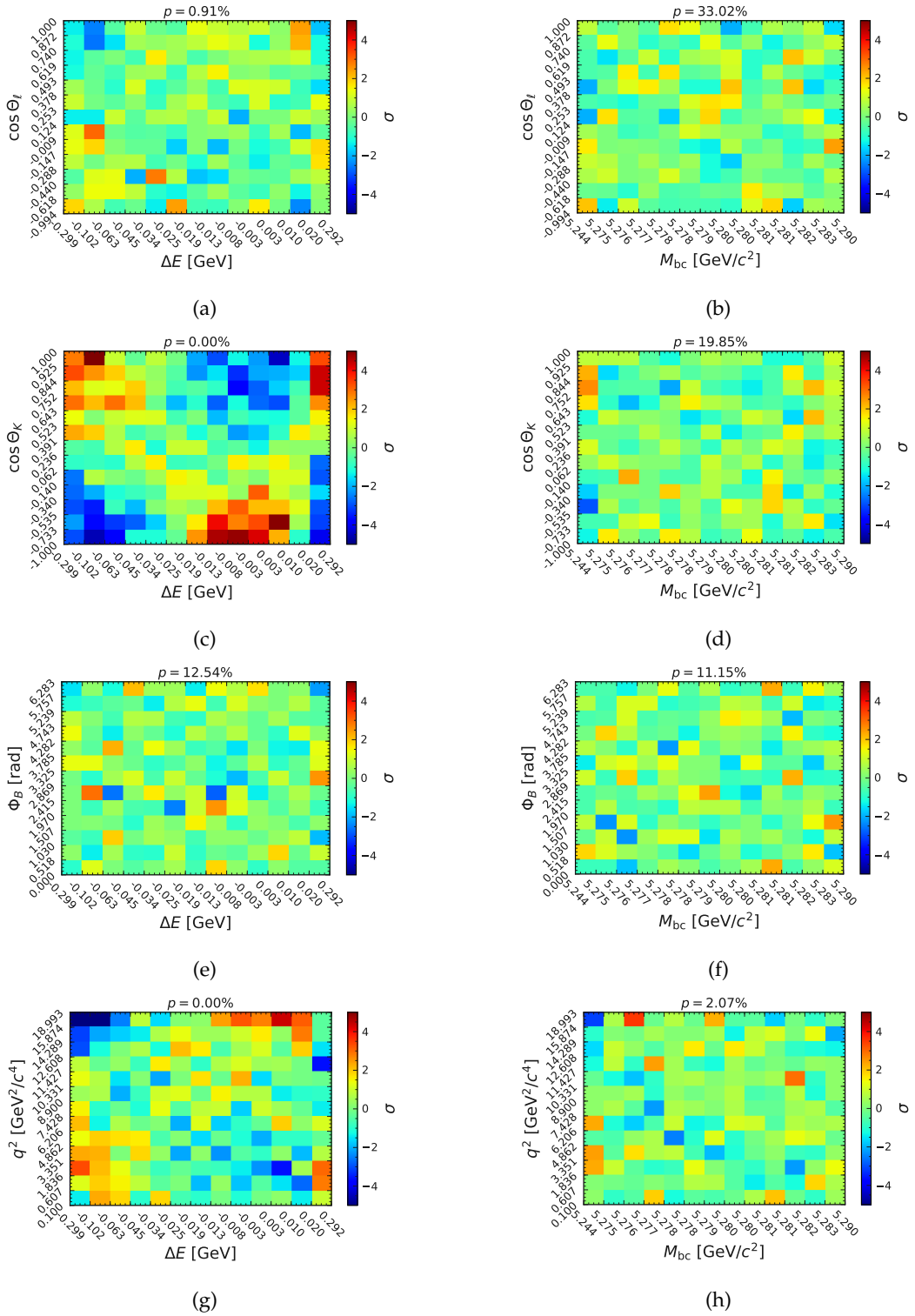
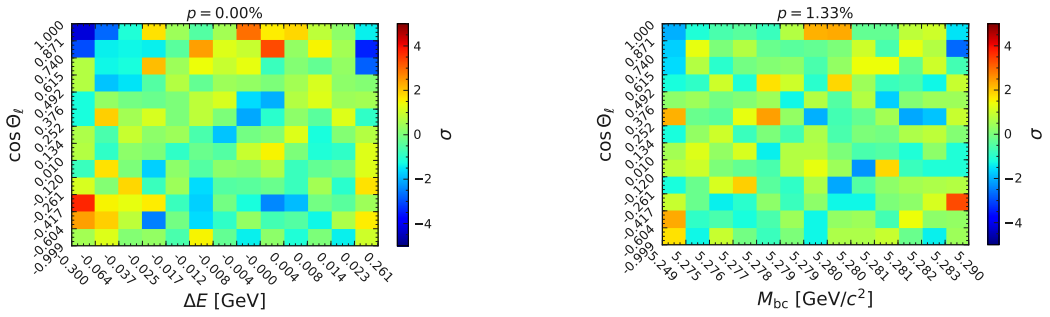
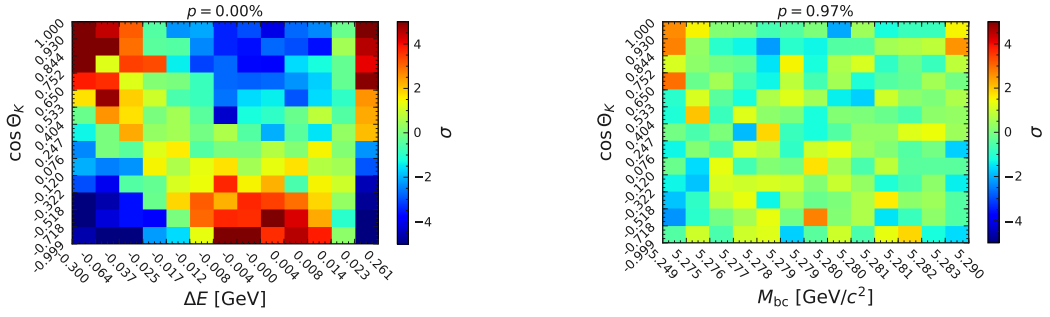


FIGURE H.13: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^+\pi^0)ee$ for signal events.



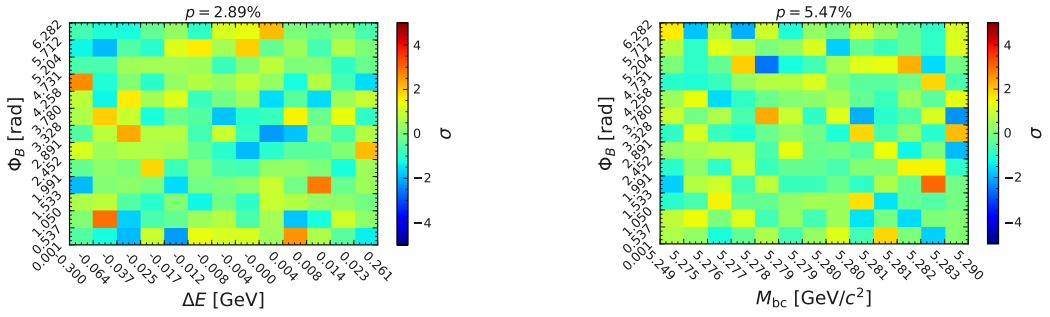
(a)

(b)



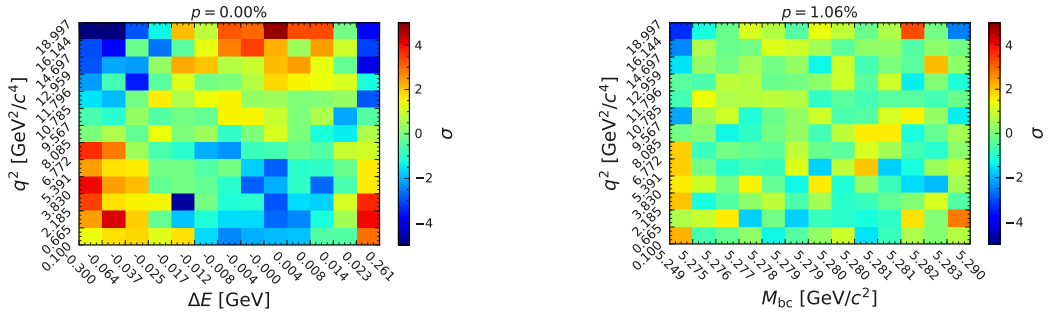
(c)

(d)



(e)

(f)



(g)

(h)

FIGURE H.14: Correlations between angles and ΔE & M_{bc} in the channel $B^+ \to K^{*+}(\to K^+\pi^0)\mu\mu$ for signal events.

H.2.2 Weights

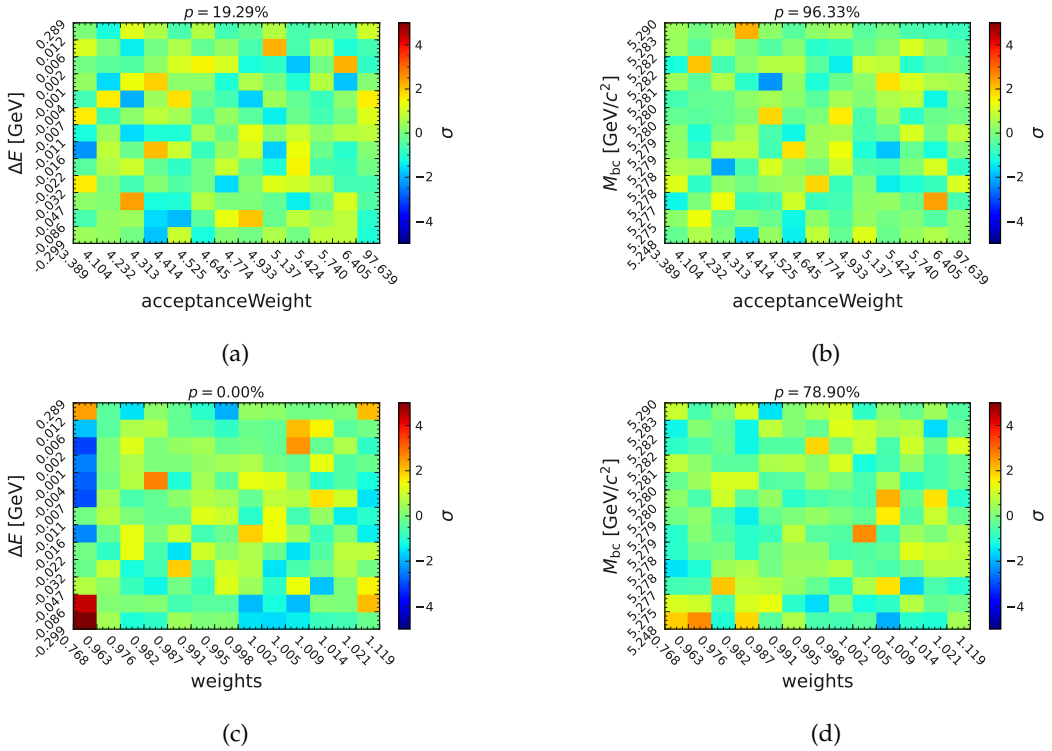


FIGURE H.15: Correlations between the $\Delta E / M_{bc}$ and acceptance and PID weights in the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ for signal events.

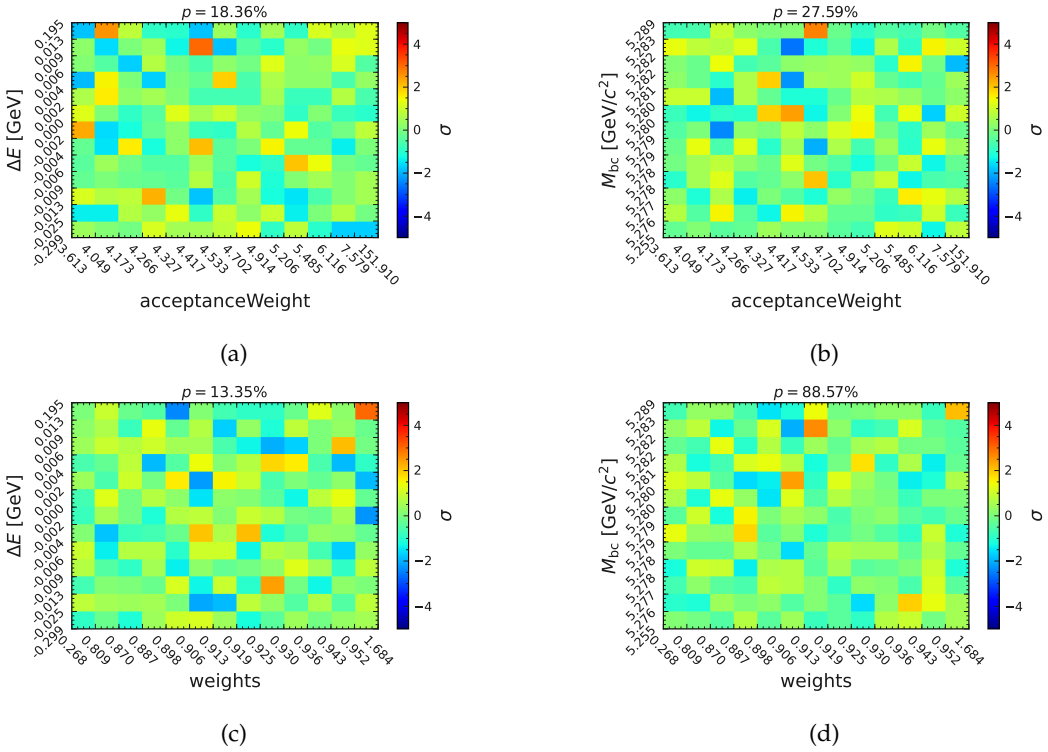


FIGURE H.16: Correlations between the $\Delta E / M_{\text{bc}}$ and acceptance and PID weights in the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ for signal events.

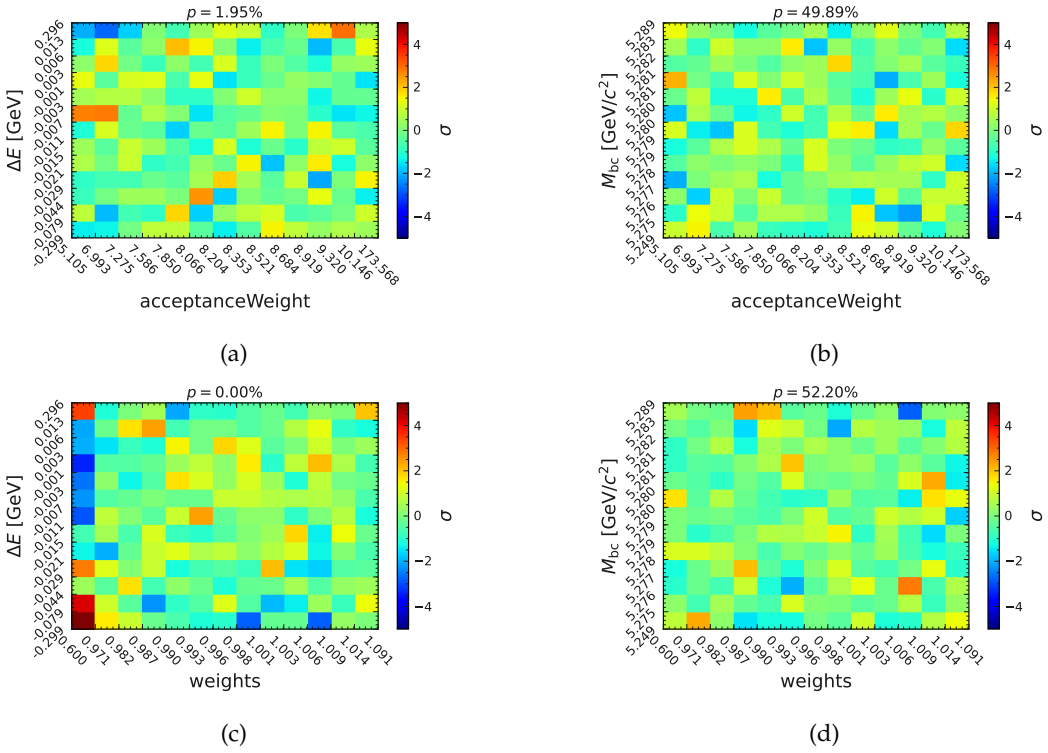


FIGURE H.17: Correlations between the $\Delta E / M_{\text{bc}}$ and acceptance and PID weights in the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ for signal events.

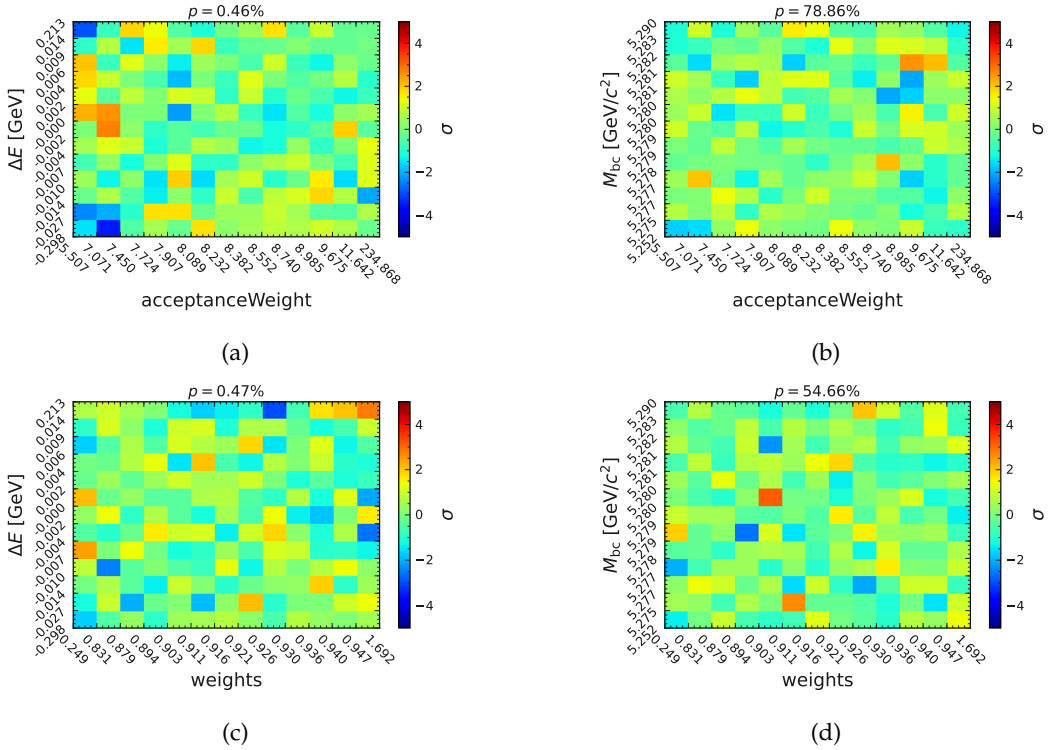


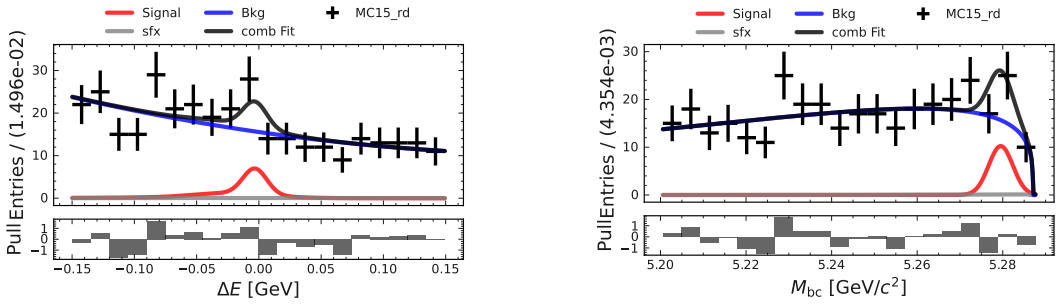
FIGURE H.18: Correlations between the $\Delta E / M_{\text{bc}}$ and acceptance and PID weights in the $B^+ \to K^{*+}(\to K^0_S \pi^+)\mu\mu$ for signal events.

Appendix I

Analysis

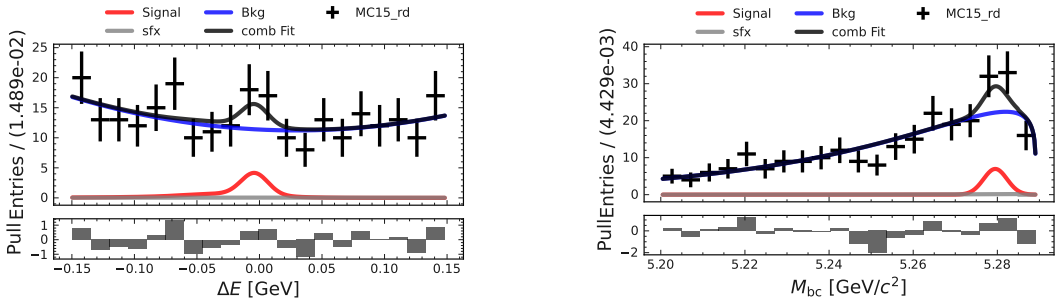
I.1 MC15_rd

I.1.1 ΔE & M_{bc} Fit Projections



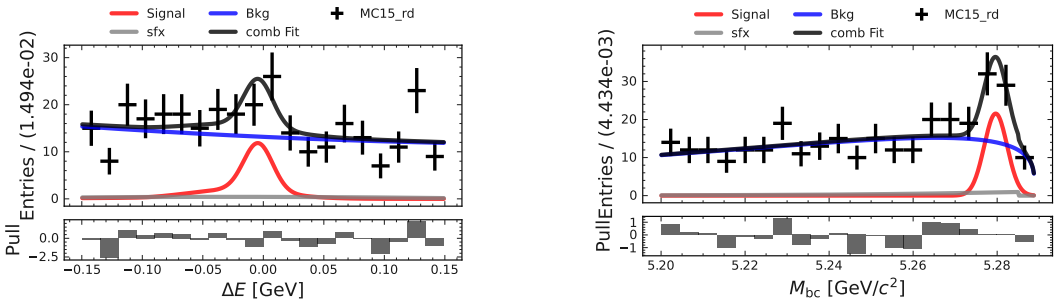
(a) (b)

FIGURE I.1: M_{bc} and ΔE fits in the q_1^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the MC15_rd dataset.



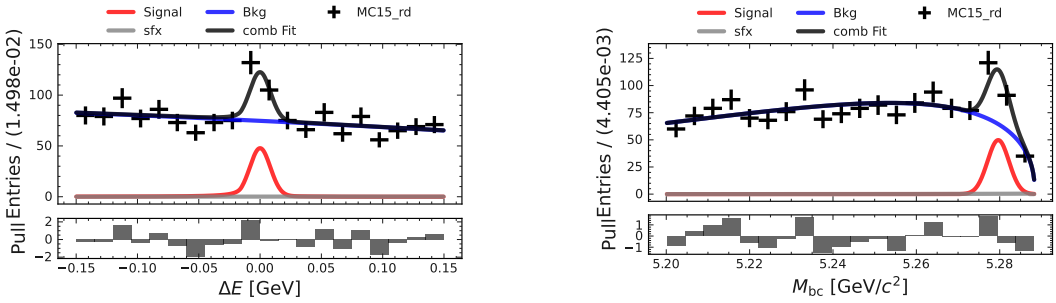
(a) (b)

FIGURE I.2: M_{bc} and ΔE fits in the q_3^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the MC15_rd dataset.



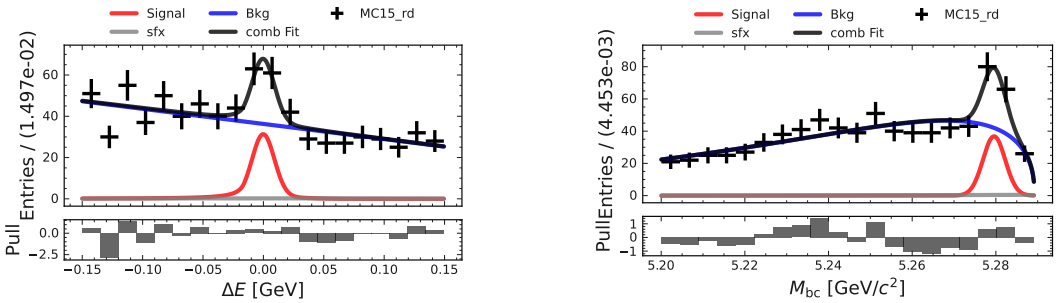
(a) (b)

FIGURE I.3: M_{bc} and ΔE fits in the q^2_4-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the MC15_rd dataset.



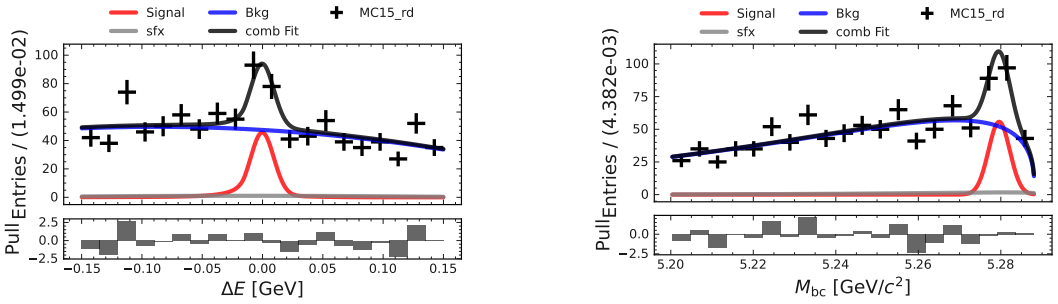
(a) (b)

FIGURE I.4: M_{bc} and ΔE fits in the q^2_1-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the MC15_rd dataset.



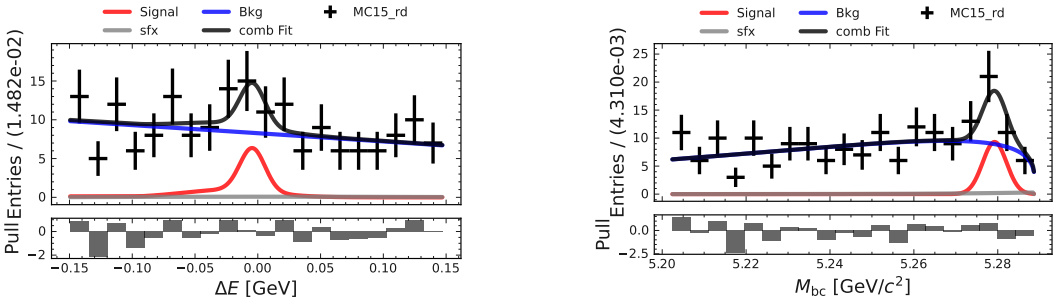
(a) (b)

FIGURE I.5: M_{bc} and ΔE fits in the q^2_3-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the MC15_rd dataset.



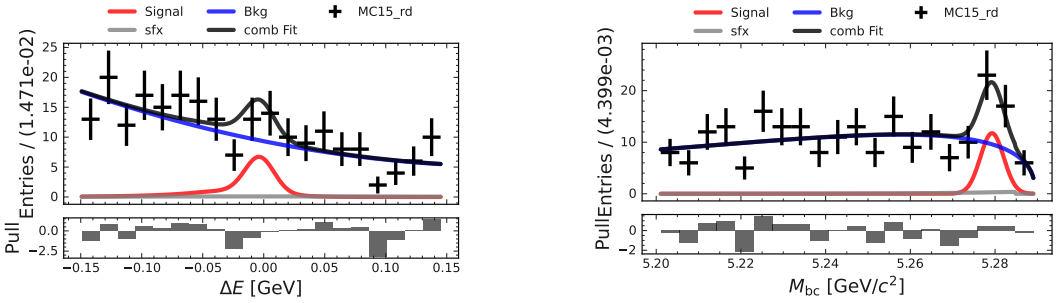
(a) (b)

FIGURE I.6: M_{bc} and ΔE fits in the q_4^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the MC15_rd dataset.



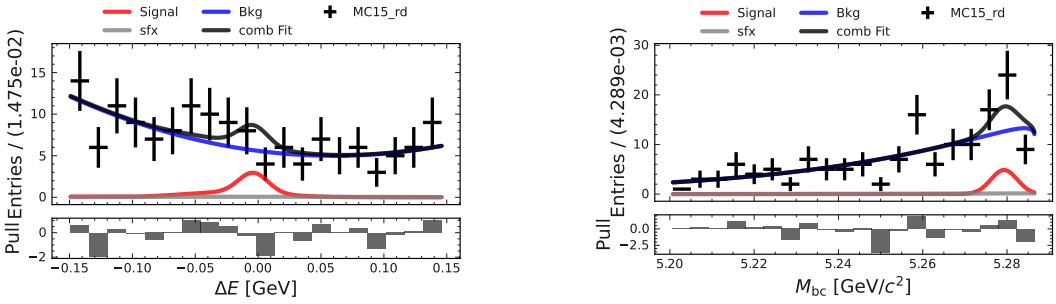
(a) (b)

FIGURE I.7: M_{bc} and ΔE fits in the q_1^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)ee$ channel with the MC15_rd dataset.



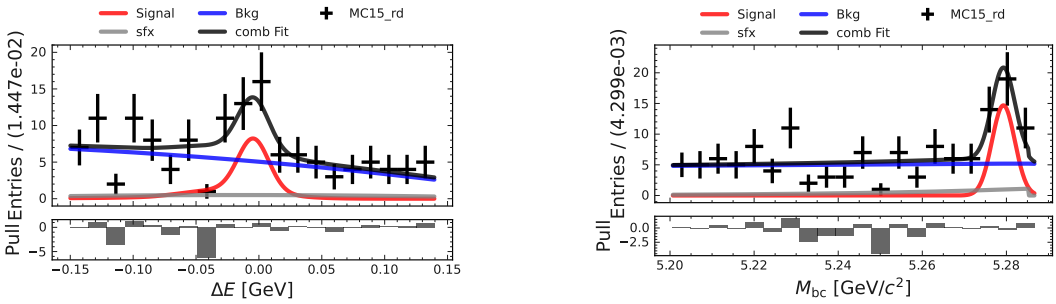
(a) (b)

FIGURE I.8: M_{bc} and ΔE fits in the q_2^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)ee$ channel with the MC15_rd dataset.



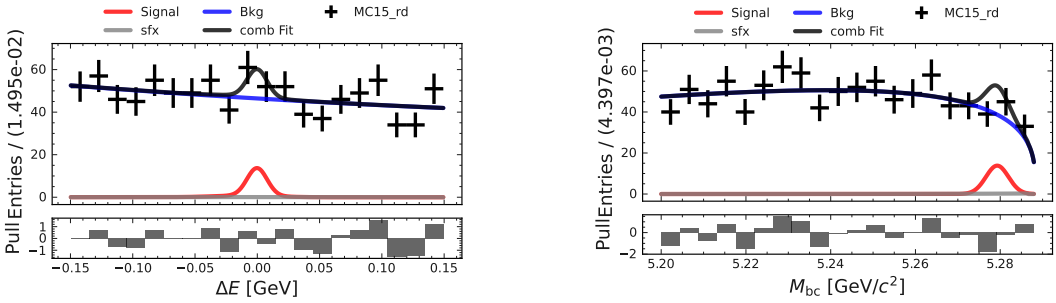
(a) (b)

FIGURE I.9: M_{bc} and ΔE fits in the q_3^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)ee$ channel with the MC15_rd dataset.



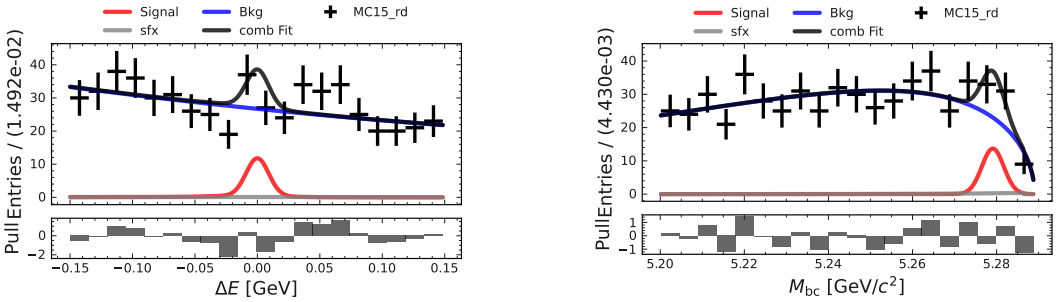
(a) (b)

FIGURE I.10: M_{bc} and ΔE fits in the q_4^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)ee$ channel with the MC15_rd dataset.



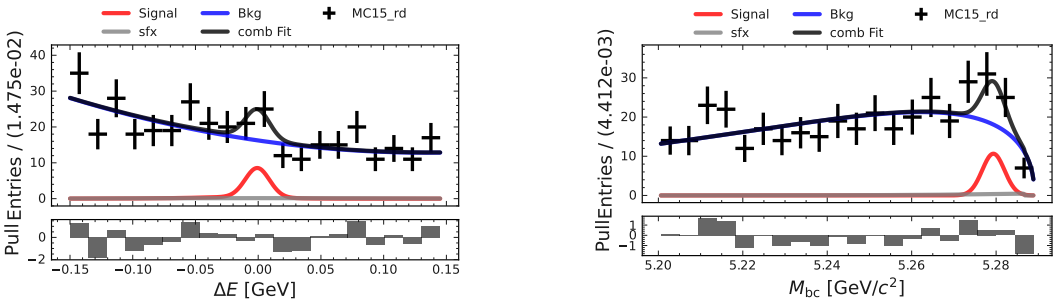
(a) (b)

FIGURE I.11: M_{bc} and ΔE fits in the q_1^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)\mu\mu$ channel with the MC15_rd dataset.



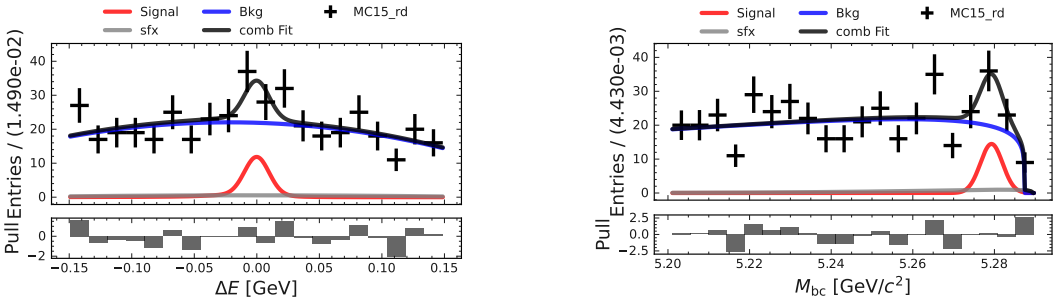
(a) (b)

FIGURE I.12: M_{bc} and ΔE fits in the q_2^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)\mu\mu$ channel with the MC15_rd dataset.



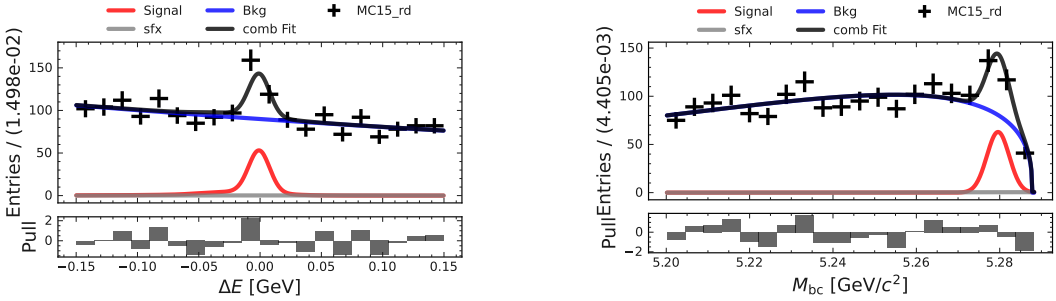
(a) (b)

FIGURE I.13: M_{bc} and ΔE fits in the q_3^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)\mu\mu$ channel with the MC15_rd dataset.



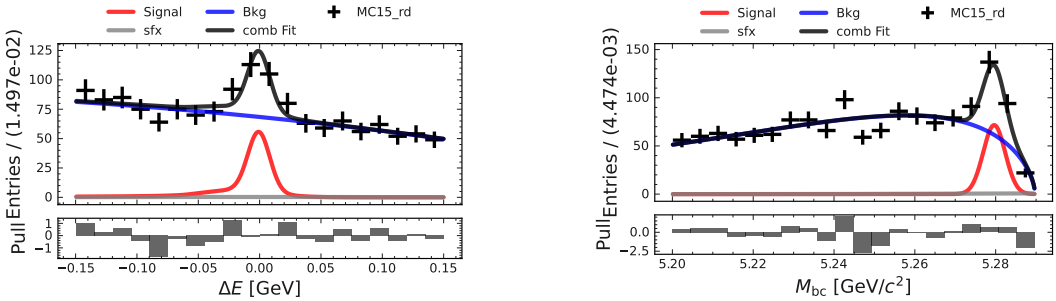
(a) (b)

FIGURE I.14: M_{bc} and ΔE fits in the q_4^2-region for the $B^+ \to K^{*+}(\to K_S^0\pi^+)\mu\mu$ channel with the MC15_rd dataset.



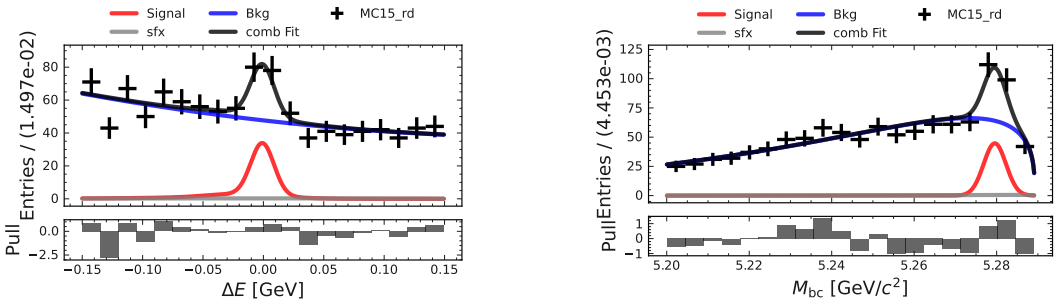
(a) (b)

FIGURE I.15: M_{bc} and ΔE fits in the q_1^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the MC15_rd dataset.



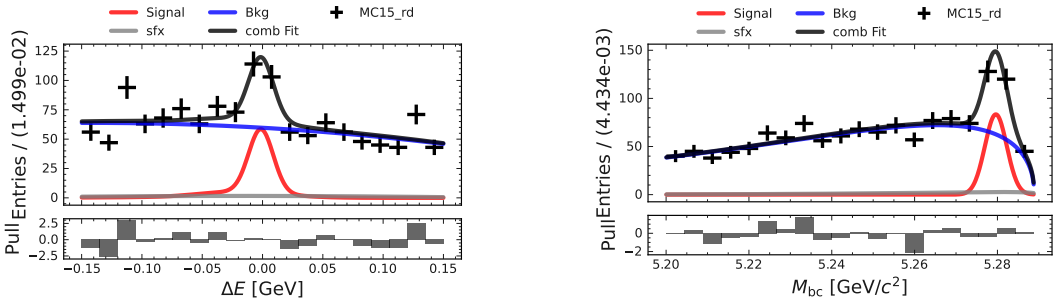
(a) (b)

FIGURE I.16: M_{bc} and ΔE fits in the q_2^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the MC15_rd dataset.



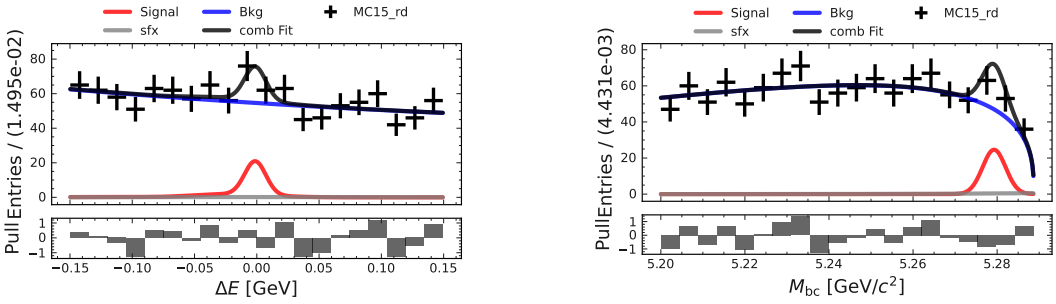
(a) (b)

FIGURE I.17: M_{bc} and ΔE fits in the q_3^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the MC15_rd dataset.



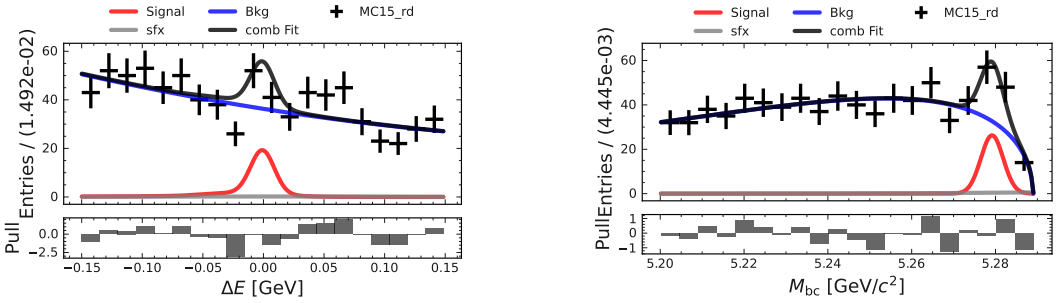
(a) (b)

FIGURE I.18: M_{bc} and ΔE fits in the q^2_4-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the MC15_rd dataset.



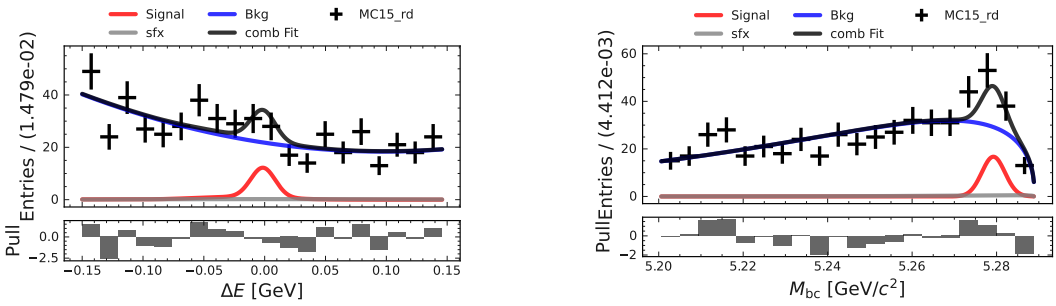
(a) (b)

FIGURE I.19: M_{bc} and ΔE fits in the q^2_1-region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel with the MC15_rd dataset.



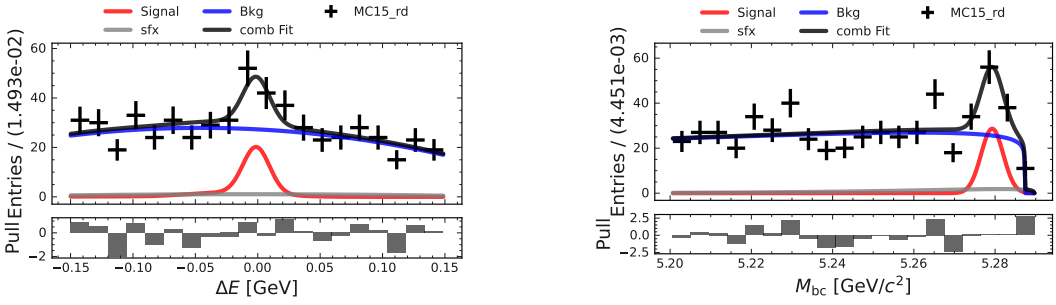
(a) (b)

FIGURE I.20: M_{bc} and ΔE fits in the q^2_2-region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel with the MC15_rd dataset.



(a) (b)

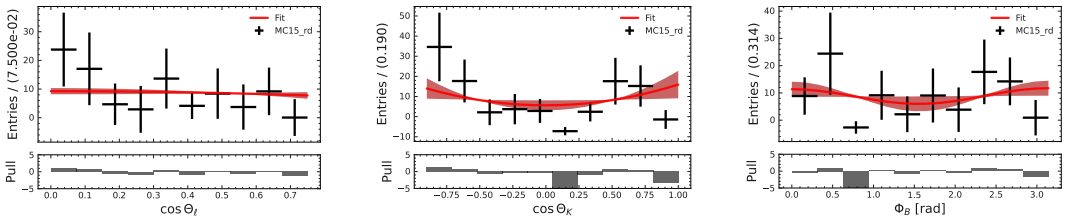
FIGURE I.21: M_{bc} and ΔE fits in the q^2_3-region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel with the MC15_rd dataset.



(a) (b)

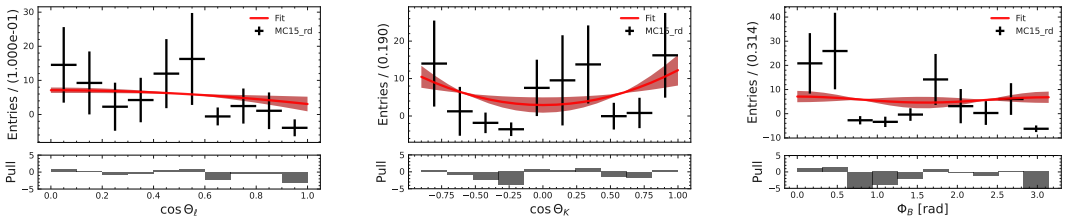
FIGURE I.22: M_{bc} and ΔE fits in the q^2_4-region for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel with the MC15_rd dataset.

I.1.2 Angular Fit Projections



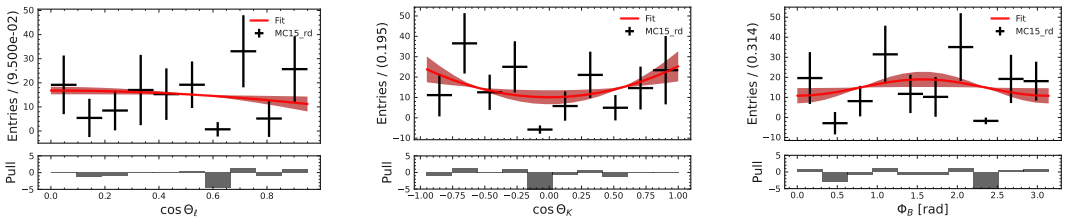
(a) (b) (c)

FIGURE I.23: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_1^2-region with the MC15_rd dataset.



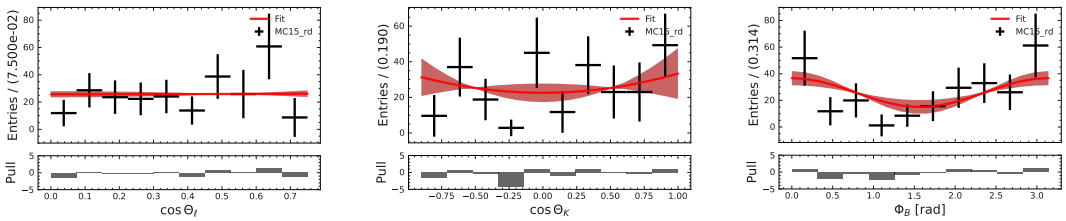
(a) (b) (c)

FIGURE I.24: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_3^2-region with the MC15_rd dataset.



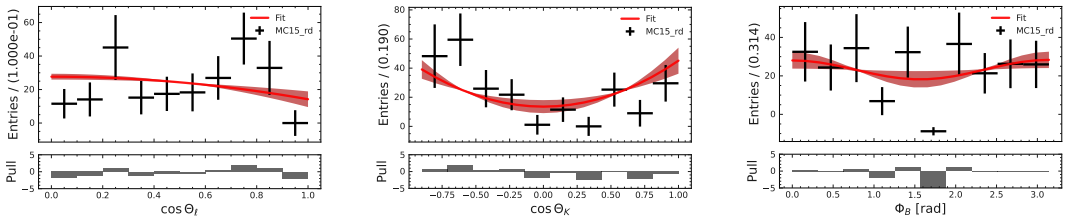
(a) (b) (c)

FIGURE I.25: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_4^2-region with the MC15_rd dataset.



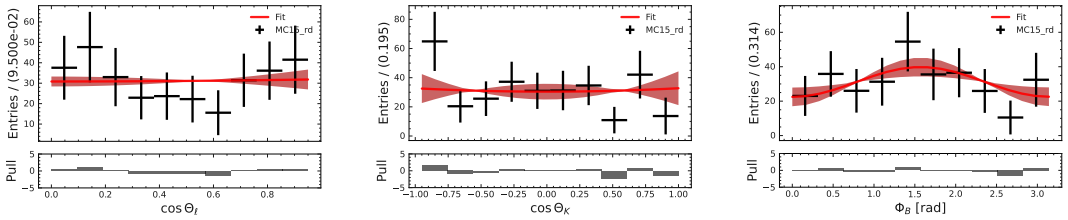
(a) (b) (c)

FIGURE I.26: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q_1^2-region with the MC15_rd dataset.



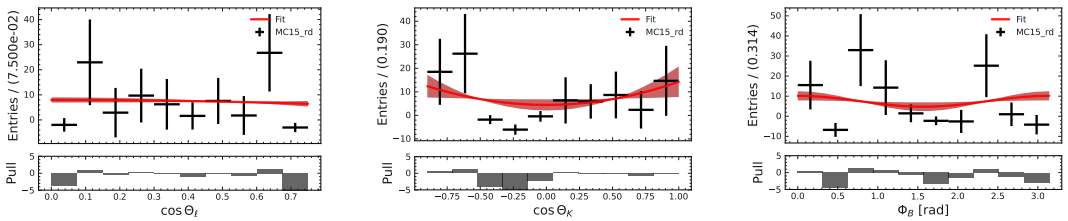
(a) (b) (c)

FIGURE I.27: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q^2_3-region with the MC15_rd dataset.



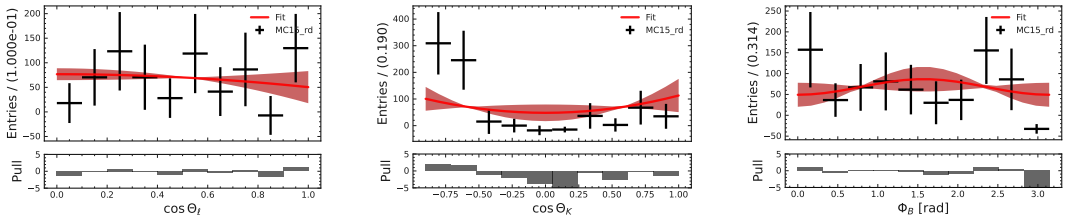
(a) (b) (c)

FIGURE I.28: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q^2_4-region with the MC15_rd dataset.



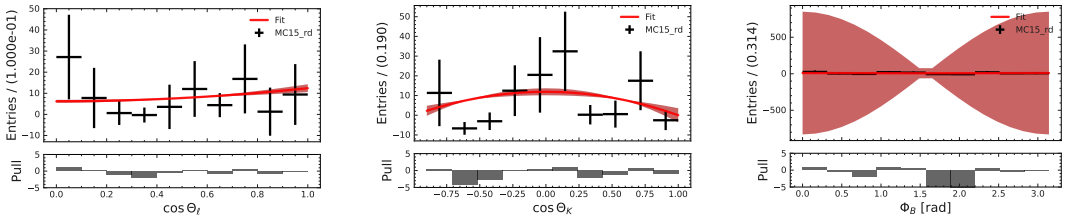
(a) (b) (c)

FIGURE I.29: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_1-region with the MC15_rd dataset.



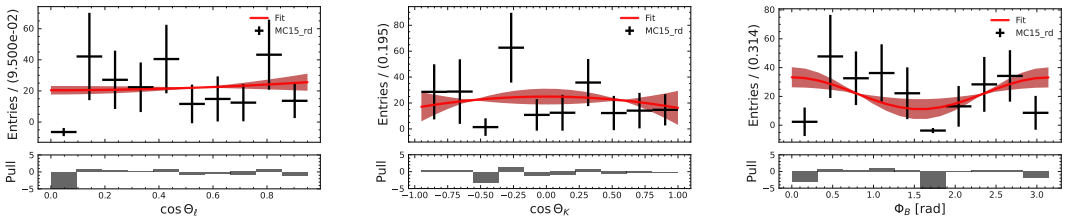
(a) (b) (c)

FIGURE I.30: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_2-region with the MC15_rd dataset.



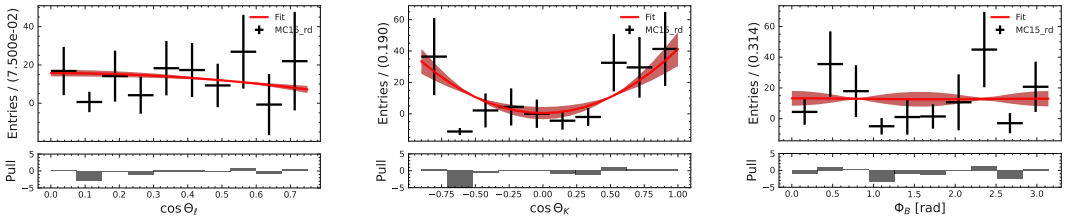
(a) (b) (c)

FIGURE I.31: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_3-region with the MC15_rd dataset.



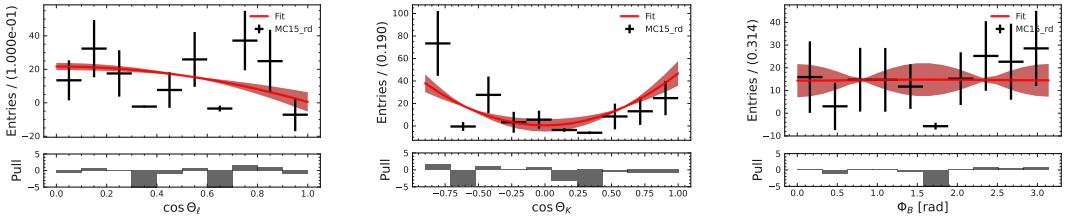
(a) (b) (c)

FIGURE I.32: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)ee$ channel in the q^2_4-region with the MC15_rd dataset.



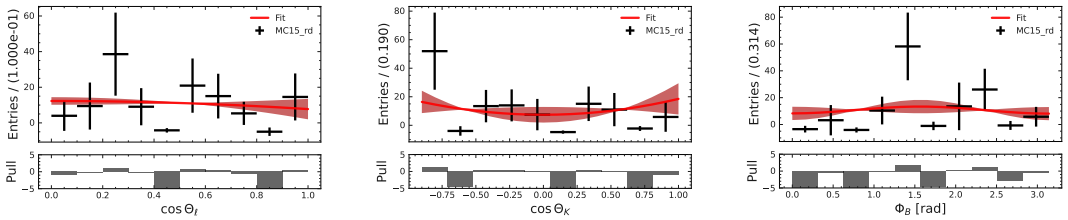
(a) (b) (c)

FIGURE I.33: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_1-region with the MC15_rd dataset.



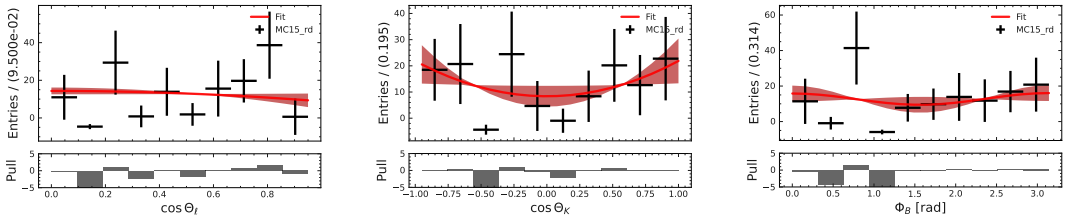
(a) (b) (c)

FIGURE I.34: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_2-region with the MC15_rd dataset.



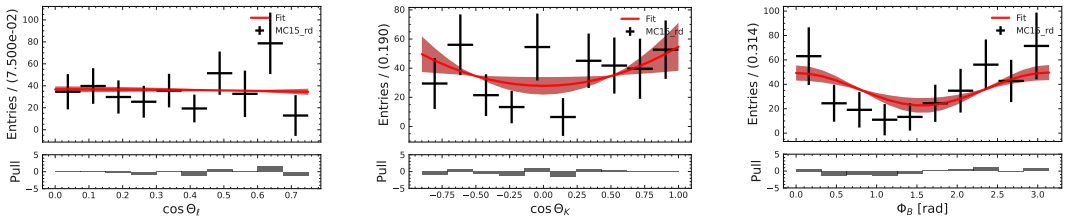
(a) (b) (c)

FIGURE I.35: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_3-region with the MC15_rd dataset.



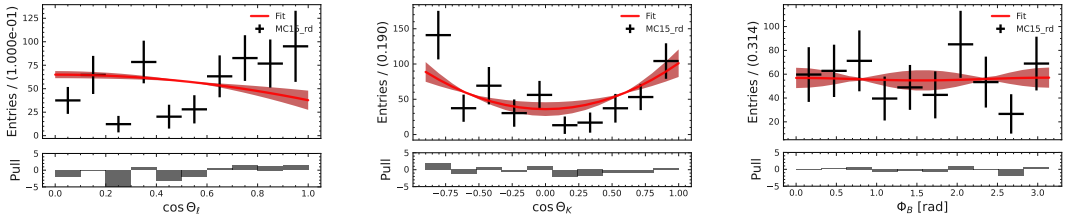
(a) (b) (c)

FIGURE I.36: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\mu\mu$ channel in the q^2_4-region with the MC15_rd dataset.



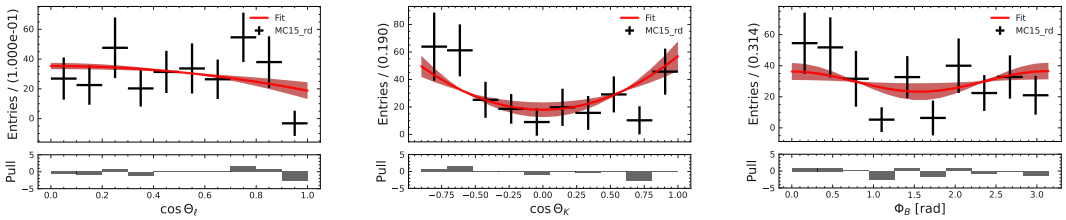
(a) (b) (c)

FIGURE I.37: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q^2_1-region with the MC15_rd dataset.



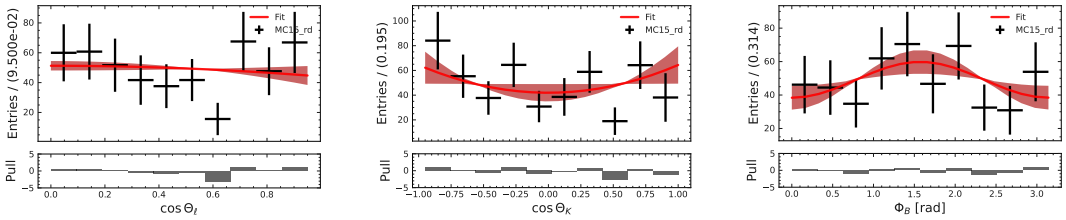
(a) (b) (c)

FIGURE I.38: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q^2_2-region with the MC15_rd dataset.



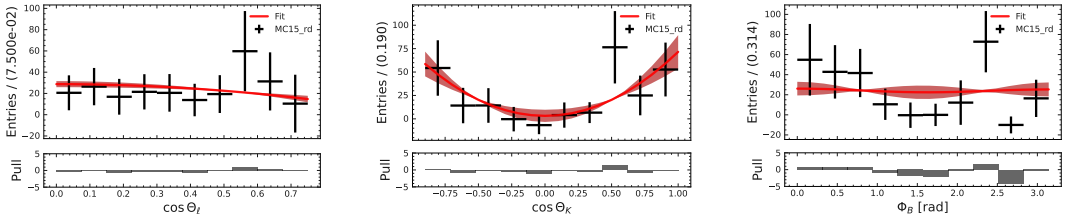
(a) (b) (c)

FIGURE I.39: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q^2_3-region with the MC15_rd dataset.



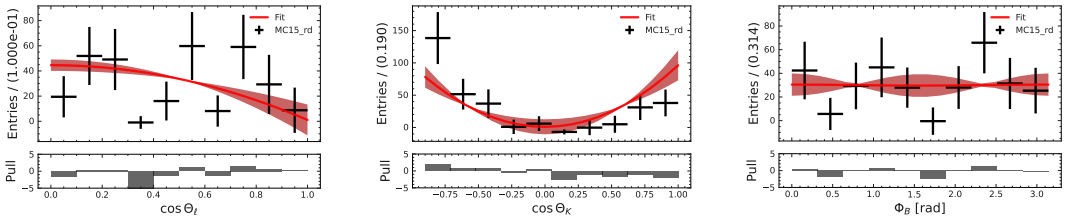
(a) (b) (c)

FIGURE I.40: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q^2_4-region with the MC15_rd dataset.



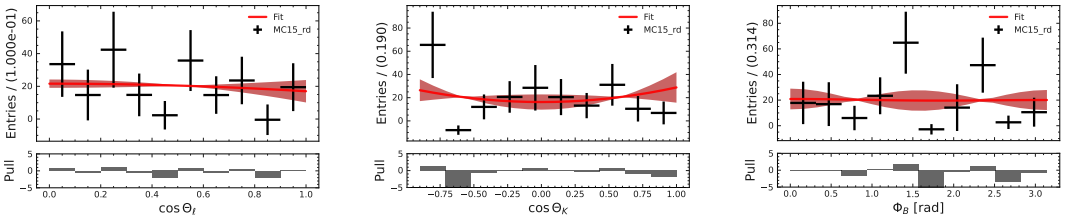
(a) (b) (c)

FIGURE I.41: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel in the q^2_1-region with the MC15_rd dataset.



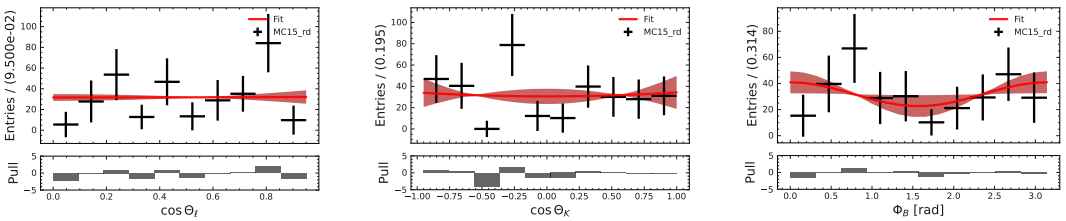
(a) (b) (c)

FIGURE I.42: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel in the q^2_2-region with the MC15_rd dataset.



(a) (b) (c)

FIGURE I.43: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel in the q^2_3-region with the MC15_rd dataset.

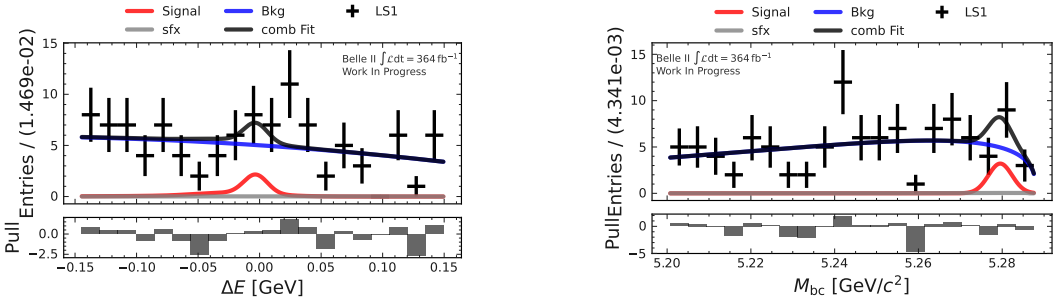


(a) (b) (c)

FIGURE I.44: Angular fit projections for the $B^+ \to K^{*+}(\to K^0_S\pi^+)\ell\ell$ channel in the q^2_4-region with the MC15_rd dataset.

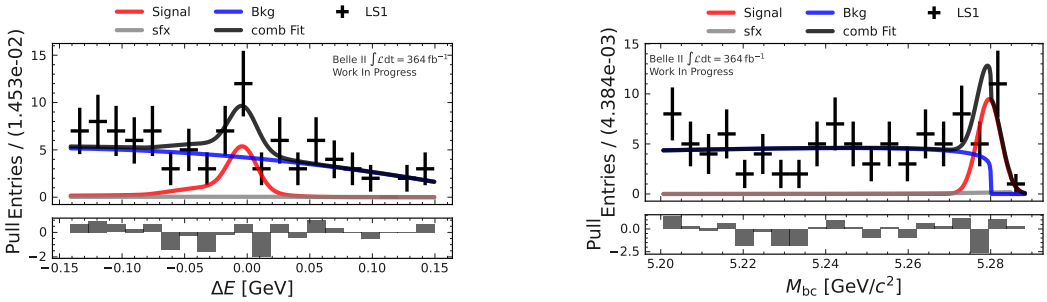
I.2 Long Shutdown 1 Data

I.2.1 ΔE & M_{bc} Fit Projections



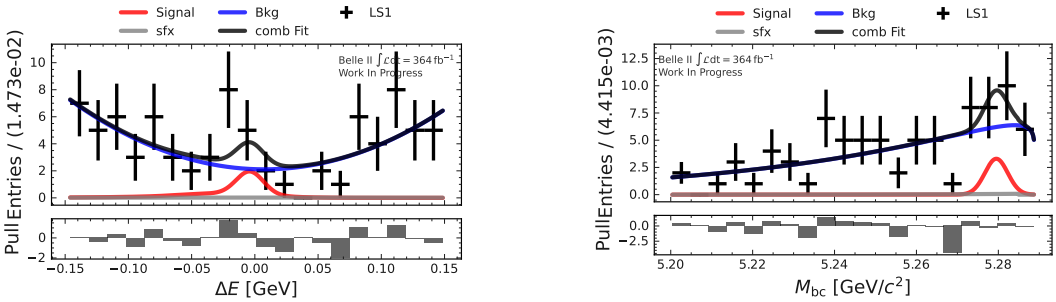
(a) (b)

FIGURE I.45: M_{bc} and ΔE fits in the q^2_1-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the LS1 dataset.



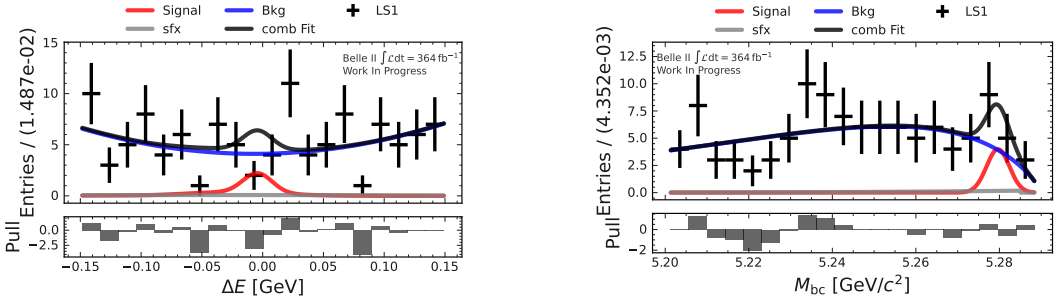
(a) (b)

FIGURE I.46: M_{bc} and ΔE fits in the q^2_2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the LS1 dataset.



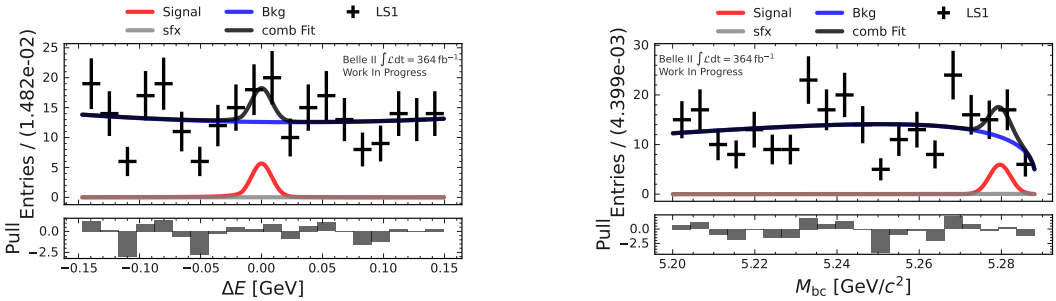
(a) (b)

FIGURE I.47: M_{bc} and ΔE fits in the q^2_3-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the LS1 dataset.



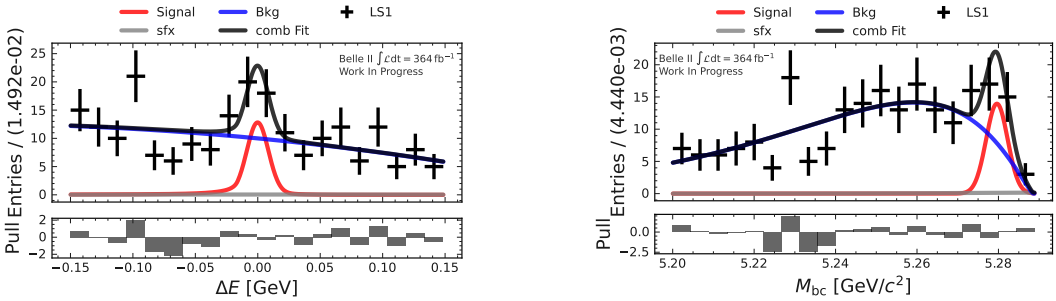
(a) (b)

FIGURE I.48: M_{bc} and ΔE fits in the q_4^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel with the LS1 dataset.



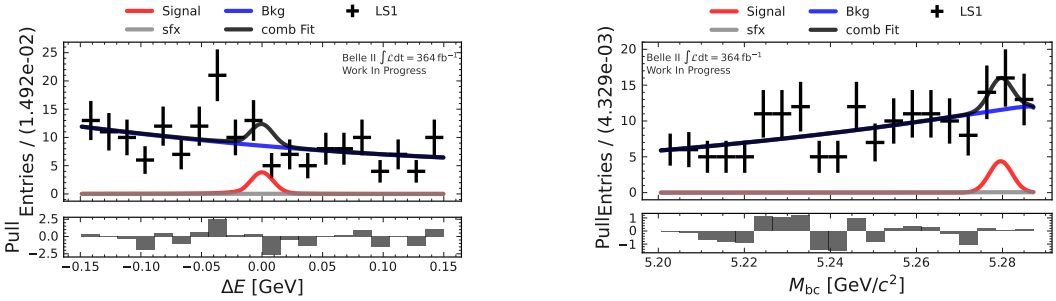
(a) (b)

FIGURE I.49: M_{bc} and ΔE fits in the q_1^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the LS1 dataset.



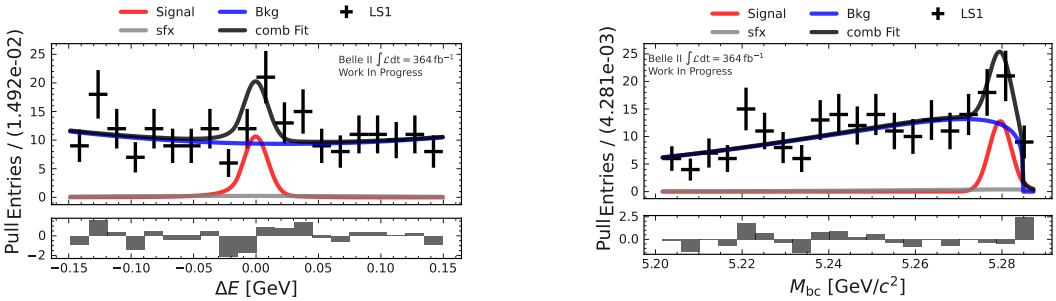
(a) (b)

FIGURE I.50: M_{bc} and ΔE fits in the q_2^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the LS1 dataset.



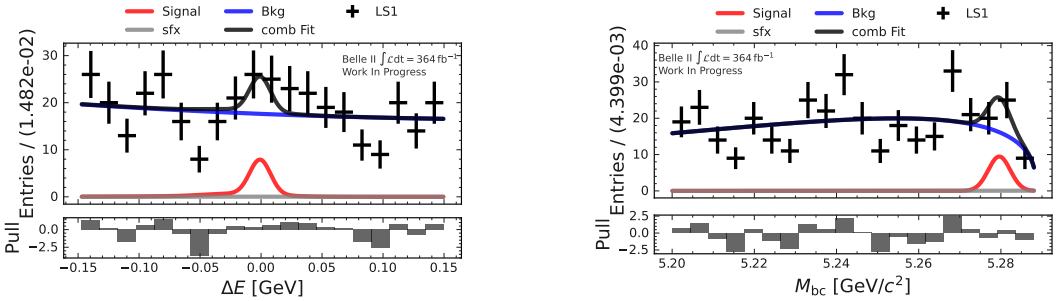
(a) (b)

FIGURE I.51: M_{bc} and ΔE fits in the q_3^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the LS1 dataset.



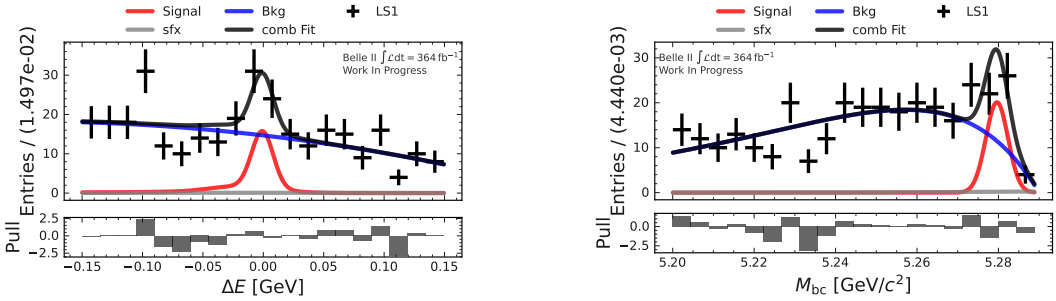
(a) (b)

FIGURE I.52: M_{bc} and ΔE fits in the q_4^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel with the LS1 dataset.



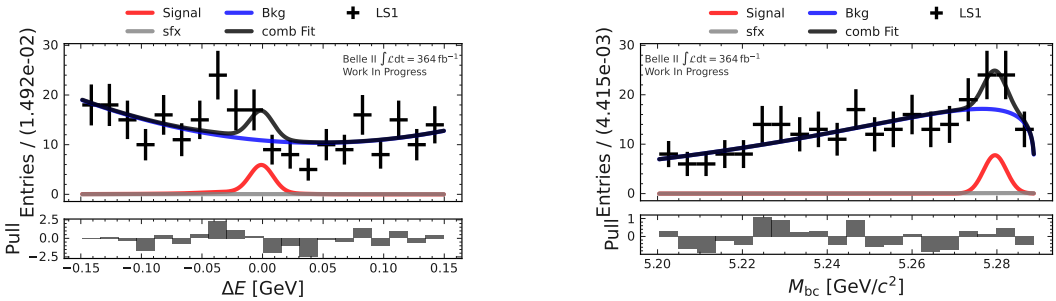
(a) (b)

FIGURE I.53: M_{bc} and ΔE fits in the q_1^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the LS1 dataset.



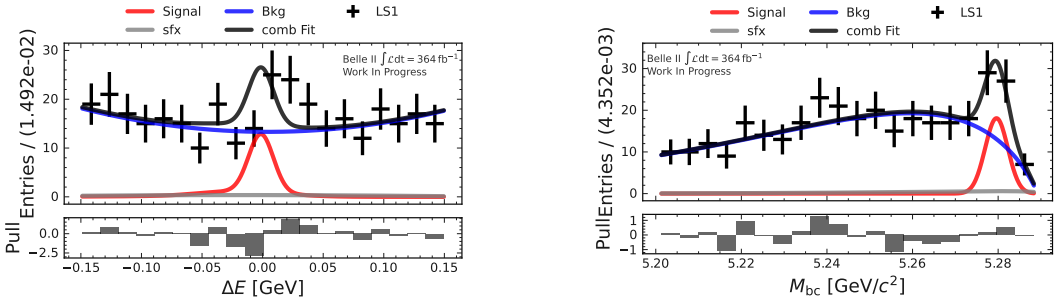
(a) (b)

FIGURE I.54: M_{bc} and ΔE fits in the q_2^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the LS1 dataset.



(a) (b)

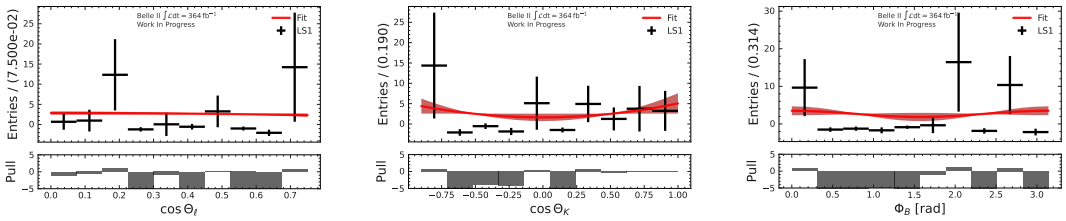
FIGURE I.55: M_{bc} and ΔE fits in the q_3^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the LS1 dataset.



(a) (b)

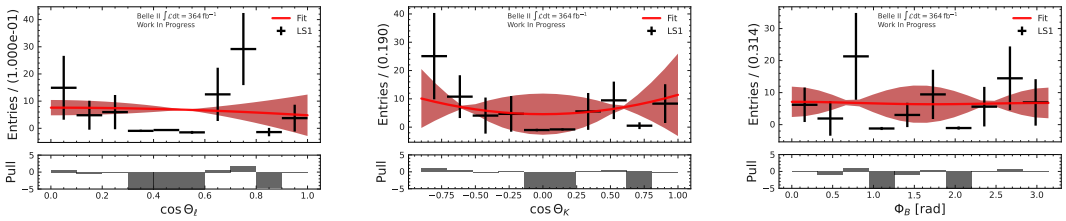
FIGURE I.56: M_{bc} and ΔE fits in the q_4^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel with the LS1 dataset.

I.2.2 Angular Fit Projections



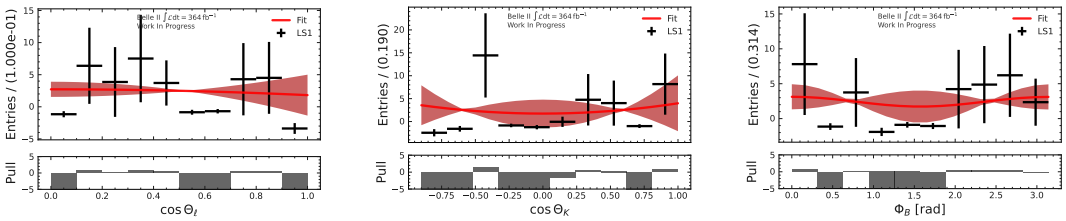
(a) (b) (c)

FIGURE I.57: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_1^2-region with the LS1 dataset.



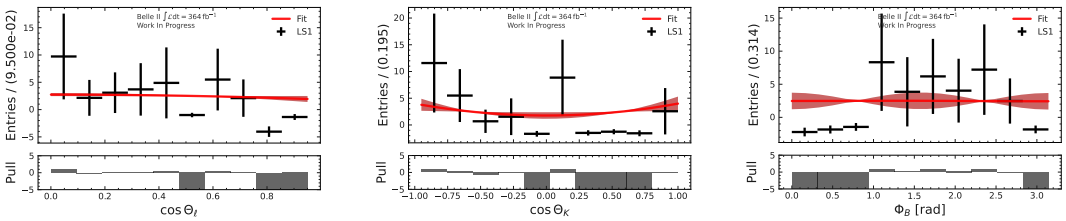
(a) (b) (c)

FIGURE I.58: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_2^2-region with the LS1 dataset.



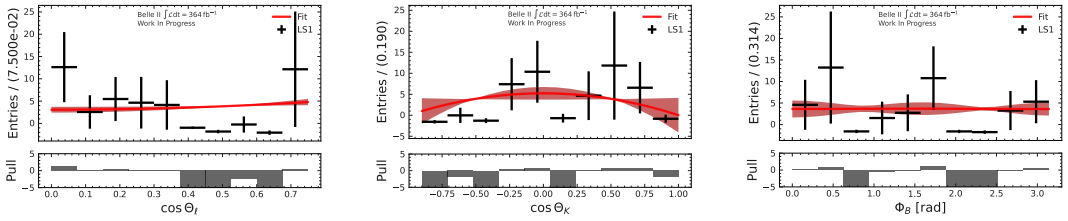
(a) (b) (c)

FIGURE I.59: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_3^2-region with the LS1 dataset.



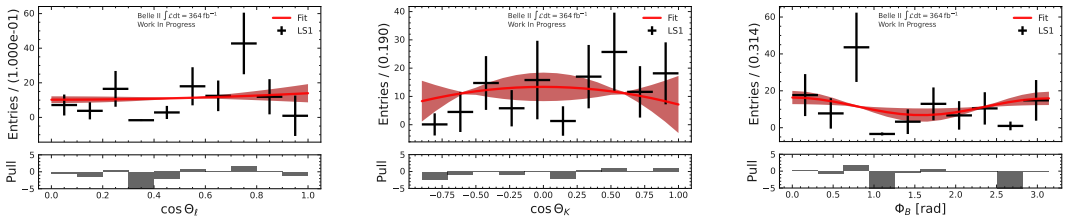
(a) (b) (c)

FIGURE I.60: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ channel in the q_4^2-region with the LS1 dataset.



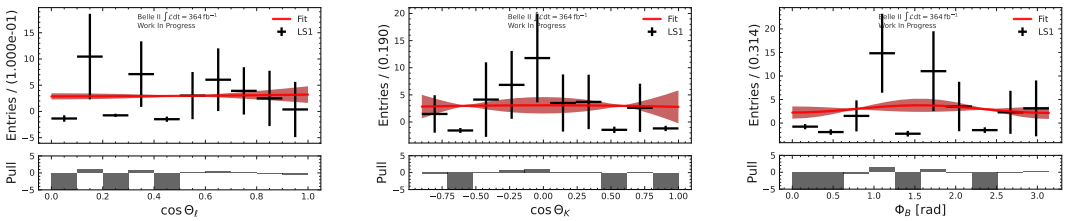
(a) (b) (c)

FIGURE I.61: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q_1^2-region with the LS1 dataset.



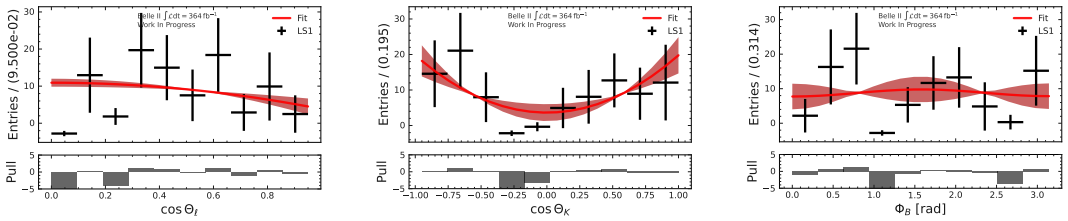
(a) (b) (c)

FIGURE I.62: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q_2^2-region with the LS1 dataset.



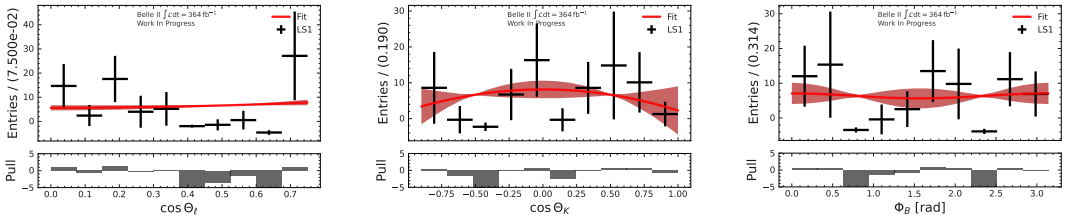
(a) (b) (c)

FIGURE I.63: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q_3^2-region with the LS1 dataset.



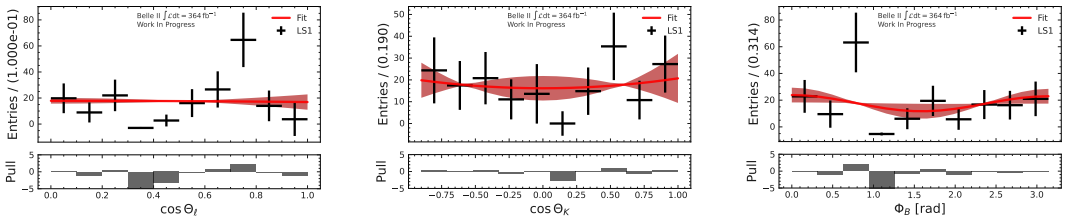
(a) (b) (c)

FIGURE I.64: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ channel in the q_4^2-region with the LS1 dataset.



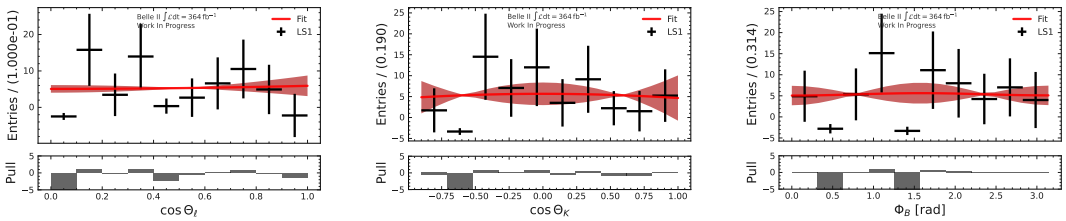
(a) (b) (c)

FIGURE I.65: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q_1^2-region with the LS1 dataset.



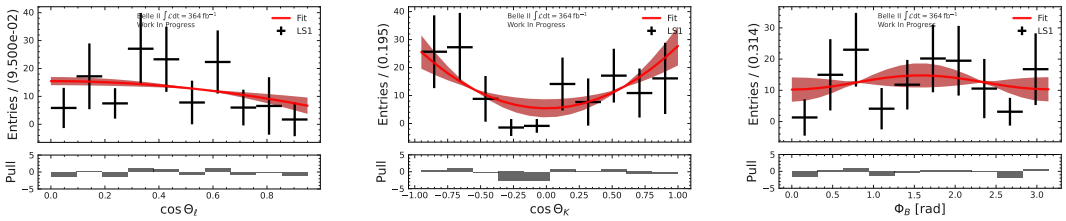
(a) (b) (c)

FIGURE I.66: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q_2^2-region with the LS1 dataset.



(a) (b) (c)

FIGURE I.67: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q_3^2-region with the LS1 dataset.



(a) (b) (c)

FIGURE I.68: Angular fit projections for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ channel in the q_4^2-region with the LS1 dataset.

Appendix J

Systematic Studies

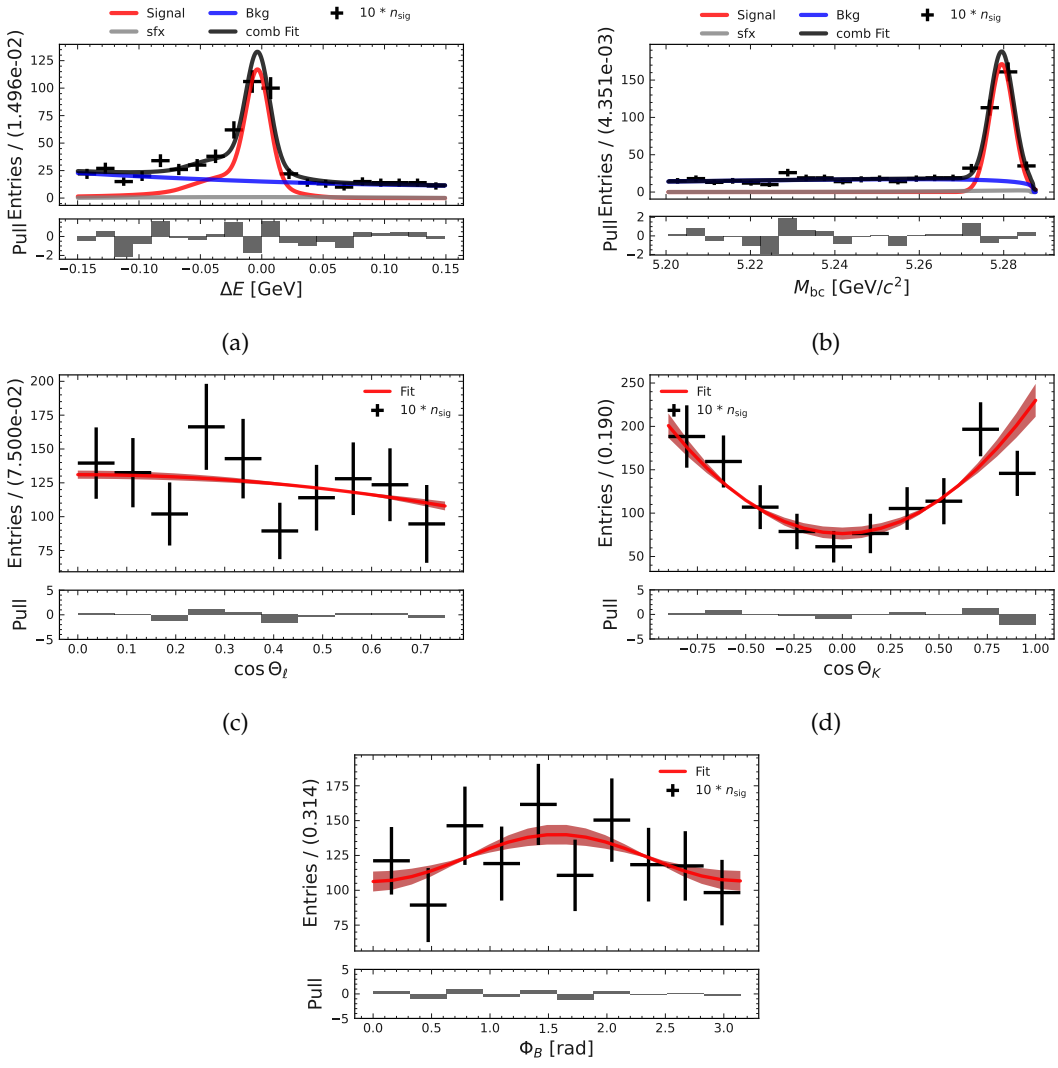


FIGURE J.1: Figures related to the fit study with $n_{\text{sig}} = 10 \cdot n_{\text{expected}}$ for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_1^2-region.

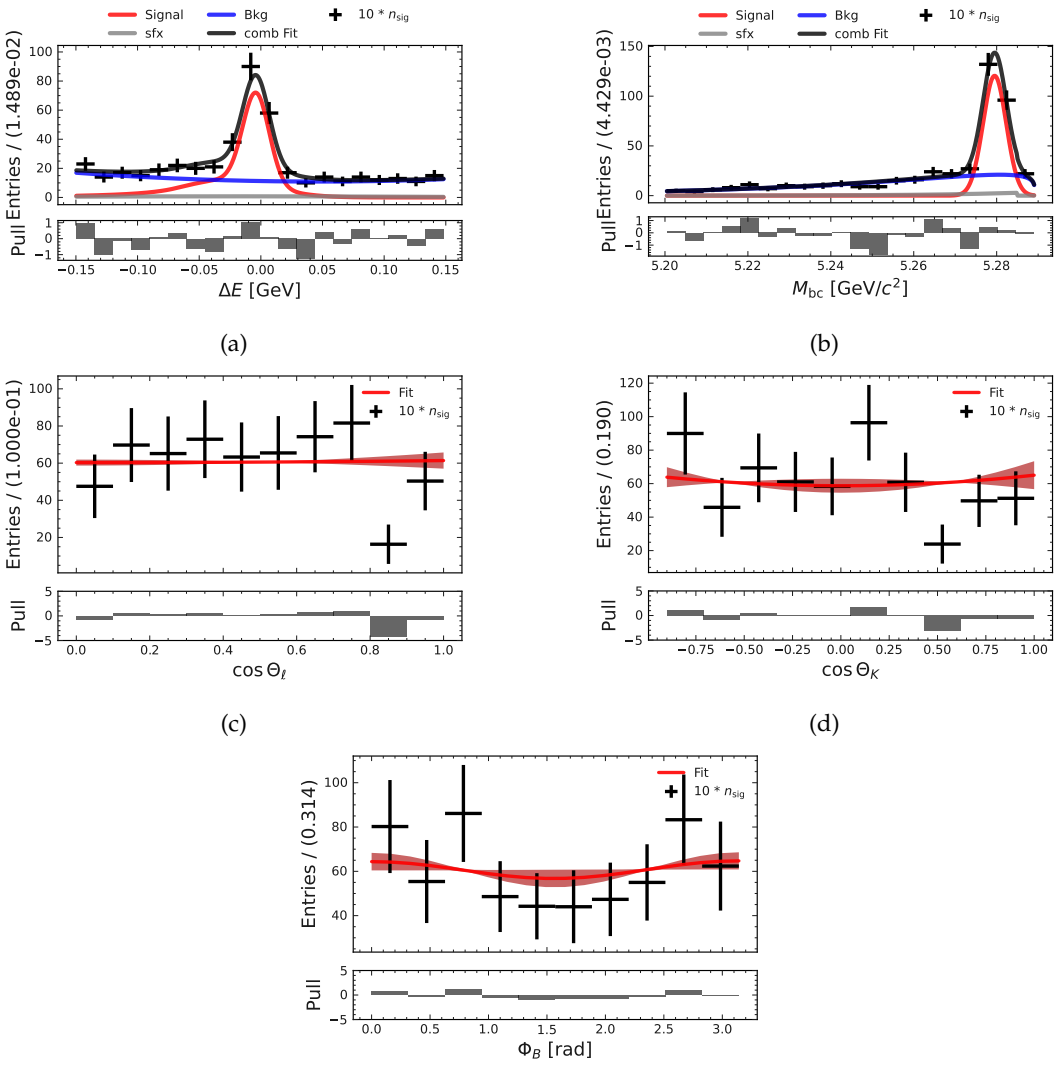


FIGURE J.2: Figures related to the fit study with $n_{\text{sig}} = 10 \cdot n_{\text{expected}}$ for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_3-region.

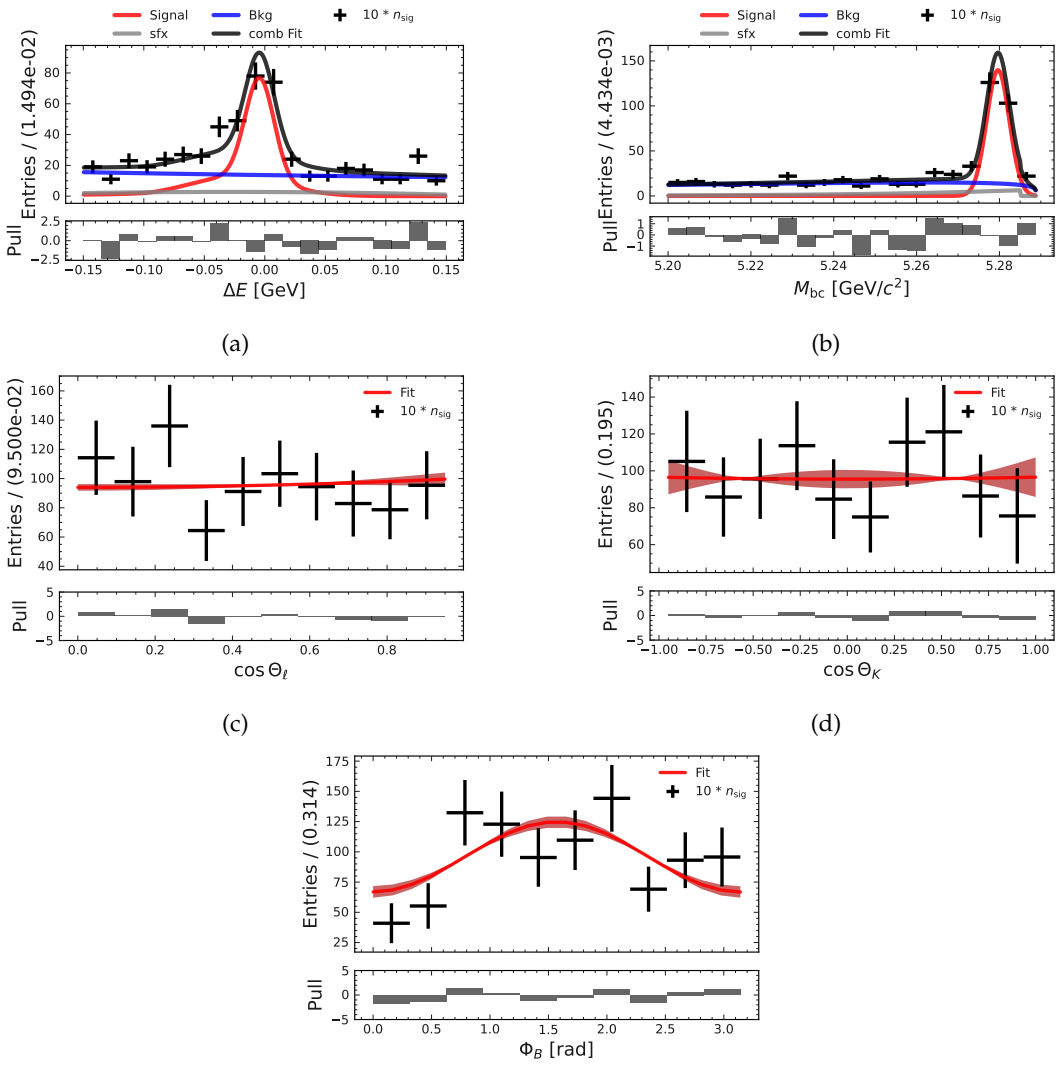


FIGURE J.3: Figures related to the fit study with $n_{\text{sig}} = 10 \cdot n_{\text{expected}}$ for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_4-region.

J.1 Fixing Background Shape

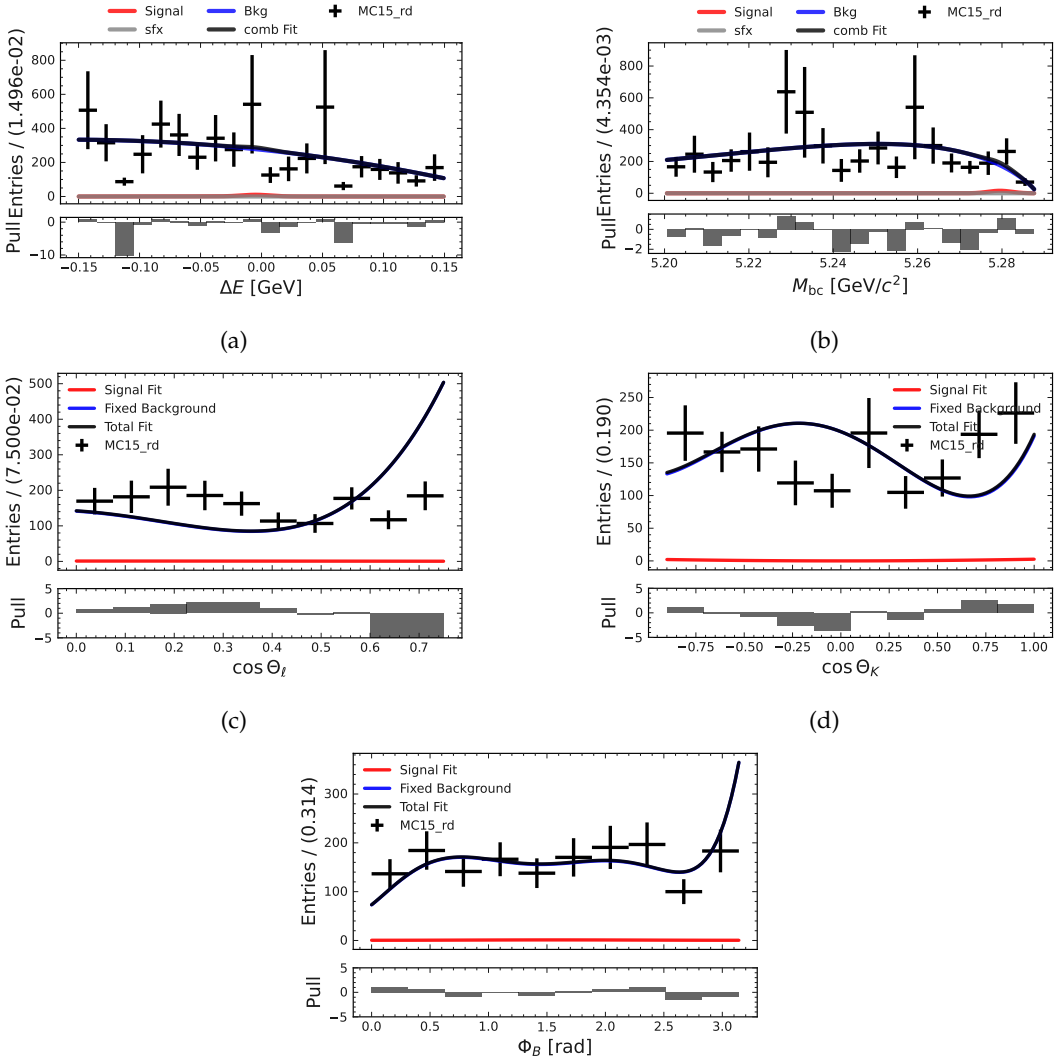


FIGURE J.4: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q_1^2-region.

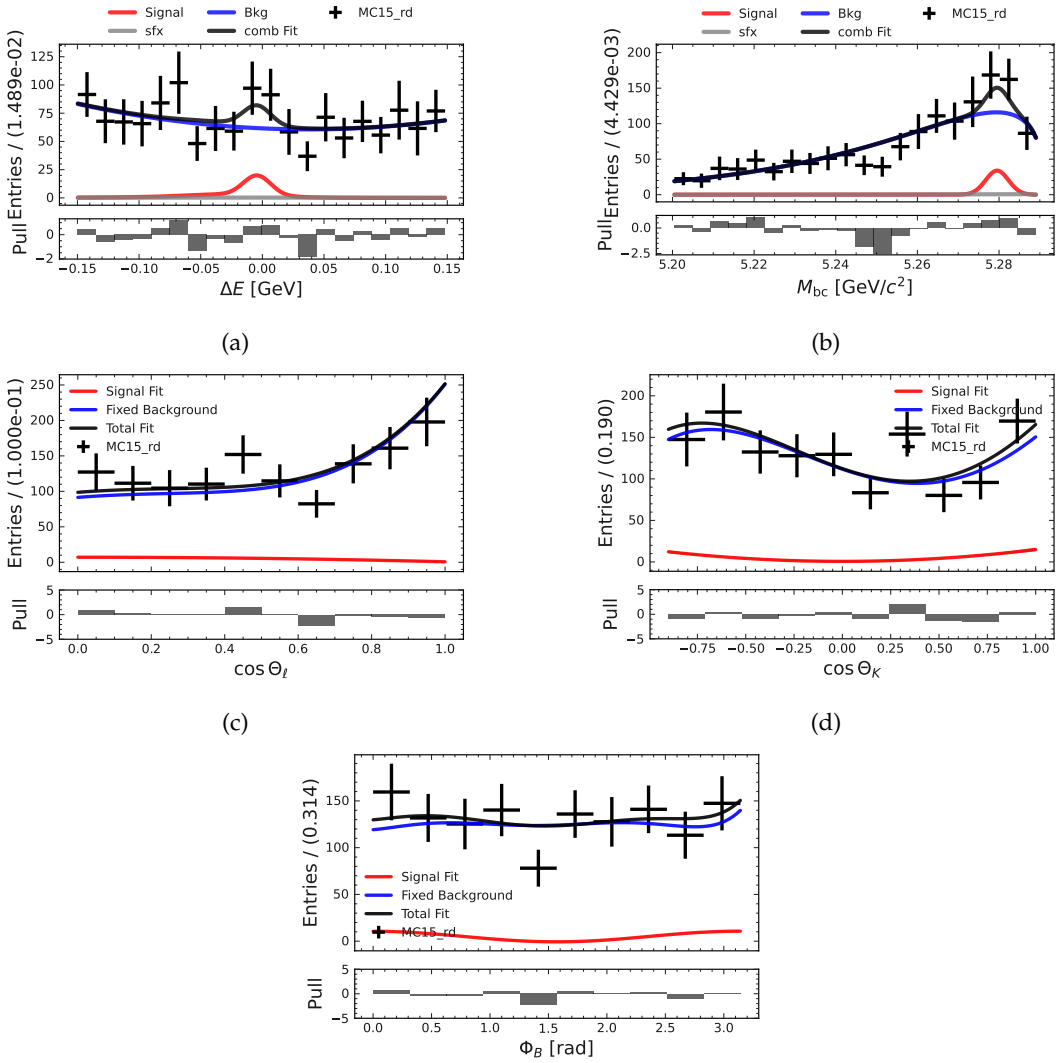


FIGURE J.5: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_3-region.

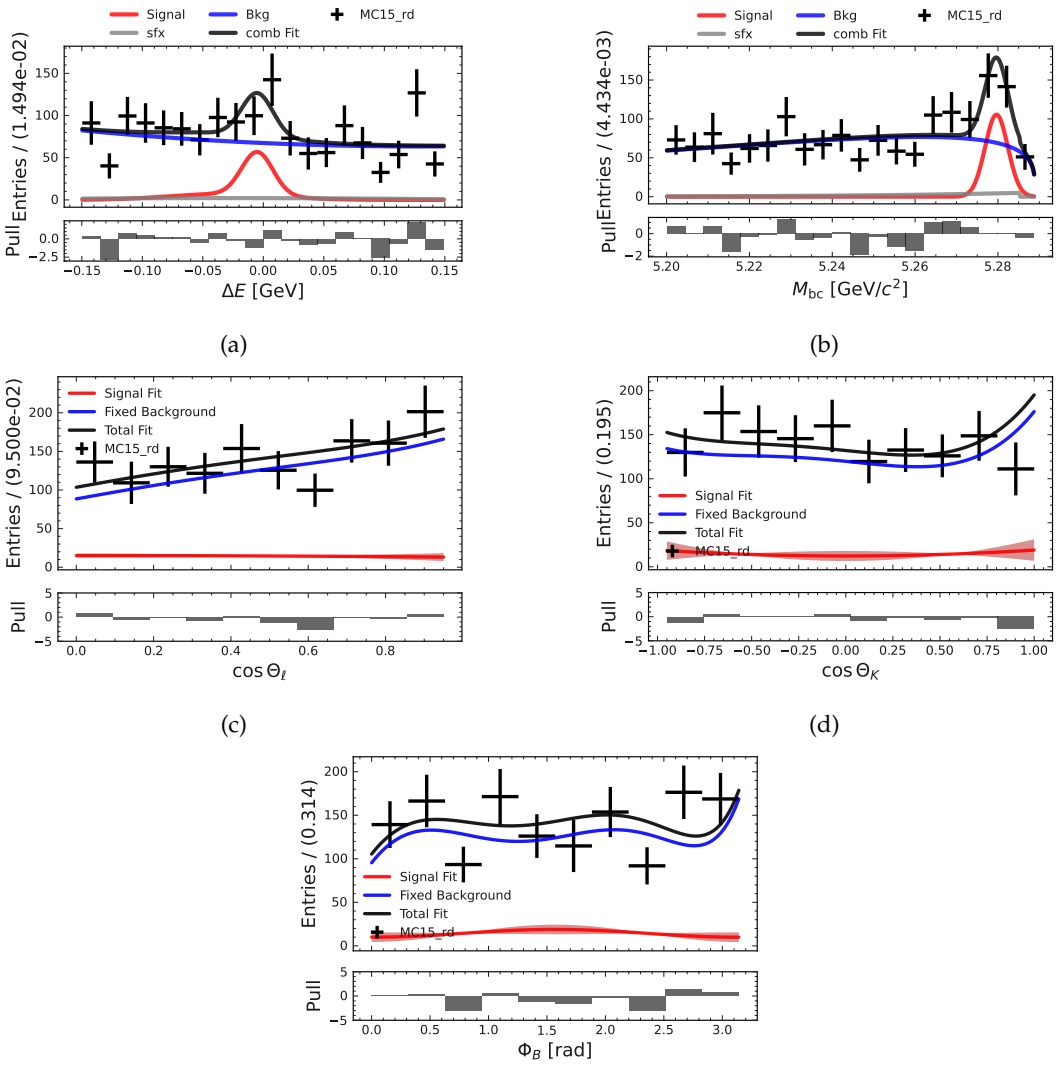


FIGURE J.6: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$ in the q^2_4-region.

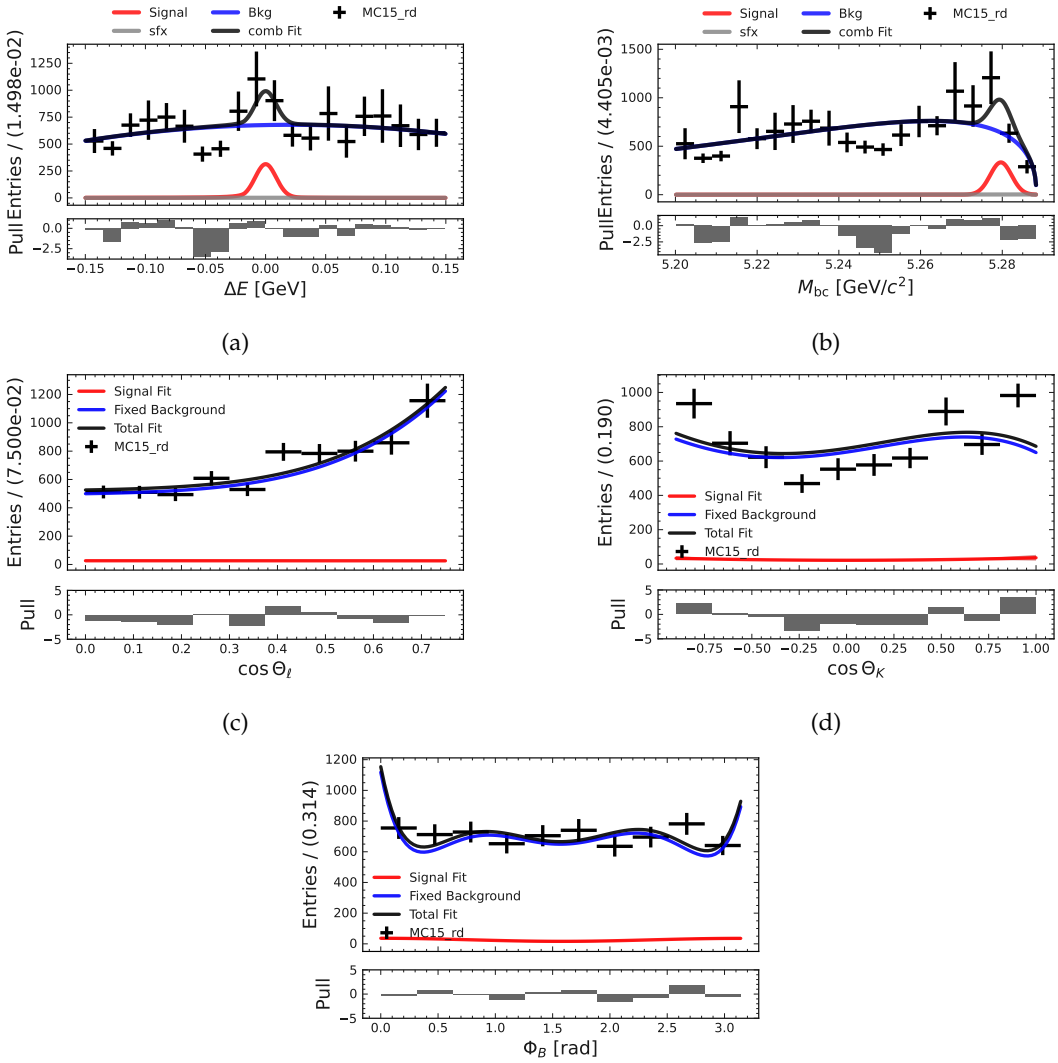


FIGURE J.7: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q_1^2-region.

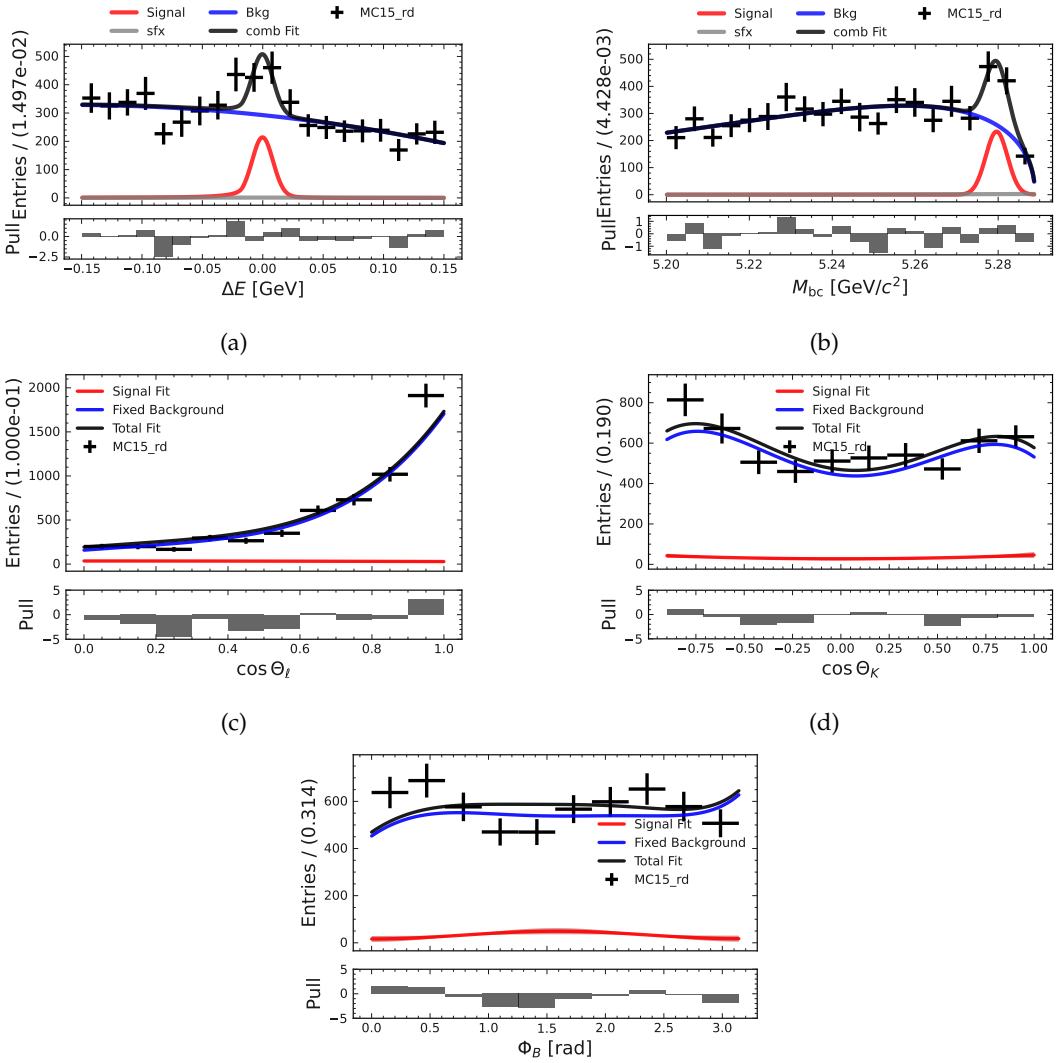


FIGURE J.8: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q^2_2-region.

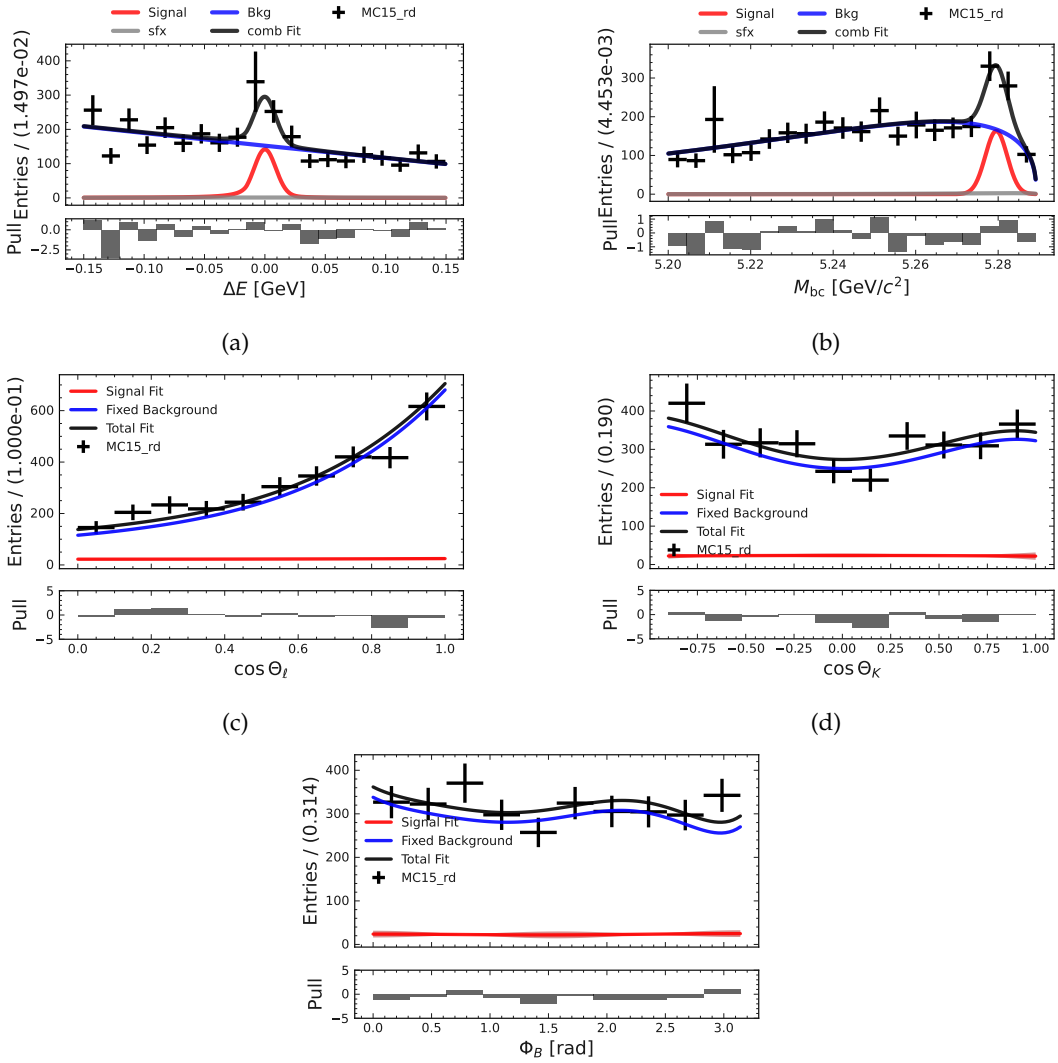


FIGURE J.9: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q^2_3-region.

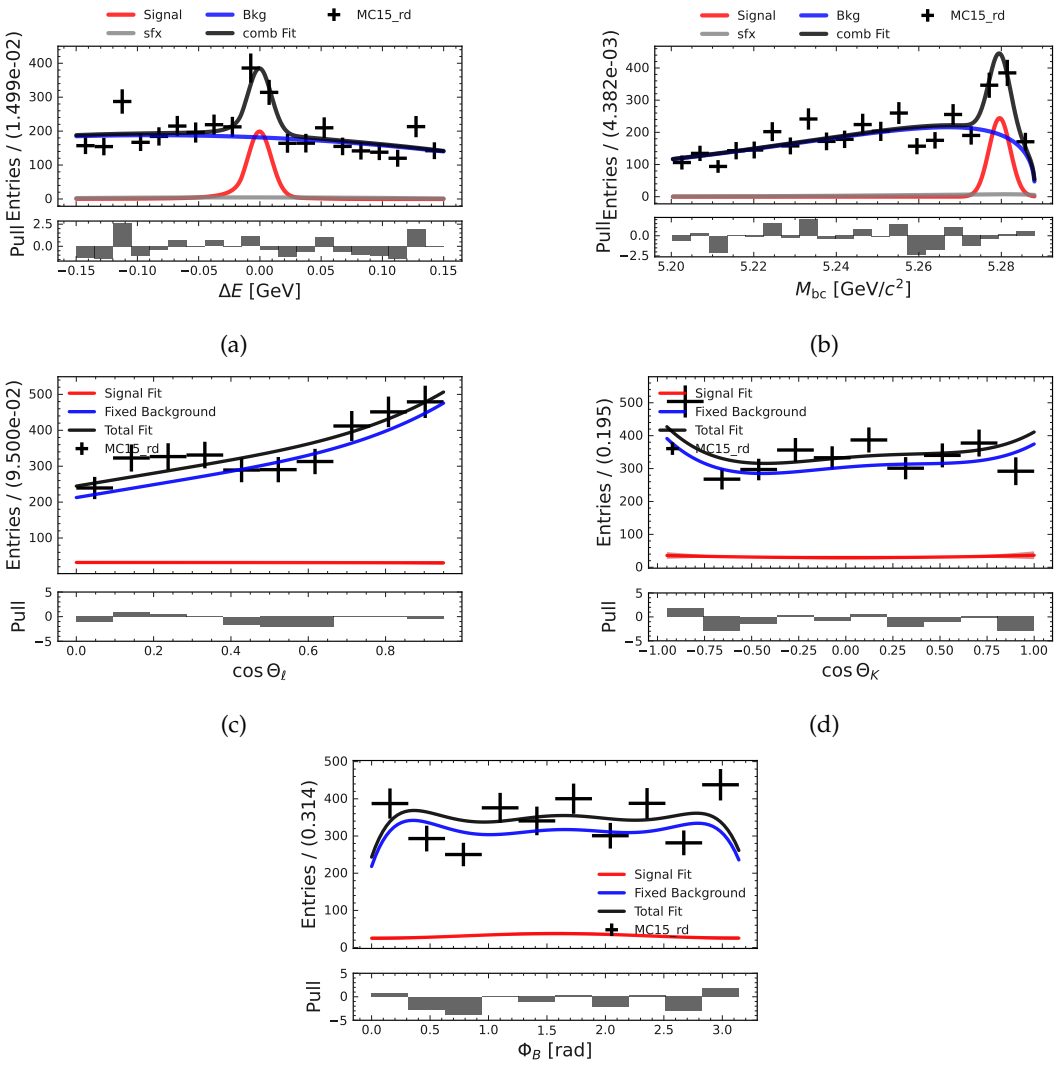


FIGURE J.10: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$ in the q^2_4-region.

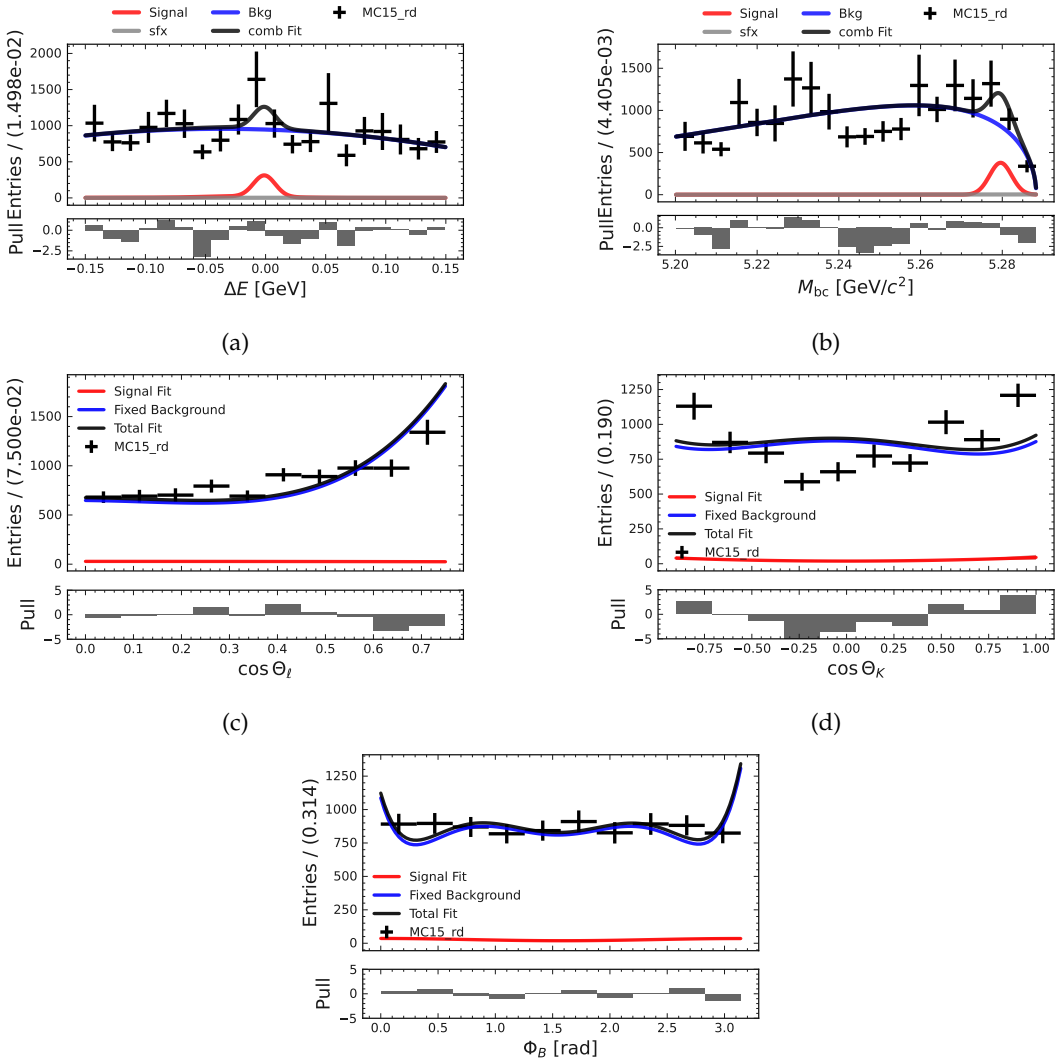


FIGURE J.11: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ in the q_1^2-region.

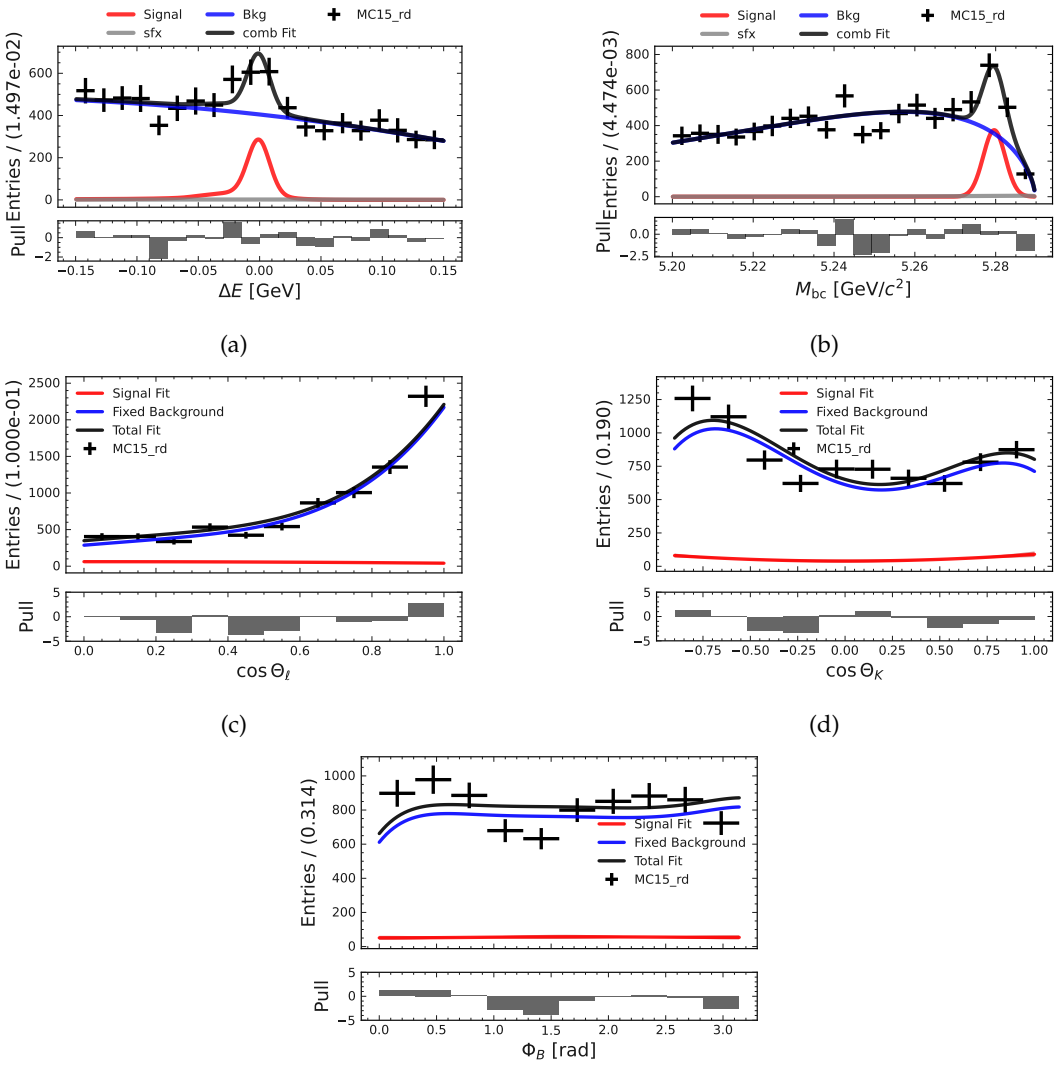


FIGURE J.12: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ in the q^2_2-region.

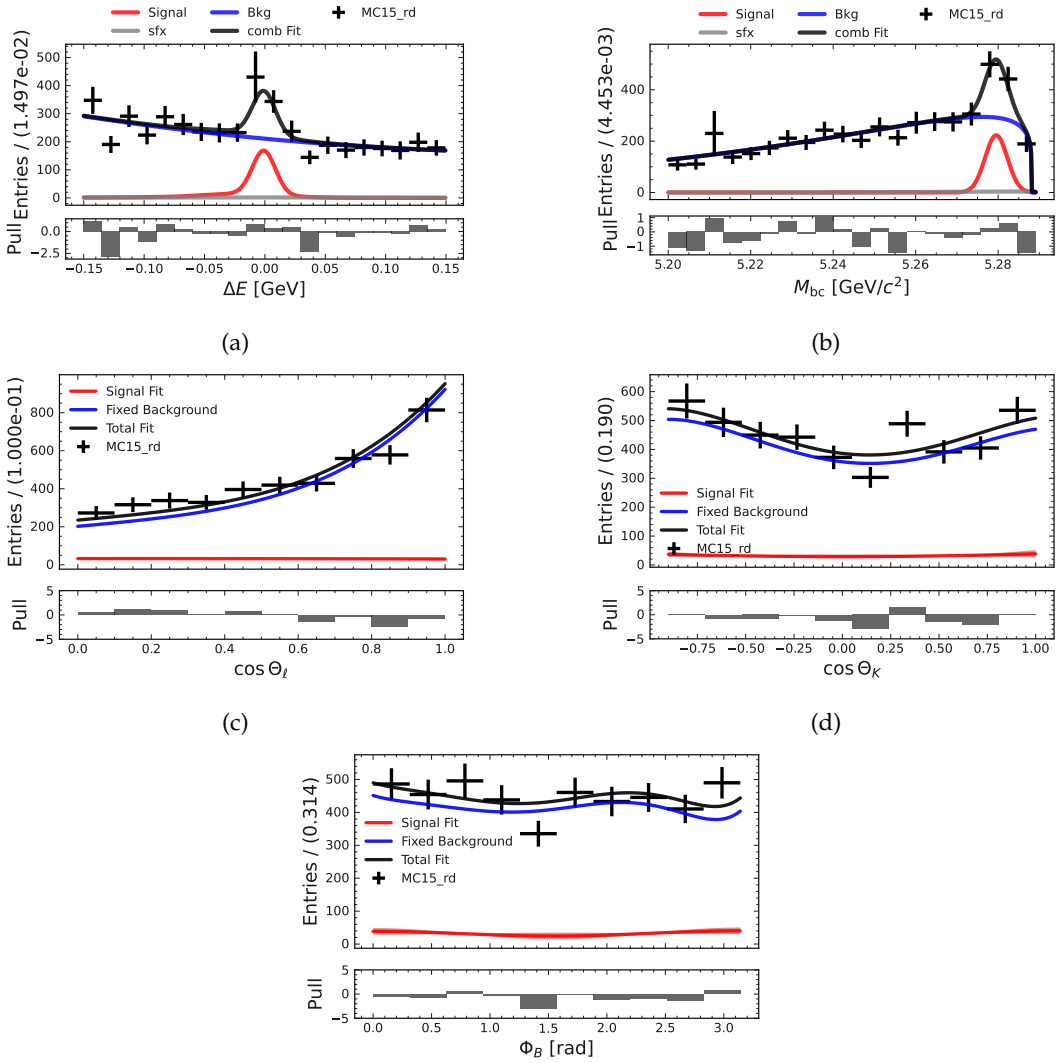


FIGURE J.13: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ in the q^2_3-region.

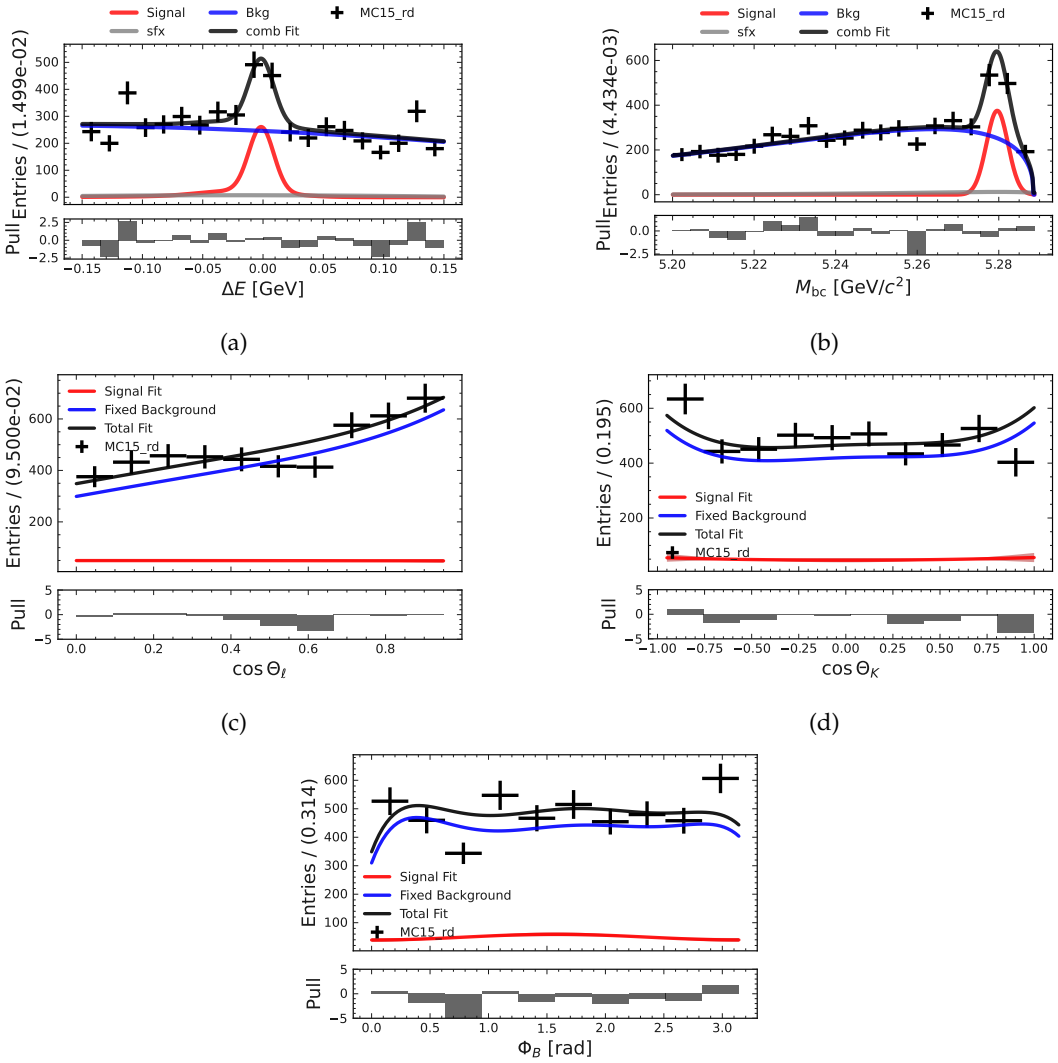


FIGURE J.14: Figures related to the fit study with fixed background shapes and no sWeights for the $B^0 \to K^{*0}(\to K^+\pi^-)\ell\ell$ in the q^2_4-region.

Appendix K

Outlook

K.1 Projection $\ell = e$

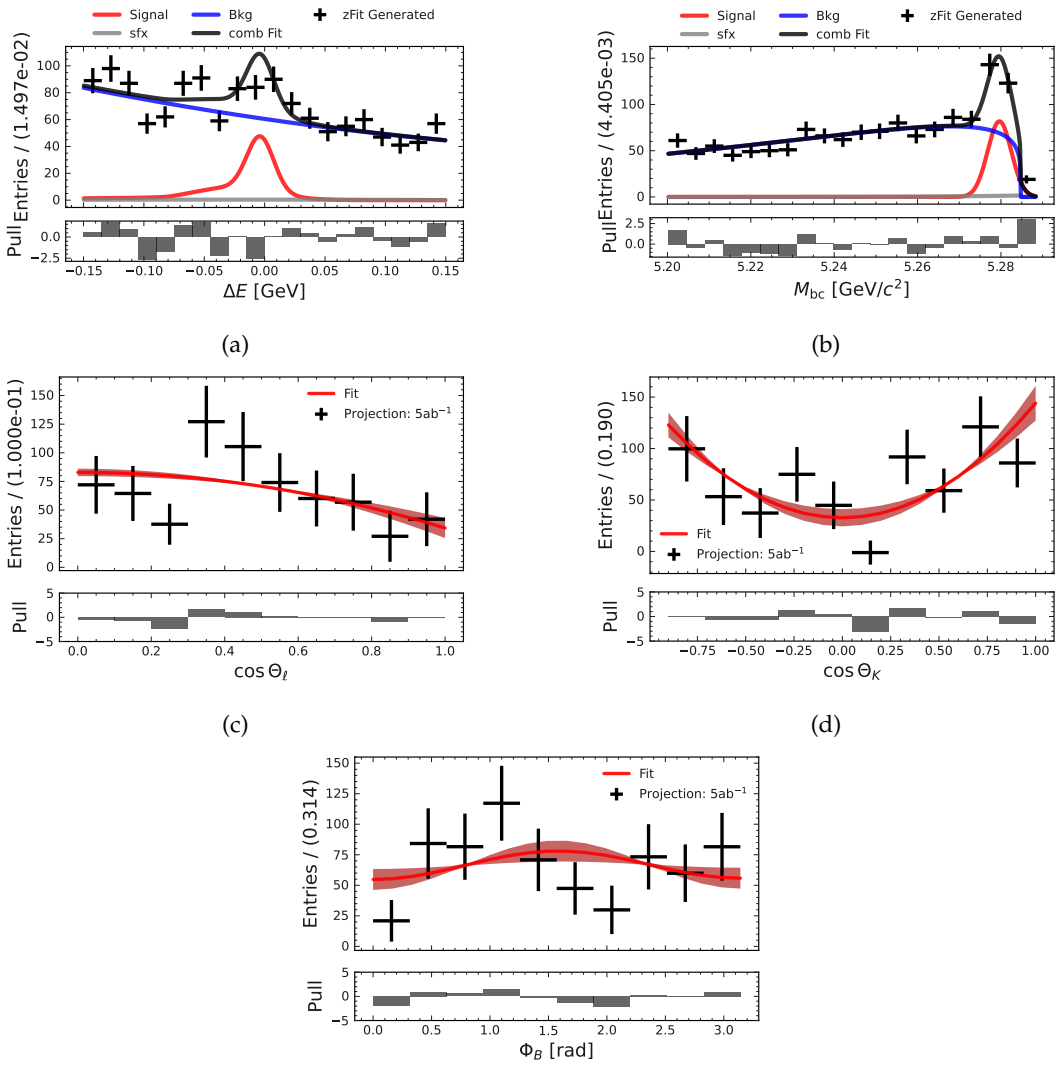


FIGURE K.1: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 5\,ab^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$.

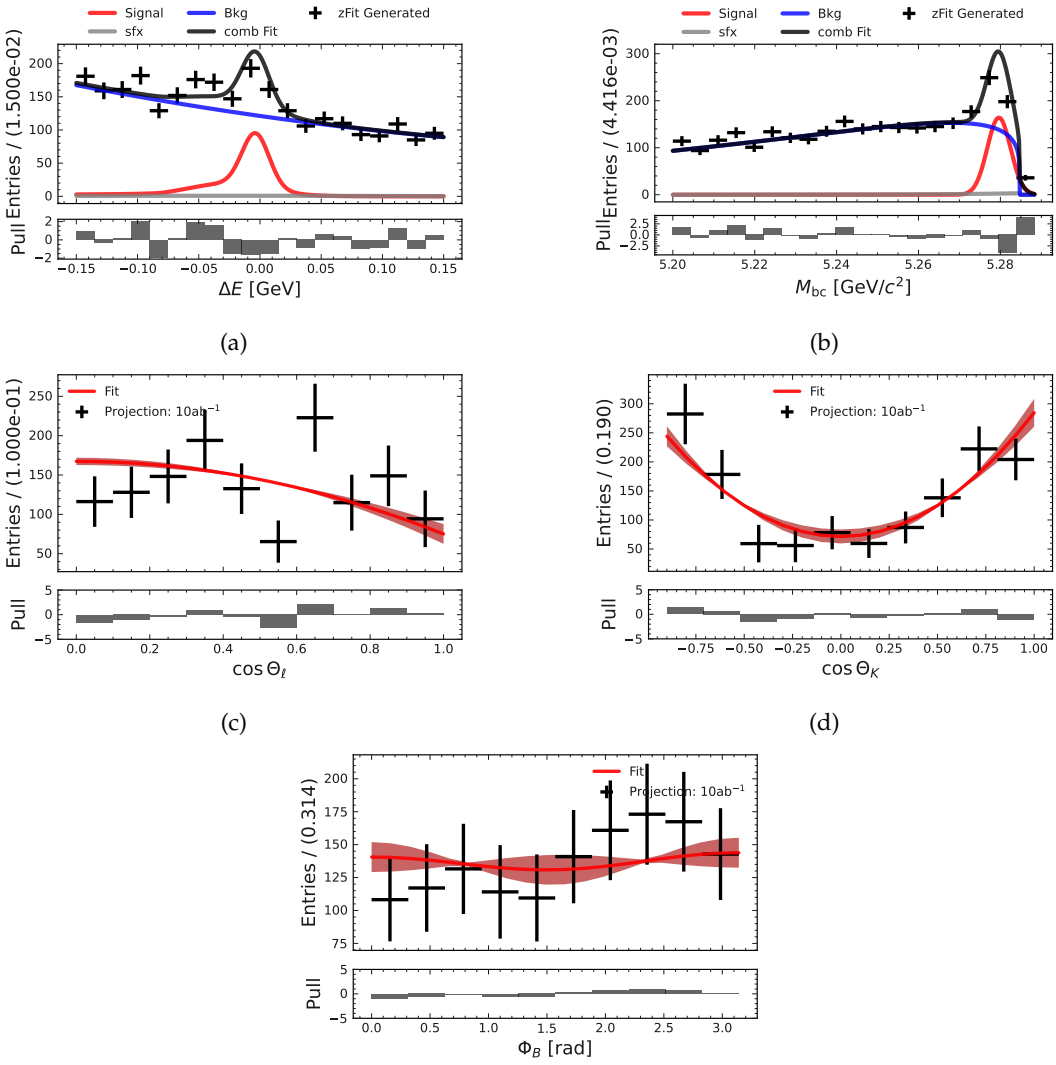


FIGURE K.2: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 10\,\text{ab}^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$.

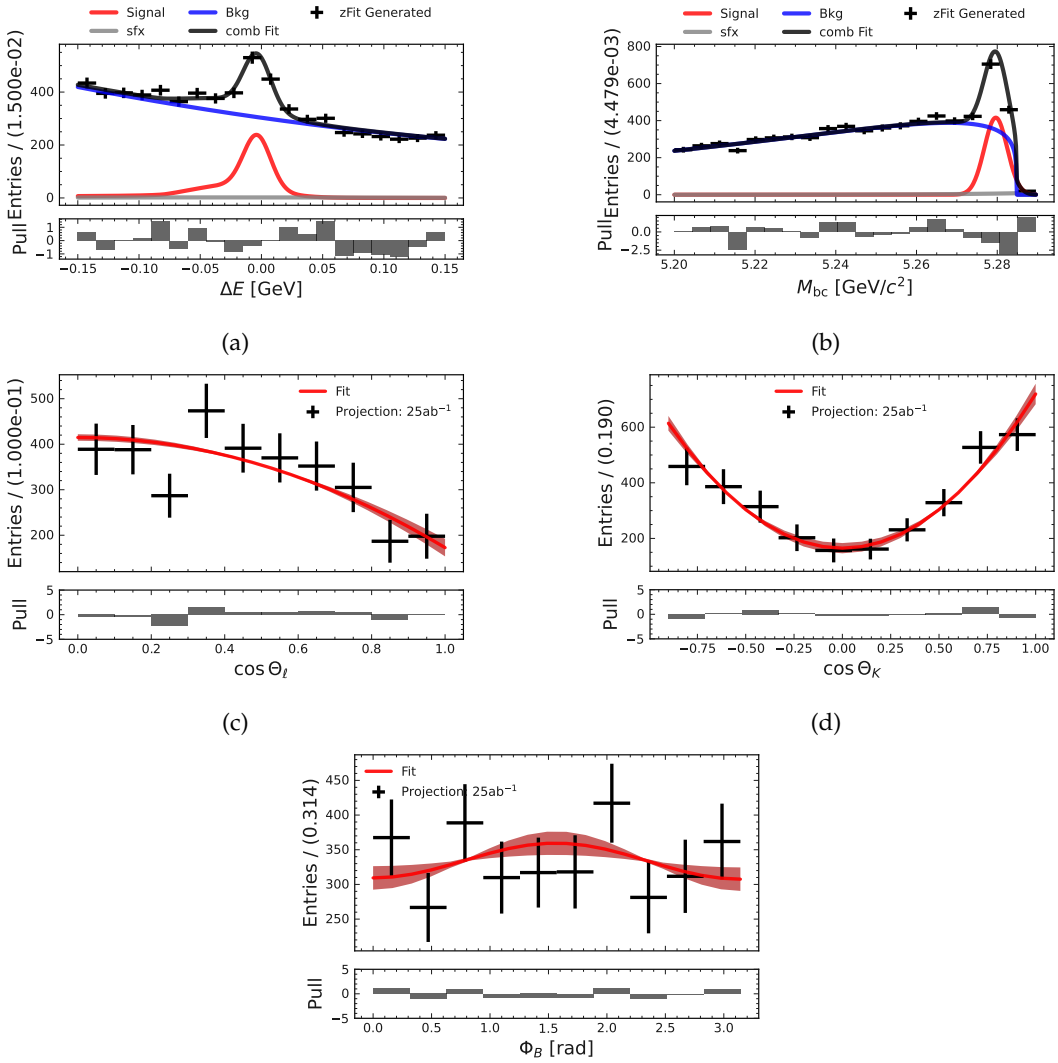


FIGURE K.3: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 25\,\text{ab}^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)ee$.

K.2 Projection $\ell = \mu$

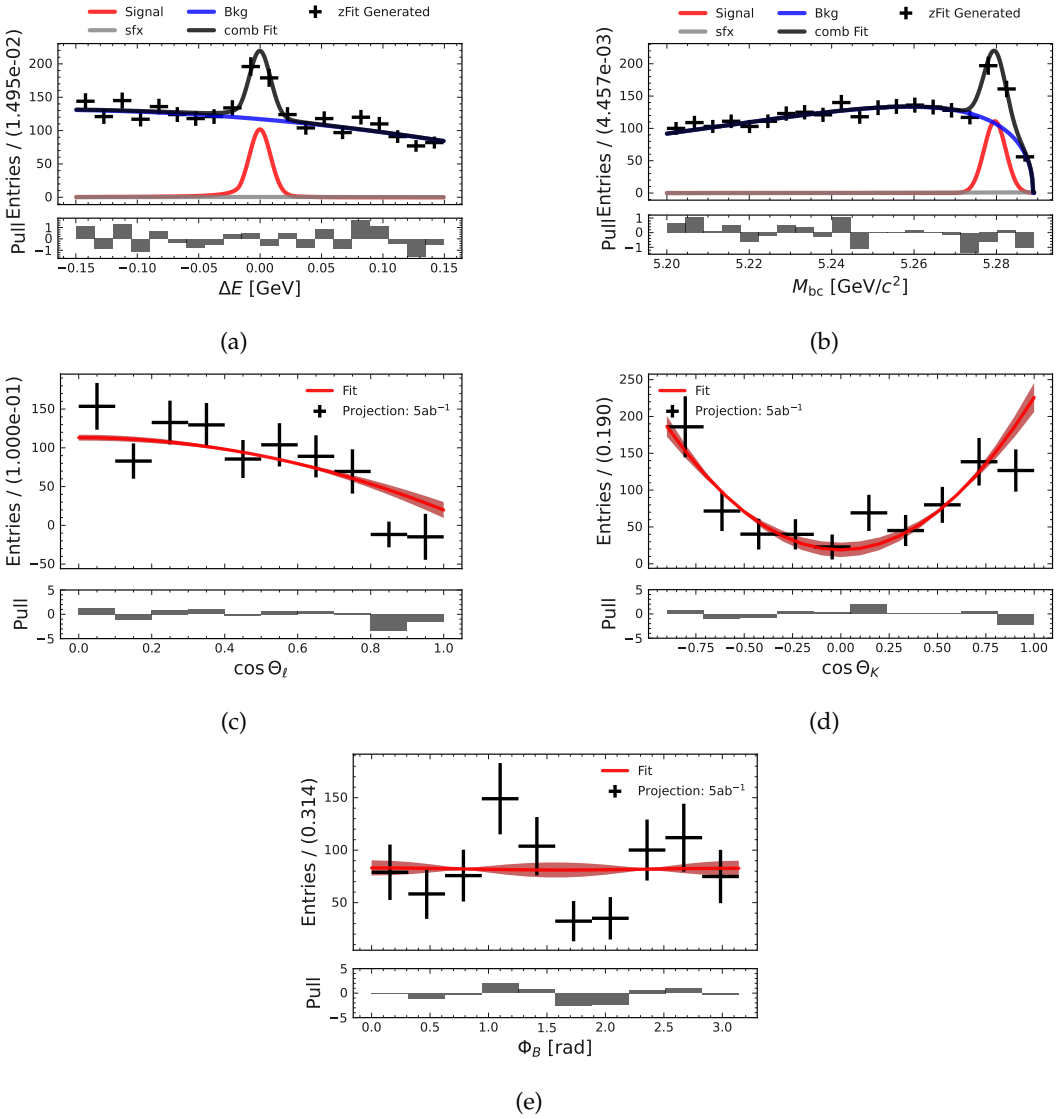


FIGURE K.4: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 5\,ab^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$.

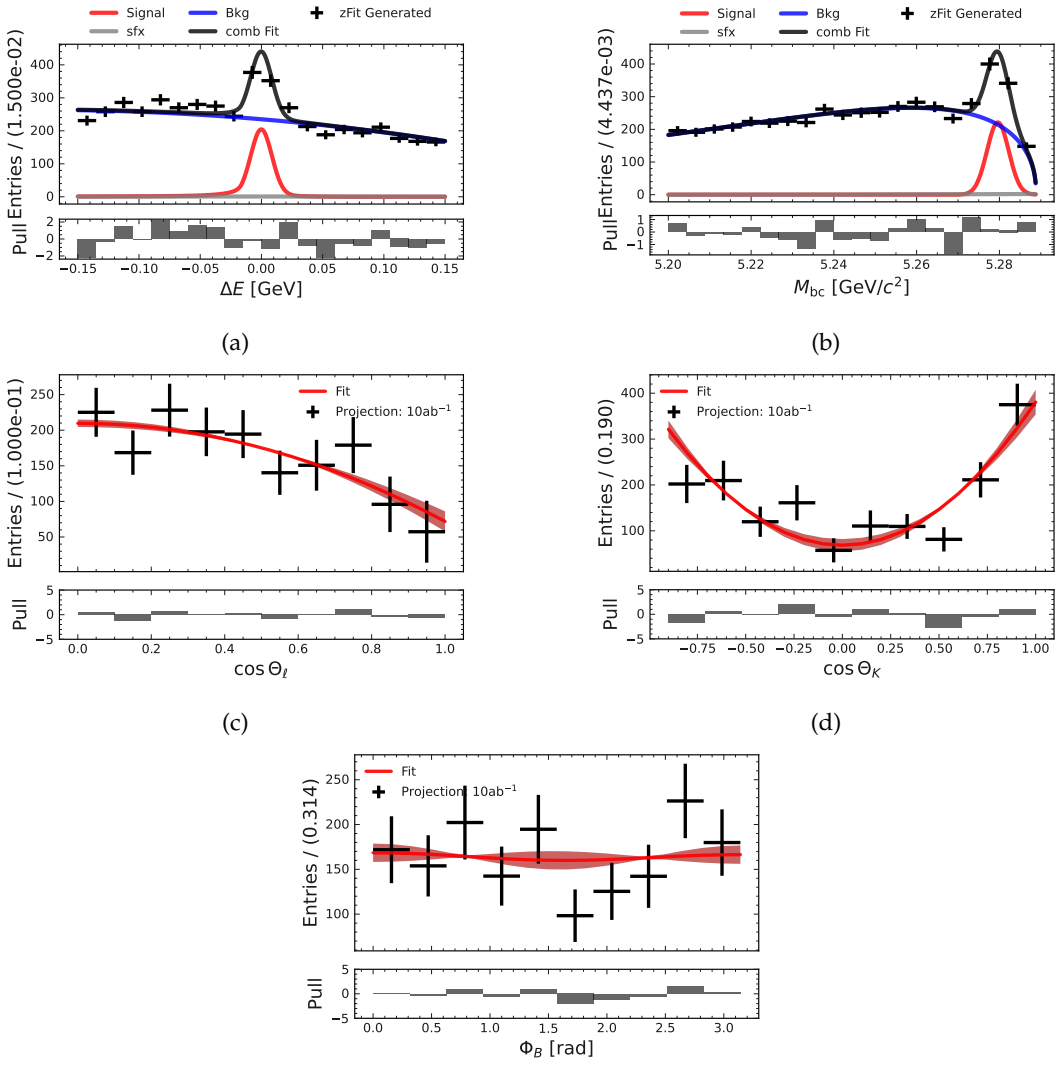


FIGURE K.5: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 10\,ab^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$.

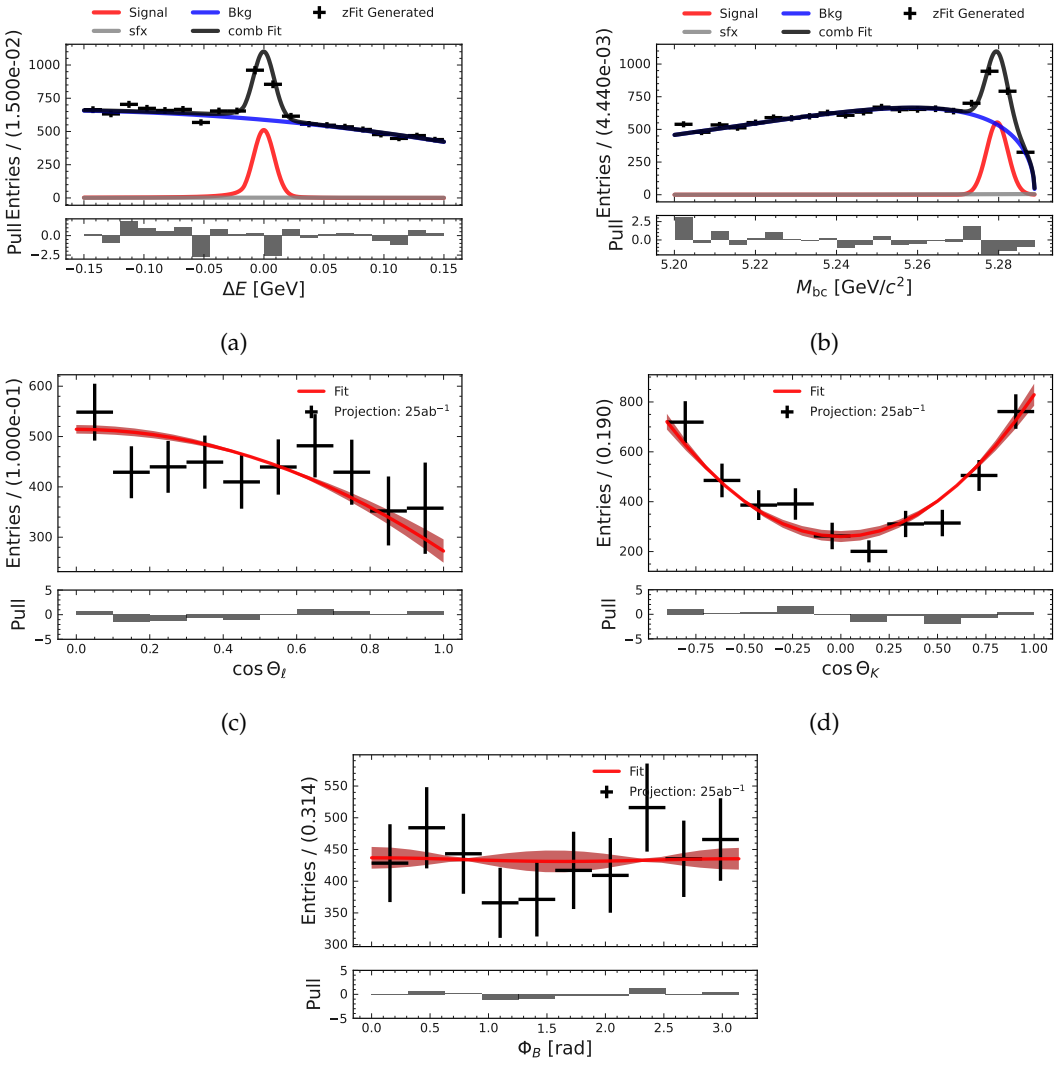


FIGURE K.6: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 25\,ab^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$.

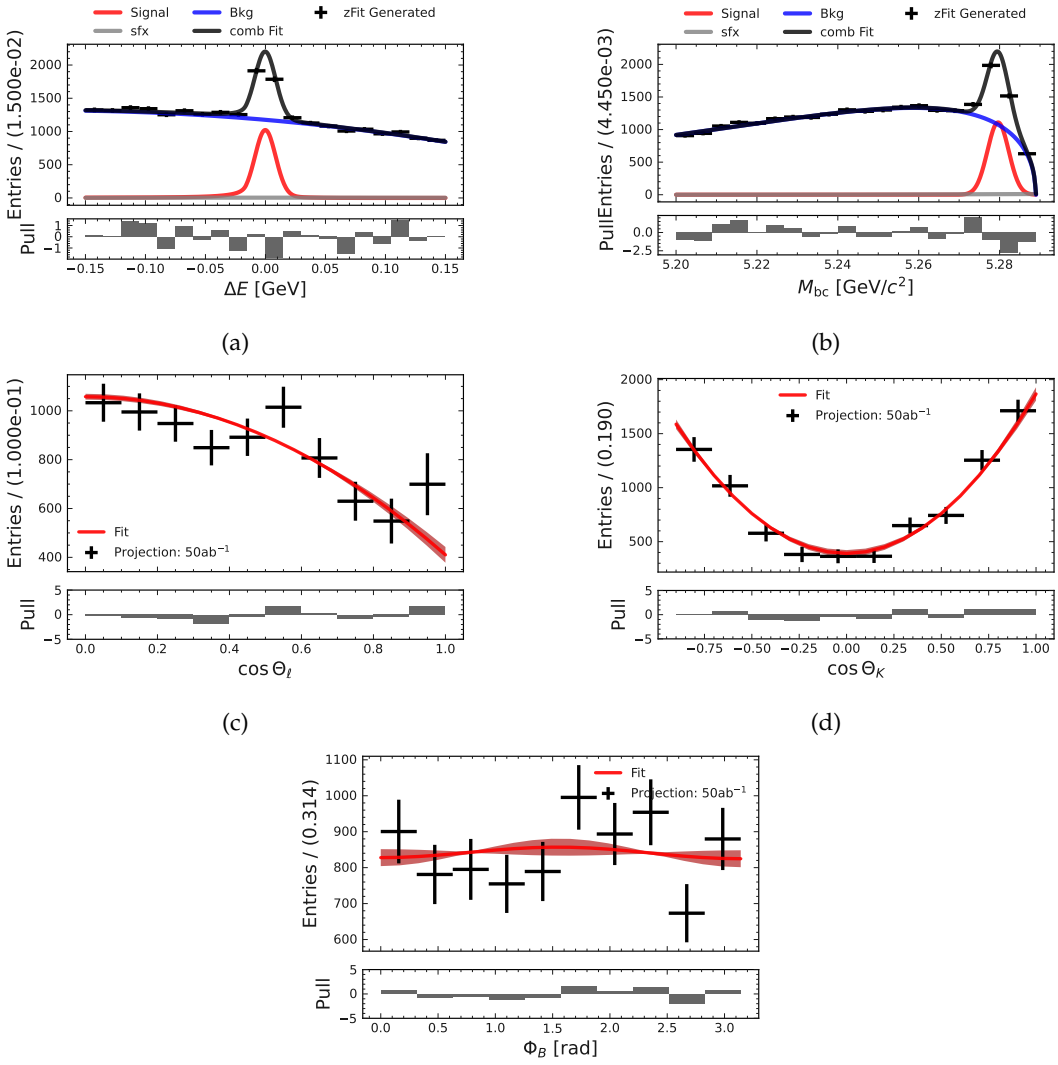


FIGURE K.7: Fits related to the projected integrated luminosity of $\int \mathcal{L}\,dt = 50\,ab^{-1}$ in the second q^2-region for the $B^0 \to K^{*0}(\to K^+\pi^-)\mu\mu$.

Part III

Additional Work for the PXD Working Group

Appendix L

Single Event Upsets Finder

> Still another time have I come to a place where it is very difficult to proceed. I ought to be hardened by this stage; but there are some experiences and intimations which scar too deeply to permit of healing and leave only such an added sensitiveness that memory reinspires all the original horror.
>
> H.P. Lovecraft, *At the Mountains of Madness* (1931)

In this chapter, some additional work for the Belle II collaboration will be presented. This work was done as a part of the qualification task. It will start with an introduction to single event upsets, continue with how to access the files to study them, because this is not trivial, and end on how to find them and how these scripts will be used in the future.

L.1 Single Event Upset

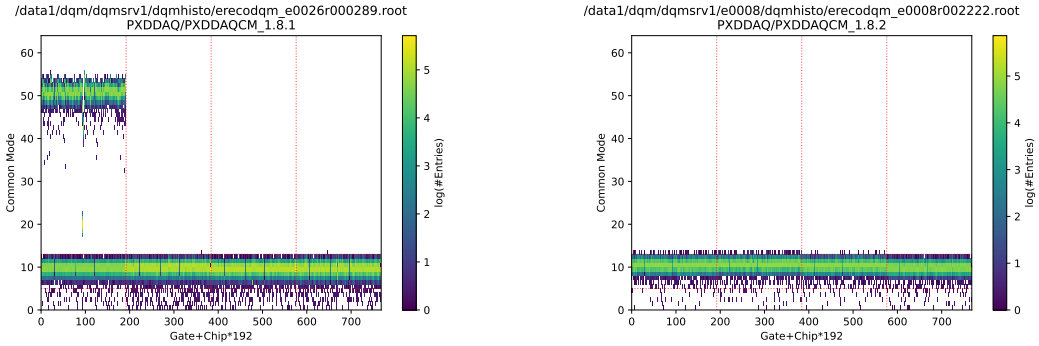
In areas with high ionizing particles, Single Event Upsets (SEUs) can occur. These SEUs are bit flips in a memory element of a semiconductor device caused by ionizing [101]. In general, no damage is done by the ionizing particle and the bit flip can be overwritten with the next writing cycle to that memory location. However, since the bit flip causes a change in the state of the device, it no longer will work as expected. In general, most SEUs are the result of cosmic particles hitting the device.

The best countermeasure is therefore the increase the shielding of the device to reduce the number of cosmic particles interacting with the device. Another method to prevent the errors caused by SEU is running multiple computers with the same scripts at the same time. This is done by NASA in their spacecrafts. Each script is run by four computers and their result is compared. If a computer has a different result compared to the others, the result of the later will be taken for further operation and the former computer will be brought back to the same state of the others [102].

Unfortunately, none of these methods can be applied to the PXD since shielding would prevent the detector from detecting and it is not possible to place a second PXD at the same position as the first to run cross checks. The chosen way to handle SEUs at the PXD is to detect their impact on the resulting data as soon as possible and then to recalibrate the PXD in order to bring it back to a working state.

L.2 SEUs at PXD

During phase 3 of the Belle II experimental data taking, a total of eight inner and two outer strips each containing two modules were installed. The impact of SEUs can be detected by observing the common mode of each detector strip of the PXD. This can be seen in Fig. L.1a. The SEU created a new band for higher common modes in the first chip. During data taking a bit flipped and from that on, all the data are filled in a common mode around 50 instead of the normal ten. This shift can happen in each chip and the offset can be any number. Knowing the resulting change induced by an SEU, a strategy can be implemented to find them. In contrast a common mode histogram without a SEU is presented in Fig. L.1b. The following sections will describe this strategy.



(a) (b)

FIGURE L.1: An example for a common mode histogram after a SEU is shown on in (a). The SEU happened in module 1.8.1. As a result, bins are filled in the top left of the histogram. A SEU can happen in each chip. A *normal* common mode histogram is shown in (b) for comparison.

L.3 Preparation

L.3.1 Input Data

Before this script can be applied to online data taking, stored files need to be analyzed. This is also a good cross check, since SEU should be found by the PXD operator observing the common mode histograms during the data taking (Only a few SEUs were found). This way, some SEUs are already known and the script must be able to find all of them.

The data for this analysis is stored on the DQM servers. All of the LS1 dataset of Belle II will be used for this offline analysis. Each of the 16 histograms within a file will be checked. Since there are about 50 000 files on the BDAQ server (*Belle II Data Acquisition Server*), the script will have to check about 800 000 histograms. Therefore, the performance of the script is an very important aspect to keep in mind. Later on, the scripts will be modified in a way that it can run during data taking in order to find SEUs as fast as possible and induce a recalibration of the PXD.

L.3.2 Accessing the Data

Unfortunately, accessing the data within the root file is not trivial. The problem is that the name of the leafs within the file contain the '/' character as shown in line 14 in the code snippet below. However, as a convention, this character indicates a branch of a file. The consequence is that it is not possible to read these files in a standard way. The uproot library will handle these names as a path with the part behind the last '/' as the leaf. The workaround solution is to get the keys of all the histograms stored in the file. These keys contain the name of the histogram among other things. If the key contains the desired name, the position of the key in the list can be stored in another list and the histogram can then be accessed by calling the position of the desired key in the list of keys.[1] After this, the histograms can be easily written out in arrays for further analysis. The following code snippet shows a working example of the workaround.

```
1  import uproot
2
3  def getListOfRelevantKeys(uprootFile, histList):
4          keys = uprootFile._keys
5          keyList = []
6          for hist in histList:
7                  for j in range(len(keys)):
8                          if str(keys[j]).__contains__(hist):
9                                  keyList.append(j)
10         return keyList
11
12 fH = uproot.open("file.root")
13 keys = fH._keys
14 histList = ['PXDDAQ/PXDDAQCM_1.1.1', 'PXDDAQ/PXDDAQCM_1.1.2']
15 keyList = getListOfRelevantKeys(fH, histList)
16 for key in keyList:
17         hists = keys[key].get()
18         num1, _, _ = hists.to_numpy()
```

L.4 SEU Regions

In order to improve the calculations to detect SEUs, the common mode histograms are divided into a total of 12 subregions. They are indicated by the red letters and numbers in Fig. L.2. It is clear, that, in this case, the SEU happened in region $A3$ but, as already mentioned, a SEU can appear in all of the 1 and 3 regions. For an histogram with no SEU, almost all entries are expected be in the 2 region. Later on the combination of the subregions $A1$, $A2$ and $A3$ will be referred to letter A subregions. The same is true for the letter subregions B, C and D.

L.5 Requirements

The strategy to select SEUs will be to use multiple sequential cuts in such a way that only common mode histograms with a SEU survive. Due to the performance requirement of the script, the order in which the cuts are applied is very important.

[1] This is not only a problem for uproot. Accessing these histogram with C++ produces the same challenge, leaving the question why this naming scheme was chosen.

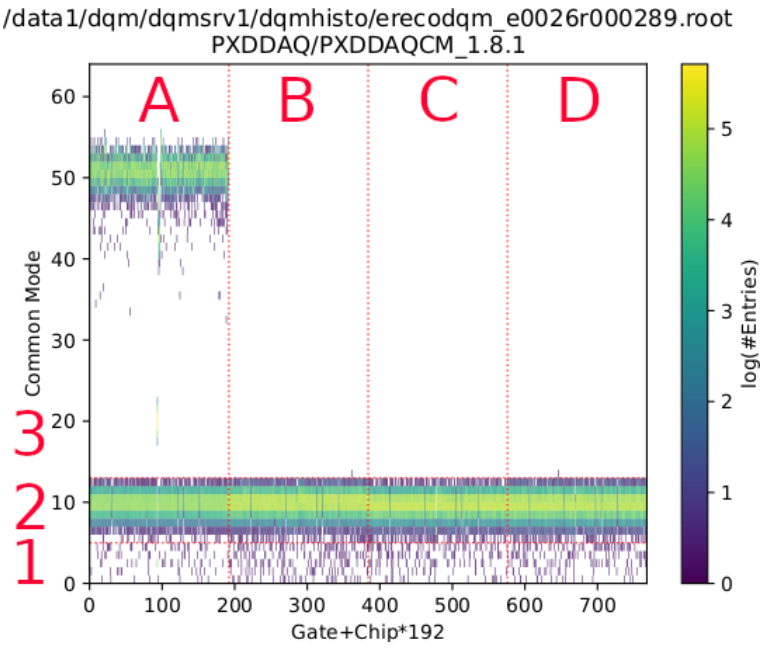


FIGURE L.2: The regions for further analysis are shown. In a normal common mode histogram almost all entries would be in subregion 2. Here, bins in subregion *A*3 are filled as a result of a SEU.

L.5.1 First Selections

The first few conditions will improve the quality of the remaining histograms. First, the top two row of the histogram, common mode 63 and 64, will be ignored for further analysis. Unfortunately, sometimes a lot of entries are dumped into these lines, messing with the following requirements. An example for this can be found in Fig. L.3.

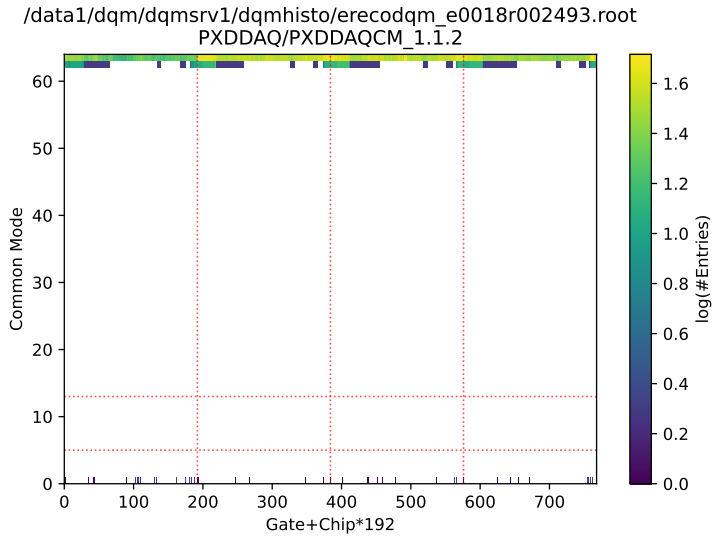


FIGURE L.3: Example for a histogram where the entries are dumped into to highest common modes.

In the next step, the histograms are manipulated a little bit. It can happen that quite a lot of events are dumped into just a few bins. This can be seen in Fig. L.1 at the $\sim$ 100th gate in the first chip and the common mode of around 20. These few bins have the highest number of entries in the whole histogram. For the further analysis the number of entries of the three bins with the highest number will be reduced to one. This will not exclude the chance to find SEUs but it will drastically improve the quality of the following selection.

In order to ensure a high enough statistic, the total number of entries must be higher than 100 000. This is a rather loose cut since most of the time the histograms contain hundreds of million entries and during normal data taking this condition is fulfilled within seconds. This selection gets rid of almost empty histograms.

After this, the most powerful requirement will be placed. The condition is that at least 0.05% of all entries are somewhere in subregion 1 or 3. This requirement removes by far the most histograms since most of them are almost exclusively filled in subregion 2.

L.5.2 First Selections on SEUs

Until now almost no information about SEUs are used. This changes with the following conditions. At least one subregions must have more than 1% of all the events compared to its corresponding region 2. For example, $(n(A1) + n(A3)) \cdot 100 > n(A2)$. This is checked for all letter subregions A, B, C and D.

Since SEUs produce an offset of the common mode band, it will produce a lot of non-zero entries in a subregion. Each chip has 192 gates and the band has a certain *width* of multiple rows. Now, the condition is that there have to be at least 350 non-zero bin entries for subregion 1 and 450 for subregion 3. This corresponds to a little less then two and three full rows in a chip, respectively.

Sometimes almost all events in subregion 1 are dumped into the first row of the common mode. This can be seen in Fig. L.4. Since SEUs can also happen in this small subregion, it would be fatal to just delete a row. Instead it is checked if the last line contains more than 10 000 times more entries than the rest of the subregion. If this is the case, the search for SEUs will be stopped for this histogram.

L.5.3 Multiple Filled Subregions

As already mentioned SEUs are extremely rare and they happened only a couple of times during data taking. Therefore, we can assume that only one SEU will happen in one module during a run. The following condition will assure that if there are too much entries in multiple subregions of 1 and 3, the histograms are thrown away. This can happen when the calibration of the PXD was bad.

The next condition is that if at least two of the letter subregions have more than 1% of their corresponding main subregion, then the histograms are dumped. As an example: Lets have $n(A_{1,3}) = n(A1) + n(A3)$. The same is true for $n(B_{1,3})$, $n(C_{1,3})$ and $n(D_{1,3})$. Now, if $n(A_{1,3}) \cdot 100 > n(A2)$ and $n(B_{1,3}) \cdot 100 > n(B2)$, then the requirement is not fulfilled and the histogram is thrown away. This will be done for every combination of letter subregions.

When a histogram survives this condition the letter subregion with a likely SEU candidate is stored. In the example shown in Fig. L.1a the information that the SEU happened in subregion A would be stored in a variable for the next step.

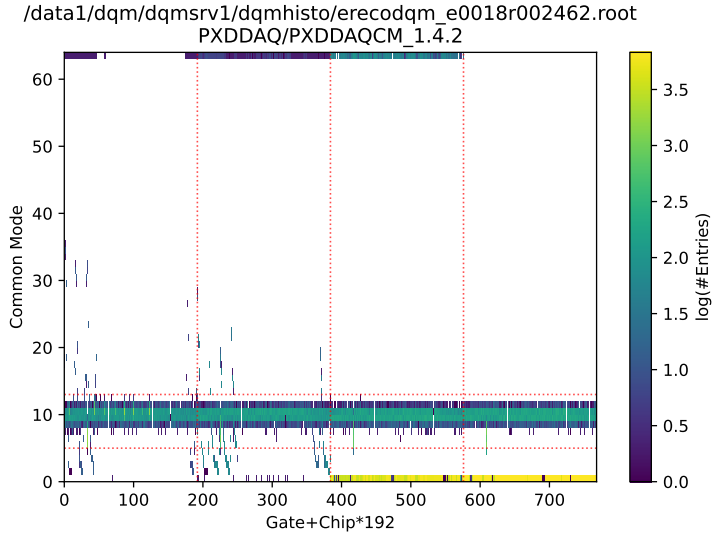


FIGURE L.4: Example for a histogram with events dumped into the bottom line.

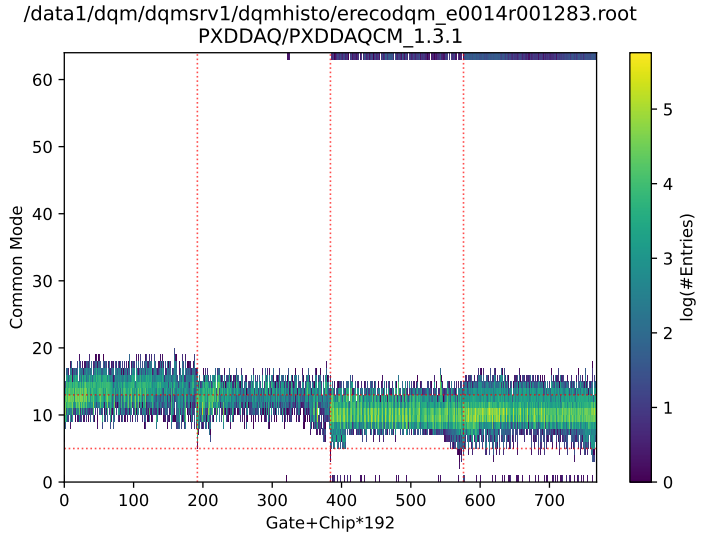


FIGURE L.5: Example for a common mode histogram with a bad calibration.

L.5.4 Fitting

Unfortunately, all the introduced requirements are not enough since there are some histograms without an SEU still passing these criteria. It was found, that most of them have *weird* curve structures in some subregion of 2. This can be seen in Fig. L.6a and L.6c. This curve distorts the band enough that it starts to spread into the subregions $A1$ and $A3$. In order to find these kind of structures, first, the average common mode value of each gate in this subregion is calculated. After this, a spline fit is performed in the 2 region with the same letter as the SEU. This is depicted in Fig. L.6b and L.6d for histograms without an SEU and in Fig. L.7 for a histogram with an SEU in the A region. The overall average value of the common mode is the first value in

the title. The second value is the value range of the spline fit. It is clear that the range of the spline fit values is far wider for the histogram with a curve compared to an SEU histograms. The final condition is that the value range of the spline fit must be smaller than 1.25. The highest range of values for a SEU was determined to be 0.7. The chosen requirement should therefore be wide enough to keep all future SEUs.

A SEU can result in a 2 subregion which is almost or completely empty. Performing a fit in such a subregion would produce nonsense. In order to prevent this, the fit is only performed when the triggered 2 region has at least 10^6 entries and more then $192 \cdot 4$ non-zero bins.

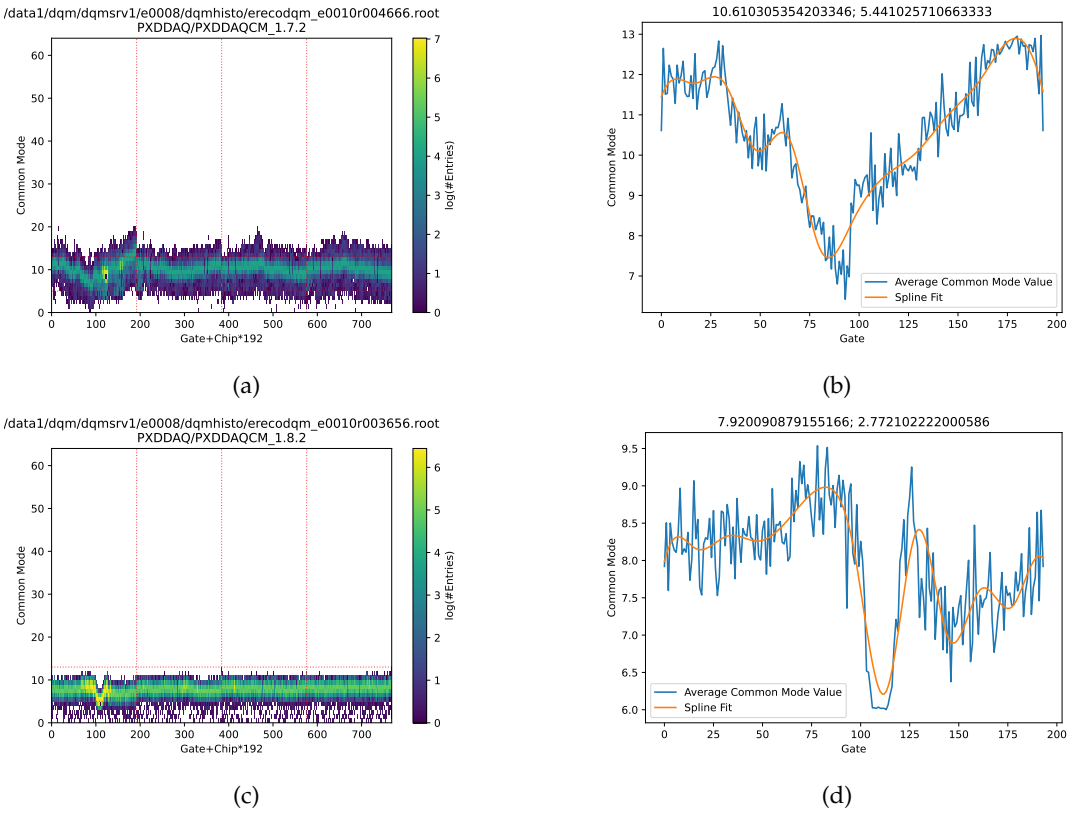


FIGURE L.6: Weird structure in subregion $A2$ can be seen in (a) and (c). The corresponding spline fit of these regions is in (b) and (d).

L.6 Surviving the Selection

All of these conditions will be applied to the provided histograms. From initially $\sim$ 800 000 histograms, only 17 pass all the requirements. Their experiment number, run number and affected module can be found in Tab. L.1. A collection of the found SEUs is depicted in Fig. L.8.

By closer analysis of the histograms it appears that there are some consecutive runs which are impacted by the same SEU. Most notably is the SEU that first happened in experiment 20, run 638 in module 1.1.2. This SEU happened during the run since there are some entries in the $D2$ region as well but all following runs are empty in this subregion and the subregion of the new entries filled after the SEU stays the same for the next four runs. Another streak occurred in experiment 26, run 821 module 1.1.1. The next found histogram is two runs down meaning that run 823 is not listed. This was checked by trying the open run 822 manually but the file was

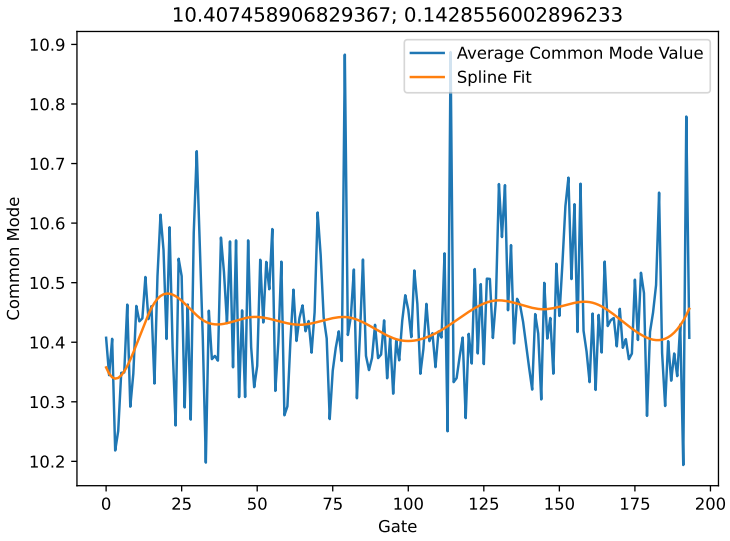


FIGURE L.7: Average common mode value of each gate in subregion *A*2 for the SEU event shown in Fig. L.1. The spline fit of these values is shown in orange. The title of this plot contains the average of all values and how much the spline fit varies in its values.

TABLE L.1: Table of common mode histograms with an SEU found in the LS1 dataset. Consecutive runs are indicated by the dashed line. The red marked row corresponds to the single non-SEU passing all criteria of the script. The corresponding histogram can be found in Fig. L.9.

#SEU	Experiment	Run Number	Module	Subregion	Fig.
1	8	2562	1.4.1	C1	L.8a
2	8	2563	1.4.1	C1	L.8b
3	18	1433	1.8.1	C3	L.8c
4	18	1441	1.8.1	C3	L.8d
5	18	1442	1.8.1	C3	L.8e
6	20	638	1.1.2	D1	L.8f
7	20	639	1.1.2	D1	L.8g
8	20	640	1.1.2	D1	L.8h
9	20	641	1.1.2	D1	L.8i
10	20	642	1.1.2	D1	L.8j
11	24	2005	1.3.1	-	L.9
12	26	289	1.8.1	A3	L.8k
13	26	821	1.1.1	D1	L.8l
14	26	823	1.1.1	D1	L.8m
15	26	837	1.1.1	D3	L.8n
16	26	839	1.1.1	D3	L.8o
17	26	1145	1.8.1	D3	L.8p

empty. This can happen when a run is restarted before data taking takes place. The same procedure was done with run 838 because runs 837 and 839 are found to be impacted by an SEU. Again, the run was empty.

It is important to note that both of these streaks happened in the same module and are only separated by a few runs. However, in the first streak the SEU impacted subregion $D1$ and the later streak $D3$. This should not be possible if only one SEU took place. In addition to that, the runs in between these streaks were checked for SEUs and none were found. It was therefore probably just a coincidence.

In addition to 16 histograms with a SEU, one common mode histogram with no SEU also passes all of the criteria. This is depicted in Fig. L.9. This histogram has some problems. The main one is that there are a lot of events dumped at high gate values in each chip. This affects all subregions. In addition to that, the band is rather wide and and all 1 subregions are therefore considered filled with non-zero entries. However, the distribution among them is in such a way, that the check for multiple filled subregions does not trigger and the histogram is therefore not thrown away. Since only one non-SEU histogram of $\sim$ 800 000 passed, the chosen criteria for the selection are kept for the future.

L.7 Online Data Taking

Up until now, the script was run on existing stored data but it is clear that such a script should be implemented in the online data taking in order to prevent these streaks since the operator were not able to find all of them and the one the operator found were not noticed as fast as possible, hence the streaks. In addition to that, with the planed increased luminosity, the probability for SEUs will also increase, making an automation more and more important.

To make an online monitoring for SEUs possible, the performance of the script must be as good as possible. For this reason, the script is written with arrays instead of pandas DataFrames. Also, the order of conditions is chosen such that it can run as fast as possible. While testing on a 1000 randomly chosen data files. The script took on average of 1.2 s to analyze a file. The time varies between about a second for files with clean common mode histograms and $\sim$ 1.8 s for histograms which reach the fitting condition. These times also include the time the script takes to load and open the root file. While taking data online, these parts will not contribute. After reaching out to experts, they confirmed that the performance is good enough and the script will be included in online data taking [103].

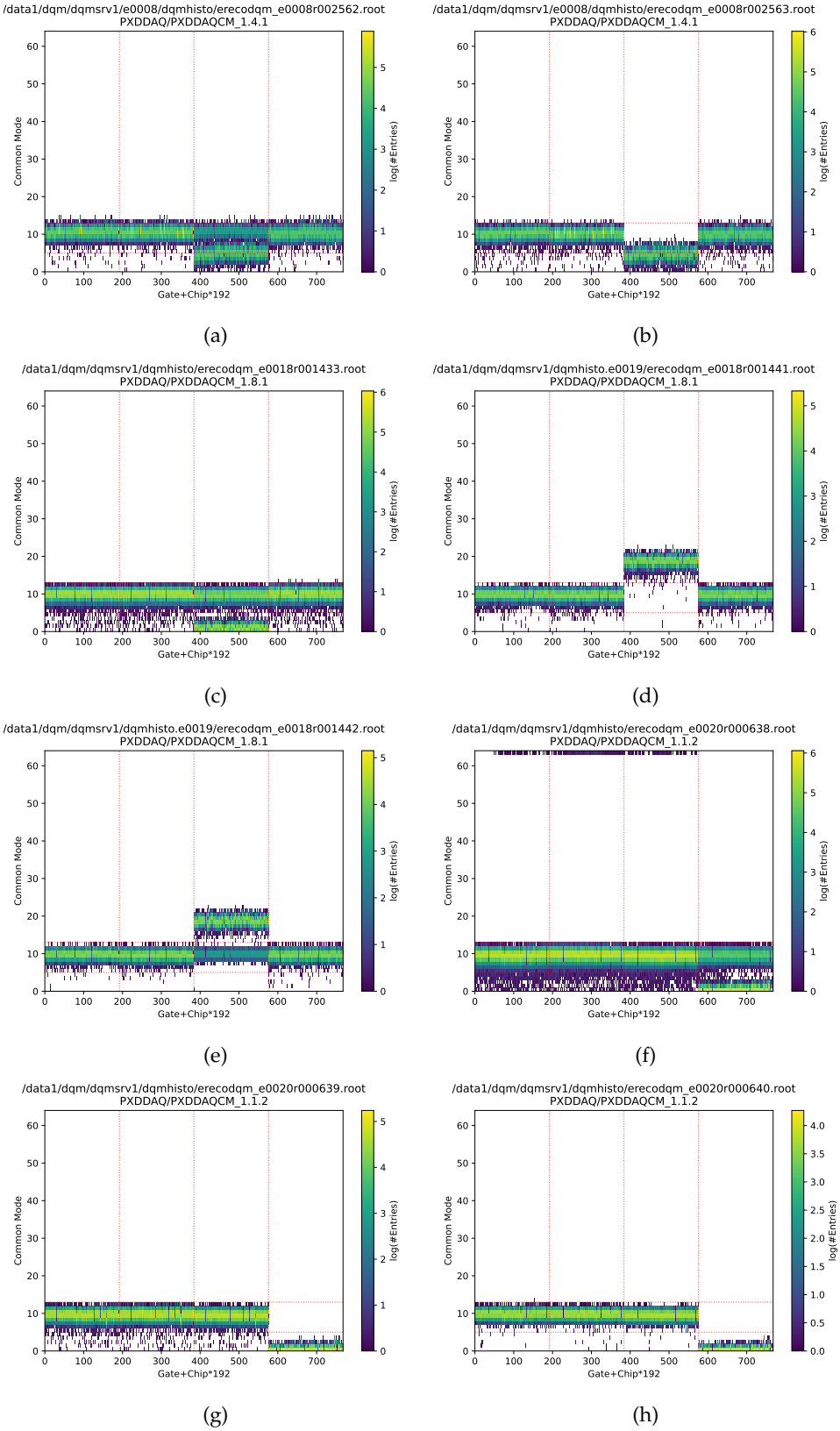
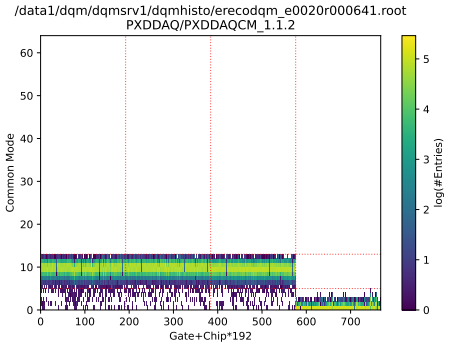
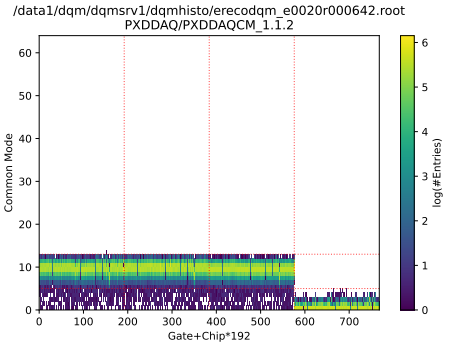


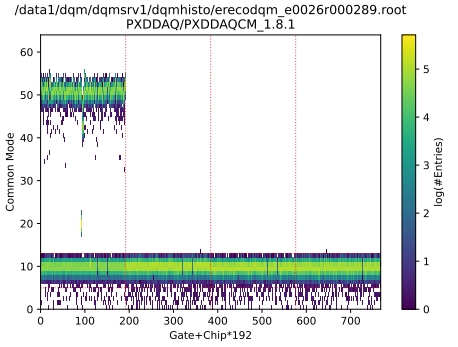
FIGURE L.8: The first eight found SEUs.



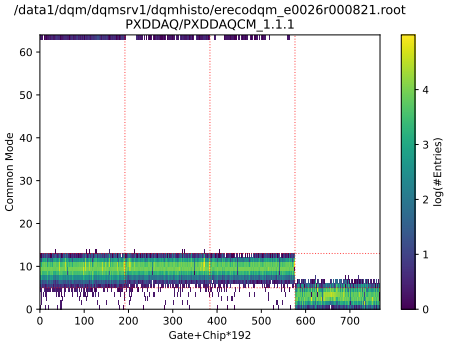
(i)



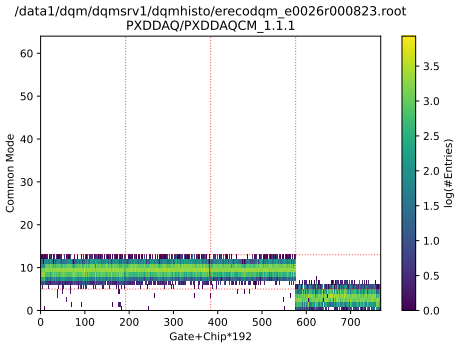
(j)



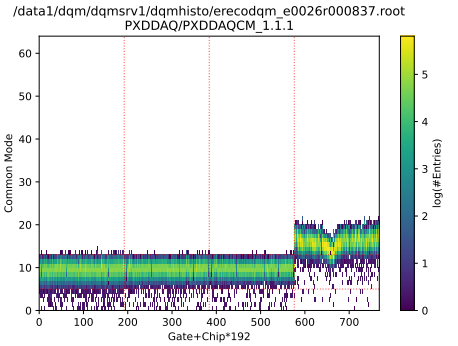
(k)



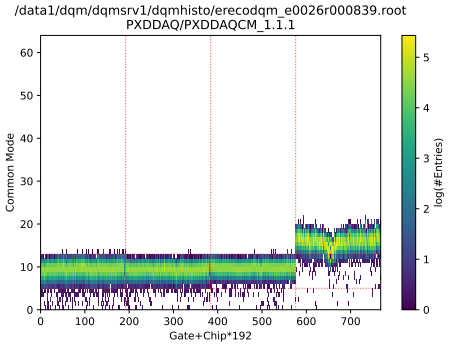
(l)



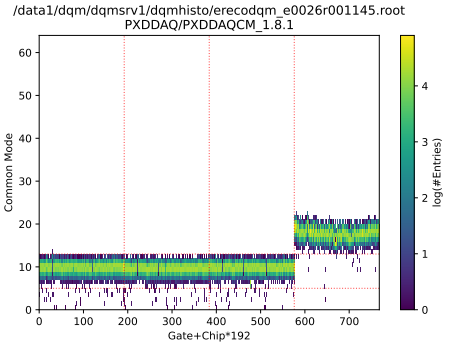
(m)



(n)



(o)



(p)

FIGURE L.8: The last eight found SEUs.

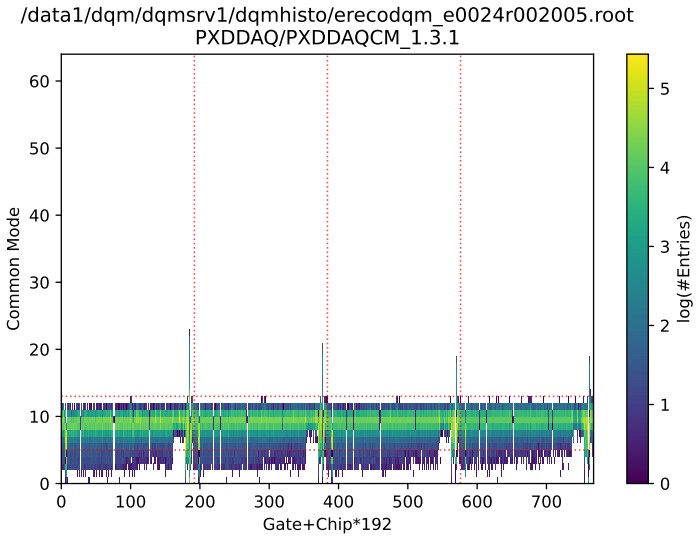


FIGURE L.9: Histogram passing all criteria without having an SEU.

Bibliography

[1] A.B. Arbuzov. "Quantum Field Theory and the Electroweak Standard Model". In: (2018). 35 pages, pp. 1–34. DOI: 10.23730/CYRSP-2017-004.1. arXiv: 1801.05670. URL: https://cds.cern.ch/record/2315477.

[2] Sheldon L. Glashow. "Partial-symmetries of weak interactions". In: *Nuclear Physics* 22.4 (1961), pp. 579–588. ISSN: 0029-5582. DOI: https://doi.org/10.1016/0029-5582(61)90469-2. URL: https://www.sciencedirect.com/science/article/pii/0029558261904692.

[3] P.W. Higgs. "Broken symmetries, massless particles and gauge fields". In: *Physics Letters* 12.2 (1964), pp. 132–133. ISSN: 0031-9163. DOI: https://doi.org/10.1016/0031-9163(64)91136-9. URL: https://www.sciencedirect.com/science/article/pii/0031916364911369.

[4] Steven Weinberg. "A Model of Leptons". In: *Phys. Rev. Lett.* 19 (21 1967), pp. 1264–1266. DOI: 10.1103/PhysRevLett.19.1264. URL: https://link.aps.org/doi/10.1103/PhysRevLett.19.1264.

[5] Nicola Cabibbo. "Unitary Symmetry and Leptonic Decays". In: *Phys. Rev. Lett.* 10 (12 1963), pp. 531–533. DOI: 10.1103/PhysRevLett.10.531. URL: https://link.aps.org/doi/10.1103/PhysRevLett.10.531.

[6] Makoto Kobayashi and Toshihide Maskawa. "CP-Violation in the Renormalizable Theory of Weak Interaction". In: *Progress of Theoretical Physics* 49.2 (Feb. 1973), pp. 652–657. ISSN: 0033-068X. DOI: 10.1143/PTP.49.652. eprint: https://academic.oup.com/ptp/article-pdf/49/2/652/5257692/49-2-652.pdf. URL: https://doi.org/10.1143/PTP.49.652.

[7] Andreas Höcker and Zoltan Ligeti. "CP Violation and the CKM Matrix". In: *Annual Review of Nuclear and Particle Science* 56.1 (2006), pp. 501–567. DOI: 10.1146/annurev.nucl.56.080805.140456. URL: https://doi.org/10.1146%2Fannurev.nucl.56.080805.140456.

[8] Yosef Nir. *CP Violation - A New Era*. 2001. arXiv: hep-ph/0109090 [hep-ph].

[9] J. Charles et al. "CP violation and the CKM matrix: assessing the impact of the asymmetric B factories". In: *The European Physical Journal C* 41.1 (2005), 1–131. ISSN: 1434-6052. DOI: 10.1140/epjc/s2005-02169-1. URL: http://dx.doi.org/10.1140/epjc/s2005-02169-1.

[10] Particle Data Group et al. "Review of Particle Physics". In: *Progress of Theoretical and Experimental Physics* 2020.8 (Aug. 2020), p. 083C01. ISSN: 2050-3911. DOI: 10.1093/ptep/ptaa104. eprint: https://academic.oup.com/ptep/article-pdf/2020/8/083C01/34673722/ptaa104.pdf. URL: https://doi.org/10.1093/ptep/ptaa104.

[11] S. L. Glashow, J. Iliopoulos, and L. Maiani. "Weak Interactions with Lepton-Hadron Symmetry". In: *Phys. Rev. D* 2 (7 1970), pp. 1285–1292. DOI: 10.1103/PhysRevD.2.1285. URL: https://link.aps.org/doi/10.1103/PhysRevD.2.1285.

[12] R. Aaij et al. "Differential branching fraction and angular analysis of the decay B 0 → K *0 μ + μ -". In: *Journal of High Energy Physics* 2013.8 (2013). DOI: 10.1007/jhep08(2013)131.

[13] R. L. Workman et al. "Review of Particle Physics". In: *PTEP* 2022 (2022), p. 083C01. DOI: 10.1093/ptep/ptac097.

[14] Wolfgang Altmannshofer et al. "Symmetries and asymmetries of $B \to K*\mu+\mu-$ decays in the Standard Model and beyond". In: *Journal of High Energy Physics* 2009.01 (2009), p. 019. DOI: 10.1088/1126-6708/2009/01/019. URL: https://dx.doi.org/10.1088/1126-6708/2009/01/019.

[15] R. Aaij et al. "Differential branching fraction and angular analysis of the decay $B0 \to K^*0\mu+\mu-$". In: *Journal of High Energy Physics* 2013.8 (2013). DOI: 10.1007/jhep08(2013)131. URL: https://doi.org/10.1007%2Fjhep08%282013%29131.

[16] R. Aaij et al. "Measurement of Form-Factor-Independent Observables in the Decay $B^0 \to K^{*0}\mu\mu$". In: *Phys. Rev. Lett.* 111 (19 2013), p. 191801. DOI: 10.1103/PhysRevLett.111.191801. URL: https://link.aps.org/doi/10.1103/PhysRevLett.111.191801.

[17] M De Cian. "Track Reconstruction Efficiency and Analysis of B0 to K*0 mu+ mu- at the LHCb Experiment". PhD thesis. University of Zurich, 2013. URL: https://doi.org/10.5167/uzh-85745.

[18] R. Aaij et al. "Angular analysis of the B 0 -> K*0 μ + μ - decay using 3 fb-1 of integrated luminosity". In: *Journal of High Energy Physics* 2016.2 (2016). DOI: 10.1007/jhep02(2016)104. URL: https://doi.org/10.1007%2Fjhep02%282016%29104.

[19] Andre Norbert Klotzbücher. "Systematic Studies to Analyze the Angular Differential Decay Rate of $B \to K^*\ell\ell$ with early Belle II Data". Master Thesis.

[20] Belle Collaboration et al. *Lepton-Flavor-Dependent Angular Analysis of $B \to K^*\ell^+\ell^-$*. 2016. arXiv: 1612.05014 [hep-ex]. URL: https://arxiv.org/abs/1612.05014.

[21] B Capdevila. "Assessing lepton-flavour non-universality from $B \to K^*\ell\ell$ angular analyses". In: *Journal of Physics: Conference Series* 873 (July 2017), p. 012039. ISSN: 1742-6596. DOI: 10.1088/1742-6596/873/1/012039. URL: http://dx.doi.org/10.1088/1742-6596/873/1/012039.

[22] R. R. Horgan et al. *Rare B decays using lattice QCD form factors*. 2015. arXiv: 1501.00367 [hep-lat]. URL: https://arxiv.org/abs/1501.00367.

[23] *Angular analysis of the $B^0 \to K^{*0}(892)\mu^+\mu^-$ decay at $\sqrt{s}$ = 13 TeV*. Tech. rep. Geneva: CERN, 2024. URL: https://cds.cern.ch/record/2899589.

[24] ATLAS Collaboration. *Angular analysis of $B_d^0 \to K^*\mu^+\mu^-$ decays in pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector*. May 2018.

[25] R. Aaij et al. "Amplitude Analysis of the $B^0 \to K^{*0}\mu\mu$ Decay". In: *Physical Review Letters* 132.13 (2024). ISSN: 1079-7114. DOI: 10.1103/physrevlett.132.131801. URL: http://dx.doi.org/10.1103/PhysRevLett.132.131801.

[26] Simon Wehle. "Angular Analysis of $B \to K^*\ell\ell$ and Search for $B^+ \to K^+\tau\tau$ at the Belle Experiment". Universität Hamburg, Diss., 2016. Dr. Hamburg: Universität Hamburg, 2016, p. 209. DOI: 10.3204/PUBDB-2016-03770. URL: https://bib-pubdb1.desy.de/record/308592.

[27] David M. Straub. *flavio: a Python package for flavour and precision phenomenology in the Standard Model and beyond*. 2018. arXiv: 1810.08132 [hep-ph]. URL: https://arxiv.org/abs/1810.08132.

[28] Danny van Dyk et al. *eos/eos: EOS Version 1.0.10*. Version v1.0.10. Sept. 2023. DOI: 10.5281/zenodo.8340859. URL: https://doi.org/10.5281/zenodo.8340859.

[29] M. Kowalski et al. "Improved Cosmological Constraints from New, Old, and Combined Supernova Data Sets". In: *The Astrophysical Journal* 686.2 (2008), pp. 749–778. DOI: 10.1086/589937. URL: https://doi.org/10.1086%2F589937.

[30] Tetsuo Abe et al. "Achievements of KEKB". In: *Progress of Theoretical and Experimental Physics* 2013.3 (Mar. 2013), 03A001. ISSN: 2050-3911. DOI: 10.1093/ptep/pts102. eprint: https://academic.oup.com/ptep/article-pdf/2013/3/03A001/4440618/pts102.pdf. URL: https://doi.org/10.1093/ptep/pts102.

[31] Ivan Heredia de la Cruz. "The Belle II experiment: fundamental physics at the flavor frontier". In: *Journal of Physics: Conference Series* 761 (Sept. 2016). DOI: 10.1088/1742-6596/761/1/012017.

[32] E Kou et al. "The Belle II Physics Book". In: *Progress of Theoretical and Experimental Physics* 2019.12 (2019). DOI: 10.1093/ptep/ptz106. URL: https://doi.org/10.1093%2Fptep%2Fptz106.

[33] Werner Herr and B Muratori. "Concept of luminosity". In: (2006). DOI: 10.5170/CERN-2006-002.361. URL: http://cds.cern.ch/record/941318.

[34] T. Abe et al. *Belle II Technical Design Report*. 2010. arXiv: 1011.0352 [physics.ins-det].

[35] P. Raimondi, D. N. Shatilov, and M. Zobov. *Beam-Beam Issues for Colliding Schemes with Large Piwinski Angle and Crabbed Waist*. 2007. arXiv: physics/0702033 [physics.acc-ph]. URL: https://arxiv.org/abs/physics/0702033.

[36] F. Abudin'en et al. "B-flavor tagging at Belle II". In: *The European Physical Journal C* 82 (2021).

[37] Adrian Vogel. "The coordinate system for LDC detector studies". In: *LC Notes. LC-DET-2005-009. pdf* (2005).

[38] *Electrons and Positrons Collide for the first time in the SuperKEKB Accelerator*. 2018. URL: https://www.ipmu.jp/en/20180426-FirstCollision.

[39] Florian Bernlochner et al. "Online Data Reduction for the Belle II Experiment using DATCON". In: *EPJ Web of Conferences* 150 (2017). Ed. by C. Germain et al., p. 00014. DOI: 10.1051/epjconf/201715000014. URL: https://doi.org/10.1051%2Fepjconf%2F201715000014.

[40] C. Marinas and M. Vos. "The Belle-II DEPFET pixel detector: A step forward in vertexing in the superKEKB flavour factory". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 650.1 (2011). International Workshop on Semiconductor Pixel Detectors for Particles and Imaging 2010, pp. 59–63. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2010.12.116. URL: https://www.sciencedirect.com/science/article/pii/S0168900210028962.

[41] *Belle II: Das Auge für die Antimaterie*. https://www.mpp.mpg.de/forschung/aufbau-der-materie/belle-ii-dem-antimaterie-raetsel-auf-der-spur/belle-ii-das-auge-fuer-antimaterie. [Accessed 06-09-2024].

[42] Harrison Schreeck et al. *Effects of gamma irradiation on DEPFET pixel sensors for the Belle II experiment*. Sept. 2021.

[43] Thomas Hauth. "Pattern Recognition at Belle II". In: (2016).

[44] P. A. Cherenkov. "Visible Radiation Produced by Electrons Moving in a Medium with Velocities Exceeding that of Light". In: *Phys. Rev.* 52 (4 1937), pp. 378–379. DOI: 10.1103/PhysRev.52.378. URL: https://link.aps.org/doi/10.1103/PhysRev.52.378.

[45] *Belle II Luminosity - Belle II - DESY Confluence — confluence.desy.de*. https://confluence.desy.de/display/BI/Belle+II+Luminosity. [Accessed 22-11-2023].

[46] *Offline Luminosity Page - DESY Confluence — confluence.desy.de*. https://confluence.desy.de/display/BI/Offline+Luminosity+Page. [Accessed 22-11-2023].

[47] 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.

[48] Project Pythia contributors. *Project Pythia*. [Accessed 18-01-2023]. 2021. URL: https://projectpythia.org/.

[49] Elisabetta Barberio, Bob van Eijk, and Zbigniew Was. "Photos — a universal Monte Carlo for QED radiative corrections in decays". In: *Computer Physics Communications* 66.1 (1991), pp. 115–128. ISSN: 0010-4655. DOI: https://doi.org/10.1016/0010-4655(91)90012-A. URL: https://www.sciencedirect.com/science/article/pii/001046559190012A.

[50] 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.

[51] *Skim Main Page — confluence.desy.de*. https://confluence.desy.de/display/BI/Skim+main+page. [Accessed 04-12-2023].

[52] Gane Samb Lo. *Mathematical Foundations of Probability Theory*. 2018. arXiv: 1808.01713 [math.PR].

[53] Norman Fenton. "Improve statistics in court". In: *Nature* 479.7371 (2011), pp. 36–37. ISSN: 1476-4687. DOI: 10.1038/479036a. URL: https://doi.org/10.1038/479036a.

[54] Luca Lista. "Practical Statistics for Particle Physicists". en. In: *CERN Yellow Reports: School Proceedings* (2017), Vol 5 (2017): Proceedings of the 2016 European School of High–Energy Physics. DOI: 10.23730/CYRSP-2017-005.213. URL: https://e-publishing.cern.ch/index.php/CYRSP/article/view/354.

[55] M. Feindt and M. Prim. "An algorithm for quantifying dependence in multivariate data sets". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 698 (2013), pp. 84–89. DOI: 10.1016/j.nima.2012.09.043. URL: https://doi.org/10.1016%2Fj.nima.2012.09.043.

[56] Erich Lohrmann Volker Blobel. *Statistische und numerische Methoden der Datenanalyse*. 1998. DOI: https://doi.org/10.1007/978-3-663-05690-4.

[57] Jonas Eschle et al. "zfit: Scalable pythonic fitting". In: *SoftwareX* 11 (2020), p. 100508. ISSN: 2352-7110. DOI: https://doi.org/10.1016/j.softx.2020.100508. URL: https://www.sciencedirect.com/science/article/pii/S2352711019303851.

[58] M. Pivk and F.R. Le Diberder. "Plots: A statistical tool to unfold data distributions". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 555.1 (2005), pp. 356–369. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2005.08.106. URL: https://www.sciencedirect.com/science/article/pii/S0168900205018024.

[59] Hans Dembinski et al. "Custom Orthogonal Weight functions (COWs) for event classification". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1040 (Oct. 2022), p. 167270. ISSN: 0168-9002. DOI: 10.1016/j.nima.2022.167270. URL: http://dx.doi.org/10.1016/j.nima.2022.167270.

[60] Carl De Boor and Carl De Boor. *A practical guide to splines*. Vol. 27. springer-verlag New York, 1978.

[61] Yann Coadou. "Boosted Decision Trees". In: *Artificial Intelligence for High Energy Physics*. WORLD SCIENTIFIC, 2022, pp. 9–58. DOI: 10.1142/9789811234033_0002. URL: https://doi.org/10.1142%2F9789811234033_0002.

[62] Tom Fawcett. "An introduction to ROC analysis". In: *Pattern Recognition Letters* 27.8 (2006). ROC Analysis in Pattern Recognition, pp. 861–874. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2005.10.010. URL: https://www.sciencedirect.com/science/article/pii/S016786550500303X.

[63] Shir Meir Lador. *What metrics should be used for evaluating a model on an imbalanced data set?* [Accessed 29-07-2024]. 2017. URL: https://towardsdatascience.com/what-metrics-should-we-use-on-imbalanced-data-set-precision-recall-roc-e2e79252aeba.

[64] John J Hopfield. "Neural networks and physical systems with emergent collective computational abilities." In: *Proceedings of the national academy of sciences* 79.8 (1982), pp. 2554–2558.

[65] IBM Cloud Education. *Neural Networks*. [Accessed 29-07-2024]. Aug. 2020. URL: https://www.ibm.com/cloud/learn/neural-networks.

[66] Ayan Nair. *Turn your computer into an Artist with AI*. [Accessed 29-01-2021]. Jan. 2021. URL: https://medium.com/mlearning-ai/turn-your-computer-into-an-artist-with-ai-8a58014bb5d4.

[67] Jan Kocoń et al. *ChatGPT: Jack of all trades, master of none*. 2023. arXiv: 2302.10724 [cs.CL].

[68] Martín Abadi et al. *TensorFlow, Large-scale machine learning on heterogeneous systems*. Nov. 2015. DOI: 10.5281/zenodo.4724125.

[69] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.

[70] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

[71] Twan van Laarhoven. "L2 Regularization versus Batch and Weight Normalization". In: *CoRR* abs/1706.05350 (2017). arXiv: 1706.05350. URL: http://arxiv.org/abs/1706.05350.

[72] Marco Peixeiro. *How to Improve a Neural Network With Regularization*. Accessed 17-01-2023. URL: https://towardsdatascience.com/how-to-improve-a-neural-network-with-regularization-8a18ecda9fe3.

[73] Alex Labach, Hojjat Salehinejad, and Shahrokh Valaee. "Survey of Dropout Methods for Deep Neural Networks". In: *CoRR* abs/1904.13310 (2019). arXiv: 1904.13310. URL: http://arxiv.org/abs/1904.13310.

[74] Vagif Aliyev. *Gradient Descent, clearly explained in Python, Part 2: The compelling code.* [Accessed 29-07-2024]. 2020. URL: https://towardsdatascience.com/gradient-descent-clearly-explained-in-python-part-2-the-compelling-code-c21ee26fbc28.

[75] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. arXiv: 1502.03167 [cs.LG].

[76] M.A. Ganaie et al. "Ensemble deep learning: A review". In: *Engineering Applications of Artificial Intelligence* 115 (Oct. 2022), p. 105151. ISSN: 0952-1976. DOI: 10.1016/j.engappai.2022.105151. URL: http://dx.doi.org/10.1016/j.engappai.2022.105151.

[77] Gao Huang et al. *Snapshot Ensembles: Train 1, get M for free*. 2017. arXiv: 1704.00109 [cs.LG].

[78] George Cybenko. "Approximation by superpositions of a sigmoidal function". In: *Mathematics of control, signals and systems* 2.4 (1989), pp. 303–314.

[79] Kurt Hornik. "Approximation capabilities of multilayer feedforward networks". In: *Neural Networks* 4.2 (1991), pp. 251–257. ISSN: 0893-6080. DOI: https://doi.org/10.1016/0893-6080(91)90009-T. URL: https://www.sciencedirect.com/science/article/pii/089360809190009T.

[80] Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Neural network approximation: Three hidden layers are enough". In: *Neural Networks* 141 (2021), pp. 160–173. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2021.04.011. URL: https://www.sciencedirect.com/science/article/pii/S0893608021001465.

[81] Dennis Elbrächter et al. *Deep Neural Network Approximation Theory*. 2021. arXiv: 1901.02220 [cs.LG].

[82] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. "Multilayer feedforward networks are universal approximators". In: *Neural networks* 2.5 (1989), pp. 359–366.

[83] Marcel Rieger. "Mastering Model Building". In: (Aug. 2023). URL: https://indico.desy.de/event/37477/contributions/149065/attachments/83647/110653/ErumData_MLSchool_ModelBuilding_2023.pdf.

[84] Wikipedia contributors. *Decision Tree — Wikipedia, The Free Encyclopedia*. [Accessed 20-09-2023]. 2004. URL: https://commons.wikimedia.org/wiki/File:Decision_Tree.jpg.

[85] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *CoRR* abs/1603.02754 (2016). arXiv: 1603.02754. URL: http://arxiv.org/abs/1603.02754.

[86] M. Dash and H. Liu. "Feature selection for classification". In: *Intelligent Data Analysis* 1.1 (1997), pp. 131–156. ISSN: 1088-467X. DOI: https://doi.org/10.1016/S1088-467X(97)00008-5. URL: https://www.sciencedirect.com/science/article/pii/S1088467X97000085.

[87] T. Kuhr et al. "The Belle II Core Software". In: *Computing and Software for Big Science* 3.1 (2018). DOI: 10.1007/s41781-018-0017-9. URL: https://doi.org/10.1007%2Fs41781-018-0017-9.

[88] J.-F. Krohn et al. "Global decay chain vertex fitting at Belle II". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 976 (2020), p. 164269. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2020.164269. URL: https://www.sciencedirect.com/science/article/pii/S0168900220306653.

[89] *Continuum Suppression (CS) — software.belle2.org*. https://software.belle2.org/development/sphinx/online_book/basf2/cs.html. [Accessed 30-07-2024].

[90] Geoffrey C. Fox and Stephen Wolfram. "Observables for the Analysis of Event Shapes in e^+e^- Annihilation and Other Processes". In: *Phys. Rev. Lett.* 41 (23 1978), pp. 1581–1585. DOI: 10.1103/PhysRevLett.41.1581. URL: https://link.aps.org/doi/10.1103/PhysRevLett.41.1581.

[91] S. H. Lee et al. "Evidence for $B^0 \to \pi^0\pi^0$". In: *Phys. Rev. Lett.* 91 (26 2003), p. 261801. DOI: 10.1103/PhysRevLett.91.261801. URL: https://link.aps.org/doi/10.1103/PhysRevLett.91.261801.

[92] *Recommendations For Hadron ID - Moriond 2023 - DESY Confluence — confluence.desy.de*. https://confluence.desy.de/pages/viewpage.action?pageId=349455022. [Accessed 07-03-2024].

[93] *Recommendations For Lepton ID - Moriond 2024 - DESY Confluence — confluence.desy.de*. https://confluence.desy.de/display/BI/Recommendations+for+Lepton+ID+-+Moriond2024. [Accessed 07-03-2024].

[94] Belle Collaboration, A. Abdesselam, and I. Adachi. *Angular analysis of $B^0 \to K^*(892)^0\ell^+\ell^-$*. 2016. arXiv: 1604.04042 [hep-ex].

[95] LHCb collaboration et al. *Determination of short- and long-distance contributions in $B^0 \to K^{*0}\mu^+\mu^-$ decays*. 2023. arXiv: 2312.09102 [hep-ex].

[96] Jigarkumar Vinodbhai Patel. "Systematic Studies on the Signal Yield Determination in the Decay $B \to K^*\ell^+\ell^-$ at Belle II". Master Thesis.

[97] "Test of lepton universality in beauty-quark decays". In: *Nature Physics* 18.3 (2022), pp. 277–282.

[98] R. Aaij et al. "Test of Lepton Universality in $b \to s\ell^+ell^-$ Decays". In: *Physical Review Letters* 131.5 (Aug. 2023). ISSN: 1079-7114. DOI: 10.1103/physrevlett.131.051803. URL: http://dx.doi.org/10.1103/PhysRevLett.131.051803.

[99] Felix Müller Björn Spruck. *Status Update SuperKEKB and PXD*. https://indico.mpp.mpg.de/event/6308/contributions/16331/attachments/11602/13016/004_MUELLER_Felix_SPRUCK_Bjoern_-_SuperKEKB_PXD.pdf. [Accessed 06-09-2024].

[100] *Luminosity*. https://www.belle2.org/research/luminosity/. [Accessed 04-09-2024]. 2024.

[101] George C Messenger et al. *Single event phenomena I*. Springer, 1997.

[102] *Computers in Spaceflight: The NASA Experience*. URL: https://history.nasa.gov/computers/Ch4-4.html.

[103] B. Spruck. "Private Commnunication".