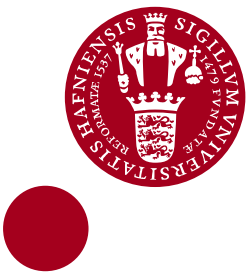
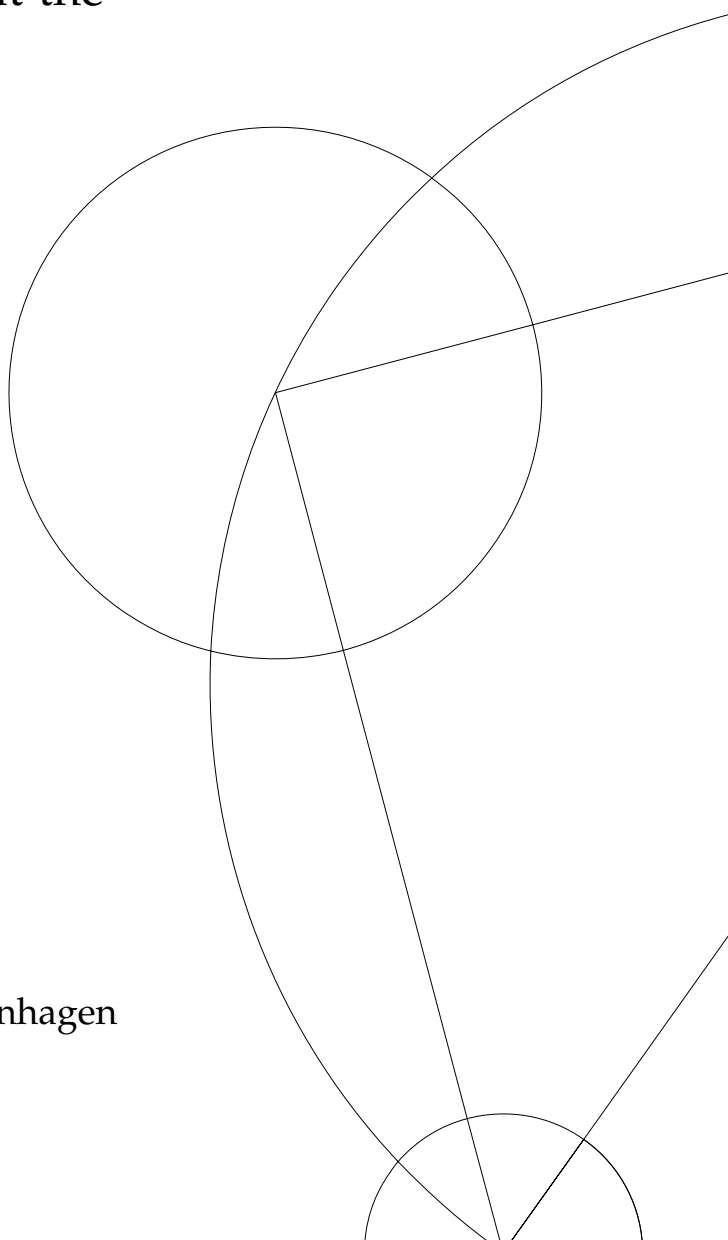
Daniel Hans Munk



B-Physics and Gradient Boosted Decision Trees

Identifying $B^0 \rightarrow K^{*0}ee$ Decays using GBDTs for the Study of Lepton Flavour Universality Violation at the ATLAS Detector, CERN

22/05/2023



MSc Thesis
Niels Bohr Institute, University Of Copenhagen

SUPERVISOR Associate Professor Troels C. Petersen

MSc Thesis
Niels Bohr Institute, University Of Copenhagen

Using Tufte-LaTeX for document design: WWW.TUFTE-LATEX.GITHUB.IO/TUFTE-LATEX/

First printing, May 2023

Abstract

This thesis is part of the Rare decays subgroup of the ATLAS B-Physics and Light States Working Group at CERN who aims to measure the $R_{K^{*0}}$ double ratio, which through Lepton Flavor Universality (LFU) can indicate if there are physics Beyond the Standard Model present in the $B^0 \rightarrow K^{*0}\ell\ell$-decay. The work of the thesis is related to the electron channel where $\ell\ell$ is an electron-proton pair with the focus of separating the two species of signal: B^0 and its antiparticle: $\overline{B}^0$ from the background. This is achieved using Gradient Boosting Decision Trees (GBDTs) trained on a mix of Monte Carlo-generated data and ATLAS data. After hyperparameter optimization and feature engineering, the GBDT model ended up with a total B^0 signal efficiency of $72 \pm 3\%$ in Monte Carlo and B^0-mass sidebands. As the B^0-mass region in the low q^2-bin ($q^2 \in [1.1, 6.0]^{\text{GeV}^2/\text{c}^4}$) is blinded at the current stage of the analysis, the signal yield for calculating the $R_{K^{*0}}$-ratio is extracted using a χ^2-fit in the high q^2-bin ($q^2 \in [6.0, 11.0]^{\text{GeV}^2/\text{c}^4}$). The fit used is a composite probability function consisting of two Gaussian probability distributions and a Bukin distribution, giving a signal yield of $N_{Sig(B^0)} = 1853 \pm (\geq 45)$ on Period K, Run 2 ATLAS data with a signal significance of 20.0 ± 0.4.
This shows GBDTs are a viable approach in separating B^0 and $\overline{B}^0$ from the background, and the hope is that the contribution of this thesis will benefit the Rare decays group in the measurement of the $R_{K^{*0}}$-ratio.

Acknowledgements

I want to express my deepest gratitude to my supervisor, Associate Professor Troels C. Petersen, for his patience with me and his dedication to answering my numerous emails. He has been a better supervisor than I could hope for.
I would also like to thank my predecessors, Mathias Ajami, for his prior work and Malthe Algren for help at the start of the analysis.
A special thank you goes out to the RK* group members who all have been helpful and kind to me when I had thousands of questions, particularly Tomas Jakoubek and Dvij Mankad, who have generously given their time outside of the weekly meetings to answer any question I might have.
I cannot forget to thank my family, friends, and colleagues for their unwavering support and encouragement while writing this thesis.
Finally, I would like to express my immense gratitude to the love of my life and fiancé, Ninna, for her love and unwavering support.

Contents

Acronyms

ALICE	A Large Ion Collider Experiment.
ATLAS	A Toroidal LHC Apparatus.
BSM	Beyond the Standard Model.
CERN	Conseil Européen pour la Recherche Nucléaire / eng: European Organization for Nuclear Research.
CKM	Cabibbo-Kobayashi-Maskawa.
CMS	Compact Muon Solenoid.
FCNC	Flavour-Changing Neutral-Current.
GBDT	Gradient Boosting Decision Tree.
GIM	Glashow–Iliopoulos–Maiani.
GNN	Graph Neural Network.
GSF	Gaussian-sum Filter.
ISR	Intersecting Storage Rings.
LEIR	Low Energy Ion Ring.
LFU	Lepton Flavour Universality.
LHC	Large Hadron Collider.
LHCb	LHC-beauty.
LINAC3	Linear Accelerator 3.
LINAC4	Linear Accelerator 4.
ML	Machine Learning.
PD	Pixel Detector.
PS	Proton Synchrotron.
PSB	Proton Synchrotron Booster.
QCD	Quantum Chromodynamics.
QFT	Quantum Field Theory.
ROC	Receiver Operating Characteristic.
SCT	Semiconductor Tracker.
SM	Standard Model.
SPS	Super Proton Synchrotron.
TRT	Transition Radiation Tracker.

Introduction

The matter we observe in our surrounding universe varies in its constituent. Naively, it follows the periodic table of the 20th century, but while this period established the atoms, it also taught us that these were not the fundamental building blocks of matter. Over the years, groundbreaking experiments and theoretical advancements have revealed deeper layers of complexity within the subatomic realm. The study of particle physics has unraveled a fascinating world where elementary particles exist, namely quarks, leptons, and bosons - each group with its own unique properties. The model which encapsulates this incredible world is called the Standard Model (SM) and is one of the most influential models in physics to this very date.

The Standard Model does not only touch upon matter, but it also explains three of the four[1] fundamental forces: the strong force, the weak force, and the electromagnetic force. The latest addition to the SM was the discovery of the Higgs boson discovered in 2012[54] at CERN in a collaboration between A Toroidal LHC Apparatus (ATLAS) and Compact Muon Solenoid (CMS).

[1] Gravitation is the force not explained by SM; however, there has been proposed an extra addition to the SM going by the name: "Graviton" which is the medium for gravity, just as the other forces have particles which act as a medium.

In short, the SM treats leptons the same way independent of the generation[2] they belong to. This means the fraction of decay into each of the generations must be equal to one, and any deviation from unity is a sign of physics Beyond the Standard Model (BSM). This particular deviation is called Lepton Flavour Universality (LFU) violation. The study of the B-meson decays, called B-physics, is widely used for testing LFU violation since the SM predicts very precisely how the B-mesons should behave. As B-mesons are rare, a deviation from the standard model would be significant and measurable.

[2] There are three generations.

This thesis is written as a part of a larger project under the *Rare decays* group[3] which is a subgroup of *ATLAS B-Physics and Light States Working Group* at ATLAS, CERN. This group studies the $B^0 \rightarrow K^{*0}\ell\ell$ decay[4] and according to SM and LFU the ratio of decays should be one: $1 \approx R_{K^{0*}} = \frac{\mathcal{B}(B^0 \rightarrow K^{0*}\mu^+\mu^-)}{\mathcal{B}(B^0 \rightarrow K^{0*}e^+e^-)}$. The equation

[3] Throughout this thesis, the Rare decay group will be called the $\boldsymbol{RK}^*$ group.

[4] $\ell\ell = \{e^+e^-, \mu^+\mu^-\}$ such that the decay is either: $B^0 \rightarrow K^{0*}e^+e^-$ or $B^0 \rightarrow K^{0*}\mu^+\mu^-$. The $+/-$ is often omitted.

for the $R_{K^{0*}}$ can be written as a function of the efficiency corrected yield: $\frac{N}{\varepsilon}(X)$ where N is the measured yield from a fit of the signal or control decay, and ε is the efficiency in selecting the signal or control decay. The control decay used are the J/ψ decay with a ratio $r_{J/\psi}$ which are measured to statistically unity[62].

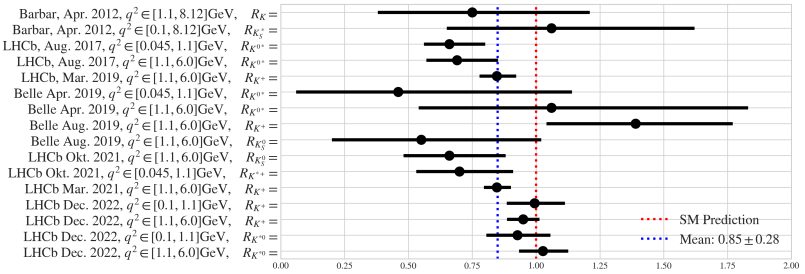


Figure 1: A historical overview of all the R_K-ratio measurements. The red line shows the SM predicted unity, and the blue shows the *naive* weighted mean (naive; since the different measurements are correlated). Both systematic and statistical uncertainties are included in the error bars. Measurements are from: [34],[62], [60], [71], [9], [63], [58] and [11] for the latest measurement at LHCb.

The earliest measurements specifically on the $R_{K^{*0}}$-ratio were done by LHCb [62] where $R_{K^{*0}} = 0.66^{+0.11}_{-0.07} \pm 0.03$ for $q^2 \in [0.045, 1.1]\text{Gev}^2/\text{c}^4$ [5] and $R_{K^{*0}} = 0.69^{+0.11}_{-0.07} \pm 0.05$ for $q^2 \in [1.1, 6.0]\text{GeV}^2/\text{c}^4$. The full historical overview of the different R_K-rato is seen in Fig. (1) where the latest measurements from LHCb are included. As seen, there is a good indication of BSM physics since the *naive* weighted mean of all historical measurements yields: $(0.85 \pm 0.28) \neq 1$. It is clear that the different experiments do not agree on the ratio. With the exception of the latest LHCb measurements from 2022, who measured the ratio close to unity with low uncertainty.
This highlights the significance of the RK* group's contribution as it will be crucial in either validating the LHCb measurement and therefore indicate no BSM physics in relation to B^0 decays or the RK* groups measurement will deviate from unity and hence indicate the opposite.

[5] Branching Ratios ($\mathcal{B}$) comes from integrating the decay rate (Γ) over the squared dilepton invariant mass (q^2), as different regions of q^2 have different properties.

This thesis focuses on the electron signal yield through the separation of background, B^0 and $\overline{B}^0$, and then the extraction of the B^0 signal yield; $N_{Sig(B^0)}$ from χ^2-fits on the separated B^0 signal. The main focus will be on the separation driven by Machine Learning. The RK* group uses two Grapth Neural Networks (GNNs) trained on non-resonant[6] MC data in the invariant B-mass region: $m_B \in [4, 5.7]\text{GeV}/\text{c}^2$ and real data in the two sideband regions: invariant B-mass $m_B \notin [4, 5.7]\text{GeV}/\text{c}^2$ or invariant B-mass $m_B \in [4, 5.7]\text{GeV}/\text{c}^2$ with the two tracks of same sign charge[7] for $q^2 \in [0.1, 6.0]\text{GeV}^2/\text{c}^4$. This thesis explores the usage of Gradient Boosting Decision Trees (GBDT) for the background vs. B^0 vs. $\overline{B}^0$ selection trained on the same data. The motivation is that GBDTs are less of a black-box model than GNNs, and the GBDT is incredibly fast at training while retaining their high performance in

[6] non-resonant means that the decay is direct where a resonant decay is a decay with an intermediate step.

[7] *Tracks* in this context means the Kaon-pion pair from the $K^{*0} \to K\pi$ decay.

classification problems.
The intention is to get equal to or better performance than the GNN approach used by the RK* group. Using various tests, each model has to be tested for its performance in selecting each of the three classes and its ability to select a signal without introducing distortion to the signal shape.

When the GBDTs have been trained, they are applied on the unblinded signal region $m_B \in [4, 5.7]^{\text{GeV}/c^2}$ of the $q^2 \in [1.1, 6.0]^{\text{GeV}^2/c^4}$-bin[8] and the best cut in the GBDTs are found using blinded a significance scan: Significance $= \frac{N_{sig}}{\sqrt{N_{Sig}+N_{Bkg}}}$. The maximum significance GBDT cuts are then used in the final fit where the signal yield is extracted; $N_{Sig(B^0)}$ for Period K, Run 2 ATLAS data.

[8] The low q^2-bin poses a challenge in that the non-resonant in $q^2 \in [0.1, 6.0]^{\text{GeV}^2/c^4}$ is rarer than the corresponding resonant channel due to the available energy in the low q^2-bin.

The work of this thesis will contribute to the RK* group by testing out alternative machine learning through GBDTs and challenging the already current ML approach. In addition, the idea is to test out new ML pipeline configurations and do feature studies that can benefit the RK* group.

The thesis is structured in three main parts: *Theory*, *Analysis*, and the *Wrap-Up*. The *Theory* contains three main theoretical areas of this thesis: The physics behind B-physics is the first of the theory section, then moving on to the ATLAS detector, where the different components are explained, and then lastly, the mathematics behind the machine learning models used. The *Analysis*-part starts with the approach used, namely what data is used and a description of the methodology. After the methodology, the analysis begins with translating the RK* GNN approach to a GBDT approach for background, B^0, and $\overline{B}^0$ selection. The last part of the analysis is dedicated to improving the GBDTs using various techniques; Feature engennering, multiple GBDTs, etc. The *Wrap-Up* reviews the analysis results, and the methodology used throughout the thesis will be revisited for discussion along with the uncertainties related to the experiment and analysis. The Wrap-UP is concluded with a summary of the findings and a discussion of how this thesis's work will extend into the future in relation to the RK* group.

In summary, this thesis aims to contribute to the extraction of the electron yield from the $B^0 \rightarrow K^{*0}e^+e^-$-decays within the ATLAS detector. This is accomplished through testing various GBDT configurations, which are trained to distinguish B^0, $\overline{B}^0$ events from the background without distorting the original mass shape of B^0 and $\overline{B}^0$. Finally, the signal yield is extracted with χ^2 fits on the B^0-mass.

Part I

The Theory Behind

1 The Standard Model

Our universe is made of matter, which comes in many different sizes and shapes and has different attributes and properties. Already in early Greece, Democritus had the idea that the world is made of tiny "atoms"[5] [1]. More than 2000 years later, in the yearly 20th century, physicists found that the universe only consists of a handful of tiny building blocks that make up all other matter. The knowledge of these tiny building blocks and the rules they obey was in the 1970s formulated into the theory, now known as the Standard Model (SM). Not only does the SM deal with matter, but it also describes three of the four fundamental forces which act upon matter.

[1] The ancient Greek word *atomos* means uncuttable.

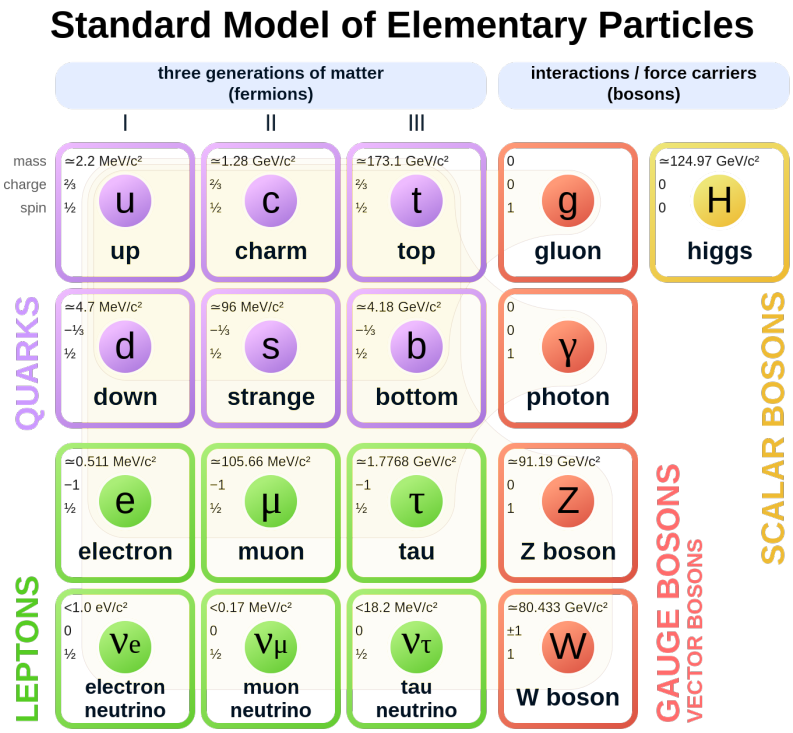


Figure 1.1: A figure of the 12 fermions and five bosons which makes up The Standard Model. Source of figure: [49].

1.1 Fundamental Particles

As seen in Fig. (1.1), there are 12 fermions[2] which are further divided into two groups: Quarks and Leptons. Fermions come with spin[3]: $\frac{1}{2}$ and obeys the Pauli Exclusion Principle[25], which states that multiple identical fermions cannot be in the same quantum state. Say a two-particle system of particle a and particle b in state $\Psi_1(a)$ and $\Psi_2(b)$ the naive quantum state would be: $\Psi = \Psi_1(a)\Psi_2(b)$, however since a and b are indistinguishable the wave-function are modified to the equation seen in Eq. (1.1).

$$\Psi = \Psi_1(a)\Psi_2(b) \pm \Psi_1(b)\Psi_2(a) \tag{1.1}$$

The "$-$" in Eq. (1.1) is required for fermions, meaning that if $a = b$, the probability wave-function vanishes, and this is the Pauli Exclusion Principle which means a and b follows Fermi-Dirac statistics[4]. The "$+$" in Eq. (1.1) is required for Bosons, which follows Bose-Einstein statistics which dictates that there can be any number of bosons at any quantum state.
Leptons and quarks have various intrinsic properties like electric charge, mass, and spin, and they are acted upon by the three fundamental forces: gravitation, weak interaction, and electromagnetism. Quarks also have *colour charge*, which means the strong interaction also acts upon it, and hence quarks differ from the lepton.
There are six types or *flavours*[5] of quarks and six for leptons. Quarks and leptons[6] are split into three generations[7] which are seen in Fig. (1.1). Starting with generation I (Gen.I), which has the smallest mass and ends with the greatest mass at Gen.III. Generation I is the most stable generation, then Gen.II and lastly Gen.III, which has the largest mass and hence is very short-lived. The stable elements we know from the periodic table[72] are all made of generation I leptons and quarks.[8] Less stable fermions will decay into more stable generations through weak interactions.[9]

One should also note that fermions and their antiparticles differ in some intrinsic properties by having the same magnitude but opposite signs.

[2] There are 24 since there is a corresponding antiparticle for each fermion.

[3] Note that "spin" and "isospin" are two different things. Spin is the angular momentum, and isospin is related to quark composition.

[4] Where the name "Fermion" comes from.

[5] In the SM - "flavours" are used instead of "types"

[6] Or just fermions.

[7] or families.

[8] *Electrons* are a lepton, while *neutrons* which are made of one up and two down quarks (udd) and *protons* which are made of two up and one down quark (uud).

[9] Which also goes under the name: the weak or weak nuclear force.

1.1.1 Composite Particles / Hadrons

The quarks are not interesting in isolation. However, the composition of quarks is of more interest, and these compositions of quarks are called *hadrons* which are held together through the strong interaction[10]. The hadrons fall into two categories: *Baryons* and *Mesons*. The rules are as follows: If a composite particle is made from an odd number of quarks, it is a baryon; if it is made of an even num-

[10] An analogy could be how molecules need the electric force to be held together.

ber of quarks, it is a meson. The most common baryons are the proton and neutron, whereas the most common mesons are: Pions (π), Kaons (K), B-mesons (B), D-mesons (D) and η-mesons (η). Note that hadrons do not contain a top quark (t) since the top quark has a lifetime of $\sim 0.5 \times 10^{-24}$s[72, p.817] due to its heavy mass (see Fig. (1.1)). The top quark does not have time to bind before it decays hence the absence of top quarks in hadrons. The hadron most interesting to this thesis is the B-meson, and a more in-depth review of it is later in the theory section.

1.2 *Fundamental Interactions*

Each of the four fundamental forces can be described by a mathematical vector field where gravity is described as a continuous field by Einstein's general theory of relativity [20] and is the only force that is not explained by the SM and therefore is an active research topic. The three others: *Electromagnetic interactions*, *strong interactions*, and *weak interactions* can be described with discrete quantum fields, and the mathematical framework is called Quantum Field Theory (QFT).
The gauge bosons (Fig. (1.1); 4^{th} column) are the mediators of the different interactions. The *strong interaction* holds quarks in hadrons together. The mediator for the strong interaction is the gluon with spin 1, which is massless. The strong interaction is called the nuclear force in relation to binding protons and neutrons to form atomic nuclei.
The theory which describes the strong interaction is called Quantum Chromodynamics (QCD). A noteworthy part of QCD is *colour confinement*. Quarks have an attribute called colour[11], and colour confinement states that free particles must have a colour charge equal to zero. [72, p. 149-176]
A quarks colour can be: red, green, or blue (see Fig. (1.2)), and an anti-quark can be: anti-red, anti-green, and anti-blue. The gluons are a mix of two colours, and baryons are made of a quark colour combination which gives a net zero colour ("white") which is the same for mesons. To form a baryon made of three quarks, there needs to be all free colours or anti-colours present, and for a meson with two quarks: a colour and its anti-colour needs to be present.

[11] This "colour" has nothing to do with the visible light spectrum.

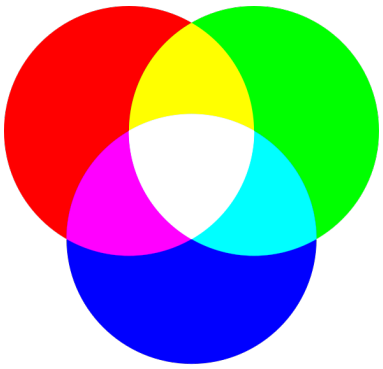


Figure 1.2: The three colours charges quarks can take: red, green, and blue. A mix of all gives white or colourless. Visual color and quark colour are not the same. This figure is shown for pedagogical reasons only. Source of figure: [19].

The weak interactions is the force that is the cause of radioactive decay. The gauge bosons which carry this interaction are the Z^0- and $W^{\pm}$-boson, which both have mass and spin 1. The W^+ is positively charged with 1 e, and The W^- is negatively charged of -1 e, and they are each others antiparticle. The Z^0 has a neutral charge and is its own antiparticle. The weak force has the property that it can

change the flavour of a quark from a down-quark to an up-quark.[12]

[12] The decay: $d \rightarrow u + W^-$.

The electromagnetic interaction is mediated by the photon with zero mass and spin one and is responsible for generating electromagnetic fields which hold electrons in their orbitals at their corresponding atomic nuclei. The result of an electron changing orbitals is light-emission/absorption[13].

[13] The very foundation of all electrical technologies.

The Higgs-boson i the latest addition to the SM. The Higgs-boson has mass and no spin. This makes it a scalar boson and not a gauge boson. The Higgs-boson in itself is a result of the Higgs-field, which gives mass to all other particles via the Brout-Englert-Higgs mechanism[22][27].
A schematic of how each particle in the SM interacts are seen in Fig. (1.3) where one can see that the Higgs particles interact with all leptons, quarks, and it interacts with itself and the W- and Z-boson. It does not interact with the gluon and photon, hence mass-less.

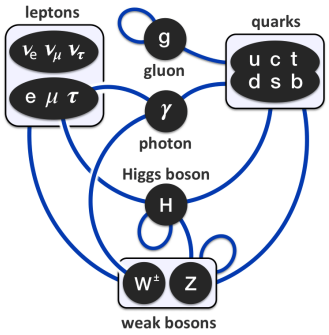


Figure 1.3: A schematic over the particle interactions in the Standard Model. Source of figure: [61, p. 9].

1.3 *Beyond the Standard Model (BSM)*

As mentioned, only three of the four fundamental interactions are explained through the SM. A "Graviton" with spin two is proposed, which could be a candidate for fusing quantum mechanics and gravitational theory into a quantum gravity theory. The hope is the "Theory of Everything, " a unified theory that can explain everything. [21]
The SM also fails to explain the asymmetry in the matter. Particles have antiparticles[14], and the question is why there is more matter than anti-matter[15]?
Then there are dark matter and dark energy, which is unaccounted for by the SM. Dark energy is a consequence of the always-increasing rate the universe is expanding. Linked to this, by observing the universe, one can calculate the visible mass' trajectories and through that, also calculate the mass needed for the movements observed. The amount of observed mass and needed mass does not match - the missing mass is called dark matter. Even though B-physics[16] are not directly related to these questions, there might be a relation with the solution of the BSM physics part of B-physics in the future.

[14] Note; some particles are their own antiparticles

[15] The simple question: "Why are we here?"

[16] Particle physics in relation to the B-meson.

2

B Physics and Lepton Universality

B physics has to do with the hadrons or, more specifically, mesons which contain a bottom quark. These mesons are called B-mesons and are seen in Tab. (2.1) and consist of a bottom anti-quark; $\overline{b}$ and either a down; u, down; d, strange; s or charm; c quark[1]. These B mesons have antiparticles composed of the anti-versions of their respective quark composition; see Tab. (2.1).

[1] Note that an $b\overline{b}$ is not a B meson but a bottomonium.

B-meson	Antiparticle	Charge	Isospin	Mass (MeV/c^2)	Mean lifetime ($\times 10^{-12}$ s)
Neutral $B^0 \mid d\overline{b}$	Anti-neutral $\overline{B}^0 \mid \overline{d}b$	0 e	$\frac{1}{2}$	5279.66 ± 0.12	1.519 ± 0.004
Charged $B^+ \mid u\overline{b}$	Anti-charged $B^- \mid \overline{u}b$	+1 e	$\frac{1}{2}$	5279.34 ± 0.12	1.638 ± 0.004
Strange $B_S^0 \mid s\overline{b}$	Anti-strange $\overline{B}_S^0 \mid \overline{s}b$	0 e	0	5366.92 ± 0.10	1.520 ± 0.005
Charmed $B_C^+ \mid c\overline{b}$	Anti-charmed $B_C^- \mid \overline{c}b$	+1 e	0	6274.47 ± 0.32	0.510 ± 0.009

Table 2.1: The four B mesons with their antiparticles and their properties. All mass and lifetime are from [72, pages: 53, 59, 68 and 70]

2.1 Lepton Flavour Universality

The Standard Model (SM) treats each of the lepton flavours equally, which means that the interaction between gauge bosons[2] and any of the three families/generations of leptons[3] must be identical and independent of the flavour as long as the equation used for calculating the interactions accounts for the difference in masses[4]. This treatment of leptons is called Lepton Flavour Universality (LFU). [42, p. 36-37] Branching ratios[5] denoted $\mathcal{B}$, can be theoretically calculated and experimentally measured to test whenever the theory matches reality. Specifically, rare decays for B-mesons are useful in detecting BSM physics. If there is a discrepancy with the predicted branching ratios for the leptons, we call it LFU Violation.

[2] With the exception of gluons.

[3] Electrons, muons or the tau; see Fig. (1.1).

[4] In other words: The fundamental interaction does not care which generation the leptons belong to.

[5] Branching ratio; $\mathcal{B}$ is defined as as follows: $\mathcal{B}(X \to a)$ means the fraction of X decays into a out of the total number of decays.

Feynman diagrams are a pictorial/graphical representation of the mathematical equations describing particles' interaction and behavior. It is mainly used in Quantum Field Theory (QFT), and the

diagrams are read the following way: input of the interaction to the left and output to the right with time flowing to the right. A Feynman diagram depicts the perturbation contribution to the probability amplitudes $\mathcal{A}$ associated with transitions, and according to superposition, the particles cannot choose which of the "diagrams" it follows; however, the total amplitude is a sum over all diagrams. From the amplitude we can get the transition probability: $P = |\mathcal{A}|^2$ and this can be calculated into decay-rates Γ which then are related to branching ratios by: $\mathcal{B}(X \to a) = \frac{\Gamma(X \to a)}{\Gamma_{\text{total}}}$.

$$R_H \equiv \frac{\int \frac{\mathrm{d}\Gamma(B \to H\mu^+\mu^-)}{\mathrm{d}q^2}\mathrm{d}q^2}{\int \frac{\mathrm{d}\Gamma(B \to He^+e^-)}{\mathrm{d}q^2}\mathrm{d}q^2} \tag{2.1}$$

A way to test LFU is by looking at the Flavour-Changing Neutral-Current (FCNC) process. This happens when a lepton changes its flavour without changing its electric charge. The SM theory states that FCNC only occurs in Feynman diagrammatic loops and not at the tree level.[6] The looping FCNC process is highly suppressed due to the Glashow–Iliopoulos–Maiani (GIM) mechanism[7] This means that transitions are rare and therefore are sensitive to new particles / BSM physics, and these new particles could increase or decrease the decay rate and, thus, the branching ratios.
The ratio used for testing changes in the decay rate is seen in Eq. (2.1) [29] where H is a hadron containing a strange quark. The decay rate, Γ, is then integrated over the di-lepton invariant mass, q^2.

[6] A basic example of a tree-level diagram is the electron-positron interaction.

[7] This mechanism is named after Sheldon Glashow, John Iliopoulos, and Luciano Maiani, who is credited for the theoretical prediction of the strange quark[24].

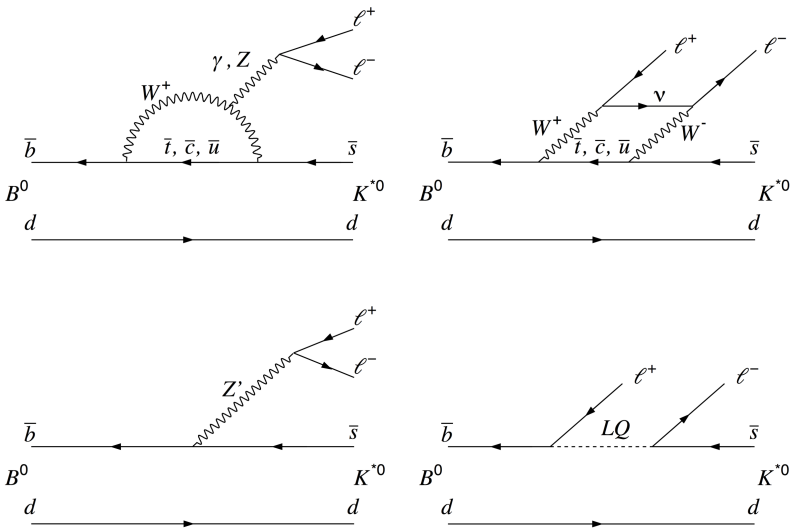


Figure 2.1: $B^0 \to K^{*0}\ell\ell$ decay Feynman diagrams. (Top left) are allowed by SM and are known as an electroweak penguin (penguin loop). (Top right) are allowed by the SM and are known as a box loop. (Bottom left) shows non-compliant SM tree-level diagram with proposed gauge boson: Z'. (Bottom right) shows non-compliant SM tree-level diagram with a leptoquark: LQ. Source of figure: [62, Fig. 1]

2.2 *The CKM Matrix*

FCNC is part of Flavour-Changing Weak Interactions, and these interactions have their properties described by the Cabibbo-Kobayashi-Maskawa (CKM) matrix, which is a generalization done by Kobayashi and Maskawa on the works of Cabibbo[33]. It comes from the Lagrangian seen in Eq. (2.2), which is the Yukawa interactions with the Higgs condensate (h.c.)[8][72, p. 261].

$$\mathcal{L}_Y = -Y^d_{ij}\overline{Q^I_{Li}}\phi d^I_{Rj} - Y^u_{ij}\overline{Q^I_L}i\epsilon\phi^* u^I_{Rj} + h.c. \tag{2.2}$$

The charge-current $W^\pm$ interactions then couple to the up and down quarks (u and d respectively) with couplings given by: $\frac{-g}{\sqrt{2}}(\overline{u_L}, \overline{c_L}, \overline{t_L})\gamma^u W^+_\mu V_{CKM}\begin{pmatrix} d_L \\ s_l \\ b_L \end{pmatrix} + h.c.$ where the CKM matrix is then seen in Eq. (2.3) which is a 3×3 unitary matrix. The CKM matrix is usually chosen in the basis of its four independent parameters: The three mixing Euler angles (θ_1, θ_2 and θ_3) and one phase angle; ϕ. The three mixing angles: (θ_1, θ_2, and θ_3) describe the amount of mixing between generations of quarks. The phase angle; ϕ is related to CP violation which is about the matter-antimatter asymmetry in the universe.

[8] The Lagrangian in itself is not the focus of the thesis and is therefore beyond the scope.

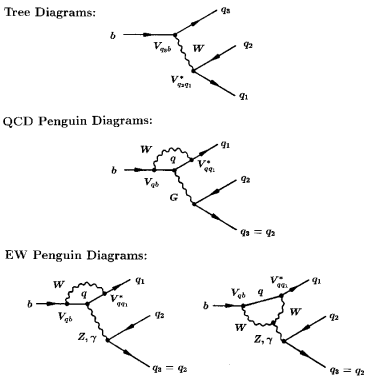


Figure 2.2: Some of the lowest order of bottom quark decays. Source of figure: [6, Fig. 22, p. 105]

$$V_{CKM} = V^u_L V^{d\dagger}_L = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \tag{2.3}$$

The CKM matrix describes the probability V_{ij} of a quark i transitioning to quark j; $i \to j$ which has another flavour[9]. The values of the CKM matrix are seen in Eq. (2.4)[72, p. 261-266]. The mixing of different quark families is called *CKM suppressed* due to the high transition rate inside each family[10].

[9] Hence the name: flavour-changing interactions.

[10] the diagonal

$$|V_{CKM}| = \begin{pmatrix} 0.97435 \pm 0.00016 & 0.22500 \pm 0.00067 & 0.00369 \pm 0.00011 \\ 0.22486 \pm 0.00067 & 0.97349 \pm 0.00016 & 0.04182^{+0.00085}_{-0.00074} \\ 0.00857^{+0.00020}_{-0.00018} & 0.04110^{+0.00083}_{-0.00072} & 0.999118^{+0.000031}_{-0.000036} \end{pmatrix} \tag{2.4}$$

The most dominant flavour-changing decay mode for bottom quarks is the $b \to cW^{*-}$ decay which is called tree or spectator decays, an example is seen in the top of Fig. (2.2). Whereas the decay $b \to u$ is suppressed relative to the transition to a bottom quark with $\left|\frac{V_{ub}}{V_{cb}}\right| \sim (0.1)^2$ which can be derived from the CKM matrix. As noted before, we are interested in Flavour-Changing Neutral-Current (FCNC) processes; however, the CKM does not allow it through tree-processes. Hence transition: $b \to s$ and $b \to d$ are only allowed through penguin diagrams(see Fig. (2.1)). All processes not a $b \to c$ decay are categorized as a *rare decay*. Note that later on,

for control channels on the R_H-ratio, J/ψ is a $b \to c$ decay, and this means the amount of decay is larger and therefore better statistics. [72, p. 908-914]

2.3 *The $B^0 \to K^{*0}\ell\ell$ Decay*

The "Rare Decays" group of "ATLAS B-Physics and Light States Working Group"[11] at ATLAS specifically looks at the $B^0 \to K^{*0}\ell\ell$ decay[12] which are a FCNC process and therefore a promising candidate to reveal BSM physics.

The Feynman diagrams are seen in Fig. (2.1) are the leading contributions from both SM and BSM. According to LFU the amplitudes from the diagrams must have equal contribution whenever $\ell\ell = \{e^+e^-, \mu^+\mu^-\}$.[13]
The B^0 has a long list of decays - in fact over 500 [72, pages: 1677-1682], and the specific decay to $K^{*0}\ell\ell$ has a branching ratio of $(1.03^{+0.19}_{-0.17}) \times 10^{-6}$ for $\ell\ell = e^+e^-$ [72, p. 1681, Γ_{542}] and $(9.4 \pm 0.5) \times 10^{-7}$ for $\ell\ell = \mu^+\mu^-$ [72, p. 1682, Γ_{543}]. [14] The choice of studying the $B^0 \to K^{*0}\ell\ell$ decay is a smart approach to investigating physics Beyond the Standard Model (BSM) with its low branching ratio. BSM reactions are expected to be rare/low amplitude; $\mathcal{A}_{BSM}$. If the SM reactions, denoted $\mathcal{A}_{SM}$ were large then $|\mathcal{A}_{BSM} + \mathcal{A}_{SM}|^2$ would reduce to $|\mathcal{A}_{SM}|^2$, menaing the the the BSM reaction would vanish. Therefore when $B^0 \to K^{*0}\ell\ell$ has a small branching ratio the $\mathcal{A}_{SM,B^0}$ is small hence the $\mathcal{A}_{BSM}$ becomes detectable[72, p. 908-914]. This introduces another challenge; lower statistics due to the rarity of the decay.

The K^{*0} has multiple decay-channel, and the one that is of specific interest is the decay into: $K^{*0} \to K^+\pi^-$. Throughout the analysis, the $K\pi$-pair poses a great challenge since the ATLAS detector can not distinguish between these two mesons, and this pair is used in the identification of the origin meson: a B^0 or a $\overline{B}^0$. The LHCb detector can, on the other hand, distinguish the Kaon from the pion, and this is the main reason for doing multiclass classification; simply because ATLAS can not detect the difference between Kaons and pions, and the hope is that a machine learning model can learn the differences.

This direct decay into a K^{*0} and two leptons are known as a *non-resonant* decay. Another category important for this thesis is the *resonant* decays where $B^0 \to K^{*0}(X \to \ell\ell)$ where X are mesons which then again decay into two leptons $X \to \ell\ell$. This could be the J/ψ, or the $\psi(2S)$ amongst others.

[11] Or just RK*-group for short.

[12] K^{*0} is short for the $K^*(892)^0$ decay which is an excitation of the neutral K^0 kaon and $\ell\ell = \{e^+e^-, \mu^+\mu^-\}$ in our case

[13] $\ell\ell = \tau^+\tau^-$ is omitted since handling this heavy lepton is hard and e, and μ gives enough statistics. Neutrinos are even worse than τ, so those are also omitted in this thesis.

[14] Why the studies of the $B^0 \to K^{*0}\ell\ell$ decay falls under the rare decays group.

2.4 The Double Ratio

Applying Eq. (2.1) on[15] $B^0 \rightarrow K^{*0}\ell\ell$ where $H = K^{*0}$ the ratio becomes $R_{K^{*0}}$ [16]. The SM predicts that $R^{SM}_{K^{*0}} \sim 0.92$ for $q^2 \in [0.045, 1.1]^{\text{GeV}^2/\text{c}^4}$[62] and $R^{SM}_{K^{*0}} \sim 1$ for $q^2 \in [1.1, 5.0]^{\text{GeV}^2/\text{c}^4}$[62]. The integration of the R_H is done over the limits: $q^2_{max} = [m(B^0) - m(K^{*0})]^2$ and $q^2_{min} = 4[m(\mu)]^2$[29] in the rest frame. From the definition of the branching ratio, we can rewrite Eq. (2.1) where the total decay rate gives the quotient of one and results in the rewritten form of $R_{K^{*0}}$ which is seen in Eq. (2.5).

[15] Throughout this thesis, the neutral B^0-meson will be mentioned a lot, and B^0 and B^0_d (because of the down-quark) will be used interchangeably.

[16] In literature this is also just known as R_{K^*}.

$$R_{K^{*0}} = \frac{\mathcal{B}(B^0 \rightarrow K^0\mu^+\mu^-)}{\mathcal{B}(B^0 \rightarrow K^0 e^+e^-)} \tag{2.5}$$

By multiplying $R_{K^{*0}}$ with the well measured $r_{J/\psi} = 1.043 \pm 0.006(\text{stat.}) \pm 0.045(\text{sys.})$[62] from the resonant $B^0 \rightarrow K^{*0}J/\psi(\rightarrow \ell\ell)$ which is statistically unity, we can rearrange $R_{K^{*0}}$ into $R_{K^{*0}} = R_{K^{*0}} \times 1 = R_{K^{*0}} \times r^{-1}_{J/\psi}$ which becomes what is seen in Eq. (2.6) where $\frac{N}{\varepsilon}(X)$ is the efficiency corrected yield[17].

[17] N is the measured yield and ε are the efficiency for the signal or control decays.

$$\begin{aligned} R_{K^{*0}} &= \frac{\mathcal{B}(B^0 \rightarrow K^{*0}\mu^+\mu^-)}{\mathcal{B}(B^0 \rightarrow K^{*0}e^+e^-)} \times \frac{\mathcal{B}(B^0 \rightarrow K^{*0}J/\psi(\rightarrow e^+e^-))}{\mathcal{B}(B^0 \rightarrow K^{*0}J/\psi(\rightarrow \mu^+\mu^-))} \\ &= \frac{\frac{N^{\mu\mu}_{sig}}{\varepsilon^{\mu\mu}_{sig}}}{\frac{N^{ee}_{sig}}{\varepsilon^{ee}_{sig}}} \times \frac{\frac{N^{ee}_{control}}{\varepsilon^{ee}_{control}}}{\frac{N^{\mu\mu}_{control}}{\varepsilon^{\mu\mu}_{control}}} = \left(\frac{\varepsilon^{ee}_{sig}}{N^{ee}_{sig}}\frac{N^{ee}_{control}}{\varepsilon^{ee}_{control}}\right) \times \left(\frac{N^{\mu\mu}_{sig}}{\varepsilon^{\mu\mu}_{sig}}\frac{\varepsilon^{\mu\mu}_{control}}{N^{\mu\mu}_{control}}\right) \end{aligned} \tag{2.6}$$

The reason for multiplying with $r_{J/\psi}$ is because this will suppress most of the systematic uncertainties, which $R_{K^{*0}}$ and $r_{J/\psi}$ have in common. The systematic uncertainties tend to cancel because the dividing efficiencies with each other will suppress efficiency uncertainties - this will make the difference in electron and moun production clearer. J/ψ has one of the largest branching ratios for the B^0 meson decay with $\mathcal{B}(B^0 \rightarrow K^{*0}J/\psi) = (1.27 \pm 0.05) \times 10^{-3}$ [72, p. 1679, Γ_{202}] and consistently decays into electron and muons with the branching ratios: $\mathcal{B}(J/\psi \rightarrow e^+e^-) = (5.971 \pm 0.032)\,\%$ [72, p. 1826, Γ_5] and $\mathcal{B}(J/\psi \rightarrow \mu^+\mu^-) = (5.961 \pm 0.033)\,\%$ [72, p. 1826, Γ_7] meaning that there will be high statistics for the $\mathcal{B}(J/\psi \rightarrow \ell\ell)$ decay making the double ratio a usefull tool for finding deviations in the SM through the B^0 meson decays.

2.5 Prior Measurements

As mentioned in the introduction, there have been multiple measurements of the R_H ratio, some of which are seen in Tab. (2.2) and a visualization of the same is seen in Fig. (1). Some of the mea-

surements are pretty far from the SM-predicted $R_K \sim 1$ in the $q^2 \in [1.1, 6]^{\text{Gev}^2/\text{c}^4}$ and others are pretty close but with greater statistically and systematic uncertainty. Until 2022 the variance of the results was quite large, and it was hard to determine if indeed there was LFU violation and therefore BSM physics.

Colab	Date	Measurement	q^2-range $(\text{GeV}^2/\text{c}^4)$	Deviation	Ref
BarBar	Apr. 2012	$R_K = 0.74^{+0.40}_{-0.31} \pm 0.06$ $R_{K^*_S} = 1.06^{+0.48}_{-0.33} \pm 0.08$	$[1.10, 8.12]$ $[0.10, 8.12]$	-	[34]
LHCb	Aug. 2017	$R_{K^{0*}} = 0.66^{+0.11}_{-0.07} \pm 0.03$ $R_{K^{0*}} = 0.69^{+0.11}_{-0.07} \pm 0.05$	$[0.045, 1.1]$ $[1.1, 6.0]$	$2.1 - 2.3\sigma$ $2.4 - 2.5\sigma$	[62]
LHCb	Mar. 2019	$R_{K^+} = 0.846^{+0.060+0.016}_{-0.054-0.014}$	$[1.1, 6.0]$	2.5σ	[60]
Belle	Apr. 2019	$R_{K^{0*}} = 0.46^{+0.55}_{-0.27} \pm 0.13$ $R_{K^{0*}} = 1.06^{+0.63}_{-0.38} \pm 0.14$	$[0.045, 1.1]$ $[1.1, 6.0]$	-	[71]
Belle	Aug. 2019	$R_{K^+} = 1.39^{+0.36}_{-0.33} \pm 0.02$ $R_{K^0_S} = 0.55^{+0.46}_{-0.34} \pm 0.01$	$[1.1, 6.0]$ $[1.1, 6.0]$	-	[9]
LHCb	Okt. 2021	$R_{K^0_S} = 0.66^{+0.20+0.02}_{-0.14-0.04}$ $R_{K^{*+}} = 0.70^{+0.18+0.03}_{-0.13-0.04}$	$[1.1, 6.0]$ $[0.045, 1.1]$	1.5σ 1.4σ	[63]
LHCb	Mar. 2021	$R_{K^+} = 0.846^{+0.042+0.013}_{-0.039-0.012}$	$[1.1, 6.0]$	3.1σ	[58]
LHCb	Dec. 2022	$R_{K^+} = 0.994^{+0.090+0.029}_{-0.082-0.027}$ $R_{K^+} = 0.949^{+0.042+0.022}_{-0.041-0.022}$ $R_{K^{*0}} = 0.927^{+0.093+0.036}_{-0.087-0.035}$ $R_{K^{*0}} = 1.027^{+0.072+0.027}_{-0.068-0.026}$	$[0.10, 1.1]$ $[1.1, 6.0]$ $[0.1, 1.1]$ $[1.1, 6.0]$	-	[11]

Table 2.2: A Timeline of a handful R_H measurements. In the *Measurements* columns: The first uncertainty is statistical, and the last is systematic uncertainty.

Then in December 2022, LHCb came with their latest result (see [11] for paper reference) with R_H ratios close to unity as predicted by SM. This new measurement of the $R_{K^{*0}}$-ratio could mean that the premise of looking into B^0 decays for physics Beyond the Standard Model was gone. As seen in Tab. (2.2), the different measurements do not all agree on the ratio; hence the validation of the LHCb results is needed with either a re-verification of the LHCb results, which indicates no Beyond the Standard Model physics or results which deviates from unity and hence indicating BSM physics.

3
ATLAS

3.1 CERN History

All historical references in section 3.1 are from CERNs own history website (with sub-pages): [64].

At the 1951 United Nations Educational, Scientific and Cultural Organization (UNESCO) [1] meeting in Paris, the decision to establish a European Council for Nuclear Research laid the foundation for the actual signed agreement that 11 countries would create what now is known as Conseil Européen pour la Recherche Nucléaire / eng: European Organization for Nuclear Research (CERN). The purpose of this organization was to do non-military research in nuclear and high-energy physics and make it publicly available.

[1] UNESCO is part of the United Nations (UN) and membership for UN automatically gives a membership for UNESCO.

The first draft of the CERN convention was completed in 1953, and the first foundation for the CERN laboratory complex where laid on the physical site on the 17th May 1954 in Geneva which were selected in 1952 to be the location of the laboratory due to Swiss neutral grounds in World War II. While the experimental part of CERN would be in Geneva, the theoretical physics would be done in Copenhagen at the Niels Bohr Institute (NBI).[2]

[2] Already from the start NBI has been part of the CERN history, and to this day there are still tight connections between NBI and CERN.

CERN were officially established in 1954 with 12 countries signing. The first accelerator, the 600 MeV synchrocyclotron, was built in 1957. This machine was used in Nuclear Research, and in 1959 the 28 GeV beam Proton Synchrotron (PS) was built and became the birth of high energy particle physics at CERN. The PS was used in 1965 for the discovery that electrons, neutrons, and protons all have anti-particles[43].

Until 1971, the proton synchrotron created a beam that collided with stationary targets; however, this changed with the Intersecting Storage Rings (ISR). The idea was to feed the beams into two storage rings and then make them collide, thus achieving higher energies.

In 1976 the Super Proton Synchrotron (SPS)[69] were finished and were operational up to 450 GeV. It is a 7 km ring, and this new

collider was the main accelerator at CERN for many years. It has contributed to many findings and, probably most noteworthy, the discovery of the W and Z bosons [18], which in 1986 awarded Carlo Rubbia and Simon van der Meer the Nobel Prize in physics. This happened after the Super Proton Synchrotron in 1979 was converted into a proton–antiproton collider. The collider technology from the Super Proton Synchrotron was paving the road for the later Large Hadron Collider (LHC) [65][3].

[3] About the word: "technology" - a lot of technologies origin at CERN. We use the internet all the time today, and the first webpage was written in 1990 and hosted on a CERN server at info.cern.ch

In January 1997, the Compact Muon Solenoid (CMS) and A Toroidal LHC Apparatus (ATLAS) were approved, and the idea behind them was general-purpose experiments. Later in February of the same year, A Large Ion Collider Experiment (ALICE) were approved, which were intended to study quark-gluon plasma. In September 1998, the LHC-beauty (LHCb) was approved, designed to study the matter-antimatter imbalance.

An important date for CERN history is the 10^{th} of September 2008 at 10.28 am. This was the first time where the particles circulated in the Large Hadron Collider, which is 27 km in circumference, and this event signaled a new era for particle physics.

The LHC has been operating and measuring collisions for chunks of time, now known as *Runs*. Run 1 was from 2009 to 2013 with collision energies up to 7 TeV [4]. In the period 2013 to 2015, there were large upgrades to LHC, and in 2015 to 2018, Run 2 was measuring collisions with 13 TeV [5]energies. From 2018 to 2022, there were big upgrades to Large Hadron Collider (LHC) named the High Luminosity[6] Large Hadron Collider project [28]. This intends to increase the luminosity by a factor of ten. In 2022 Run 3 started and is still running.

[4] Each beam with 3.5 TeV.

[5] Each beam with 6.5 TeV.

[6] Luminosity is a word used for collision rate.

3.2 *From Accelerator to the ATLAS Detector*

All technical numbers in section 3.2 for the following accelerators are found in the references: Proton Synchrotron[67], LINAC4/2[36], LINAC3[35], LHC[65], Super Proton Synchrotron[69], LEIR[66] and PSB[68]. For further in-depth information on all present and past detectors at CERN, see [2].

Before collisions are measured at the ATLAS detector, the particles have been underway through different systems. There are two starting points for the LHC. One is the Linear Accelerator 3 (LINAC3)[35] which provides lead ($_{82}Pb$) ions and sends them into the Low Energy Ion Ring (LEIR)[66]. The lead Ions are accelerated using conducting cavities where electric fields creates confined waves which in turn creates a potential that accelerates the ions. Using magnets, the beams can be focused. The LEIR strips the lead ions from remaining electrons so only the nuclei remain before entering the Proton Synchrotron (PS).

Another entry to the Proton Synchrotron (PS)[67] are through Lin-

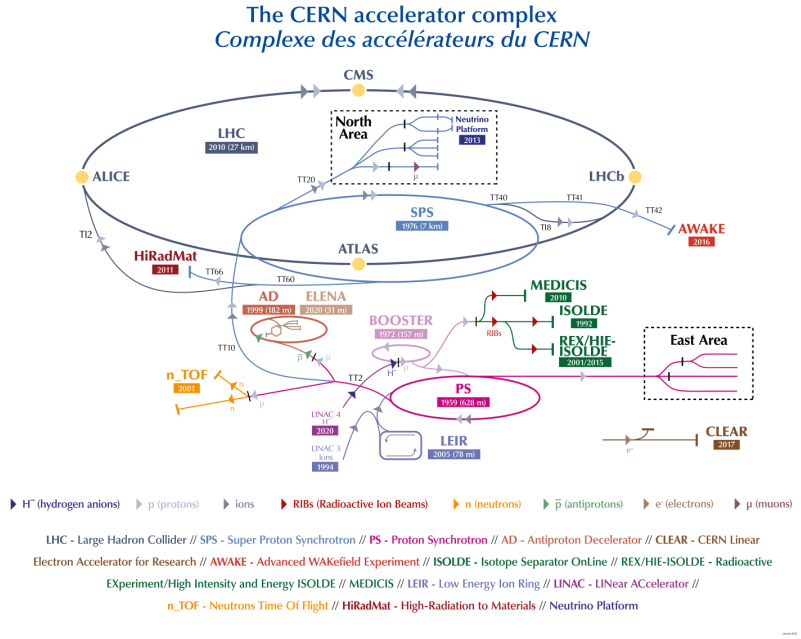


Figure 3.1: The CERN complex as of 2022. All different components of the accelerator complex are seen in the figure legend below the figure. Most notable for this thesis are the LHC and the ATLAS detector. Source of figure: [37]

ear Accelerator 4 (LINAC4)[36] [7], which accelerates hydrogen ($_{00}H$) which are negatively ionized. LINAC4 is the proton provider to LHC and uses, like LINAC3, radio frequency cavities to accelerate the ions. In multiple stages the ions are accelerated from 3 MeV $\rightarrow$ 50 MeV $\rightarrow$ 100 MeV to finally 160 MeV. The ions have their electrons removed before entering the Proton Synchrotron Booster (PSB)[68]. During the upgrade in 2019, Linear Accelerator 4 (LINAC4) replaced the older version 2, which had been operating for over 40 years. The older version provided beams at energies equal to 50 MeV. With LINAC2 the booster accelerated the protons to 1.4 GeV before injecting them into the PS whereas LINAC4 boosts the proton beam to 2.0 GeV.[8]

[7] Operational since 2020

[8] Note that LINAC2 were operated under Run 2.

The Proton Synchrotron (PS) has a circumference of 628 meters with 100 dipole magnets and 277 electromagnets to bend the beam. With Radiofrequency cavities, the beams are boosted to 26 GeV[9]. The beam is then lead into the Super Proton Synchrotron (SPS), which are at a size of 7 km circumference with boosting capabilities up to 450 GeV[10].

[9] In addition to protons, it has accelerated alpha particles (helium nuclei), oxygen, sulfur, argon, xenon, and lead nuclei, electrons, positrons, and antiprotons.

[10] 1317 electromagnets and 744 dipole magnets are used to bend the beam.

Finally, the beam is injected into LHC, which is the largest accelerator at CERN with 27 km circumference. It has two beams moving in opposite directions, and each beam is then accelerated up to 6.5 TeV [11]. The LHC is exceptional in that it is a state-of-art instrument. LHC is one of the world's largest vacuum systems and with its three vacuum system parts; (1) The beam vacuum, which is at 10^{13} atm due to avoidance of gas molecule colliding with acceler-

[11] A design maximum of 7.0 TeV.

ated particles. (2) The vacuum system for the cryomagnets, and (3) the vacuum system for helium distribution. The cryogenic system is kept at 1.9 K for super-fluid helium temperature. There are 9593 superconducting magnets bending the beams into a focused beam [12]. The LHC also consists of 2×8 superconducting cavities with multiple functions. One is to accelerate the beam using radio frequency, which delivers 2 MV at 400 Hz. The other function is to pack the beam into 2808 proton bunches with consist of 1.1×10^{11} protons. These bunches are created due to the pulsing nature of the radio frequency. It tightens the bunches so that higher luminosity is reached, maximizing the number of collisions.

[12] A focused beam is indeed needed when one wants to make protons collide.

The luminosity are defined as $\frac{dR}{dt} = \mathcal{L}\sigma_p$ [13] where LHC has $\mathcal{L} = 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$. The integration becomes the collision which is a cross-section area measured in inverse femtobarns, $1\,\mathrm{fb}^{-1} = 10^{39}\,\mathrm{cm}^{-2}$. The speed of the protons are 0.999999991 c [14] and the particles have traveled the 27 km tube 11245 pr second.

[13] Where $\frac{dR}{dt}$ are the events pr second, σ_p are the cross-section and $\mathcal{L}$ are the luminosity.

[14] c is understood as the speed of light at 299792458 m/s [45].

The final destination for the beams is in the detector of interest, namely the A Toroidal LHC Apparatus (ATLAS) detector, where the number of collisions is up to 1 billion collisions per second.

3.3 *The ATLAS Detector*

The ATLAS detector is the largest[15] particle detector in the world with its 46 m long / 25 m diameter cylindrical shape seen in Fig. (3.2). It is situated 100 m underground[16] and weights around 7000 tonnes. The detector is constructed as multiple layers, each contributing to the detection of the results of collisions, namely showers of particles, their individual trajectory, momentum, and energy. Over 1 billion collisions happen inside the detector, and only a small fraction[17] has the measured quality to be stored in the CERN network.

The reference: [16, Main page and its subpages.] are used for The ATLAS Detector section 3.3 and the subsections: 3.3.1, 3.3.2, 3.3.3, 3.3.4, , 3.3.7.
Additionally [70, Main page and its subpages.] are used for reference for subsections: 3.3.7 and 3.3.5 in section 3.3.

[15] volume-wise

[16] Which utilises the earth to shield from radiation

[17] One in a million.

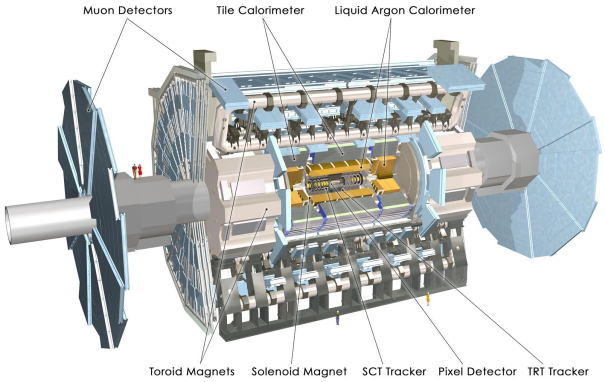


Figure 3.2: The ATLAS detector in 2008 for Run 2 - a computer-generated image with labels. Source of figure: [56]

3.3.1 *The Inner Detector*

The inner detector is the first part of the ATLAS detector that interacts with particle-collision debris. It consists of three components: Pixel Detector (PD), Semiconductor Tracker (SCT), and Transition Radiation Tracker (TRT). The inner detector measures the electrically charged particles' direction, momentum, and charge.

Moving from the collision point and outward, the first is the Pixel Detector (PD). The PD works by particles going through the detector and leaves energy in the four layers of silicon pixel where the pixel sizes are $50 \times 400\,\mu\text{m}^2$ for external layers and $50 \times 250\,\mu\text{m}^2$ for innermost layer yielding a resolution up to $10\,\mu\text{m}^2$. The detector has 92 million pixels which is an area of $\sim 1.9\,\text{m}^2$, which makes it capable of determining the origin and momentum of the detected particle.

[18] or "tubes".

Moving on to the next part of the inner detector, which is the SCT which aims to reconstruct tracks of the charged particles from the collisions. This detector has 4088 modules of $\sim$ 6 million readout strips of silicon sensors. These are organized in layers, so particles must go through at least four readout strips. The precision is up to $25\,\mu\text{m}^2$.

[19] Transition radiation is when an object with a charge moves at a constant speed in a non-uniform or non-stationary medium or near such. An example would be a charged particle moving between two different mediums, giving off radiation.

The last part of the inner detector is the TRT which is made of 300000 straws[18] each 4 mm in diameter with a $30\,\mu\text{m}^2$ gold-plated wolfram ($_{74}W$) wire in the center surrounded by gas. When charged particles go through the straw, they ionize the gas and create an electrical signal used for re-creating the tracks. In addition, it also gives information on the particle type obtained from the transition radiation effect[19].

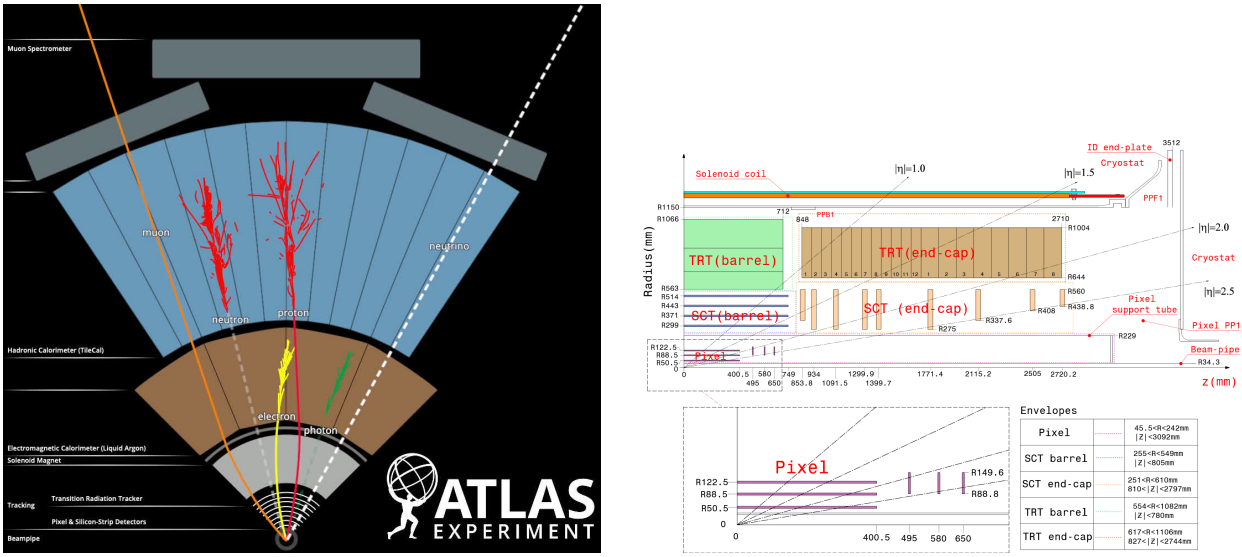
3.3.2 *The Magnet System*

The ATLAS detector has a superconducting[20] magnet system which aims to measure the momentum and charge of charged particles. This is done by bending the trajectories of the particle residuals from the collisions. The magnet consists of the Central Solenoid Magnet, which encapsulates the inner detector. This magnet aims to measure momentum and has the dimensions: 5.3 m long and 2.4 m in diameter weighing over 5 tonnes The solenoid gives a 2 T magnetic field storing 38 MJ which is achieved by 9 km niobium-titanium (Nb-Ti) superconducting wire.

[20] Operating temperature is 4.5 K.

The Toroid Magnet System consists of one big[21] barrel toroid which surrounds the central part of the detector. It is made of 8 coils providing in total up to 4 T with its over 56 km superconducting

[21] The largest toroidal magnet ever constructed with a length of 25.3 m.



(a) (b)

Figure 3.3: Figure (a): A cross-section of the ATLAS detector where the interaction of different particles are seen. Source of figure: [46].
Figure (b): A schematic of the ATLAS Inner Detector where each element are located compared to the pseudorapidity. Source of figure: [12].

wire, which aims to measure muon momentum. The two end-cap toroidal magnets ensure that muons leaving the detector close to the beamline are also measured.

3.3.3 *The Calorimeters*

The calorimeters are designed such that particles[22] that pass through, deposits their energy in the calorimeter (seen in Fig. (3.3)(a)). There are two calorimeters in ATLAS. The innermost is the Liquid Argon (LAr) Calorimeter[23] which measures the energy of electrons, photons, and hadrons. This is done by multiple metal[24] layers of that absorb and convert particles into showers of lower energy-particles. These showers ionize the liquid argon (temp: $-184\,^{\circ}\mathrm{C}$) and give off an electrical current. Due to the honeycomb structure of the inner calorimeter, almost no particles escape. Therefore, retracing energies from the shower makes a reconstruction of the original incoming particles' energy possible. The inner calorimeter is specialized for electron and photon measurements.

Surrounding the LAr calorimeter is the Tile Calorimeter[25] which measures hadronic particle energies which do not get stopped in the LAr calorimeter. Using layers of steel and sparkling plastic tiles, the steel layers create showers of particles upon interacting with incoming particles, and the plastic tiles produce photons which are turned into electric currents. These currents are proportional to the original particles' energy. There are 420000 plastic tiles, and the tile calorimeter weighs 2900 tonnes, making it the heaviest part of *ATLAS*. As seen in Fig. (3.3)(a), muons and neutrinos are not recorded in the LAr Calorimeter or the tile Calorimeter.

[22] Not all types.

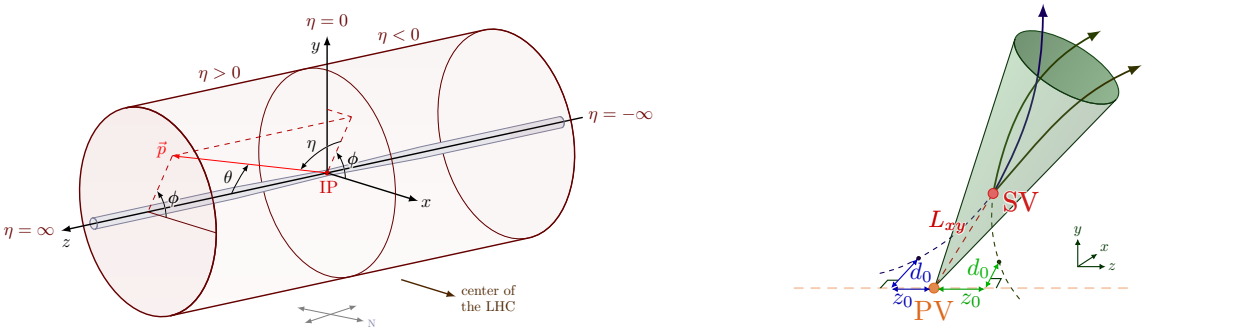
[23] also known as the Electromagnetic Calorimeter (ECAL) or EM Calorimeter.

[24] Wolfram ($_{74}W$), copper ($_{29}Cu$) or lead ($_{82}Pb$).

[25] Also known as the Hadronic Calorimeter (HCAL).

3.3.4 *Muon Spectrometer*

To measure muons, the ATLAS detector has a muon spectrometer that measures the momentum of muons since muons typically do not interact with the Inner detector or the Calorimeter. The muon spectrometer consists of 4000 individual muon champers using four different technologies: *Thin Gap Chambers*, which are used for triggering and measuring 2nd coordinate at the end of the detector. *Resistive Plate Chambers*, which also is used for triggering and 2nd coordinate measurements, is used in the detector's central region. *Monitored Drift Tubes*, which measures curved tracks and lastly, *Cathode Strip Chambers*, which measures coordinates at the end of the detector.



(a) (b)

Figure 3.4: Figure (a): The ATLAS coordinate system where the beamlines also top of the z-axis and with the origin as the collision center. Source of figure: [53], the figure is edited such the compass fits the ATLAS detector.
Figure (b): B jet from pp-collision at the primary vertex (PV) in the ATLAS detector. SV is the secondary vertex where the B meson decays. L_{xy} are the distance between PV and SV. d_0 and z_0 are the distance between PV and SV projected onto the x-axis and z-axis, respectively. Source of figure: [52].

3.3.5 *The ATLAS Coordinate System*

The ATLAS coordinate system is the same as for the CMS detector with the origin at the point of collisions. Following Fig. (3.4)(a), the z-axis is the beamline, and the x-y plane is a cross-section cut in the detector with the y-axis pointing upwards. An interesting quantity on Fig. (3.4)(a) is the pseudorapidity; η defined in Eq. (3.1). The reason η is used instead of θ is that the difference of pseudorapidity; $\Delta\eta$ is Lorenz invariant[26].

$$\eta = -\ln\left[\tan((theta2)\right] \tag{3.1}$$

This quantity is then used in other factors such as in the Angular Distance between particles, which are defined as $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$ which also are Lorentz Invariant. Another noteworthy quantity is the radiation length X_0, which is defined in Eq. (3.2) defined according with CERN technical papers: [26].

$$X_0 = \frac{716.4}{Z(1+Z)\ln(287\cdot Z^{-1/2})}\,\mathrm{g/cm^3} \tag{3.2}$$

The radiation length; X_0 is the mean length in which an electron's

[26] The quantity is invariant or unchanged regardless of the observer's reference frame.

energy is reduced by a factor of $1/e$ [27]. With these state-of-the-art detectors, there are still sections of the detector that have better readout than other areas. These areas can be categorized as functions of pseudorapidity which are seen in Fig. (3.3)(b). Here it is seen that inside the Inner Detector for the best readout, the values of pseudorapidity would be: $|\eta| <= 2.5$. Another important factor is the supporting material inside the ATLAS detector. The amount of support structure can be quantized by the amount of radiation length of the material. Where a peak in the material means the signal is probably more distorted in this region. An important example of this is seen in Fig. (3.5) in the area $1.5 < X_0 < 1.75$ and especially for $|\eta| > 3.5$.

[27] The physics behind is when an electron is near an atom, it will be affected by the electromagnetic field of the atom. This interaction will produce photons; hence the free electron will loos energy. This phenomenon is called **Bremsstrahlung**.

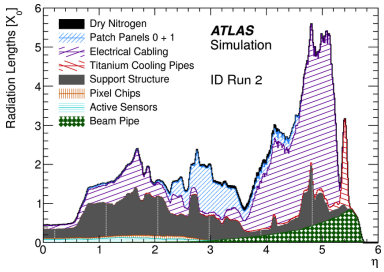


Figure 3.5: A schematic of the Run2 material budget of the ATLAS detector. Source of figure: [47, figure 3, p.2].

3.3.6 *The Trigger System*

The LHC provides ATLAS with collisions that would take up to 60 million megabytes per second, which is impossible to store. One way is to have a trigger system that sorts the collisions before they are stored in CERN servers. The triggers take event data at $\sim$ 40 MHz to $\sim$ 100 kHz. The trigger system is made of hardware and software triggers; the first trigger; is Level-1 Trigger (L1), which is purely hardware based. The L1 trigger uses data-event from the detector such as total energy in the Calorimeter, the multiplicity of certain objects above a threshold, or topological conditions[28]. The L1 trigger takes the data direct from the detector at $\sim$ 40 MHz and ramps it down to $\sim$ 100 kHz within a latency of 2.5 μs. In this short period, 2.5 μs - the event data is in buffer storage, and if the event passes the L1 trigger, it gets sent to the second trigger. The L1 trigger also identifies the Region-of-Interest (RoIs) in η and ϕ, which also are given to the second trigger.

[28] Could be invariant mass or angular distance.

The second trigger stage, the High-Level Trigger (HLT) or Level-2 Trigger (L2), is software based. The L2 trigger uses ultra-fast algorithms which do an early accept/reject of the data. Then more CPU-intensive algorithms take the accepted data and do a second filtering. These algorithms also do a basic reconstruction of tracks and associate them with energies from the calorimeter. These algorithms are executed on a computing farm made of $\sim$ 40000 selection applications, also known as Processing Units (PUs), which on average take an accept/reject selection within $\sim$ 500 ms. The number of events outputted by the L2 trigger is around 1000 events per second. The software on the L2 trigger is based on the Athena framework[10], which is developed and maintained by the CERN collaboration. When the trigger system fully accepts an event, all associated data is collected and the event is passed on for proper

reconstruction.

The trigger settings used in this thesis are:
HLT_e5_lhvloose_nodo_bBeexM6000t and
HLT_2e5_lhvloose_nodo_bBeexM6000t[39].
A breakdown of the triggers: *HLT* means it is on the HLT/L2 trigger, *e5* and *2e5* means that the trigger requires at least one and two electron(s) with transverse energy larger than $5\,\mathrm{GeV}/c^2$ receptively. *lhvloose* means the identification and isolations criteria for the electrons are "loose". *nodo* means there is no d_0 requirement in the trigger (see Fig. (3.6)). Lastly, *bBeexM6000t* tells the trigger that the event needs to have 2 electrons with an invariant mass lower than $6\,\mathrm{GeV}/c^2$.

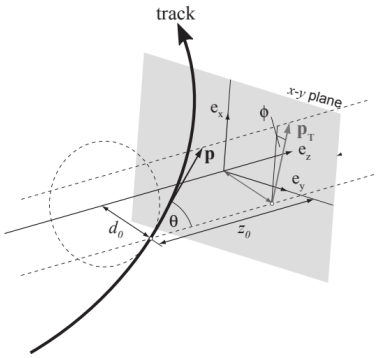


Figure 3.6: Track coordinates ATLAS. Source of figure: [70]

3.3.7 *Track Reconstruction*

Track reconstruction is an essential part of the detector system. Since the proton-proton collisions are not one-at-a-time but come in bunches, multiple collisions happen at each bunch crossing, i.e., pile-up. After proton-proton collisions, hadrons are formed by the quarks from the hadronization process and these hadron showers form a cone called a jet; see Fig. (3.4)(b).
Track reconstruction is not only used for the reconstruction of charged particles. However, it is part of almost all other parts of the reconstruction process: Reconstruct leptons, finding primary and secondary vertices, jet flavour tagging[29], and pile-up removal for when jets are overlapping resulting in smeared signal. In Eq. (3.3), the parameter tuple is the primary tracking parameter. For reference, see Fig. (3.6); d_0 and z_0 are the distance from the primary vertex to the closest point of the track. ϕ and the θ of the track momentum, where the primary vertex is the reference frame. q is the charge, and p is the magnitude of the momentum of the reconstructed track.

$$\left(d_0, z_0, \phi, \theta, \frac{q}{p}\right) \tag{3.3}$$

The tracking system consists of two *paths*; the first is an inside-out, which starts with nearby signals in the PD and SCT is converted to clusters which again are converted to $3D$ space-points. These $3D$ space points are then used to form seeds containing 3 points that can create a track-line to the vertex of interest. Iteratively, the seeds are being expanded into trajectories using an adaptive Kalman filter that finds adjacent clusters and smooths the trajectory. This is not the most precise way. However, it is fast, and this method creates track candidates. Further refinement of the track candidates is done with a global χ^2-fit followed by a Gaussian-sum Filter (GSF)[30] After this, the other *path* is taken: Outside-in. This is to increase

[29] Finding which quarks are in the jet.

[30] GSF is based on Kalman filters takes the non-linear effect of bremsstrahlung into account.

acceptance from particles produced in the Region-of-Interest (RoI): $|\eta| < 2.46$ [1]. Here track candidates start from the TRT and cross-check to be in the RoI. Then the same is done as in the other *path*; seed-expansion, adaptive Kalman, and χ^2-fits.

3.3.8 Seed-Cluster Reconstruction

The reconstruction of the electrons is a two-step process. The electron deposits their energy in the EM calorimeter, which can be divided into a grid of 200×256 elements (towers) of size $\Delta\eta \times \Delta\phi == 0.0250.025$[1]. The idea is that each tower consists of three layers and a presampler[31] and the energy deposited in each layer is summed to total tower energy and is calibrated to EM-scale. An algorithm finds each seed cluster[32] by a sliding window of $\Delta\eta \times \Delta\phi = 3 \times 5$ by scanning over adjacent towers. The algorithm stops when all towers have been searched. If two seed clusters are present[33] the alorithm discards the one with lowest transverse energy; E_T. However, if the energy is within 10%, the tower with the highest energy in the original tower is kept. This is in place to ensure that each electron candidate is reconstructed using only a single set of energy deposits in the EM Calorimeter; hence it will improve the ability to measure the momentum and direction of the electron accurately. These seed clusters are then used to determine the electron candidates.

[31] The pre-sampler is a small detector in before the calorimeter which corrects for lost energy due to non-detector material interactions.

[32] A seed cluster is a region in the EM calorimeter where there is the energy deposited above a threshold.

[33] overlap of $\Delta\eta \times \Delta\phi = 5 \times 9$.

3.3.9 Elecron Identification

Identifying electrons[34] is essential for the analysis. A combination of the energy deposited in the EM calorimeter and the track reconstruction is used to identify if the electrons come from the primary vertex (PV) or secondary vertex (SV) or are unrelated to the decay of interest and hence are discarded[1]. This non-related electron that needs to be removed could be an electron which, due to bremsstrahlung, emits a photon, creating an electron-proton pair. This will result in a shower of electrons, which needs to be removed since it distorts the signal. The track-cluster matching is done by a likelihood-based algorithm (likelihood function: Eq. (3.4)), which can be calculated for a signal; S or background; B prompt electrons[35] by using their respective probability density functions for each variable, P_i - extracted from simulated distributions.

[34] or positrons.

[35] Promt electron are electrons directly from pp-collisions.

$$L_{S(B)}(\mathbf{x}) = \prod_{i=1}^{n} P_{S(B),i}(x_i) \tag{3.4}$$

From the likelihood, the inverse[36] distriminant for each event can be calculated: $d'_L = -15^{-1} \ln(d_L^{-1} - 1)$ where $d_L \frac{L_S}{L_S + L_B}$ is the dis-

[36] The discriminant has a sharp peak at 0 and 1. Hence the inverse is a better choice for computational stability.

criminant. The discriminant can be used to define cuts: VeryLoose, Loose, Medium, and Tight, which are different values for how strict the selection should be. The idea behind the LLH is that electrons will deposit most of their energy at the front of the calorimeter. Some other particles, like pions, tend to create broader electron showers, extending deeper into the calorimeter. Then the LLH algorithm uses this information to assign a probability that each track is associated with each seed cluster in the EM Calorimeter. The source of the misidentification of electrons could be multiple electrons close to each other, so there will be a track-cluster mismatch. Hence the algorithm also looks into the isolation[37] of electrons and adds a threshold for separation between electrons. The efficiency of the electron reconstruction is important to this thesis due to the $R_{K^{*0}}$-ratio where the *efficiency* corrected yield is used.

[37] Isolation means that the electron is separated from other particles measured in momenta and positions.

3.4 Efficiencies

This detailed review of the ATLAS detector and its components is not only for understanding where the data comes from; it is also essential for estimating the efficiencies of various stages of the detector. As the focus of this thesis is on the extraction of the electron signal yield, the efficiency is the other part of the $R_{K^{*0}}$-ratio (Eq. (2.6)). The total efficiency is seen in Eq. (3.5), which is the equation used to estimate the efficiencies of LHCb 2022 $R_{K^{*0}}$-ratio measurements[44].

$$\varepsilon_{tot} = \varepsilon_{geo} \times (\varepsilon_{MVA} \times \varepsilon_{Pre-select} \times \varepsilon_{Trig} \times \varepsilon_{PID}) \tag{3.5}$$

The detector-dependent[38] efficiencies are ε_{geo}, $\varepsilon_{Pre-select}$, ε_{Trig} and ε_{PID}.

[38] or just a part of the above section on the ATLAS detector.

ε_{geo} is the fraction of signals generated within geometric acceptance in the detector; η, ϕ, etc. $\varepsilon_{Pre-select}$ the fraction of signals which passed various cuts based on some requirements which could be variables based on particle identification, vertex quality, and kinematics, etc.
ε_{Trig} is the efficiency on the trigger system; namely the high-level trigger and the low level trigger.
ε_{PID} is the fraction of signals correctly identified; electron or muons.
The ε_{MVA} is the efficiency of the Machine Learning selectron of B^0, $\overline{B}^0$, and background.

3.5 ATLAS Data

As mentioned in subsection 3.3.6, the data are stored, and the road to storage is a bit different depending on the origin of the data

if it is from Monte Carlo simulations, denoted *MC* or actual data obtained from the ATLAS detector.
Starting with the MC path as seen in Fig. (3.7), the *generation* is the first step. This step uses a range of different MC models that tries to encapsulate the properties of the particles. Usually, multiple models are used in the generation process and are merged to get as close as possible to the complex structures from Quantum Chromodynamics. The *Simulation* step is where detector-interactions are simulated, and the *digitization* step is the detector output. These steps are equivalent to the *collision/trigger* steps of the data path described earlier in this thesis.

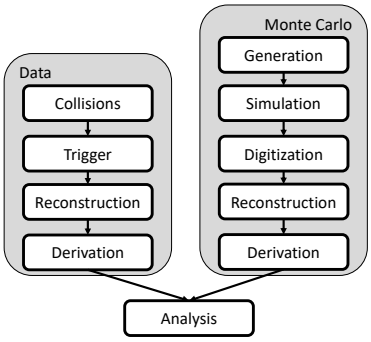


Figure 3.7: A diagram over the data/MC path until it is ready for analysis. Source of figure inspiration: [7, p. 3]

Data after the *reconstruction* step are stored in *.root*[39]. The raw reconstructed data are stored in Analysis Object Data (AOD) and extended with metadata and additional information to eXtended Analysis Object Data (xAOD). *Athena*[10][40] are then used to take the raw data and cut it down into smaller bits such that the different groups at ATLAS don't have to use extra computer power to process data of no interest. This is called the *derivation* step. The size of xAOD is about petabytes, and it can't be read into memory on regular computers as it is. The following is done in the derivation step to create Derived Analysis Object Data (DxAOD/DAOD) based on criteria made by the ATLAS research groups who needs the DAODs: *Skimming*, *slimming*, *thinning* and *argumentation*. *Skimming* is the action of removing whole events. *Thinning* removes objects inside the events, and *slimming* removes variables inside objects. The last is *argumentation*, which creates new information based on existing knowledge. The DAODs are of terabyte size.

Further, each subgroup *slims*, *skims* and *thins* the DOAD into n-tuples[41] which are meant for high-performance analysis where the data are used with higher frequency. An example of this could be a machine learning analysis, such as the analysis of this thesis.

[39] A storage framework which is developed by CERN. It has a hierarchical structure like a tree in which variables as branches and entries in the branches are single events. Metadata can also be stored in this format.

[40] A modular analysis framework developed by CERN

[41] N-tuples are of megabyte/gigabyte size.

4
Machine Learning

The main component of this thesis is the implementation of Machine Learning (ML) to filter B^0 and $\overline{B}^0$ from the background which is highly nontrivial, and before heading into the analysis, review of the ML theory is needed.

Machine Learning is the science of programming computers such that they learn from data and is a branch of Artificial Intelligence (AI). The spam filter was one of the first implementations of ML the broad public met. The spam filter has learned from thousands of flagged spam emails from users such that nowadays, spam emails are automatically filtered. On the other hand, if the spam filters were hard-coded[1], every new attempt to make spam emails must be caught by creating new rules/if-statement[2]. Luckily, this is done automatically with ML.

[1] Lines after lines of *if* statements.

[2] And the spam filter developers have to send out update after update for all eternity.

ML can be divided into multiple categories such as **Supervised Learning** where the programmer labels the training data. These labels can either be classes for a *classification* problem or the labels are continuous, and the problem is *regression*. There are no labels in **Unsupervised Learning**, and the algorithm has the learn with no labeled data. A widespread algorithm in the category is *clustering* algorithms. **Semisupervised Learning** is an approach where it is too costly to label all data; hence, a stacked model of both a supervised and unsupervised algorithm is usually used. **Reinforcement Learning** is a bit different from the other since here, the programmer defines an agent who operates inside its world with a given rule set and a score function. Then reinforcement learning agent tries to optimize this score function within its programmed world.

There are many advantages to using ML; however, it is not some magic that can do everything. The programmer has to be aware of the multiple problems with ML and how to prevent these problems from affecting the end result. Some of these problems could be:

- Not enough data

- Data that does not represent/correlate with the objective
- Data of insufficient quality
- Over-fitting to training data
- Under-fitting to training data

This thesis uses mainly supervised ML, and therefore, from hereon, this chapter will use the notation of reference [51] to create a more formal language around ML. Let T be the task of the ML algorithm; the goal is to make the computer learn the mapping: $f : (\mathbf{x} \in \mathbb{R}^D) \rightarrow (\mathbf{y} \in \mathbb{Y})$ where $\mathbf{x}$ is the features[3], D is the dimensionality of the input and the number of features. $\mathbf{y}$ is the output vector, called labels or targets. The training set; $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N$ where N is the sample size.

[3] Some also use the terms: *predictors*.

4.1 Gradient Boosted Decision Trees

The main ML algorithm used in this thesis is the Light Gradient Boosting Machine (LightGBM) package managed by Microsoft (git repository: [13]) original paper by Guolin et al. in 2017 [32].

4.1.1 Decision Trees

LightGBM is an ensemble method that uses decision trees as weak learners and combines them into a strong learner. A decision tree starts at the root node as seen in Fig. (4.1) and partitions the training set into smaller and smaller sets until an approximated target value is reached.

The algorithm goes as follows: Let $\mathcal{D}_i = \{(\mathbf{x}_n, \mathbf{y}_n) \in N_i\}$ be the set that has reached node i.
Let $\mathcal{D}_i^L = \{(\mathbf{x}_n, \mathbf{y}_n) \in N_i | y_{n,j} \leq t\}$ and $\mathcal{D}_i^R = \{(\mathbf{x}_n, \mathbf{y}_n) \in N_i | y_{n,j} > t\}$ be the partitioned sets from $\mathcal{D}_i$ dependent threshold t_j for feature j.[4] A greedy method is used to find the optimal pair of (j, t_j) by optimizing Eq. (4.1) where $\mathcal{T}_j$ is the set of possible thresholds for feature j.

$$(j_i, t_i) = \underset{\{j \in 1,\ldots,D\}}{\operatorname{argmin}} \left[\min_{t \in \mathcal{T}} \left[\frac{|\mathcal{D}_i^L(j,t)|}{|\mathcal{D}|} c(\mathcal{D}_i^L(j,t) + \frac{|\mathcal{D}_i^R(j,t)|}{|\mathcal{D}|} c(\mathcal{D}_i^R(j,t) \right] \right] \tag{4.1}$$

At each node i, there is a region space defined by $R_i = \{\mathbf{x} | x_i \leq/> t_i, \ldots\}$[5]. By partitioning the 2D-input space, this approach creates a M-dimensional output space dependent on the number of leaf nodes. This results in a piece-wise linear approximation to the training set labels and the mapping mentioned

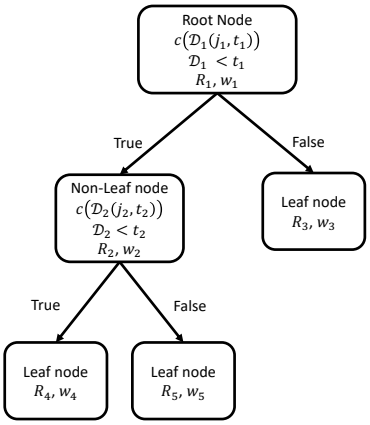


Figure 4.1: Example of a decision tree in training; $\mathcal{D}$ is input training set, $c(\mathbf{X})$ is the cost function, t_i is non-leaf threshold, R_i is the partitioned space of $\mathcal{D}$ at leaf i and w_i is the weight of leaf i.

[4] This notation is for regression tasks. For classification; the partitions has the form: $\mathcal{D}_i^L = \{(\mathbf{x}_n, \mathbf{y}_n) \in N_i | y_{n,j} = t\}$ and $\mathcal{D}_i^R = \{(\mathbf{x}_n, \mathbf{y}_n) \in N_i | y_{n,j} \neq t\}$.

[5] The symbol $\leq/>$ here just denotes that it can either be $x_i \leq t_i$ or $x_i > t_i$.

prior becomes: $f(\mathbf{x};\theta) = \sum_{j=1}^{J} w_j \mathbb{I}(\mathbf{x} \in R_j)$ where the weights $w_i = \frac{\sum_{n=1}^{N} y_n \mathbb{I}(\mathbf{x} \in R_i)}{\sum_{n=1}^{N} \mathbb{I}(\mathbf{x} \in R_i)}$ is the output from node i at region R_i[6]. The most common cost functions; $c(\mathcal{D})$ are either MSE (Eq. (4.2)) for regression problems and entropy[7] (Eq. (4.4)) or Gini-index (Eq. (4.3)) for classification.

[6] Which is just the mean of the training labels of the region; R_i.

[7] or log-loss.

$$\text{MSE:} \quad c(\mathcal{D}_i) = \frac{1}{N} \sum_{n=1}^{N} (y_n - f(\mathbf{x}_n;\theta))^2 \tag{4.2}$$

$$\text{Gini-index:} \quad c(\mathcal{D}_i) = 1 - \sum_{c} \hat{\pi}_{ic}^2 \tag{4.3}$$

$$\text{Entropy:} \quad c(\mathcal{D}_i) = -\sum_{c=1}^{C} \hat{\pi}_{ic} \log(\hat{\pi}_{ic}) \tag{4.4}$$

$$\text{where } \hat{\pi}_{ic} = \frac{1}{|\mathcal{D}|} \sum_{n \in \mathcal{D}_i} \mathbb{I}(y_n = c) \text{ for class } c$$

4.1.2 *Boosting*

Decision trees have many advantages[8]; however, their disadvantages need to be addressed: Due to their greediness, they are inaccurate and unstable[9]. A method to overcome this is by using boosting, which in its essence, is a sequential fitting of trees. Boosting takes a *weak learner*; F_m which is a tree at stage $m \in \mathcal{M}$, then F_{m+1} is trained on the residual errors of F_m resulting in *strong learner* after $M \in \mathcal{M}$ steps a. Since trees depend on each other by fitting the last iteration, the boosted trees has reduced bias of the strong learner and has greater stability than one tree [51].

[8] Intuitive and easy to interpret with the possibility of printing the trees and they can do both regression and classification.

[9] Even a tiny change in the data set can lead to an entirely different tree.

Gradient boosting on decision trees is usually done on regression trees. The weak learner at stage $m \in \mathcal{M}$; $F_m(\mathbf{x})$ (see Eq. (4.5)) which is a sum of weights ($w_{j,m}$) assigned to the leaves (indexed over $1, \ldots, j, \ldots J_m$) of the underlying decision tree.

$$F_m(\mathbf{x}) = \sum_{j=1}^{J_m} w_{j,m} \mathbb{I}(\mathbf{x} \in R_{j,m}) \tag{4.5}$$

To obtain the weights $w_{j,m}$, we use a loss function $\ell(y_i, f_{m-1}(\mathbf{x}_i) + w)$[10], where y_i is the target value of the i-th training example, $f_{m-1}(\mathbf{x}_i)$ is the predicted value of y_i at stage $m-1$, and w is the weight to be optimized which are represented by is represented by $\hat{w}_{j,m} = \underset{w}{\operatorname{argmin}} \sum_{x_i \in R_{jm}} \ell(y_i, f_{m-1}(\mathbf{x}_i) + w)$. Summarized: the boosting process results in a sequence of decision trees where each tree is fitted to the residuals of the previous tree. The final boosted model (strong learner) is now a sum of these decision trees (weak learners), where each tree contributes weights to the final prediction.

[10] See Tab. (4.1) for a list of loss-functions.

The algorithm for gradient boosting is seen in Algorithm 1 (Source: Algorithm 10 [51, p. 612])[11]

[11] Note that ν are the *learning rate*.

Algorithm 1: Gradient boosting

Initialize $f_0(\mathbf{x}) = \underset{F}{\operatorname{argmin}} \sum_{i=1}^{N} L(y_i, F(\mathbf{x}_i))$;

for $m = 1 : M$ **do**

  Compute gradient residual:

$$r_{i,m} = -\left[\frac{\partial L(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)}\right]_{f(\mathbf{x}_i)=f_{m-1}(\mathbf{x}_i)};$$

  Use weak learner to compute:

$$F_m = \underset{F}{\operatorname{argmin}} \sum_{i=1}^{N} (r_{i,m} - F(\mathbf{x}_i))^2;$$

  Update: $f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \nu F_m(\mathbf{x})$;

Return: $f(\mathbf{x}) = f_M(\mathbf{x})$

Name	Loss	Gradient: $-\frac{\partial \ell(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)}$
Squared error	$\frac{1}{2}(y_i - f(\mathbf{x}_i)$	$y_i - f(\mathbf{x}_i)$
Absolute error	$\lvert y_i - f(\mathbf{x}_i)\rvert$	$\operatorname{sign}(y_i - f(\mathbf{x}_i))$
Exponential loss	$\exp(-\tilde{y}_i f(\mathbf{x}_i))$	$-\tilde{y}_i \exp(-\tilde{y}_i f(\mathbf{x}_i))$
Binary logloss	$\log(1 + \exp(-\tilde{y}_i f(\mathbf{x}_i)))$	$y_i - \pi_i$
Multiclass logloss	$-\sum_c y_{ic} \log(\pi_{ic})$	$y_{ic} - \pi_{ic}$

Table 4.1: Table of commonly used loss functions. For binary classification: $\tilde{y}_i \in \{-1, +1\}$ and $\pi_i = \sigma(2f(\mathbf{x}_i))$. For regression: $y_i \in \mathbb{R}$. Source: [48]

4.2 *LightGBM*

As already mentioned, the *GBDT*-algorithm used in this thesis is the LightGBM algorithm maintained by Microsoft[13]. This algorithm utilizes two different algorithms, which makes it different from other boosted decision tree algorithms such as the well-known XGBoost[8].

One of the main features of the LightGBM algorithm is that tree growth is leaf-wise and not level-wise, which most other competitors use. Leaf-wise growth finds the most optimal leaf, splits it, and moves on to the next. This gives high precision; however, it is a double-edged sword since it also introduces a high chance of over-fitting.

There are three main modifications that LightGBM utilizes. The first is *histogram-based tree growth*. During the growth of each individual tree, the features are transformed into histograms, and the (j_i, t_i) from Eq. (4.1) is calculated based on histograms and not the outright values (seen in Fig. (4.2)). This makes finding the threshold faster due to histogram-binning since the number of possible thresholds, t_i, is reduced. This also introduces the possibility of

errors if the bin granularity is too coarse and the underlying distribution is not correctly encapsulated.

In addition to *histogram-based tree growth*, LightGBM also the Gradient-based One-Side Sampling (GOSS)-algorithm. The GOSS algorithm modifies the basic gradient algorithm by selecting the top a% gradients with the largest values. The hypothesis follows: *Large gradients are more important for learning than small gradients*. The GOSS algorithm also uses randomly selected b% of the remaining small gradients to avoid over-fitting and avoid a skew in gradient distribution. The effect of downsampling the gradients is a reduced computation time needed in training and a focus on the under-trained trees. An important thing to note when using the GOSS algorithm; it is primarily effective for large samples and many weak learners in the ensemble. The outline of the algorithm is found in Algorithm 2.

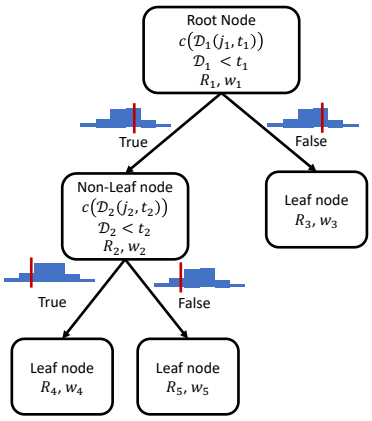


Figure 4.2: Example of decision tree with histogram-based tree growth. Which is a modified version of Fig. (4.1).

Algorithm 2: Gradient-based One-Side Sampling

Input: a, b - ratio of large and small gradient respectively.;
for $k = 1$ *: Iterations* **do**
  Compute gradients: $\forall i \in N$:
  $$r_{i,m} = -\left[\frac{\partial L(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)}\right]_{f(\mathbf{x}_i)=f_{m-1}(\mathbf{x}_i)};$$
  Sort gradients i descending order: $A = \{r_{i,m} \leq r_{jm} \leq \ldots\}$;
  Pick: $A_{top} = A[: a \times N]$ and
  $A_{bot} = \text{Random}(b\% \text{ of } A[a \times N :]$;
  Create a new tree w. weights calc from:
  $r_{i,m} \times \frac{1-a}{b} : r_{i,j} \in A_{bot}$;
  Calc. new iteration of weak learner with weights from
  $r_{i,m} \in A_{top} \cup A_{bot}$;

The last modification LighGBM uses is the Exclusive Feature Bundling (EFB)-algorithm. This algorithm aims to reduce the number of features by bundling them together by chaining the two algorithms: *Greedy Bundling* and *Merge Exclusive Features*. The *Greedy Bundling*-algorithm computes the conflict matrix[12], which measures feature overlap. The algorithm then finds the features with the most conflict and combines the two features into a new feature. This is done iteratively to reduce redundancy in the feature space. The detailed pseudo-code can be seen in [32, Algorithm 3 & 4]. A summary of the Greedy Bundling is seen in Algorithm 3. The *Merge Exclusive Feature* algorithm is chained together with the *Greedy Bundling*-algorithm since the idea behind it is to take the low-conflict features which were not merged in the *Greedy Bundling*-algorithm and bundles features together which has non-overlapping non-zero values in the conflict matrix. This offsets the features in the same bundle such that the original feature distribu-

[12] A conflict matrix measures feature-pair conflict. For all feature pairs; (i, j), calculate the number of samples where both features have non-zero values: $(x_{i,k}, x_{j,k}) \neq (0, 0)$ for the same sample.

tions are not distorted, and the resulting histograms of the features after EFB should approximately be the same as if bundling were not implemented.

Algorithm 3: Greedy Bundling

```
Create graph; G of features with weighted edges - weight are
  calculated based on feature conflict;
Sort features based on vertex degree -descending;
for i = 1 : Features do
    if Feature Conflict < Threshold then
        Add to existing feature bundle;
    if Feature Conflict > Threshold then
        Create a new feature bundle;
```

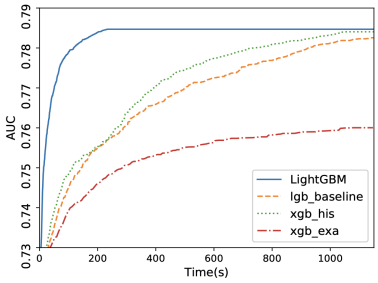


Figure 4.3: AUC vs. time for a binary classification task with 10M rows of data and 700 features columns. *xgb_exa* are a pre-sorted algorithm and *xgb_his* are a histogram-based algorithm where xgb stands for XGBoost. *LightGBM* are with the GOSS and EFB algorithms and *lgb_baseline* are without. Source: [32, Figure 1, p. 7]

The implementation of GOSS and EFB is what makes LightGBM unique, and they give a significant speedup: For the histogram-based tree growth, the speed goes from $\mathcal{O}(\#Data \times \#Features)$ to $\mathcal{O}(\#Bins \times \#Features)$ for finding optimal leaf-split threshold.
For the GOSS algorithm, the speed-up goes as: $\mathcal{O}(\#Data \times \#Features)$ to $\mathcal{O}(\#Samples \times \#Features)$ and for Exclusive Feature Bundling: $\mathcal{O}(\#Data \times \#Features)$ to $\mathcal{O}(\#Data \times \#Bundles)$. The combined speedup are then: $\mathcal{O}(\#Samples \times \#Bundles)$. An example of this speedup with comparisons is seen in Fig. (4.3), which are from the original LightGBM paper[32].

4.3 Hyper-Parameter Optimization

This thesis uses the default setting/parameters for LightGBM[13, Git version: 3.3.2 with Python 3.10.8] unless other is stated. The speedups, as mentioned, are not only reliant on the number of features and samples, It also depends on the number of cpu-cores for multiprocessing, and all LightGBM models mentioned in his thesis are trained and run on a High-Perfomance Computer (HPC) hosted by the University of Copenhagen[59] which has 48 cores. This makes it possible to achieve significant speedup. When training a LightGBM model, the number of cores used for parallel learning is *n_jobs*. Speed is not the only important parameter; the precision, stability, and training time for LightGBM primarily depend on the choice of hyperparameters. An example of some of the hyperparameters to choose from is found in Tab. (4.2).
Finding the best combination of hyperparameters in the hyperparameter-hyperspace has historically been very hard since many hyperparameters can be correlated, and locating the optimal set of hyperparameters is highly nontrivial. Looking at Tab. (4.2), there is a

Hyperparameter	Description
num_leaves	The maximum number of leaves allowed in each tree.
max_depth	limit the maximum depth allowed for the trees.
num_iterations	The number of boosting iterations.
early_stopping_round	Parameter for stopping training if performance on validation is not increased for #early_stopping_round.
lambda_l1	L1 (or Lasso) regularization.
lambda_l2	L2 (or Ridge) regularization.

Table 4.2: A small sample of hyperparameters for LightGBM from the LightGBM docs: [48]

6D hyperspace with infinite combinations. One way is using the Optuna[3, Git version: 3.0.4 with Python 3.10.8] optimization framework. This framework uses various smart algorithms to find the optimal hyperparameter combination without brute-forcing all hyperspace-parameter combinations. This is achieved by two processes/algorithms: Sampling and pruning.

The default sampler and the one used in this thesis is the Tree-structured Parzen Estimator (TPE) sampler[4]. The TPE sampler takes the search space and partitions it into a tree-structured search space. The TPE algorithm recursively partitions the search space into a *good* and a *bad* space/tree nodes. This makes the algorithm focus on the suitable regions of the hyperparameter space, and computation time is decreased.
The TPE sampler uses a Gaussian kernel density estimator[13]: $p(x|B) = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{\sqrt{2\pi}\sigma_i}\exp -\frac{(x-x^{(i)})^2}{2\sigma_i^2}$ to estimate the hyperparameter probability density function[14]. The TPE algorithm is seen in algorithm[15]: 4. If the objective is not singular, the TPE algorithm is modified accordingly; see reference [4].

[13] Kenel Density Enstimator or Parzen Window Density Estimator.

[14] where M observations with set: $\mathcal{B} = \{x^{(1)}, ..., x^{(M)}\} \subset \mathcal{H}$, where $\mathcal{H}$ is the hyperparameter space.

[15] The pseudo-algorithm: 4 is inspired by [55, Algorithm 1].

Algorithm 4: TPE Sampler in Optuna

Require: A tree-structured search space $\mathcal{X}$, An objective function $f : \mathcal{X}^k \to \mathbb{R}$ (lightGBM model metric) where k is the number of hyperparameters and a set of starting observations: $\mathcal{D} = \{(\mathbf{x}(1), y(1)), \ldots, (\mathbf{x}(k), y(k))\}$ where $y = f(\mathbf{x})$.
Select an quantile: $\gamma \in (0, 1)$;
for n_{iter} *iterations* **do**
  $\mathcal{D}_\ell = \{(\mathbf{x}, y) \in \mathcal{D} \mid y \in \text{best-}\lfloor \gamma |\mathcal{D}| \rfloor\}$;
  $\mathcal{D}_g = \mathcal{D} \setminus \mathcal{D}_\ell$;
  for *1 to* n_{hyper} *total hyperparameters* **do**
    With kernel density estimation; construct $\ell(x_i)$ where $\{x_i \mid (\mathbf{x}, y) \in \mathcal{D}_\ell\}$;
    With kernel density estimation; construct $g(x_i)$ where $\{x_i \mid (\mathbf{x}, y) \in \mathcal{D}_g\}$;
    Find x_i^* which maximizes the Expected Improvement: $EI(x_i) = \left(\gamma + (1 - \gamma)\frac{g(x_i)}{\ell(x_i)}\right)^{-1}$;
  $\mathcal{D} = \mathcal{D} \cup \{(\mathbf{x}^*, y^* = f(\mathbf{x}^*))\}$, where $\mathbf{x}^*$ is the vector of the x_i^*s.

On top of using a sampler and not a simple grid search, the Optuna library uses *pruning*. Pruning is a technique that discards the models that do not perform above a certain threshold. The pruner used in this thesis is the *Median pruner*, which after N finished optimization runs calculates the median, and those models that under-performs are discarded, and the rest are kept.

4.3.1 Verstack

LightGBM and Optuna are two libraries that integrate very well into each other and instead of calling these two libraries individually, a very clever and well-made Python library called Verstack[73, Git version: 3.0.4 with Python 3.10.8] combines LightGBM and Optuna. The Verstack library is used for training all LightGBM models in this thesis. Verstack cleverly searches the hyperparameter space and has high precision and low training time. Even though this thesis uses Verstack, a local version of Verstack is used, which is more cluster-safe[16] The Verstack python library is an ML swiss-army knife, and only the LGBMTuner[17] are used for this thesis. Hence the branched-out version of Verstack used in this thesis only contains the LGBMTuner-method[18]. This branched-out version of Verstack can be found in the git-repository of this thesis[19] since it is not yet made into a complete pull request.

[16] Avoiding the use of all cores since multiple people are using the shared 48-core HPC cluster at the University of Copenhagen.

[17] The python method which combines LightGBM and Optuna.

[18] In addition to making Verstack cluster-safe, the logging was also changed such it was written to a file.

[19] Gitlab repository: [50].

Part II

Analysis

5
The Aim and Previous Work

At the beginning of the thesis, the RK* group focused mainly on the electron yield; $N_{Sig(B^0)}$ of the $B^0 \rightarrow K^{*0}e^+e^-$ decay. Efficiency studies and the signal yield on muons and the control channel: J/ψ were only in the infancy. Hence, this thesis focused on the electron yield due to data availability.

The analysis will focus on separating signal vs. background, where signals are divided into B^0 and $\overline{B}^0$ since the ATLAS detector can not distinguish between Kaons and Pions. This is not an easy task since separating the background from the signal can distort the signal mass distribution and the remaining background such that large uncertainties will be introduced in the fits used for the extraction of the signal yield.

A simple diagram of the hypothesis of this thesis is seen in Fig. (5.1). The signal and background are mixed on the left side, and under the histogram, a $2D$ coordinate system with two axes is seen. One axis is the pre-selection[1] cuts and the other is the distribution of $m(B^0)$ (seen in the histogram above). The hypothesis is that by applying multiple cuts (or applying GBDTS) on different variables, the signal would be well separated from the background (righthand side of Fig. (5.1)) and ready for the fit for the yield extraction.

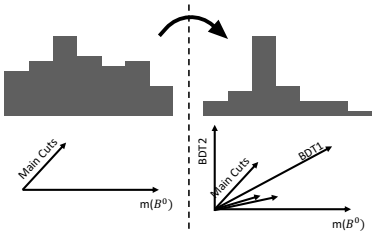


Figure 5.1: A Diagram of the separation of signal ($m(B^0)$) from the background using various variable cuts or GBDTs.

[1] or Main cuts.

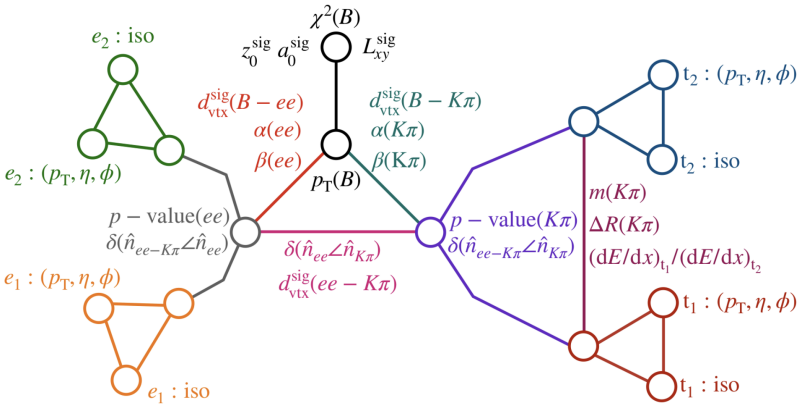


Figure 5.2: The RK* group GNN architecture. Each branch represents the physical aspects of the B^0 decay: A B-vertex branch etc. All the variables/features can be seen in Tab. (5.1). Source of figure: [30]

The RK* group uses two Graph Neural Network (GNN)s of the form seen in Fig. (5.2) for their signal-background separation. The GNN architecture represents the decay topology: a branch consisting of electron one, a branch for electron two, etc. A complete overview of the features the GNN uses can be seen in Tab. (5.1).
A GNN can be a powerful tool and is an excellent deep-learning model when features are linked and have mutual dependencies. The biggest downside of GNNs are their long training time. As mentioned, the RK* group uses two GNNs, and each uses $\sim$ **4**h on training. Having signal/background separation algorithms that take this long is not preferable when doing studies of variables and trying out different feature configurations.

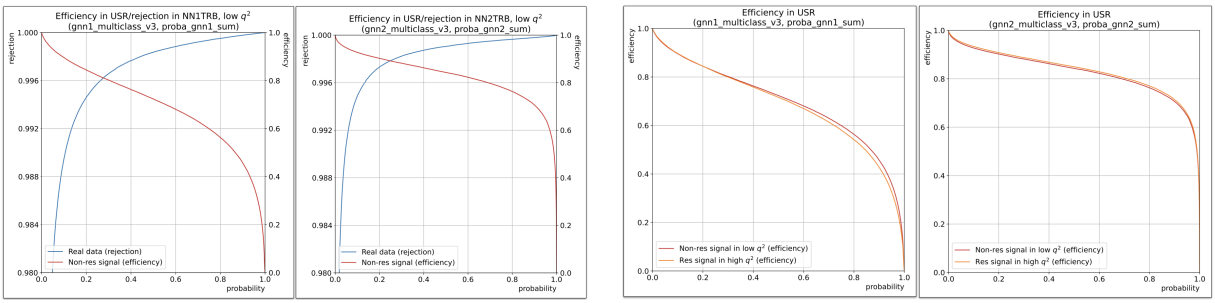


Figure 5.3: Efficiency of the two GNNs from the RK* group. Starting from the left: Figure (a) GNN1, which is trained on signal region and mass-sideband, which shows low di-electron (q^2_{low}) signal efficiency and background rejection as a function GNN output probability. (b) Shows GNN2 trained on signal region and same-sign charge sideband. Again it shows signal efficiency and background rejection. Figure (c) Shows only signal efficiency with the difference in high and low di-electron bands for GNN1. For the right-most figure; (d) shows the same as (c) with GNN2 instead. Source of figure: [39].

These training times of $\sim$ **4** h per GNN were the factor that gave birth to the hypothesizes that signal/background separation done by GBDTs could be a faster and just as precise. Additionally, GBDTS could be used to do feature studies in which features are the most significant in separating signal vs. background.

At the 10$^{\text{th}}$ of August 2022 at the RK* weekly meetings, the performance of the two GNNs was presented and is seen in figure Fig. (5.3). The leftmost two figures show the efficiency in selecting signal and rejection of background for the two GNNs. The first GNN are trained on MC signal region and real data background in B-mass sidebands. The second GNN is trained in the MC signal region with real data background data defined such that two tracks (Kaon/Pion) have the same sign charge.
The benchmark seen in Fig. (5.3) will be used throughout this thesis when a GBDT model needs to be tested and benchmarked.

Variable	Description
$p_T(e_1)$, $p_T(e_2)$, $p_T(trk_1)$, $p_T(trk_2)$	Transverse momentum
$\eta(e_1)$, $\eta(e_2)$, $\eta(trk_1)$, $\eta(trk_2)$	Pseudorapidity
$\phi(e_1)$, $\phi(e_2)$, $\phi(trk_1)$, $\phi(trk_2)$	The angle between the particle trajectory and the plane perpendicular to the beamline
$\text{ISO}_{c40}(e_1)$, $\text{ISO}_{c40}(e_2)$ $\text{ISO}_{c40}(trk_1)$, $\text{ISO}_{c40}(trk_2)$	The sum of energy from other particles in a cone of radius=0.4
P-value(ee), P-value$(K\pi)$	The probability that ee or $K\pi$ are produced by background.
$\angle(ee - K\pi \vert ee) - \text{plane}$ $\angle(ee - K\pi \vert K\pi) - \text{plane}$	The angle between the vector-sum of $ee - K\pi$ vs. ee or $K\pi$.
a_0^{sig}, z_0^{sig}, L_{xy}^{sig}	These are the point closest to the primary vertex: a_0 is the distance perpendicular to the beam, z_0 are along the beamline, and L_{xy} is the distance between the decay vertex and primary vertex. Significance is the distance divided by its uncertainty.
$\chi^2(B)$, $p_T(B)$	Transverse momentum for the B-meson and its good-of-fit from the reconstruction.
$\alpha(ee)$, $\beta(ee)$, $d_{vtx}^{sig}(B - ee)$	$\alpha(ee)$ is the angle of the electron pair. $\beta(ee)$ is the velocity of the center of mass of the electron pair. $d_{vtx}^{sig}(B - ee)$ is the distance between the ee and B vertex divided by its uncertainty
$\alpha(K\pi)$, $\beta(K\pi)$, $d_{vtx}^{sig}(B - K\pi)$	Same as above, just with $K\pi$ instead of ee.
$\angle(ee \vert K\pi) - \text{plane}$, $d_{vtx}^{sig}(ee - K\pi)$	The angle between the vector-sum of ee vs. $-K\pi$ and the significant distance between the ee and $K\pi$ vertex.
$m(K^+\pi^-)$, $m(K^-\pi^+)$	invariant mass of the kaon pion decay (and its antiparticle)
$\Delta R(K\pi)$, $(\frac{dE}{dx})_{t1}/(\frac{dE}{dx})_{t2}$	The angular separation of $K\pi$ and the fraction consist of $\frac{dE}{dx}$, which are the energy loss in the detector at different layers.

Table 5.1: Features/varaibles used in the RK* GNN. To the left, the variable/feature is written mathematically, and to the left is an explanation of the variable.

6
Methodology

For reproducibility, the data used in the analysis and the pre-selection cuts applied before the analysis are reviewed. These cuts are chosen by the RK* group and hence adopted for this analysis. The Methodology is also about dealing with multiple candidates per event, e.g., multiplicity, which is also covered in this section. Lastly, the analysis methodology is reviewed with the ML testing pipeline and the fitting routine.

6.1 Data Preparation

As mentioned, Run 3 is the current run period or "LHC Run" as written in Fig. (6.1). This figure shows the naming convention concerning the different periods, blocks, etc. For this analysis, **LHC Run 2 is used, and the period of interest is Period K**, which is the primary period the analysis has been used on. This is due to the first periods ramping up in luminosity and, therefore, insufficient quality. Later periods have better data quality. Hence period K is the earliest used. Other periods which can be used are L, M, N, O, and Q. [14].

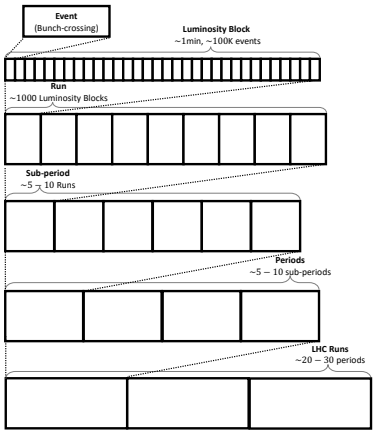


Figure 6.1: A diagram of how data is divided by ATLAS. Figure inspired by [7, p. 10]

The MC data used in this thesis are seen in Tab. (6.1) and Tab. (6.2). In Tab. (6.1), one can see the MC samples and the decay, denoted signal. The signal is separated into two parts: Non-Resonant and Resonant, where the only difference is that there is an intermediate J/ψ between the B^0 and the ee. The n-tuples containing these have truth labels, so it is possible to distinguish the decays. A simulated MC background decays list is seen in Tab. (6.2).

Category	Decay	DSID
Non-Resonant	$B^0 \to K^{*0}ee$	300590
Non-Resonant	$\overline{B^0} \to \overline{K^{*0}}ee$	300591
Resonant	$B^0 \to K^{*0}J/\psi(ee)$	300592
Resonant	$\overline{B^0} \to \overline{K^{*0}}J/\psi(ee)$	300593

Table 6.1: Table of Monte Carlo signal samples divided into Resonant and Non-Resonant with their DSID ("Dataset Identification").

Decay	DSID	Anti-Decay	DSID
$B^+ \to \pi^+ J/\psi(ee)$	300718	$B^- \to \pi^- J/\psi(ee)$	300719
$B^+ \to K^+\pi^0(ee\gamma)$	300722	$B^- \to K^-\pi^0(ee\gamma)$	300723
$B^+ \to \pi^+\pi^0(ee\gamma)$	300724	$B^- \to \pi^-\pi^0(ee\gamma)$	300725
$B^+ \to \pi^+\eta(ee\gamma)$	300726	$B^- \to \pi^-\eta(ee\gamma)$	300727
$B^+ \to K^+\eta(ee\gamma)$	300730	$B^- \to K^-\eta(ee\gamma)$	300731
$B^0 \to K^+\pi^- J/\psi(ee)$	300734	$\overline{B^0} \to K^-\pi^+ J/\psi(ee)$	300735
$B^0 \to K^+\pi^-\psi(2S)(ee)$	300738	$\overline{B^0} \to K^-\pi^+\psi(2S)(ee)$	300739
$B^0 \to K^+\pi^-\pi^0(ee\gamma)$	300742	$\overline{B^0} \to K^-\pi^+\pi^0(ee\gamma)$	300743
$B^0 \to K^{*0}\eta(ee\gamma)$	300744	$\overline{B^0} \to \overline{K^{*0}}\eta(ee\gamma)$	300745
$B^0 \to K^{*0}\pi^0(ee\gamma)$	300748	$\overline{B^0} \to \overline{K^{*0}}\pi^0(ee\gamma)$	300749

Table 6.2: Table of Monte Carlo background samples with their DSID ("Dataset Identification").

6.1.1 *Group Designated Pre-selection Cuts*

The RK* group applies a series of pre-selection[1] cuts to the n-tuples and stores the data into feather-files[2]. These cuts are seen in Tab. (6.3) with an explanation of why the extra cuts are implemented. As mentioned, there are two signals: $m(B^0)$ and $m(\overline{B}^0)$, and the question is which are the correct ones. The RK* has used a modified B-mass feature to take this into account by using the mass of the K^{0*}-meson, which is set to $891.66\,\mathrm{MeV/c^2}$ and this modified mass is seen in Eq. (6.1) and is designated: $m(K^{*0}_{\text{closer}})$.

[1] Throughout the pre-selection cuts are also called main cuts or group cuts.

[2] feather files; .ftr are a file storage format which stores arrays and has high read/write speed and is ideal for ML studies.

$$m(K^{*0}_{\text{closer}}) = \begin{cases} m(B^0) & \text{for } |m(K^+\pi^-) - 891.66\,\mathrm{MeV/c^2}| < |m(K^-\pi^+) - 891.66\,\mathrm{MeV/c^2}| \\ m(\overline{B^0}) & \text{for } |m(K^+\pi^-) - 891.66\,\mathrm{MeV/c^2}| \geq |m(K^-\pi^+) - 891.66\,\mathrm{MeV/c^2}| \end{cases} \tag{6.1}$$

Table 6.3: Table of cuts used by the RK*-group which are applied to both data and MC.

Variable-Cut	Description
$p_T(e_1) > 5\,\text{GeV}$ $p_T(e_2) > 5\,\text{GeV}$	The transverse momentum (p_T) for electron has to be above $5\,\text{GeV/c}$. This threshold is set height enough to filter out low-energy background particles of no interest.
$p_T(trk_1) > 5\,\text{GeV}$ $p_T(trk_2) > 500\,\text{MeV}$	The transverse momentum (p_T) for tracks has to be above $500\,\text{MeV/c}$. - same argumentation as for the electrons. Just with a smaller threshold.
$\lvert\eta(e_1)\rvert < 2.5$ $\lvert\eta(e_2)\rvert < 2.5$	Setting the absolute pseudorapidity (η) of the two below 2.5 refers to the part of the detector with the highest resolution and detection potential (Hit in the SCT and TRT in the inner detector).
$\lvert\eta(trk_1)\rvert < 2.5$ $\lvert\eta(trk_2)\rvert < 2.5$	Setting the absolute pseudorapidity (η) of the two tracks below 2.5 - same argumentation as for the electrons.
$m(ee) < 7\,\text{GeV/c}^2$	Setting the invariant mass of the electron-pair ($m(ee)$) below $7\,\text{GeV/c}^2$. This is a bit over the double of the J/ψ mass which is $m_{J/\psi} = 3096.900 \pm 0.006\,\text{MeV/c}^2$ such that resonant signals can pass, and non-interesting particles are removed.
$GSF(OK) = \text{True}$	The Gaussian-Sum Filter "OK" variable is set to True means that the electron identification and reconstruction are of good quality
$m(B^0) \in [3, 6.5]\text{GeV/c}^2$ AND $m(K^{*0}) \in [690, 1110]\text{MeV/c}^2$ OR $m(\overline{B^0}) \in [3, 6.5]\text{GeV/c}^2$ AND $m(\overline{K^{*0}}) \in [690, 1110]\text{MeV/c}^2$	This band is in both the desired range where the "true" mass of the B^0 is: $m(B^0) = 5279.66 \pm 0.12\text{MeV/c}^2$ and for the K^{*0} is $m_{K^0} = 895.55 \pm 0.20\text{MeV/c}^2$ [72]. It is assumed the K^0 comes from the two tracks: $K^{\pm}\pi^{\mp}$ and the electron pair. The same for the antiparticle decays. The reason for the boolean: "OR" between the decays and anti-decays.
Quality(e_1)=Loose Quality(e_2)=Loose	There are multiple cuts: Tight, medium, and loose. The quality of the electrons is set to *loose* cuts.
Quality(trk_1)=Loose Quality(trk_2)=Loose	There are multiple cuts: Tight, medium, and loose. The quality of the tracks is set to *loose* cuts.
Quality(e_1)-Shower=Loose Quality(e_2)-Shower=Loose	There are loose criteria for the identification quality of electrons from electron showers. There are multiple cuts: Tight, medium, and loose.
$\Delta R(ee) > 0$	Most electron pairs come from the J/ψ; however, another big fraction will also come from photon decays ($\gamma \rightarrow ee$). Since photons have mass: 0, the electron pair will have a high probability of having the same direction of the γ. Hence the threshold of the angular separation of the two electrons is above 0.1.
$q(e_1) \times q(e_2) < 0$	The electrons need opposite charges since decay pairs are of interest.
$ISO_{c40}(e_1) < 100\text{GeV/c}^2$ $ISO_{c40}(e_2) < 100\text{GeV/c}^2$	the ISO_{c40} is the measured energy from other particles within a cone of radius: 0.4 from the electron of interest.

6.1.2 Handling Multiplicity

As seen in figure 6.2, there are, on average, 24.7 candidates per event, with a maximum of 1650 candidates for one event for Period K after the pre-selection. This means the candidate multiplicity needs to be handled, or the ML model will, in its training, learn to reject almost everything since the actual signal would get flooded by poor candidates rather than learn to separate the signal from the background, which is hard to distinguish.
The RK* groups solution to the multiplicity-problem are by using the six candidates in the background with the highest $L_{xy}^{sig}/\chi^2(B)$ (see Eq. (6.2)) which would be most similar to the signal. The reasoning behind this is seen in Fig. (6.3) where the truth-efficiency is seen as a function of L_{xy}^{sig}, $\chi^2(B)$ and $L_{xy}^{sig}/\chi^2(B)$ using the MC signal dataset with DSID: 300590 (see Tab. (6.1)). The six best $L_{xy}^{sig}/\chi^2(B)$ captures both the goodness of the B-vertex ($\chi^2(B)$) and the significance of the distance between the primary- and decay-vertex by a 99.1% in truth efficiency in MC.

$$N_{Candidates}^{Multiplicity} = 6 \text{ best } L_{xy}^{sig}/\chi^2(B) \tag{6.2}$$

The "six best $L_{xy}^{sig}/\chi^2(B)$"-approach is only for training the GBDTs and is not used for the application of the GBDTs. The MC signal is all truth-matched for the training; hence the problem of multiplicity is not a problem here.

6.2 Analysis Methodology

The approach of this thesis mirrors the RK*'s with a few changes where, with the most noteworthy change: replacing the GNNs with GBDTs. For reference, the entire training pipeline is seen in the flow diagram in Fig. (6.4). The first step: "Main Cut and Candidate Selection", is a Python script that takes n-tuples as input and outputs feather files ready for the ML pipeline. The script applies the pre-selection cuts defined by the RK* group seen in Tab. (6.3). It also does feature engennering[3] and locates the six best $L_{xy}^{sig}/\chi^2(B)$ in the background for training.

6.2.1 Mass Regions

The $q^2 \times m(B^0)$-space is divided into multiple areas (with additional requirements; $\chi^2(B)$, $q(e)$ and $q(trk)$) as seen in Fig. (6.5)(a). The reason for this is that some of the data is blinded such that the result is not overfitted. As seen in Fig. (6.5)(a), the data are divided

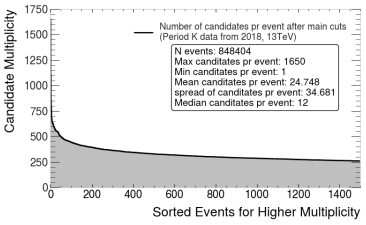


Figure 6.2: Multiplicity of period K data after main cuts.

N	$L_{xy}^{sig}/\chi^2(B)$	L_{xy}^{sig}	$\chi^2(B)$
1	89.4%	91.3%	83.4%
2	95.6%	96.1%	91.1%
3	97.3%	97.6%	94.1%
4	98.3%	98.4%	95.9%
5	98.8%	98.9%	97.0%
6	99.1%	99.2%	97.7%
7	99.3%	99.3%	98.2%
8	99.4%	99.5%	98.6%
9	99.6%	99.6%	98.9%
10	99.7%	99.7%	99.1%
20	99.9%	99.9%	99.9%
30	100%	100%	100%

Figure 6.3: Truth efficiency using MC signal data (DSID 300590) with different proxy features. Table from: [39].

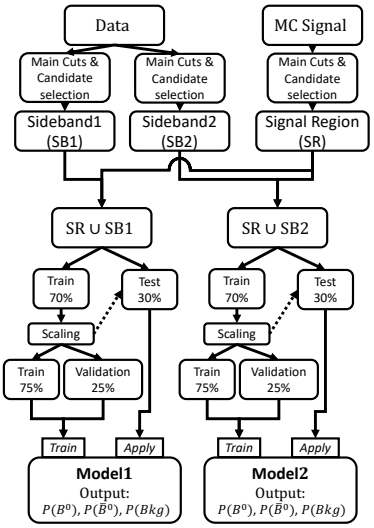
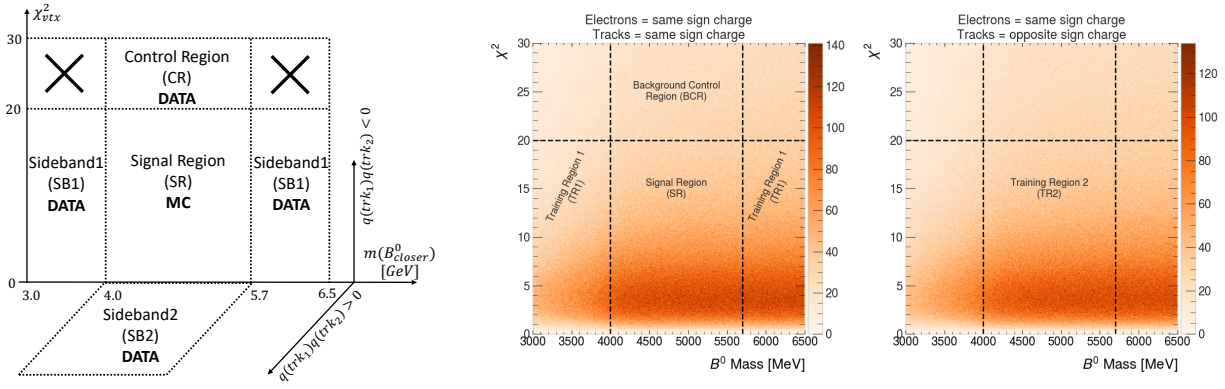


Figure 6.4: The ML Pipeline for training the models.

[3] More on this later.



(a) (b)

Figure 6.5: Figure (a): The RK* group B mass region cuts. Source of original figure: [30], modified by the author of this thesis.
Figure (b): Showing the Period K density in the different mass region cuts in a combined q^2_{low} and q^2_{high}.

into multiple areas: Signal Region (SR), Sideband1 (SB1), and Sideband2 (SB2), and these exist for both q^2_{low} and q^2_{high}. In Fig. (6.5)(b) the event density for combined q^2_{low} and q^2_{high} is seen for the different mass-region CUTS.

As seen in Fig. (6.4), the GBDTs are trained on $\{\text{SR}^{\text{MC}} \cup \text{SB1}^{\text{data}}\}$ and $\{\text{SR}^{\text{MC}} \cup \text{SB2}^{\text{data}}\}$ for q^2_{low} where SR^{MC} is only containing non-resonant truth-matched decays for the training and the SB1^{data}/SB2^{data} contains the six best $L^{sig}_{xy}/\chi^2(B)$ for each event.

As seen in Fig. (6.6), the density for the signal is high in q^2_{high} where the density in q^2_{low} for the signal is lower. This is because the energy is significantly higher in q^2_{high}, and the higher energy allows for the production of heavier particles hence resonant decays which poses as a statistical challenge in the q^2_{low}-bin.

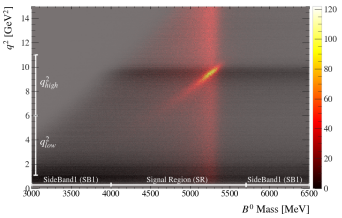


Figure 6.6: Density of MC Signal in $q^2 \times m(B^0)$-space (red/yellow) overlaid over the density of Period K (grey). It is clear that the signal is strongly present in q^2_{high} compared to q^2_{low}.

The rest of Fig. (6.4) shows how the data are used with respect to the models. Here model1 and model2 are the two GBDTs. For scaling, many different types of scalers exist, and the one used by this thesis is the RobustScaler of the Scikit-learn python library [17], which are defined as seen in Eq. (6.3). This scaler does not assume the distribution of the features is normally distributed, and the scaling is robust against outliers. The scaling is fitted to the training data and then applied to the test data to avoid leaking information.

$$x'_i = \frac{x_i - \text{Quantile}_{50\%}(x)}{\text{Quantile}_{75\%}(x) - \text{Quantile}_{25\%}(x)} \tag{6.3}$$

The output of each model is multiclass probabilities: $P_1(B^0)$, $P_1(\overline{B}^0)$ and $P_1(Bkg)$[4] which are under the rule: $P_1(B^0) + P_1(\overline{B}^0) + P_1(Bkg)$ = 1. These outputs are then used to make three decisions: *Global Candidate Selection*, *Local Candidate Selection*, *Mass Hypothesis Selection*, which are described in Tab. (6.4). In Fig. (6.7), the selection rules are visually applied to an event X.

[4] The subscript notation can vary depending how well the different models are distinguished.

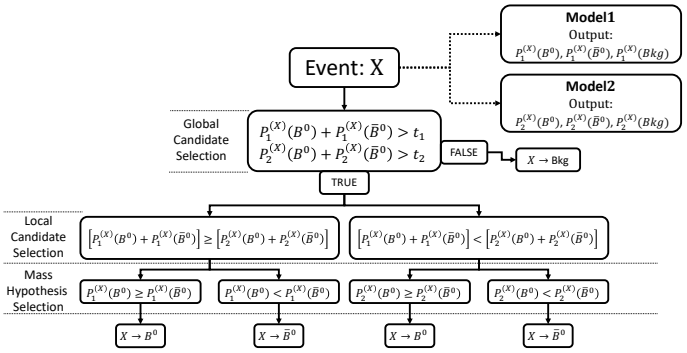


Figure 6.7: A visualization of how the Global, Local, and Mass Hypothesis Selection Rules are used to sort events. The rules are seen in Tab. (6.4)

GBDT Selection Rules	
Global Candidate Selection	Classify either candidate as signal or background whenever above/below threshold t. $P_1(B^0) + P_1(\overline{B}^0) > t_1$ $P_2(B^0) + P_2(\overline{B}^0) > t_2$
Local Candidate Selection	Find the best candidate within an event by the maximum signal probability. $\max \left\{ \begin{matrix} P_1(B^0) + P_1(\overline{B}^0), \\ P_2(B^0) + P_2(\overline{B}^0) \end{matrix} \right\}$
Mass Hypothesis Selection	If the mass are assigned the value; $m(B^0)$ or $m(\overline{B}^0)$ depending on the maximum of the two signal probabilities. Take model P_i which maximizes the Local Candidate Selection and assign mass as: $m = \begin{cases} m(B^0) & \text{if } P_i(B^0) = \max\{P_i(B^0), P_i(\overline{B}^0)\} \\ m(\overline{B}^0) & \text{if } P_i(\overline{B}^0) = \max\{P_i(B^0), P_i(\overline{B}^0)\} \end{cases}$

Table 6.4: Global, Local and Mass Hypothesis Selection Rules

6.2.2 *ML Testing*

After each model has been trained, a comprehensive testing-scheme are applied to ensure that the model performance is satisfying and to monitor the different aspects of the performance. Four different testing suites are applied to the trained GBDTs: (1) *LightGBM* Testing Suite, (2) *Signal vs. Background* Testing Suite, (3) *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite and lastly the (4) *Mass Shape* Testing Suite seen in Fig. (6.8).

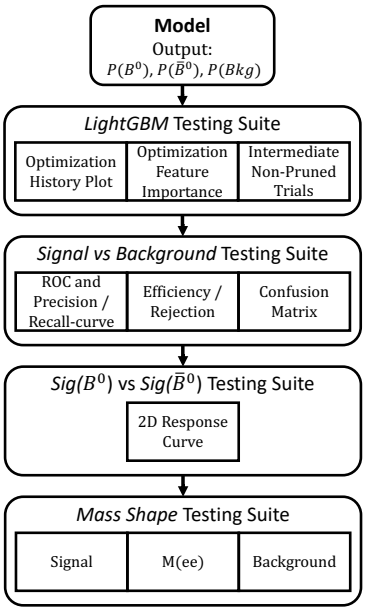


Figure 6.8: This figure depicts how the output of the models is tested using four different testing suites.

LightGBM Testing Suite

The *LightGBM* Testing Suite consists of three subtests to see if the LightGBM hyperparameters are tuned correctly for the highest performance possible. The first test is the *Optimization History Plot*: which shows the objective value vs. the number of trials which identifies stagnation periods and shows if the optimization is minimized. The next test is the *Optimization Feature Importance*. This test shows which hyperparameters were the most important in minimizing the objective value. The last test is the *Intermediate Non-Pruned Trials* which shows non-pruned intermediate objective

values during the optimization and indicates the absence of overfitting if the objective is minimized and converged.

***Signal vs Background* Testing Suite**

The *Signal vs. Background* Testing Suite consists of 3(+1) tests to quantify the model's ability to separate combined signal; Sig $= (B^0 + \overline{B}^0)$ from the background.

Test one[5]: *1D response curve* - Histograms of combined signal and background probabilities[6] on a logarithmic scale. The probability-axis(x-axis) is logit transformed using $\text{logit}(p) = \ln(p/(1-p))$ where $\text{logit}(p) \in (-\infty, \infty)$ to enhance visibility of overlapping densities.

Test two: *ROC-curve and Precision/Recall-curve* - Quantifies the combined signal vs. background classification for all thresholds. The Receiver Operating Characteristic (ROC) Curve plots the true positive rate (TPR)[7] which is the probability of detection against the false positive rate (FPR)[8] which are the probability of a false alarm. The optimal classification is (FPR=0, TPR=1,) and the AUC-score is the area under the curve where $AUC = 0.5$ is a random classifier, and $AUC = 1$ is a perfect The Precision/Recall-curve is the precision (PPV)[9], which is the proportion of positive predictions that are correct versus the Recall(TPR). The optimal classification is (TPR=1, PPV=1), and the Average Precision (AP) is the area under the Precision/Recall-curve where $AP = 0.5$ is a random classifier, and $AP = 1$ is a perfect classifier.

Test three: *Efficiency/Rejection* - shows the efficiency of selecting combined signal and the rejection of background as a function of probability threshold.

Test four: *Confusion matrix* - Measures the model's performance in classifying B^0, $\overline{B}^0$, and background for one specific threshold.

***Sig*(B^0) *vs Sig*($\overline{B}^0$) Testing Suite**

This testing suite is used to see how well the multi-class classifier can predict if it is a B^0 or $\overline{B}^0$ given either the B^0 or the $\overline{B}^0$ MC samples.

***Mass Shape* Testing Suite**

The *Mass Shape* Testing Suite is used to see if the mass distributions get distorted when GBDT cuts are applied. The importance of this test is significant since the end goal is the extraction of the signal yield. If the signal gets smeared or the background starts peaking in the signal region due to GBDT cuts will introduce large errors in the signal yield and in the worst-case scenario; the extraction of the signal yield will not be possible. An example of the worst-case scenario is seen in Fig. (6.9).

[5] not mentioned in Fig. (6.8), so this is the "+1".

[6] As mentioned; The model outputs a probability (P) to be a class ($P \in [0, 1]$).

[7] Also called the sensitivity, recall or hit rate

[8] FPR is also known as fall-out.

[9] Also known as the Positive Predictive Value

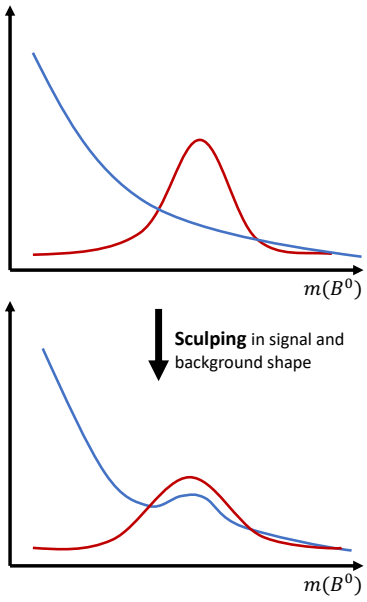


Figure 6.9: An example of the worst-case scenario of mass sculpting. The blue line is the background, and the red is the signal. The top figure is the raw background + signal. When cuts are applied to the data, the worst-case scenario is a peaking background and a smeared signal (bottom figure). This makes the fitting and hence the extraction of the yield a hard task and introduces larger errors in the yield, and in the end, the $R_{K^{*0}}$-ratio will have a larger error.

6.2.3 *Feature Engineering*

All features used in this thesis originate in the n-tuples, and all n-tuple features which are not modified start with "*BeeKst*". A full review of the feature engennering calculations is seen in Tab. (6.5). The symbol of each part and the feature name are present, along with the equation for calculating the specific feature. To compare the result of this thesis with the RK*-group, the values have to be the same though they might not be the most up-to-date values from PDG-group values[72].

Some of the equations in Tab. (6.5) use various vertexes derived from *BeeKst* features. The equations to calculate the vertexes are found in Appendix A.1; equations Eq. (A.1) - Eq. (A.24). Some calculations in Tab. (6.5) use the significant distance between two vertices, the equation for calculating this is seen in Eq. (6.4)

$$d^{sig}(vtx1, \epsilon(vtx1), vtx2, \epsilon(vtx2)) = \frac{|(vtx2 - vtx1)|}{\sqrt{\begin{bmatrix}(vtx2_x - vtx1_x)^2\\(vtx2_y - vtx1_y)^2\\(vtx2_z - vtx1_z)^2\end{bmatrix} \cdot \begin{bmatrix}(\epsilon(vtx2)_x - \epsilon(vtx1)_x)^2\\(\epsilon(vtx2)_y - \epsilon(vtx1)_y)^2\\(\epsilon(vtx2)_z - \epsilon(vtx1)_z)^2\end{bmatrix}}} \tag{6.4}$$

6.2.4 *Feature Importance*

For any Ml model, it is standard procedure to do feature importance to see which features impact the model the most. This thesis uses tree feature importance techniques and then combines the feature importance into one.

Native LightGBM Feature Importance

LightGBM uses the "split" to determine feature importance. The GBDT algorithm splits each tree based on a feature to minimize the loss function. Split-feature importance is the total number across the ensemble the GBDT has used a specific feature to split a tree. The idea is the features used frequently are important to the model.

Permutation Feature Importance

The algorithm for permutation feature importance is:
(1) apply a trained model; *Model* on a chosen set: X. Then one uses a chosen metric: *METRIC* to benchmark the performance; P.
(2) Shuffles a single feature randomly and re-apply *Model* again.
$X \xrightarrow{\text{Shuffle feature i}} X'$, METRIC(MODEL(X')) = P'.
(3) Compare increase/drop in performance: P-P'. Step 2 is done n times to reduce statistical fluctuations.
This method is very computationally expensive. However, this feature importance shows a direct correlation between outputted increase/decrease for any feature for any chosen Metric. The metric mainly used in this thesis is the AUC score.

Table 6.5: Engineered features used in Analysis.

$p_T(e_1)$:	$positron\_pT = BeeKst\_electron0\_pT$
$p_T(e_2)$:	$electron\_pT == BeeKst\_electron1\_pT$
$p_T(trk_1)$:	$trackPlus\_pT = BeeKst\_meson0\_pT$
$p_T(trk_2)$:	$trackMinus\_pT = BeeKst\_meson1\_pT$
$\eta(e_1)$:	$positron\_eta = BeeKst\_electron0\_eta$
$\eta(e_2)$:	$electron\_eta == BeeKst\_electron1\_eta$
$\eta(trk_1)$:	$trackPlus\_eta = BeeKst\_meson0\_eta$
$\eta(trk_2)$:	$trackMinus\_eta = BeeKst\_meson1\_eta$
$\phi(e_1)$:	$positron\_phi = BeeKst\_electron0\_phi$
$\phi(e_2)$:	$electron\_phi == BeeKst\_electron1\_phi$
$\phi(trk_1)$:	$trackPlus\_phi = BeeKst\_meson0\_phi$
$\phi(trk_2)$:	$trackMinus\_phi = BeeKst\_meson1\_phi$
$ISO_{c40}(e_1)$:	$positron\_iso\_c40 = BeeKst\_electron0\_iso\_c40$
$ISO_{c40}(e_2)$:	$electron\_iso\_c40 == BeeKst\_electron1\_iso\_c40$
$ISO_{c40}(trk_1)$:	$trackPlus\_iso\_c40 = BeeKst\_meson0\_iso\_c40$
$ISO_{c40}(trk_2)$:	$trackMinus\_iso\_c40 = BeeKst\_meson1\_iso\_c40$
$\chi^2(B)$:	$B\_chi2 = BeeKst\_chi2$
$m(K\pi)$:	$diMeson\_Kpi\_mass = BeeKst\_kaonPion\_mass$
$m(\pi K)$:	$diMeson\_piK\_mass = BeeKst\_pionKaon\_mass$
$m(K\pi) - m(\pi K)$:	$diMeson\_Kpi\_piK\_mass\_diff = (diMeson\_Kpi\_mass - diMeson\_piK\_mass$
$Mean(m(K\pi), m(\pi K))$:	$diMeson\_Kpi\_piK\_mass\_avg = (diMeson\_Kpi\_mass + diMeson\_piK\_mass)/2$
$p_T(B)$:	$B_pT = vtx\_Bd\_p4.pt, (\sqrt{p_x^2 + p_y^2})$
a_0^{sig}:	$a0\_significance = BeeKst\_a0\_minA0/dBeeKst\_a0\_minA0\_err$
z_0^{sig}:	$z0\_significance = BeeKst\_z0\_minA0/dBeeKst\_z0\_minA0\_err$
L_{xy}^{sig}:	$Lxy\_significance = BeeKst\_Lxy\_minA0/dBeeKst\_Lxy\_minA0\_err$
$P - value(ee)$:	$diElectron\_pvalue = 1 - CDF_{\chi^2}(\chi^2 = BeeKst\_diElectron\_chi2, nDoF = BeeKst\_diElectron\_nDoF)$
$P - value(K\pi)$:	$diMeson\_pvalue = 1 - CDF_{\chi^2}(\chi^2 = BeeKst\_diMeson\_chi2, nDoF = BeeKst\_diMeson\_nDoF)$
$d_{vtx}^{sig}(B - K\pi)$:	$d\_B\_diMeson\_significance = d^{sig}(vtx\_Bd\_vtx, vtx\_Bd\_vtx\_err, vtx\_diMeson\_vtx, vtx\_diMeson\_vtx\_err)$ Eq. (6.4)
$d_{vtx}^{sig}(B - ee)$:	$d\_B\_diElectron\_significance = d^{sig}(vtx\_Bd\_vtx, vtx\_Bd\_vtx\_err, vtx\_diLepton\_vtx, vtx\_diLepton\_vtx\_err)$ Eq. (6.4)
$d_{vtx}^{sig}(ee - K\pi)$:	$d\_diMeson\_diElectron\_significance = d^{sig}(vtx\_diMeson\_vtx, vtx\_diMeson\_vtx\_err, vtx\_diLepton\_vtx, vtx\_diLepton\_vtx\_err)$ Eq. (6.4)
$\Delta R(K\pi)$:	$dR\_trackPlus\_trackMinus = \sqrt{((\Delta\phi)^2 + (\Delta\eta)^2)}$, $\Delta\phi = (vtx\_m0\_K\_p4 - vtx\_m1\_pi\_p4)_\phi$ and $\Delta\eta = (vtx\_m0\_K\_p4 - vtx\_m1\_pi\_p4)_\eta$
$\alpha(K\pi)$:	$diMeson\_angle\_alpha\_sym = \lvert \frac{\pi}{2} - \cos^{-1}\left(\frac{vtx\_n1 \cdot vtx\_Kpi\_p4}{\lvert vtx\_n1\rvert \lvert vtx\_Kpi\_p4\rvert}\right) \rvert$
$\beta(K\pi)$:	$diMeson\_angle\_beta\_sym = \lvert\lvert \frac{\pi}{2} - \cos^{-1}\left(\frac{vtx\_n3\_pmm \cdot vtx\_B2diMeson}{\lvert vtx\_n3\_pmm\rvert \lvert vtx\_B2diMeson\rvert}\right) \rvert - \frac{\pi}{2} \rvert$
$\alpha(ee)$:	$diElectron\_angle\_alpha\_sym = \lvert \frac{\pi}{2} - \cos^{-1}\left(\frac{vtx\_n1 \cdot vtx\_diLepton\_p4}{\lvert vtx\_n1\rvert \lvert vtx\_diLepton\_p4\rvert}\right) \rvert$
$\beta(ee)$:	$diMeson\_angle\_beta\_sym = \lvert\lvert \frac{\pi}{2} - \cos^{-1}\left(\frac{vtx\_n3\_pee \cdot vtx\_B2diLepton}{\lvert vtx\_n3\_pee\rvert \lvert vtx\_B2diLepton\rvert}\right) \rvert - \frac{\pi}{2} \rvert$
$(\frac{dE}{dx})_{t1}/(\frac{dE}{dx})_{t2}$:	$tracks\_dEdx\_ratio = \frac{BeeKst\_meson0\_pixeldEdx}{BeeKst\_meson1\_pixeldEdx}$
$(\frac{dE}{dx})_{t1} - (\frac{dE}{dx})_{t2}$:	$tracks\_dEdx\_diff = BeeKst\_meson0\_pixeldEdx - BeeKst\_meson1\_pixeldEdx$
$\angle(ee - K\pi \lvert ee)$-plane:	$angle\_vtx_plane\_ee\_plane = \cos^1\left(\frac{vtx\_n1 \cdot vtx\_n2\_ee}{\lvert vtx\_n1\rvert \lvert vtx\_n2\_ee\rvert}\right)$
$\angle(ee - K\pi \lvert K\pi)$-plane:	$angle\_vtx\_plane\_mm\_plane = \cos^1\left(\frac{vtx\_n1 \cdot vtx\_n2\_mm}{\lvert vtx\_n1\rvert \lvert vtx\_n2\_mm\rvert}\right)$
$\angle(ee \lvert K\pi)$-plane:	$angle\_vtx\_plane\_mm\_plane = \cos^1\left(\frac{vtx\_n2\_ee \cdot vtx\_n2\_mm}{\lvert vtx\_n2\_ee\rvert \lvert vtx\_n2\_mm\rvert}\right)$
$\angle(Q(ee, K\pi) \lvert K\pi)$-plane:	$angle\_pp\_plane\_mm\_plane = \cos^1\left(\frac{vtx\_n1\_pp \cdot vtx\_n2\_mm}{\lvert vtx\_n1\_pp\rvert \lvert vtx\_n2\_mm\rvert}\right)$
$\angle(Q(ee, K\pi) \lvert ee)$-plane:	$angle\_pp\_plane\_ee\_plane = \cos^1\left(\frac{vtx\_n1\_pp \cdot vtx\_n2\_ee}{\lvert vtx\_n1\_pp\rvert \lvert vtx\_n2\_ee\rvert}\right)$

SHAP Feature Importance

SHAP values or (SHapley Additive exPlanations) is a way to measure feature importance which comes from cooperative game theory[38]: The concept is as follows: The Shapley value for a player (a feature) is a measure of the average marginal contribution for a player to all possible coalitions (coalitions are the subsets of features) that could be formed in the game.
If X is a feature vector, $f(X)$ is the output of the model and $S \subset X$ with indices $\{1, 2, \ldots, n\}$ so which means $f(S)$ is the output of the model only with features contained in S. SHAP value for feature i is calculated the following way: The SHAP values are calculated as seen in Eq. (6.5) where one sums over all subsets S where feature i is possible to be in. One then does a linear regression between the original model: $f(x)$ and $g(x')$, which has the features alternated such that $f(x) \approx g(x') = \phi_0 \sum_i \text{SHAP-value}_i z'_i$ where z'_i is the coefficient for feature i.

$$\text{SHAP-value}_i(X) = \frac{1}{(2^n - 1)} \sum_{S, i \in S} (f(S) - f(S \setminus i)) \tag{6.5}$$

Lastly, in this thesis, the SHAP values were used to find global interpretability by taking the mean of the absolute value of the SHAP-value$_i$ values: Eq. (6.6). The hypothesis is that large absolute Shapley values mean that the feature has a big contribution and is important.

$$\text{Importance of feature J} = \frac{1}{n} \sum_{i}^{n} \mid \text{SHAP-value}_J^{(i)} \mid \tag{6.6}$$

Summed Feature Importance

A summed feature importance is created by scaling[10] each of the three feature importance between $[0, 1]$. The scaled features are then summed feature-wise into a summed feature importance.

[10] The actual feature importance value is not interesting, only the order of the important features and the difference between them.

6.2.5 *Fitting Routine*

Since SR^{data} in q^2_{hig} is un-blinded at the current stage of the RK* analysis, the extraction of the signal yield is done the following way:
Different background mass-cut candidates are scaled to SB1^{data} in q^2_{high} to indicate which are the most similar. After this, different background probability function candidates are fitted to the just-found background distribution, and the best candidate is selected; PDF_{Bkg}.
A signal probability distribution are fitted to the SR^{MC} in q^2_{high}; PDF_{Sig}. The combined Signal and Background PDF; $PDF_{Sig} + PDF_{Bkg}$ are then blindly fitted to a grid of different GBDT thresh-

olds in the un-blinded $\mathrm{SR}^{\mathrm{data}}$ q^2_{hig} and the set of GBDT cuts which maximizes: Significance $= \frac{N_{sig}}{\sqrt{N_{sig}+N_{Bkg}}}$ is used. Then PDF_{Bkg} is fitted to maximum significance GBDT-cut background distribution, and the same is done for the PDF_{Sig} on maximum significance GBDT-cut $\mathrm{SR}^{\mathrm{MC}}$ in q^2_{high}. At last, the combined signal and background fit; $PDF_{Sig} + PDF_{Bkg}$ on maximum significance GBDT-cut $\mathrm{SR}^{\mathrm{data}}$ q^2_{high}, and the signal yield is extracted.

7
GNN to GBDT

The first configurations used with GBDTs are the "*2GNN to 2GBDT*"-approach which is a translation of the GNN approach used by the RK* group where the GNNs are substituted with GBDTs. The number of training samples used is seen in Tab. (7.1). The total amount of truth-matched signals is 20000 where each sideband contains 700000 event candidates[1].

[1] This number of events for both signal and background are also used for the GNNs.

Table 7.1: Configurations for "2GNN to 2GBDT"

Number of Events used in training	
Number of Events	Data-set Info
10000	Non-Resonant B^0_d, q^2_{low} (DSID: 300590)
10000	Non-Resonant $\overline{B}^0_d$, q^2_{low} (DSID: 300591)
700000	Sideband1 (Period K, q^2_{low})
700000	Sideband2 (Period K, q^2_{low})

The training of the two GBDTs was executed using: Verstack[73][2] and the number of trials trained was the default 100 for each GBDT, and the training time for 25-35 CPU cores was about $\sim$ 10 min. The input features used for training have the distributions seen in Fig. (7.1), which are log-scaled and normalized. The data are separated into SR^{MC}, $SB2^{data}$, and $SB2^{data}$. This is so it is possible to see the features normalized with no y-xis log-scale in Fig. (A.1) in appendix A.2.

[2] LightGBM integrated with Optuna.

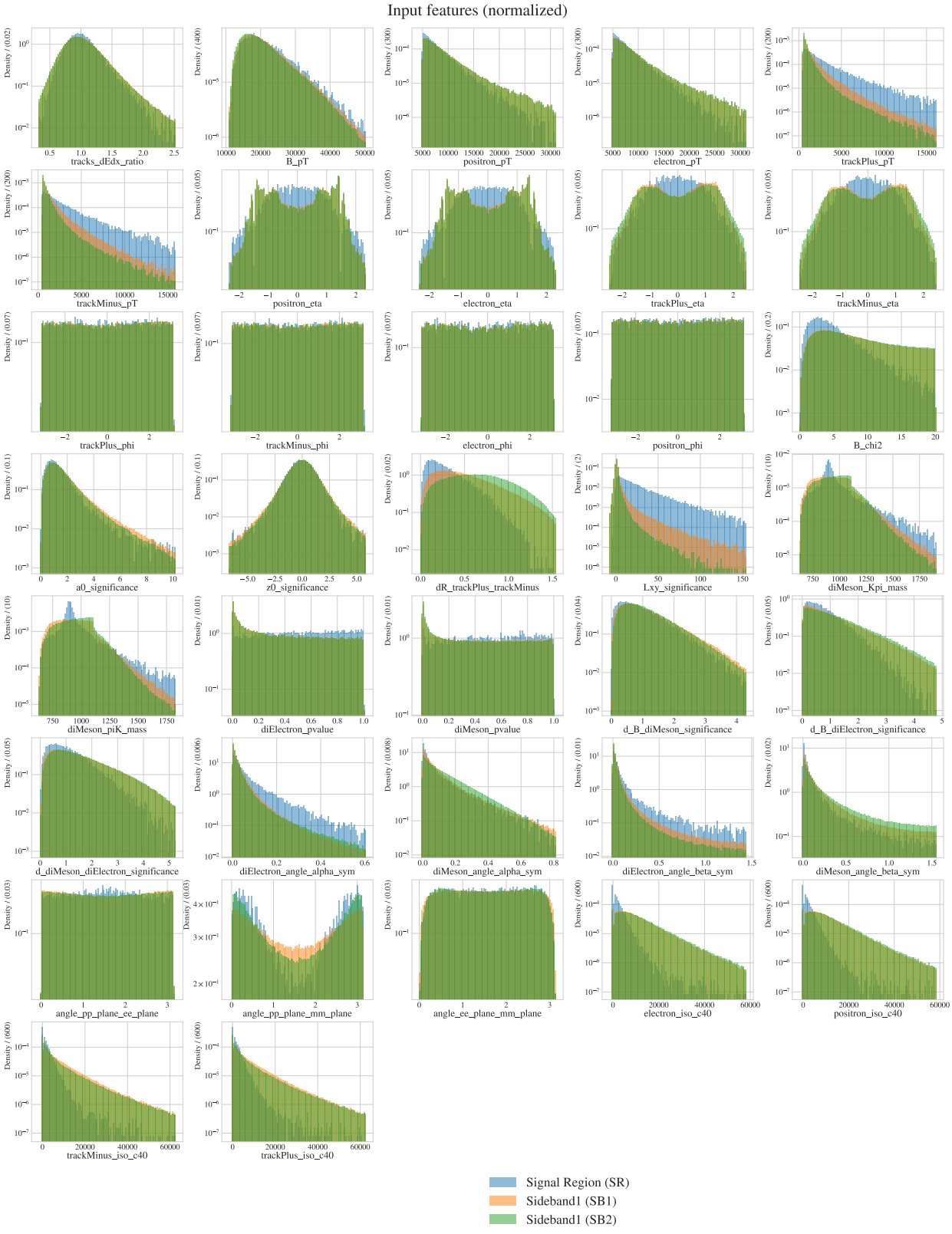


Figure 7.1: Normalized, log-scaled features used in "2GNN to 2GBDT". The plotted features are all separated into Signal Region; SR^{MC}, Sideband1; $SB1^{data}$, and Sideband2; $SB2^{data}$.

The training of the 2 GBDTs is initialized with multi-log loss. For reproducibility, the specific parameters for the LightGBM classi-

fication boosters (both GBDT1 and GBDT2) are seen in Tab. (A.1) in appendix A.2. If nothing else is stated, the default values in LightGBM and Optuna are used along with the initialization of seed-value= 42 when it is possible to provide a seed.

After training, the *LightGBM* Testing Suite was applied on both GBDTs, which are seen in Fig. (7.2) and Fig. (7.3), as Fig. (7.3) is small a larger version can be seen in Fig. (A.3) in appendix A.2. The main takeaway is that the Optimization History Plot for both GBDTs has its objective lowered[3] as well as the Intermediate Values plot also has nice convergences. From this, it can be concluded that GBDT1 and GBDT2 on "2GNN to 2GBDT" both converged to a minimum, and training was successful with *num_leaves* and *min_sum_hessian_in_leaf* as the most important hyperparameter for both GBDTs during hyperparameter optimization.

[3] Note that for GBDT1, the #Trials stop at 50 since it found a minimum and early stopping were triggered to avoid over-fitting.

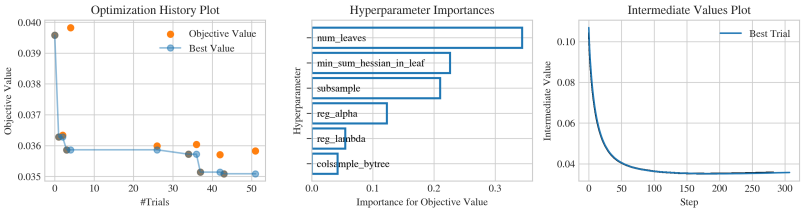


Figure 7.2: *LightGBM* Testing Suite on "2GNN to 2GBDT"-GBDT1 which shows the training of GBDT1 is succesfull.

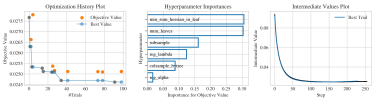


Figure 7.3: *LightGBM* Testing Suite on "2GNN to 2GBDT"-GBDT2 which shows the training of GBDT2 is succesfull.

For both GBDTs, the *Signal vs. Background* is applied on the test set (see Fig. (7.4) and Fig. (7.5)). Since it is the test set and later this test will be applied to all events where all selection rules are applied, only a few notes on these figures: The separation is quite good overall with $AUC = 0.993$ and $AP = 0.901$ for GBDT1 and $AUC = 0.996$ and $AP = 0.956$ for GBDT2 with GBDT2 slightly better overall that GBDT1 in the test set.

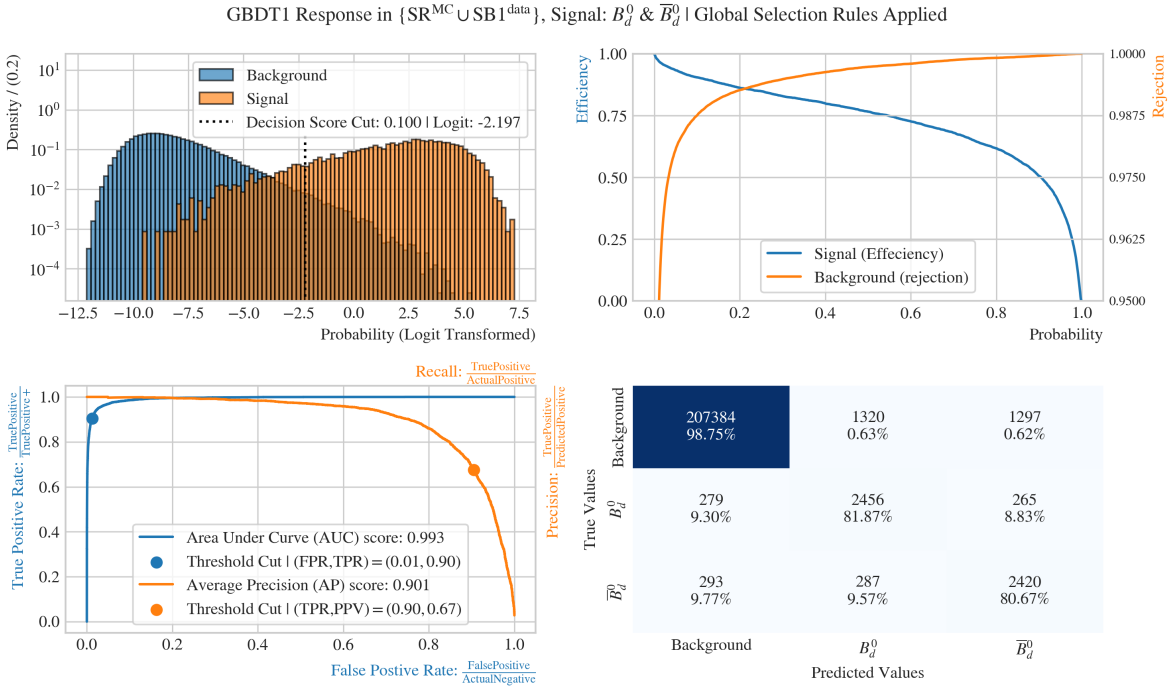


Figure 7.4: *Signal vs Background* Testing Suite on "2GNN to 2GBDT"-GBDT1 in $\{SR^{MC} \cup SB1^{data}\}$ test set. The GBDT1 shows a good performance in separating Signal(B^0+$\overline{B}^0$) vs. Background with $AUC = 0.993$ and $AP = 0.901$.

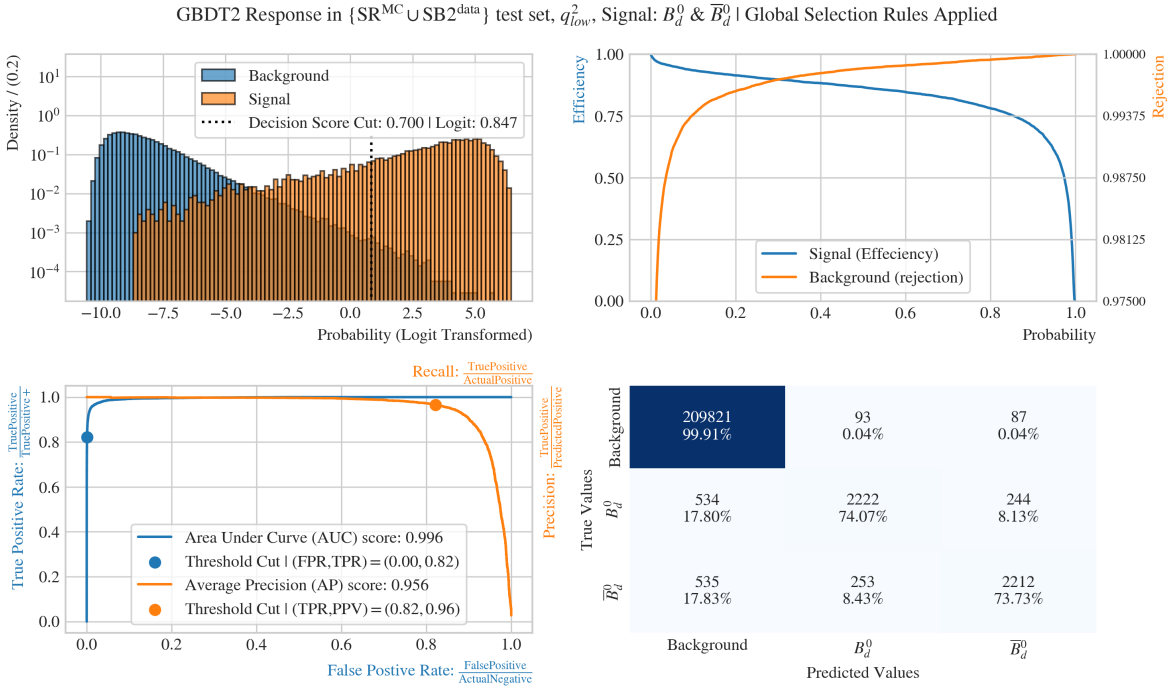


Figure 7.5: *Signal vs Background* Testing Suite on "2GNN to 2GBDT"-GBDT2 $\{SR^{MC} \cup SB2^{data}\}$ test set. The GBDT2 shows a good performance in separating Signal(B^0+$\overline{B}^0$) vs. Background with $AUC = 0.996$ and $AP = 0.956$.

The next test on the test-set is the *Sig*(B^0) *vs. Sig*($\overline{B}^0$) Testing Suite. The figures are seen in Fig. (7.7) and Fig. (7.8). Please refer to Fig. (7.6) for an interpretation of the 2D response curves.
For both Fig. (7.7) and Fig. (7.8), there are high-density spots at

$(0,0)$, $(0,1)$ and $(0,0)$, $(0,1)$ which means both GBDTs predicts with high precision, however, the events which are in $(0,0)$ signifies that the GBDT does not capture all aspects of the signal properties so some signal is strongly classified as background with $\sim$ 100% probability.

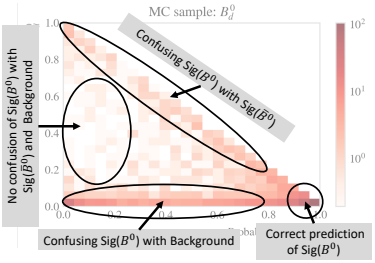


Figure 7.6: An interpretation of 2D Response Curves with Sig(B^0): Fig. (7.7)(left-most) as an example. Source of figure inspiration: [39].

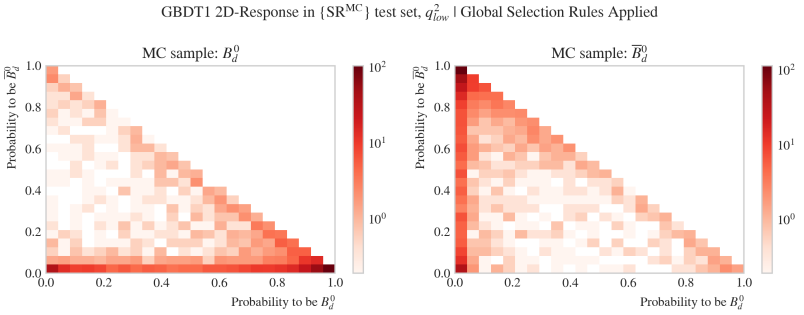


Figure 7.7: *Sig(B^0) vs Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT"-GBDT1 with density on log-scale. The figure shows good B^0 and $\overline{B}^0$ selection for GBDT1 in MC.

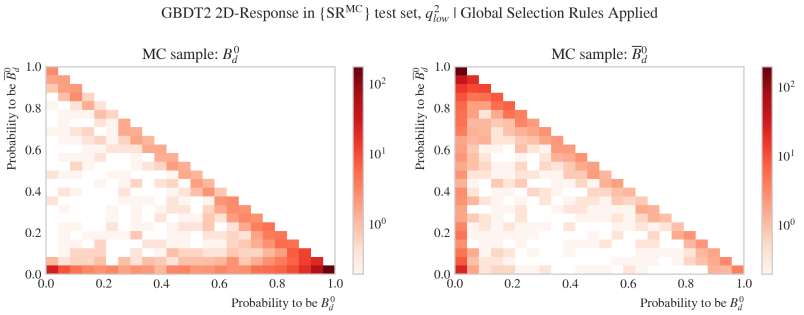


Figure 7.8: *Sig(B^0) vs Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT"-GBDT2 with density on log-scale. The figure shows good B^0 and $\overline{B}^0$ selection for GBDT2 in MC.

The last use-case for the test set is feature Importance, and these are seen in Fig. (7.9) and Fig. (7.10). Both the train- and test sets are used for SHAP and Permutation Importances since it is interesting to see if they align in the train and test sets. At the same time, the training-set feature importance is for seeing which features are important under training, and the test-set is purely a measure of feature importance under prediction, which are the main feature importance of interest and also the one that the summed feature importance is based on. The most important features for both GBDT1 and GBDT2 are: [L_xy_significance, diMeson_Kpi_mass, diMeson_piK_mass].

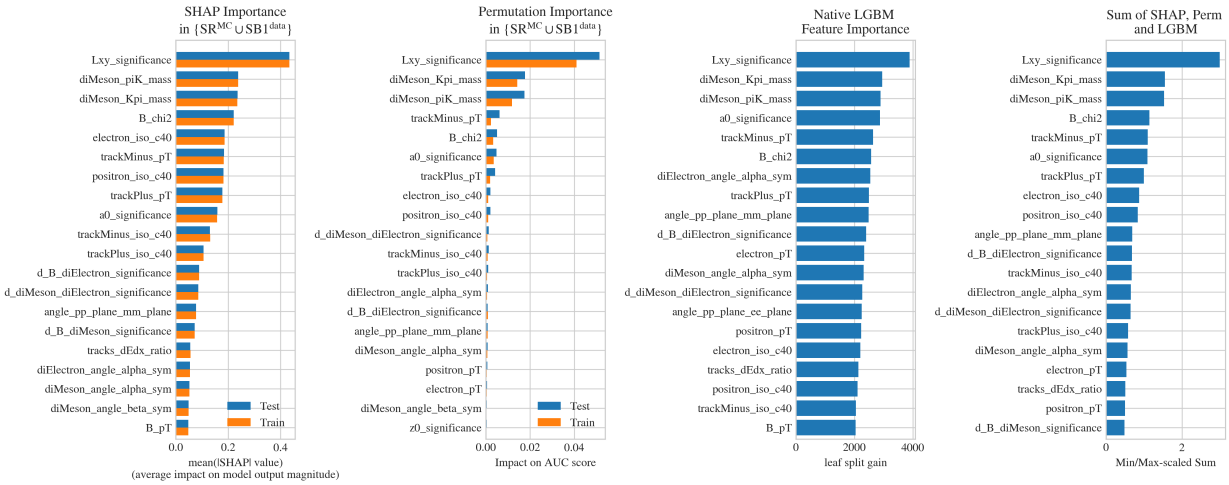


Figure 7.9: 20 of the highest scoring feature importances on "2GNN to 2GBDT"-GBDT1 for both training-set and test-set. The five most important features in the summed feature importance are: [L_xy_significance, diMeson_Kpi_mass, diMeson_piK_mass, B_chi2, trackMinus_pT]

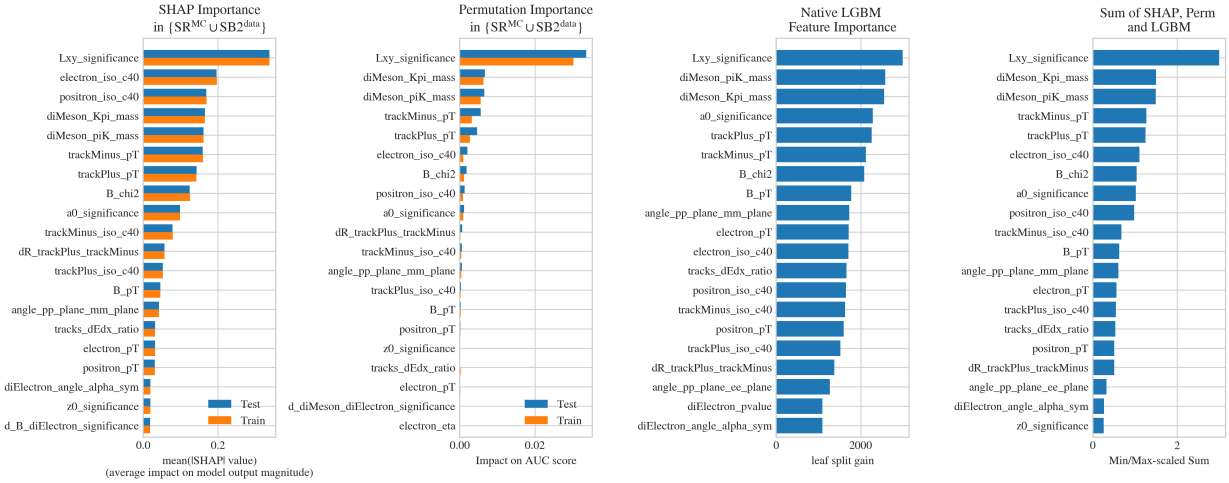


Figure 7.10: 20 of the highest scoring feature importances on "2GNN to 2GBDT"-GBDT2 for both training-set and test-set. The five most important features in the summed feature importance are: [L_xy_significance, diMeson_Kpi_mass, diMeson_piK_mass, trackMinus_pT, trackPlus_pT]

The "2GNN to 2GBDT"-GBDTs pass all testing suites for the train set. However, the "real" test is in the actual $\{SR^{MC} \cup SB1^{data}\}$ and $\{SR^{MC} \cup SB2^{data}\}$ sets where MC is not truth-matched, and all candidates for sidebands are preset. This means the next test needs to check for the performance of the *Global Candidate Selection*, *Local Candidate Selection*, and *Mass Hypothesis Selection*. The number of data used in the "real" test is seen in Tab. (7.2).

Number of Events after main CUTS and region-mass cuts.		
Data-set	Number of Events	Data-set Info
SR =	803317	ALL MC Signal Files ∣ main CUTS ∣ SR-cut ∣ q^2_{low}
SB1 =	2036874	ALL Period K Files ∣ main CUTS ∣ SB1-cut ∣ q^2_{low}
SB2 =	2036874	ALL Period K Files ∣ main CUTS ∣ SB2-cut ∣ q^2_{low}

Table 7.2: Configurations for applying "2GNN to 2GBDT"-GBDTs after they have been tested on the test set.

The *Signal vs. Background* Testing Suite and *Sig(B^0) vs. Sig($\overline{B}^0$)* Test-

ing Suite are applied on the SR^{MC}, $SB1^{data}$, and $SB2^{data}$} to check the classification ability of the two GBDTs which now have to do *Local Candidate Selection* on top of *Global Candidate Selection*. The *Signal vs. Background* Testing Suite for both GBDTs in Fig. (7.11) and Fig. (7.12) shows good separation for both GBDTs and they both handle the *Local Candidate Selection* with $AUC = 0.974$ and $AP = 0.947$ for GBDT1 and $AUC = 0.981$ and $AP = 0.965$ for GBDT2. Again GBDT2 has better performance than GBDT1; however, the background rejection is significantly worser for both GBDTs compared to the test set.

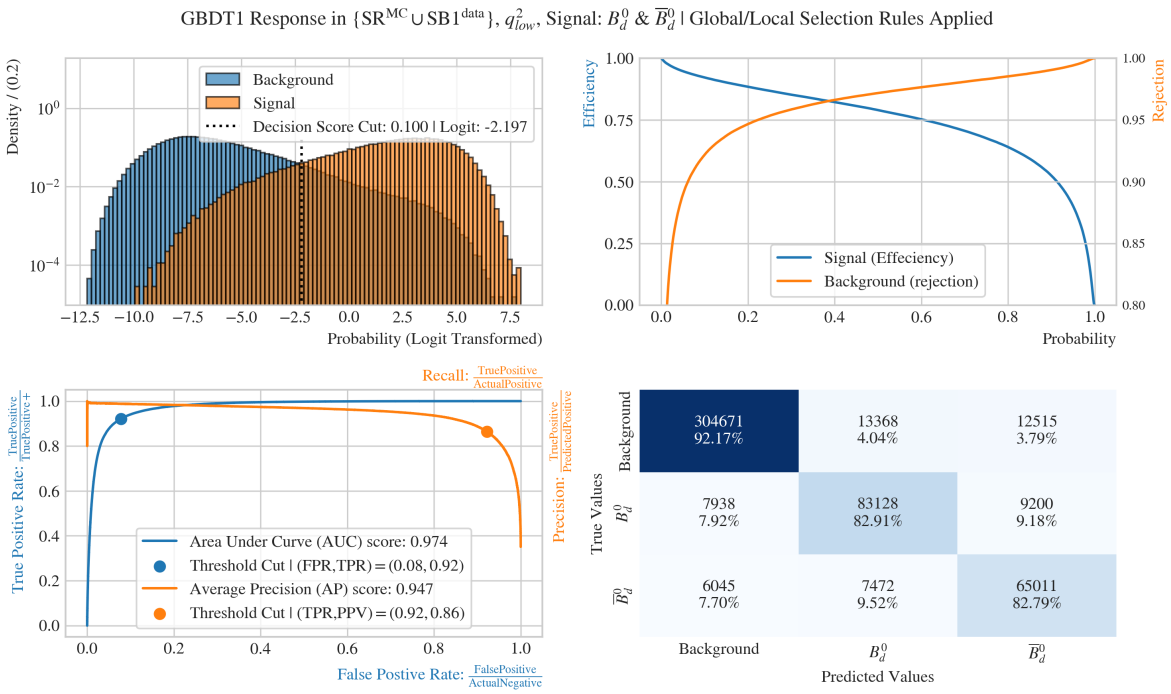


Figure 7.11: *Signal vs Background* Testing Suite with "2GNN to 2GBDT"-GBDT1 on non-train {SR^{MC} ∪ $SB1^{data}$}. The GBDT1 shows a good performance in separating Signal(B^0+$\overline{B}^0$) vs. Background with $AUC = 0.974$ and $AP = 0.947$.

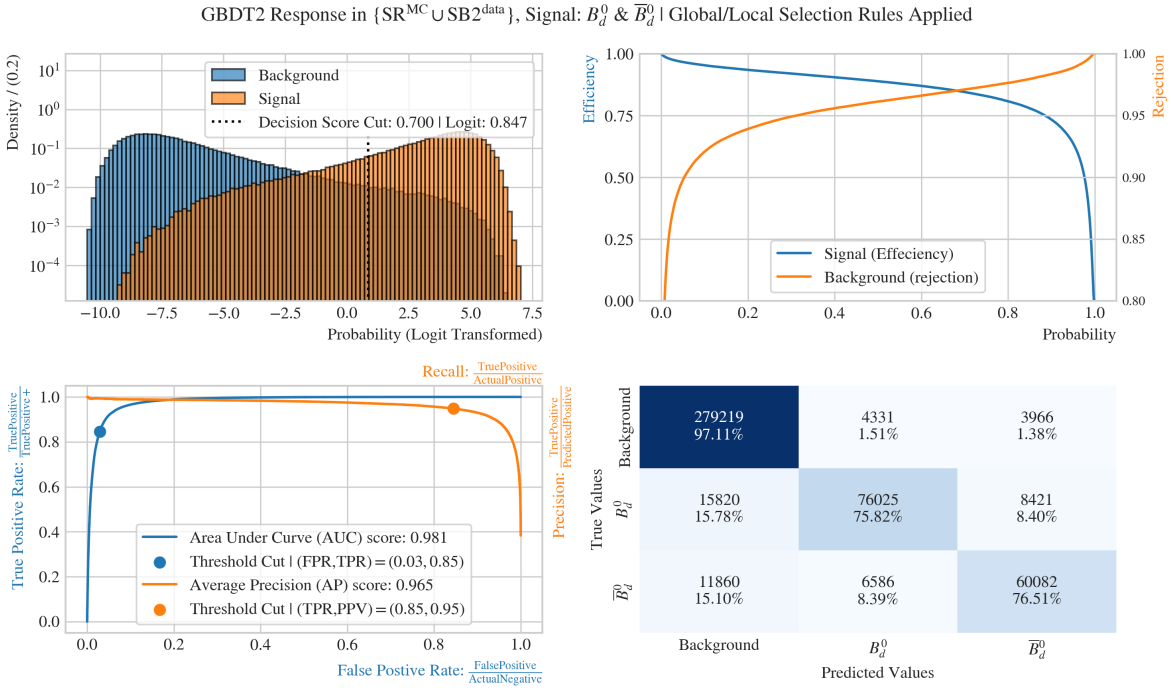


Figure 7.12: *Signal vs Background* Testing Suite with "2GNN to 2GBDT"-GBDT2 on non-train $\{\text{SR}^{\text{MC}} \cup \text{SB2}^{\text{data}}\}$. The GBDT2 shows a good performance in separating Signal(B^0+$\overline{B}^0$) vs. Background with $AUC = 0.981$ and $AP = 0.965$.

Sig(B^0) *vs Sig*($\overline{B}^0$) Testing Suite are seen in the margin figures: Fig. (7.13) and Fig. (7.14). If the reader wants to inspect these on a larger scale, they are found in appendix A.2; Fig. (A.4) and Fig. (A.5).
The takeaway from the testing suite is that the density at $(1,0)$ and $(1,0)$ for B^0_d and $\overline{B}^0_d$ respectively in both Fig. (7.13) and Fig. (7.14) is very high with respect to the rest of the plot. Just as for the train-set *Sig*(B^0) *vs. Sig*($\overline{B}^0$)-test, again, there are signals strongly classified as background at $(0,0)$ just as with the test-sets.

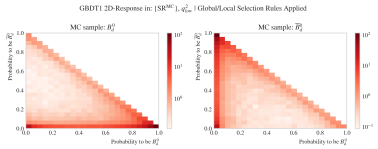


Figure 7.13: *Sig*(B^0) *vs Sig*($\overline{B}^0$) Testing Suite on "2GNN to 2GBDT"-GBDT1 with density on log-scale.

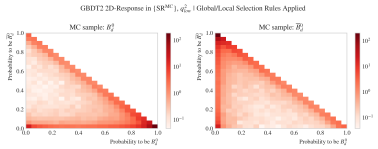


Figure 7.14: *Sig*(B^0) *vs Sig*($\overline{B}^0$) Testing Suite on "2GNN to 2GBDT"-GBDT2 with density on log-scale.

The last of the testing suite is the *Mass Shape* Testing Suite, which is an application of all three candidate selections: *Global-*, *Local-*, and *Mass Hypothesis Candidate Selection*. As seen in Fig. (7.15), a distortion of the signal is present as a drop in efficiency which is seen in the lower part of the plot due to a threshold-cut scan. The shape at the peak for both GBDT1 and GBDT2 for B^0_d-decay has nearly no distortion. However, the sides have a more significant drop in efficiency - especially in the upper part of the B mass: $m(B) > 5300\,\text{MeV}/c^2$. There is also a drop in efficiency on the right side: $m(B) < 5000\,\text{MeV}/c^2$; however, the drop-off is less steep. The result for both GBDTs is that the tighter the cut is in the Global Candidate Selection, the sharper the peak is for the Signal.
For the background in Fig. (7.16), there is no sculpting seen. This means the GBDTs do not learn any particular shape in the background, and it stays flat in efficiency.
For $m(ee)$ in 7.17, there is also a very flat efficiency line hence no

distortion for $m(ee)$ in q^2_{low}-bin.

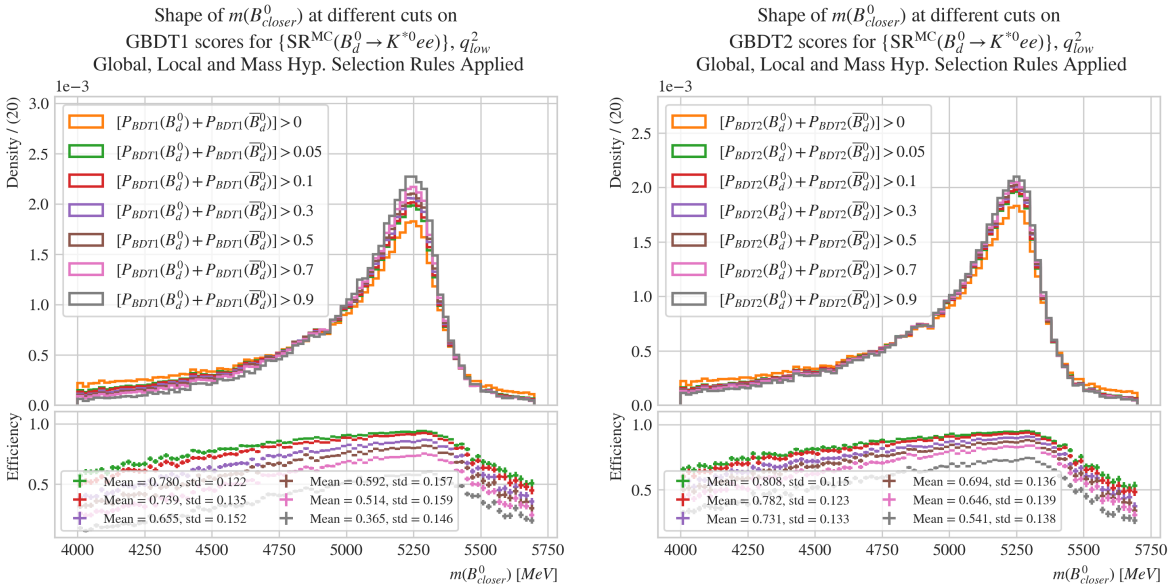


Figure 7.15: *Mass Shape* Testing Suite for Signal on "2GNN to 2GBDT". As seen, the signal distribution shape is retained around $m(B^0) \in 5250\,\text{MeV}/c^2$ and then falls off as seen in the efficiency in the lower part. Especially in the upper part of the B-mass, there is a sharp drop-off. This distortion is not dire, however, it could lead to a larger error in the extracted signal yield.

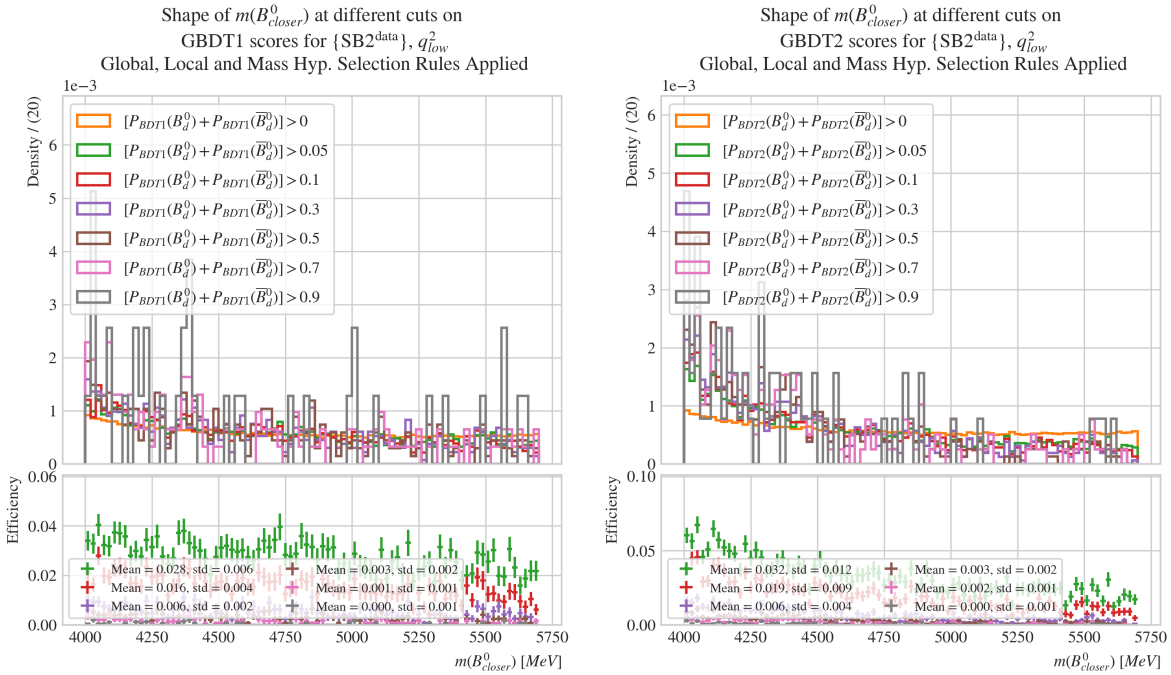


Figure 7.16: *Mass Shape* Testing Suite for background on "2GNN to 2GBDT". As seen, there is little to no background sculpting which is seen in the efficiency spread in the lower plot for the different cuts.

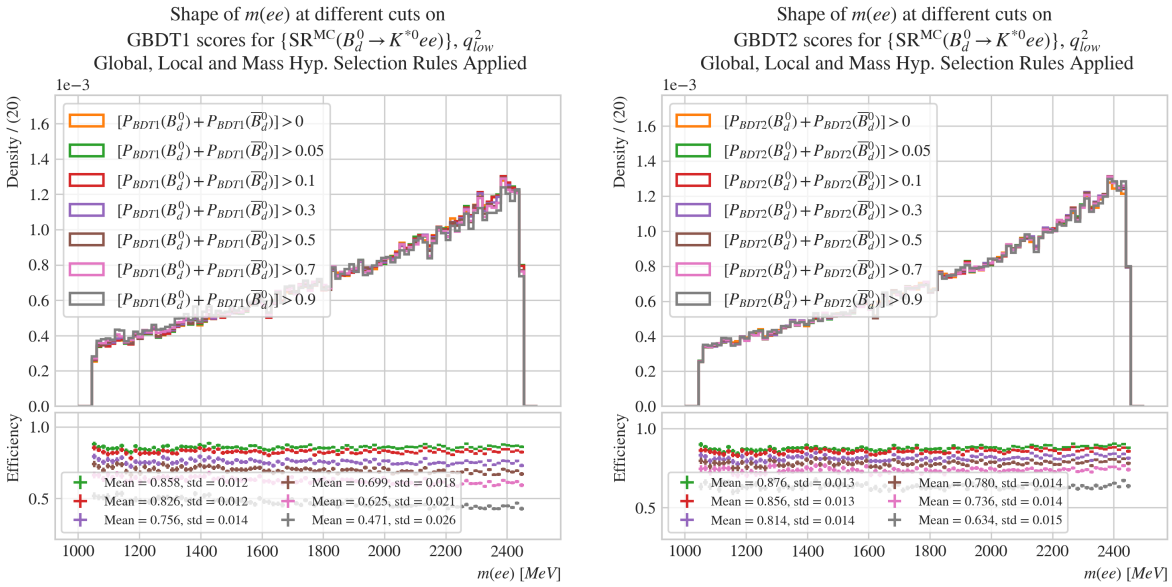


Figure 7.17: *Mass Shape* Testing Suite for $m(ee)$ on "2GNN to 2GBDT". As seen in the efficiency, there is no sculpting in the $m(ee)$-shape.

As the GBDT are trained on non-resonant decays, it is important to see if there is a discrepancy between resonant and non-resonant in monte carlo. This is seen in Fig. (7.18), which depicts that both efficiency lines are almost on top of each other; this means GBDTs have not learned a difference in the two q-bins. For a larger image of Fig. (7.18), please see Fig. (A.6) in appendix A.2.

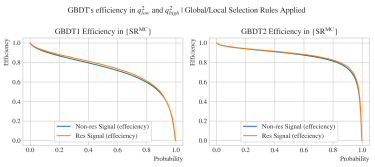


Figure 7.18: Resonant vs Non-Resonant signal efficiency plots for both the q^2_{low} and q^2_{high} bin for both GBDTs in "2GNN to 2GBDT".

7.0.1 *Benchmarking*

For benchmarking, Fig. (7.18) and the benchmark signal efficiency plot: Fig. (5.3)(c-d) are overlaid, and the result is seen in Fig. (7.19). The figure shows that the "2GNN to 2GBDT"-approach has better signal efficiency than the GNNs except for probabilities over 0.92. From $[0.92, 1.00]$, the GNNs beats the GBDTs in signal efficiency. In the following section, efforts are made to get a higher efficiency than the GNNs for all probabilities.

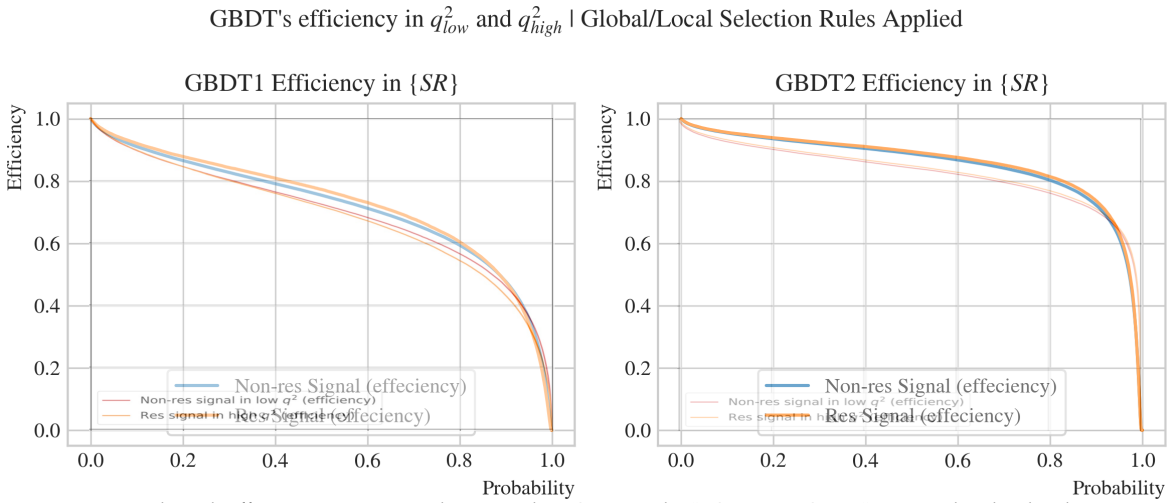


Figure 7.19: Benchmark efficiency comparison between the 2GNN vs the "2GNN to 2GBDT"-approach. The thin lines are the GNNs and the thick line is the GBDTs. The GBDTs have higher signal efficiency for all probability thresholds except for $P \in [0.92, 1.00]$.

7.0.2 *Fitting and the Signal Yield*

The following fitting steps are used to extract the B^0-signal yield in the un-blinded SR^{data}, q^2_{high} for Period K data.

(1): Scale potential backgrounds to blinded SB1^{data}, q^2_{high} and find best χ^2.
(2): Find the best probability distribution that fits the background, which scales best to SB1^{data}.
(3): Fit signal probability distribution to SR^{MC}, q^2_{high}.
(4): Do blinded Sig+Bkg fit significance scan over a grid of GBDT cuts in SR^{data}, q^2_{high} Period K data. Locate the GBDT cuts which maximize: significance $= \frac{N_{Sig}}{\sqrt{N_{Sig}+N_{Bkg}}}$.
(5): Redo individual background probability distribution fit and signal probability distribution fit on their respective distributions where the distributions are cut with the GBDT cuts, which maximizes the significance. The SR range are reduced from $[4000, 57000]$ to $[4250, 5700]$ and is denoted SR^* for more precise fits.
(6): Fit the total Sig+Bkg to the maximum GBDT cut on SR^{data}, q^2_{high} Period K data in and extract the signal yield: N_{Sig}.

The fit-routine is by the Iminuit Python library[15] where the MIGRAD-routine[31] is called twice to do χ^2-fits. The data used in the fitting are seen in Tab. (7.2). [4] After the two MIGRAD minimization routines have been used to minimize the χ^2, the HESSE algorithm is run to calculate the Hessian matrix at the minimum χ^2 to get accurate uncertainties on the fitting parameters.
For *step (1)*, the different potential backgrounds are scaled by multiplying the distributions with a scaling parameter, α, and then doing a χ^2-fit. The idea is to find the background which looks most like the background in SB1^{data}, q^2_{high}, and the lowest χ^2 value quantifies the best fit. As seen in Fig. (7.20), the lowest value is $\chi^2 = 2439.14$, which is the SB2^{data} in data. This means the fitting of the back-

[4] All histograms has Poisson errors.

ground distribution is done on $\mathrm{SB2}^{\mathrm{data}}$, q^2_{high}.

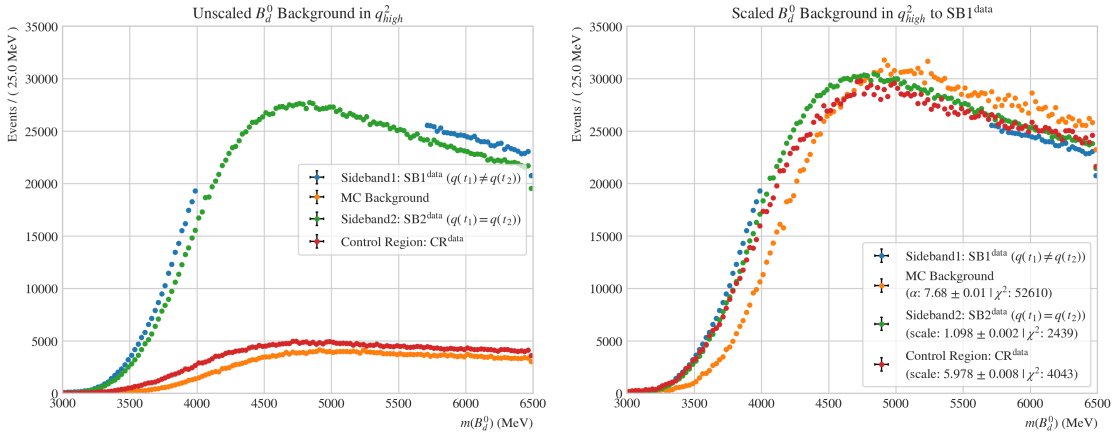


Figure 7.20: Background scaling to $\mathrm{SB2}^{\mathrm{data}}$, q^2_{high} using χ^2 fit with a single scaling parameter: α. The righthand side is the unscaled distribution. The lefthand side is the scaled distributions where the $\mathrm{SB2}^{\mathrm{data}}$, q^2_{high} is the distribution which comes closest to $\mathrm{SB1}^{\mathrm{data}}$, q^2_{high} with $chi^2 = 2449$ and $\alpha = 1.098 \pm 0.002$.

All background fits are done within the SR-$m(B^0)$ window, and each function used is individually normed such they are truncated distributions in the SR-$m(B^0)$ window. The distributions tested are Eq. (7.1), Eq. (7.2), Eq. (7.3), Eq. (7.4), Eq. (7.5) and Eq. (7.6) and this are seen in Fig. (7.21).

$$\tan^{-1}(x) + \mathrm{Pol}_1(x) = w\tanh^{-1}(x') + (1-w)(a(x'') + b) \text{ where: } x' = \frac{x-\mu_{\tan}}{1000}, x'' = \frac{x-\mu_{\mathrm{Pol}_1}}{1000}, w \in [0,1] \tag{7.1}$$

$$\tanh(x) + \exp(x) = w(\tanh(x') + s) + (1-w)\lambda\exp(-\lambda x'') \text{ where: } x' = \frac{x-\mu_{\tanh}}{1000}, x'' = \frac{x-\mu_{\exp}}{1000}, w \in [0,1] \tag{7.2}$$

$$\tan^{-1}(x) + \exp(x) = w(\tanh^{-1}(x') + s) + (1-w)\lambda\exp(-\lambda x'') \text{ where: } x' = \frac{x-\mu_{\tanh}}{1000}, x'' = \frac{x-\mu_{\exp}}{1000}, w \in [0,1] \tag{7.3}$$

$$\mathrm{Pol}_3(x) = a(x')^3 + b(x')^2 + c(x') + d \text{ where: } x' = \frac{x-\mu}{1000} \tag{7.4}$$

$$\mathrm{Pol}_4(x) = a(x')^4 + b(x')^3 + c(x')^2 + d(x') + e \text{ where: } x' = \frac{x-\mu}{1000} \tag{7.5}$$

$$\mathrm{Erf}(x) + \exp(x) = w\left(\frac{2}{\sqrt{\pi}}\int_0^z \mathrm{e}^{-t^2}\mathrm{d}x' + s\right) + (1-w)\lambda\mathrm{e}^{-\lambda x''} \text{ where: } x' = \frac{x-\mu_{\mathrm{erf}}}{1000}, x' = \frac{x-\mu_{\mathrm{epx}}}{1000}, w \in [0,1] \tag{7.6}$$

In Fig. (7.21), one can see all the background proposed background distributions, and the PDF with the lowest χ^2 is the $\mathrm{Erf}(x) + \exp(x)$ background distribution (see Eq. (7.6)). In addition, one can see shape parameters with their errors along with the Goodness-of-Fit (GoF) parameters in Fig. (7.21).

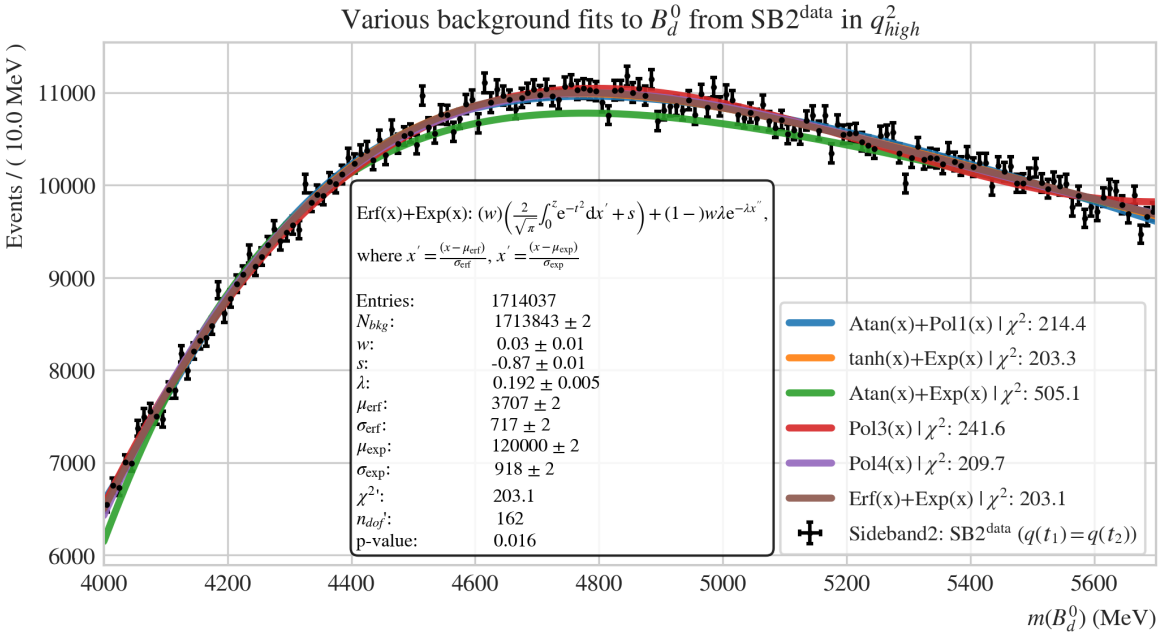


Figure 7.21: Background distribution fits on B^0: SB2$^{\text{data}}$, q^2_{high}. The best fit is a $Erf(x) + Exp(x)$ function. Note that SB2 is reduced into the range: $m(B^0) \in [4000, 5700]^{\text{MeV}/c^2}$.

The RK* group had already proposed signal distribution: a combination of two Gaussian distributions and a Bukin distribution with a shared peak for both Gaussian and the Bukin PDF. The signal PDF is defined in Eq. (7.7) where the Gaussian component is seen in Eq. (7.8) and the Bukin component in Eq. (7.9). Both are individually normed and truncated to the SR-$m(B^0)$ window. The fit of the signal pdf is seen in Fig. (7.22). With a p-value of 0.13 and the MIGRAD-routine returning a convergence report; the fit can be concluded to fit the SR$^{\text{MC}}$, q^2_{high} well.

$$\text{DoubleGauss}(x) + \text{Bukin}(x) =$$
$$w_2 \times ((w_1 \times \text{Gauss}_1(x; \mu_{\text{Peak}}, \sigma_1)) + (1 - w_1) \times \text{Gauss}_2(x|\mu_{\text{Peak}}, \sigma_2)) + ((1 - w_2) \times \text{Bukin}(x|\mu_{\text{Peak}}, \sigma_p, \xi, \rho_1, \rho_2) \tag{7.7}$$

Where :

$$\text{Gauss}_i(x; \mu_{\text{Peak}}, \sigma_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{1}{2}\left(\frac{x - \mu_{\text{Peak}}}{\sigma_i}\right)^2\right) \tag{7.8}$$

$$\text{Bukin}(x; \mu_{\text{Peak}}, \sigma_p, \xi, \rho_1, \rho_2) = \begin{cases} A_1(x; \mu_{\text{Peak}}, \sigma_p, \xi, \rho_1) & \text{if } x \leq x_1 \\ A_2(x; \mu_{\text{Peak}}, \sigma_p, \xi, \rho_2) & \text{if } x \geq x_2 \end{cases} \tag{7.9}$$

$$\text{where} \quad x_{1,2} = \mu_{\text{Peak}} + \sigma_p\sqrt{2\ln 2}\left(\frac{\xi}{\sqrt{\xi+1}} \mp 1\right) \quad \text{and}$$

$$A_i(x; \mu_{\text{Peak}}, \sigma_p, \xi, \rho_i) = \exp\left[\frac{\xi\sqrt{\xi^2+1}(x - x_1)\sqrt{2\ln 2}}{\sigma_p(\sqrt{\xi^2+1} - \xi)^2 \ln(\sqrt{\xi^2+1} + \xi)} + \rho_i\left(\frac{x - x_i}{\mu_{\text{Peak}} - x_i}\right)^2 - \ln 2\right]$$

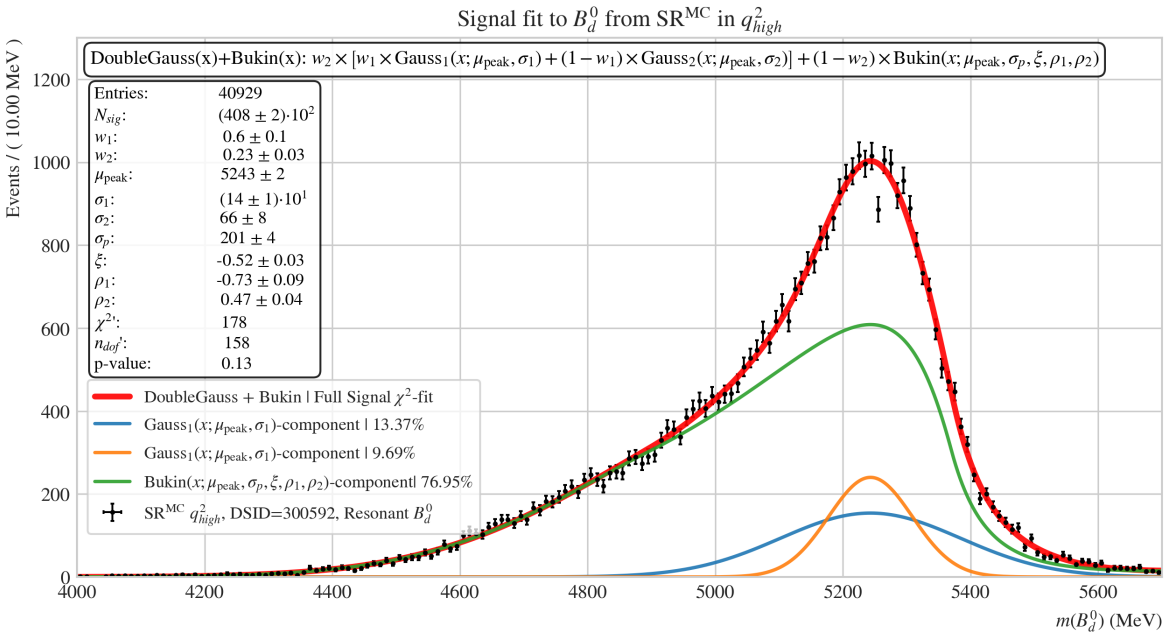


Figure 7.22: Signal fit: Double Gauss + Bukin PDF fit on B_d^0; SR^{MC} in the q^2_{high}-bin.

Following the RK* fitting group procedure, the total fit: Signal PDF + Background PDF has some shape parameters floated, and some are fixed. For the signal PDF: The shared peak; μ_{Peak} are floated. The sigmas $(\sigma_1, \sigma_2, \sigma_p)$ are also floated; however, their ratios are fixed. All other shape parameters are fixed to the MC fit, see Fig. (7.22) for fixed shape parameter values). For the background PDF, all shape parameters are floated.

Step (4) is about the blinded signal+background fit on SR^{data} period K data in the q^2_{high}-bin iteratively creating a significance scan by extracting Significance $= \frac{N_{Sig}}{\sqrt{N_{Sig}+N_{Bkg}}}$. The significance scan is seen in Fig. (7.23) and note that if the scan value is set to zero, the fit did not converge and is a white space on the plot. The blinded significance scan gave the following cuts in the GBDTs: $GBDT1 = 0.1$ and $GBDT2 = 0.7$ with a significance value of $\frac{N_{Sig}}{\sqrt{N_{Sig}+N_{Bkg}}} = 25.6$.

Step (5) and step (6) are seen in Fig. (7.24) and Fig. (7.25) where the GBDT cuts, which maximize the found significance (GBDT1, GBDT2)=(0.1, 0.7), are applied to the distributions before the fits. As noted in the Signal Range are reduced unto SR*: $m(B^0) \in [4250, 5700]\text{MeV}/c^2$ for better fits. This smaller fit range is still compliant with the RK* group approach.

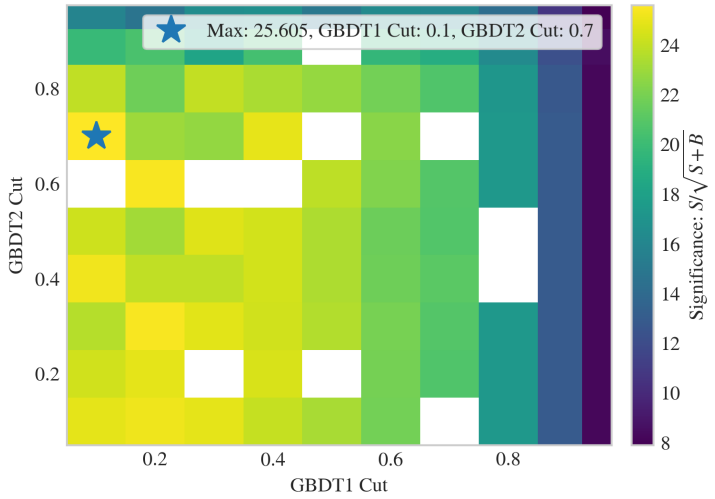


Figure 7.23: Blinded Significance scan on B_d^0 q^2_{high} $\mathrm{SR}^{\mathrm{data}}$ Period K data with fixed signal PDF shape parameters and free background PDF shape parameters (except the weight). The Grid used are $\mathcal{M} \times \mathcal{M}$ where $\mathcal{M}$ =[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95].

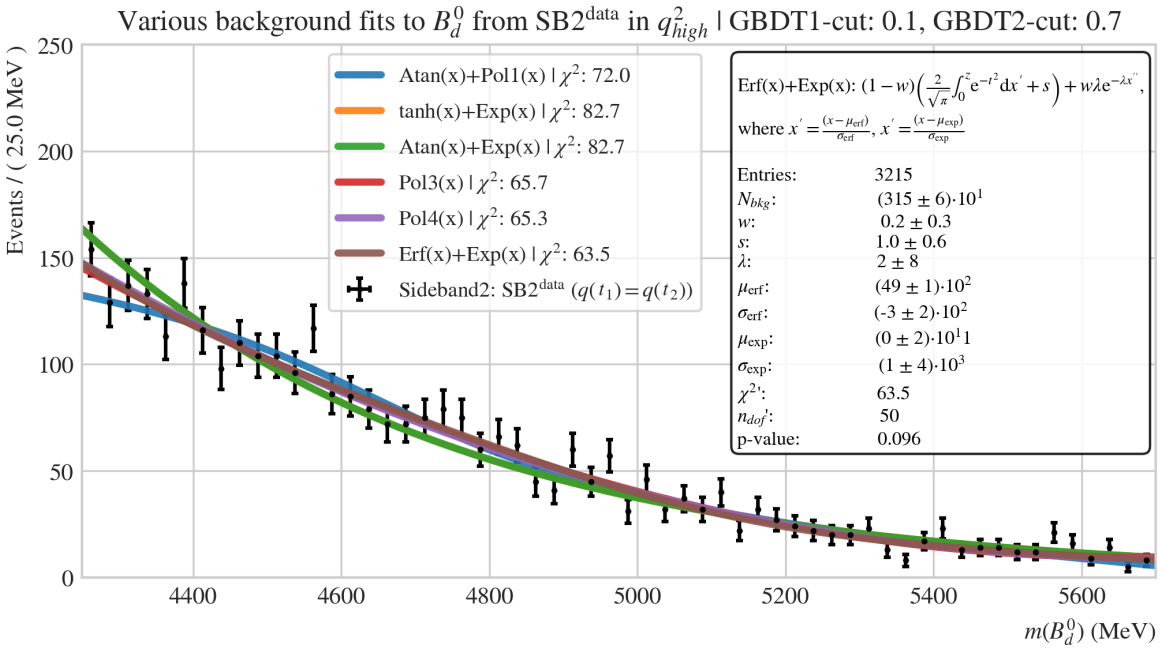


Figure 7.24: Background distribution fits on B^0 $\mathrm{SB2}^{\mathrm{data}}$, q^2_{high} with (GBDT1,GBDT2)=(0.1, 0.7). The best fit is again the $\mathrm{Erf}(x) + \mathrm{Exp}(x)$ function. Note that SB2 is reduced into the range: $m(B^0) \in [4250, 5700]^{\mathrm{MeV}/c^2}$.

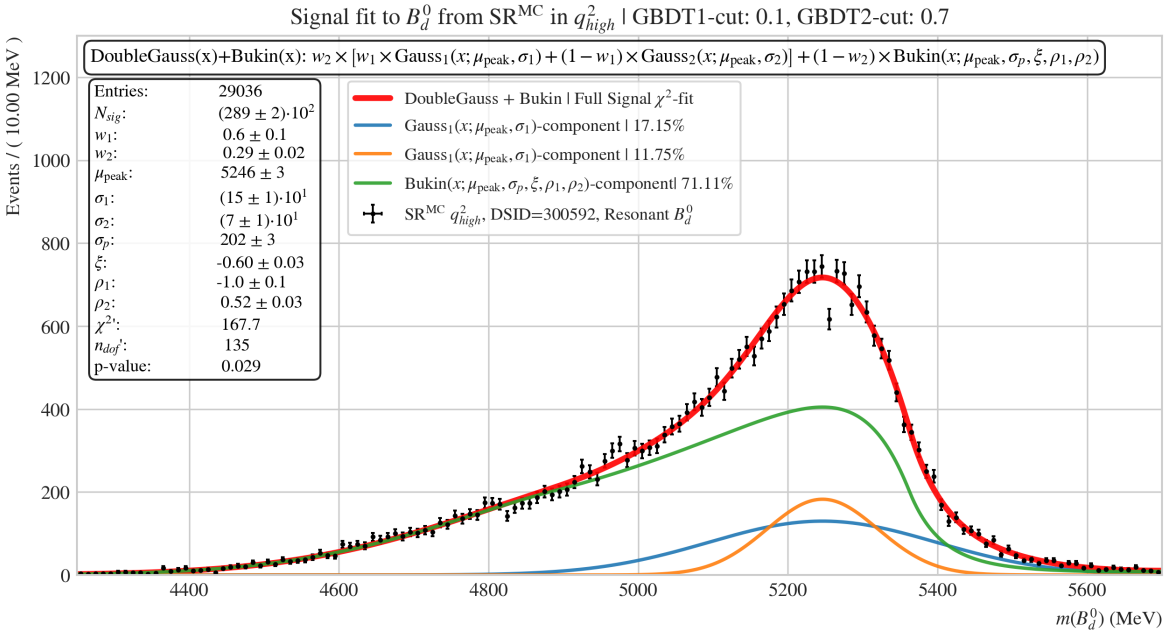


Figure 7.25: Signal fit: Double Gauss + Bukin PDF fit on B_d^0; SR$*^{MC}$ in the q^2_{high}-bin with (GBDT1,GBDT2)=(0.1, 0.7).

The final fit is a fit of the type: $N_{Sig} \times PDF_{Sig} + N_{Bkg} \times PDF_{Bkg}$ such that the number of signals can be extracted and used in the R_{K*0}-double ratio. Again - as in the significance scan, the Signal PDF (Eq. (7.7)) has its shape parameters fixed with floated mean and sigmas, whereas the background PDF has free shape-parameters except for the weight between the Erf(x) and Exp(x)[5]. The total fit is seen in Fig. (7.26). The fit converged with GoF-parameters: $\text{p} - \text{value} = 1 - \chi^2_{CDF}(\chi^2 = 66.2, N_{dof} = 48) = 0.04$, which is just short of the standard rejection threshold of 0.05. As seen in Fig. (7.26), something is wrong with the error of $\mu_{erf} = 35 \pm (61 \times 10^9)$. This indicates the challenges when fitting combined[6] background. The final signal yield is: $N_{Sig} = (58 \pm 9) \times 10$.

[5] The fixing of the background fit weight is a deviation from the RK* procedure, however, it made the χ^2-fit converge.

[6] background consist of many different decays.

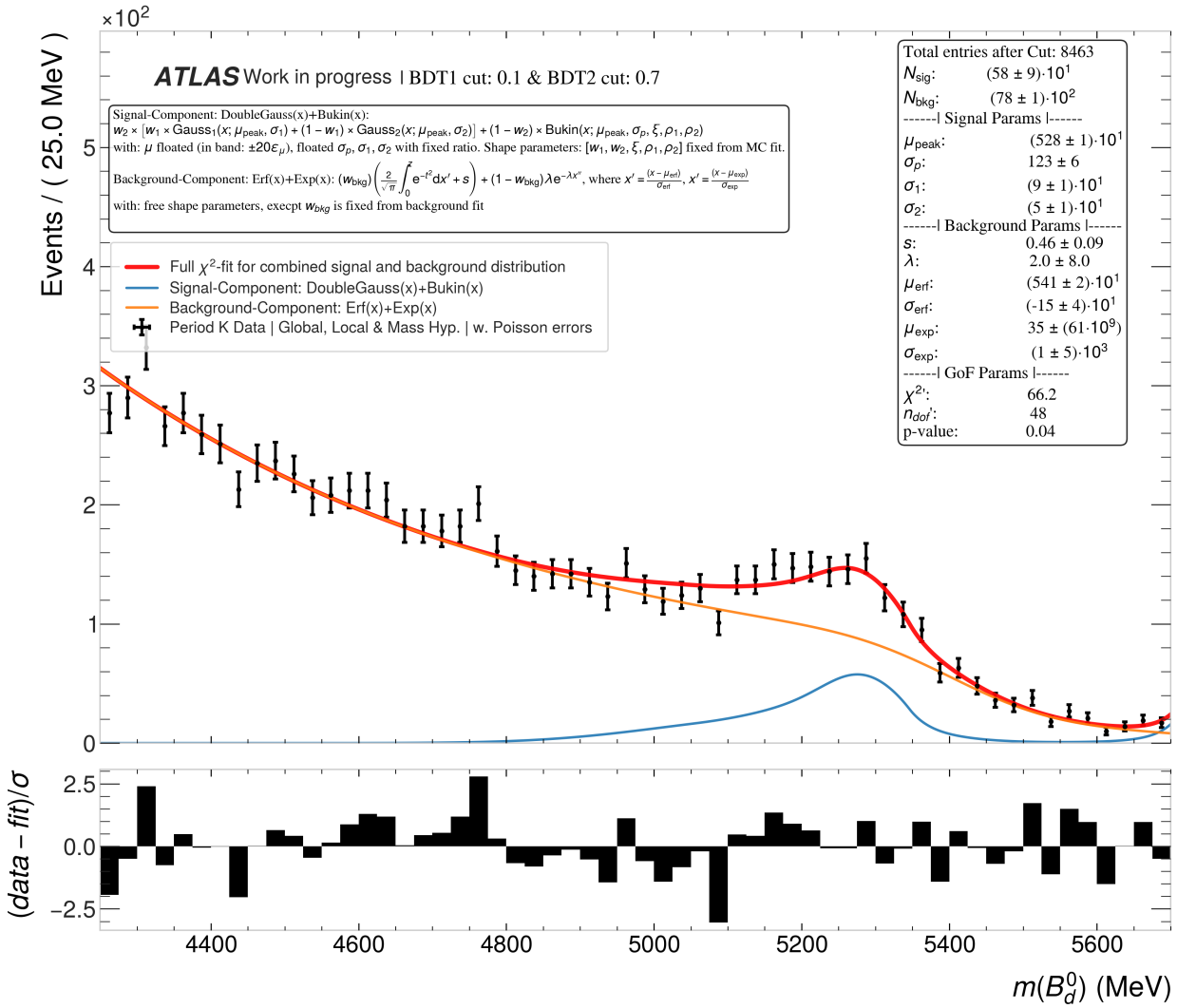


Figure 7.26: Full fit (Sig+Bkg) on B_d^0 period K, q^2_{high}. The threshold cuts for both GBDTs come from the maximum significance scan in Fig. (7.23). The fit is just short of the standard p-value threshold of: 0.05. This might be due to the signal shape distribution going up again for $m(B^0) > 5600\,\text{MeV}/c^2$ which is not ideal.

Looking back, each of the *Signal vs. Background* Testing Suite figures (Fig. (7.11) and Fig. (7.12)) they show the maximum significance scans GBDT cuts/thresholds: (GBDT1, GBDT2)=(0.1, 0.7). For GBDT1, the specific cut on the ROC and Precision/Recall curve at GBDT1-cut: 0.1 are (FPR, TPR, PPV)=(0.08, 0.92, 0.86). This means for a GBDT1-cut at 0.1, the GBDT correctly identifies 92% of the signals (TPR); however, it also incorrectly classifies 8% of the background events as signal (FPR). Lastly, 86% of the classified signals are actually true signals (PPV).
For the GBDT2-cut at 0.7, the (FPR, TPR, PPV)=(0.03, 0.85, 0.95) means the GBDT correctly identifies 85% of the signals (TPR); however, it also incorrectly classifies 3% of the background events as signal (FPR). Lastly, 95% of the classified signals are actually true signals (PPV).

This concludes the "2GNN to 2GBDT" approach and the above analysis shows that it is a viable approach however not satisfactory enough hence it can be further developed.

8
The Search for Better Performance

This chapter focuses on improving the performance of the "2GNN to 2GBDT" approach. There are many paths to take when a model, such as "2GNN to 2GBDT", needs to be improved. The chosen paths used in this thesis are seen in Fig. (8.1), which shows a roadmap of different approaches tried out to achieve better performance. The (✓) marks that the approach works, and the (X) denotes the approach does not work. Firstly the two approaches: "2GNN to 3GBDT" and the "2GNN to 2GBDT w. enriched MC background" which failed, will be shown in the next section. However, only an explanation of how it was done and where it failed will be provided. Please see appendix A.3 for each of the various testing suites from the "2GNN to 3GBDT" approach and see appendix A.4 the testing suites of the "2GNN to 2GBDT w. enriched MC background" approach. After exploring the two failed approaches, the "2GNN to 2GBDT" approach is expanded to extra features, thus ending the analysis with a final improved "2GNN to 2GBDT" approach.

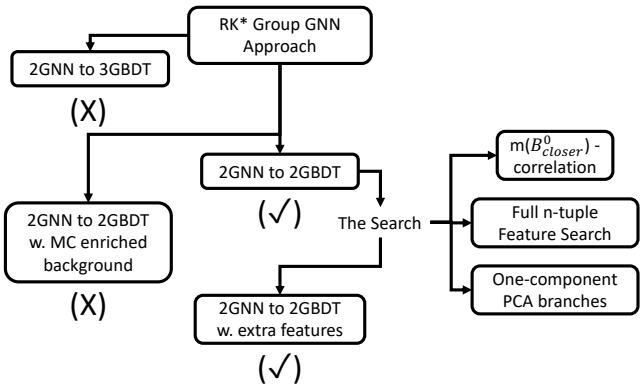


Figure 8.1: A roadmap to better performance for the GBDT model.

8.1 2GNN to 3GBDT

The "2GNN to 3GBDT" is a modification of the "2GNN to 2GBDT" approach, which only does two-class classification where the two first GBTs are trained as the original "2GNN to 2GBDT" approach

with the exception of the B^0 and $\overline{B}^0$ is not distinguished. Lastly, a third GBDT is trained in selecting either B^0 or $\overline{B}^0$, substituting the *Mass Hypothesis Candidate Selection*. The two GBDTs, which are trained on $\{\text{SR}^{\text{MC}} \cup \text{SB1}^{\text{data}}\}$ and $\{\text{SR}^{\text{MC}} \cup \text{SB2}^{\text{data}}\}$ respectively with normal binary log-loss. The last GBDT; *GBDT3* are training against 80000 truth-machted MC signal in SR^{MC} with 50% B^0 and 50% $\overline{B}^0$. The GBDT parameters are seen in Tab. A.2 in appendix A.3.

All three GBDTs are trained and signal efficiency benchmarks are seen in Fig. (8.2), the signal efficiency is higher for GBDT1 and GBDT2 relative to the "2GNN to 2GBDT" approach. This might be because the GBDTs can focus on learning the pure combined signal properties and do not have to predict multiple signal species. The other tests from the testing suite[1] can be seen in Fig. (A.3) in appendix A.3 for the interested reader.

[1] Except for the *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite since there is only one signal.

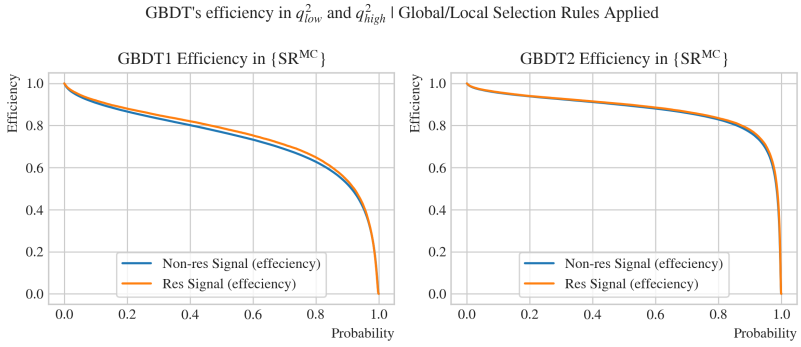


Figure 8.2: Resonant vs Non-Resonant signal efficiency plots for both the q^2_{low} and q^2_{high} bin for both GBDTs in "2GNN to 3GBDT". Both GBDT1 and GBDT yields good classifying performance.

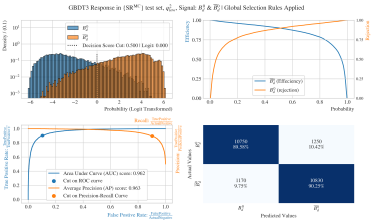


Figure 8.3: *Signal vs Background* Testing Suite on "2GNN to 3GBDT"-GBDT3 test-set which shows good performance on test set in separating B^0 and $\overline{B}^0$.

In the roadmap (Fig. (8.1)), there is an (X) under. The reason for this is when it comes to the prediction of Sig(B^0) vs. Sig($\overline{B}^0$) for GBDT3. In Fig. (8.3), the GBDT3 does a great job separating the two signals. This is quantified with $AUC = 0.962$ and $AP = 0.963$.

However, when applying GBDT3 on the non test-set SR^{MC}, it does not detect the differences in the Sig(B^0) vs. Sig($\overline{B}^0$) which is seen in Fig. (8.4). The AUC score is below 0.5 and this is worse than a random classifier. The AP score is a bit better but still not satisfactory. This implies that the GBDT3 could be over-fitted to the training set (Fig. (8.3)).

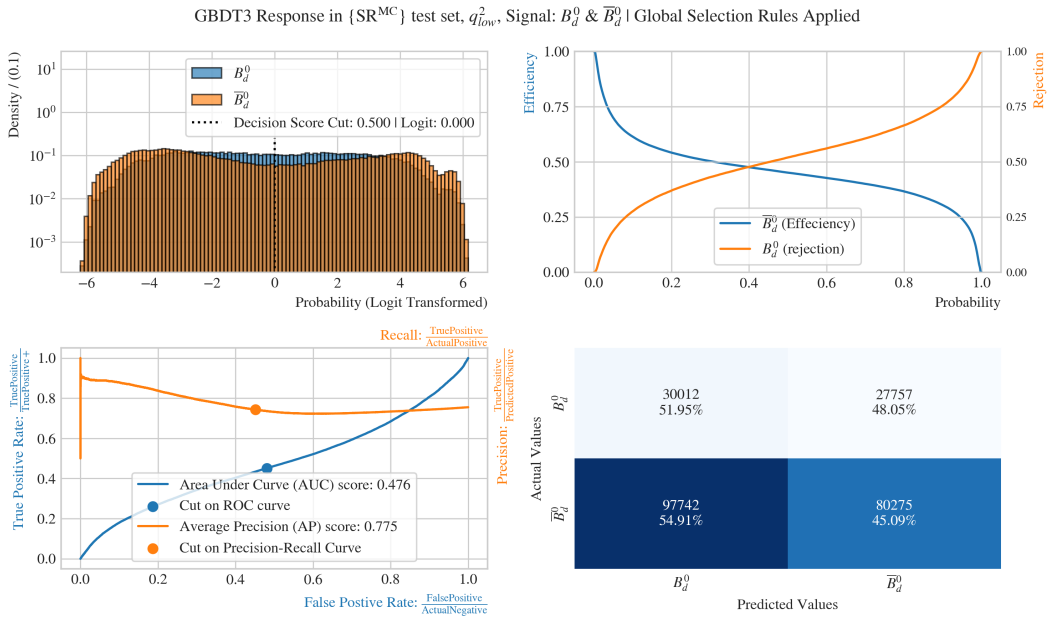


Figure 8.4: *Signal vs Background* Testing Suite on "2GNN to 3GBDT"-GBDT3 which shows that GBDT3 does does not classify B^0 vs. $\overline{B}^0$ well on the non-test set. Specifically seen in the AUC and AP quantities ($AUC < 0.5$).

This implies a dead end for the "2GNN to 2GBDT" in the analysis (X), and the following approach tested is the "2GNN to 2GBDT w. enriched MC background"-approach.

8.2 2GNN to 2GBDT w. enriched MC background

The "2GNN to 2GBDT w. enriched MC background" approach or from now on; the "*Enriched 2GNN to 2GBDT*" tries to do the same as "2GNN to 2GBDT"; however, this approach uses MC background data in the signal region to make the GBDTs better at separating signal since the background it trains on has events from the signal region and not only the sidebands. Whereas the "2GNN to 2GBDT" approach uses 700000 sideband events, the "Enriched 2GNN to 2GBDT" approach uses 400000 sideband events and 300000 MC backgrounds.
This means the GBDTs are trained on: $\{\text{SR}^{\text{MC}} \cup \text{SB1}^{\text{data}} \cup \text{Bkg}^{\text{MC}}\}$ and $\{\text{SR}^{\text{MC}} \cup \text{SB2}^{\text{data}} \cup \text{Bkg}^{\text{MC}}\}$ where the same Bkg^{MC} are used in the enrichment of the both sidebands. The parameters used for the two GBDTs are seen in Tab. A.3 in appendix A.4.
Another testing suite was developed for this approach: the *Sideband vs. MC Background* Testing Suite, which is closely related to the *Signal vs. Background* Testing Suite. This modified testing suite can be seen in Fig. (A.23) and Fig. (A.25) in appendix A.4 for the two GBDTs. The idea behind this test was that even though the GBDTs would train to be good performing classifiers, there was a problem if the GBDTs also learned the difference in sideband and

MC background - hence the test.

This approach ended with the performance on the signal efficiency, which is seen in Fig. (8.5). The lack of performance stopped this direction, and the focus was reverted to the simple "2GNN to 2GBDT".

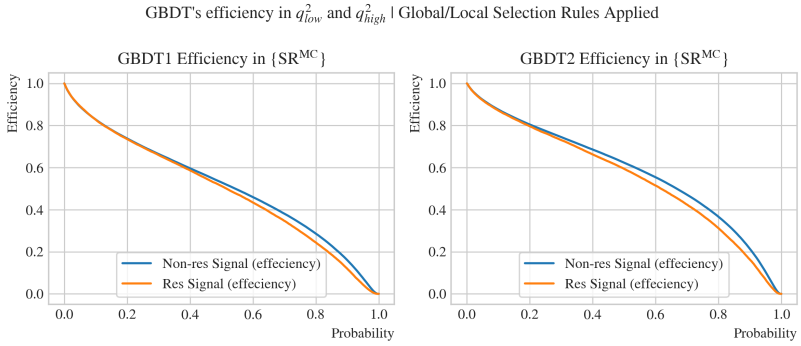


Figure 8.5: Resonant vs Non-Resonant signal efficiency plots for both the q^2_{low} and q^2_{high} bin for both GBDTs in "Enriched 2GNN to 2GBDT". It is clear the drop in signal efficiency for both GBDTs.

8.3 *The Search*

With both "2GNN to 3GBDT" and "Enriched 2GNN to 2GBDT" approached as dead ends, which were approaches where either the machine learning framework was altered or the training set was changed, the approach became the search for new features to add to the already good performing "2GNN to 2GBDT" model with the hypothesis: A GBDT model is only as good as the features it is given. The *search* for new features was a threefold process with the *$m(B^0_{closer})$-correlation* very tightly connected to the *Full n-tuple Feature Search*.

8.3.1 *$m(B^0_{closer})$-correlation*

The *$m(B^0_{closer})$-correlation* is about finding which features in the n-tuples correlate the most with the B-mass and then remove them from the list of potentially interesting features which could be added to the "2GNN to 2GBDT"-model for leakage avoidance. The standard approach would be using Pearson's Correlation Coefficient to find the features mostly correlated with $m(B^0_{closer})$. The correlation matrix is seen in Fig. (8.6) where the white lines represent non-numeric features.
There are 627 features either directly un-altered from the n-tuples or feature-engineered features, shown in Fig. (8.6). After removing $m(B^0_{closer})$ along with duplicates, non-numeric features and highly $m(B^0_{closer})$ Pearson correlated features; the feature search space ended up with 403 features[2].

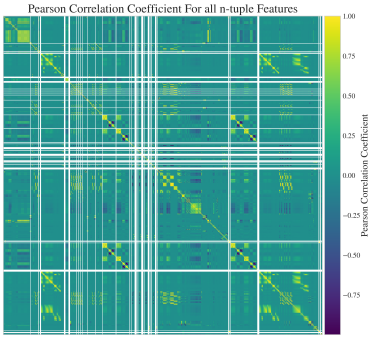


Figure 8.6: Pearson Correlation Coefficient for all 627 n-tuple features and extra added experimental features.

[2] Some extra features were removed simply by "visually seeing" the MC samples were not aligned with the actual data.

An RMSE Regressor with the Verstack Python library[73] with 100 defualt trials and 70%/30% train/test-set scaled with Scikit-Learn's Robust Scaler (Eq. (6.3)) are applied with the $m(B^0_{closer})$ as the target of the regression.
The idea is to apply the feature Importance ranking explained earlier[3] for the regressor. This is another approach in calculating correlations and hopefully, it would pick up the deeper correlations of the features. The regressor is applied on 70000 events from the SR^{MC}, $SB1^{data}$, and $SB2^{data}$ mass regions with the standard pre-selection cuts. The parameters and result from the training are seen in Tab. A.4 in appendix A.5. The 15 most important features for each of the three mass-cuts: SR^{MC}, $SB1^{data}$, and $SB2^{data}$ are seen in Fig. (A.35), Fig. (A.36) and Fig. (A.37) in appendix A.5 respectively. These importances are used in the step of cycling through all 403 features to find the best features to add to the "2GNN to 2GBDT" model.

[3] The only change is on the permutation Importance, which has the AUC score substituted with RMSE.

8.3.2 *Full n-tuple Feature Search*

The strategy for finding the best features is outlined in Fig. (8.7), which are the "2GNN to 2GBDT" model over and over again, slowly cutting away the too-good-to-be-true features which would train GBDT1 and GBDT2 into perfect classifiers. The AUC and AP scores are used to quantify a *perfect classifer* by values: $AUC = 1$ and $AP = 1$.
Starting with 403 features, the most leaking features are seen in Fig. (A.38) in appendix A.6. These features mainly consist of B masses and info-features along with obvious leaking[4] features.
The strategy of Fig. (8.7) is then carried out for $N = 6$ times where each time the features removed are seen in: Fig. (A.39), Fig. (A.40), Fig. (A.41), Fig. (A.42) and Fig. (A.43) in appendix A.6. The 6^{th} run of the "2GNN to 2GBDT" with the additional features are found in appendix A.6 where the figures of the *LightGBM* Testing Suite, *Signal vs. Background* Testing Suite and the *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing suite are applied. In addition the best 35 features for $SR^{MC} \cup SB1^{data}$ and $SR^{MC} \cup SB2^{data}$ are seen in Fig. (A.50) and Fig. (A.51) in appendix A.6.

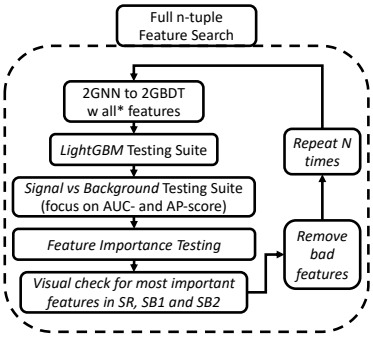


Figure 8.7: The full n-tuple feature search strategy is displayed as a looping diagram starting at "2GNN to 2GBDT w. all* features" where the all* means that some are discarded during the iterations.

[4] Leaking means that the classifier knows information about the B mass, this means "leaking" and "correlated with $m(B^0_{closer})$" are in this context used for the same.

The result of *the search* is the five features seen in Fig. (8.8), which are the five most important non-leaking features in the n-tuples/feature-engineered features:
tracks_dEdx_diff, diMeson_Kpi_piK_mass_avg, diMeson_Kpi_piK_mass_diff, angle_vtx_plane_mm_plane, angle_vtx_plane_ee_plane, and the explanation/calculations of these features are found in Tab. (6.5).

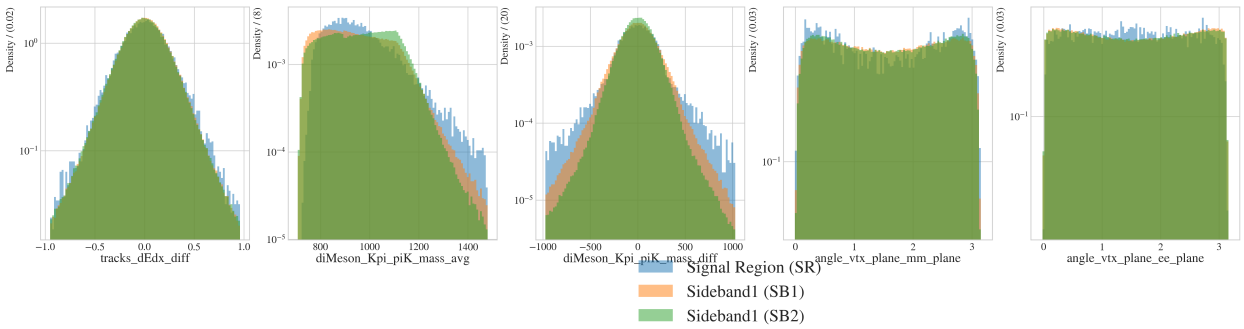


Figure 8.8: Best features with no leaking properties after iteration 6 of the full n-tuple feature search: [tracks_dEdx_diff, diMeson_Kpi_piK_mass_avg, diMeson_Kpi_piK_mass_diff, angle_vtx_plane_mm_plane, angle_vtx_plane_ee_plane]. Note that the explanation/calculations of these features are found in earlier shown Tab. (6.5)

8.3.3 *One-component PCA branches*

Another way to introduce new features is through Principal Component Analysis (PCA). This dimensionality reduction technique compresses multi-dimension data into smaller dimensions by projecting the data onto the eigenvectors with the most contributing eigenvalues. The steps are the following for one-component PCA:

- Scale data: $\mathbf{X}$ since PCA are sensitive to data-scale.
- Calculate covariance matrix: $Cov(\mathbf{X}) = \mathbb{E}[(\mathbf{X} - \mathbb{E}[\mathbf{X}])((\mathbf{X} - \mathbb{E}[\mathbf{X}])^T]$
- Solve for eigenvalues: $\mid Cov(\mathbf{X}) - \lambda\mathbb{I} \mid = 0$
- For eigenvalues: $\lambda_1, \lambda_2, \ldots$ find maximum eigenvalue: $\lambda' = max\{\lambda_1, \lambda_2, \ldots\}\ \lambda'$
- Calcualte corresponding eigenvector $\mathbf{v}$: $(Cov(\mathbf{X}) - \lambda'\mathbb{I})\mathbf{v} = 0$
- Project $\mathbf{X}$ onto $\mathbf{v}$: $\mathbf{X}' = \mathbf{X}\mathbf{v}$

The idea is that the GNN architecture is created based on the decay topology hence each of the 11 branches would be perfect for One-component PCA analysis, which is seen in Fig. (8.9) (color-coded). The new features are made by the Scikit-Learns PCA method and are fitted to the training set and applied to the rest of the data[5]. The fit and transform of the PCA are done after the Robust Scaler(Eq. (6.3)) has scaled the data.

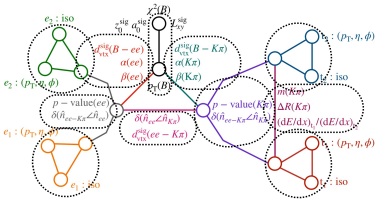


Figure 8.9: A translation of the GNN architecture to 11 one-component PCA. Circles represent which features are bundled together to form a one-component PCA. There are 11 circles.

[5] The same way the scaler is applied.

8.4 *The Final Model*

In total, the "2GNN to 2GBDT" is extended with 16 extra features, namely eleven "GNN branch" one-component PCAs and five non-leaking features seen in Fig. (8.8). The name of this extended feature approach is dubbed: "2GNN to 2GBDT w extra features".

"2GNN to 2GBDT w extra features" uses the exact same approach as the original "2GNN to 2GBDT" hence not much explanation is needed. The "2GNN to 2GBDT w extra features" approach passes all testing suites, which are seen in appendix A.7 with the classifying performance for GBDT1 of $AUC = 0.978$, $AP = 0.951$ and for GBDT2: $AUC = 0.982$ and $AP = 0.964$ on the non test-set. This means both classifiers perform as they should (this is also seen in figures Fig. (8.10) and Fig. (8.11).

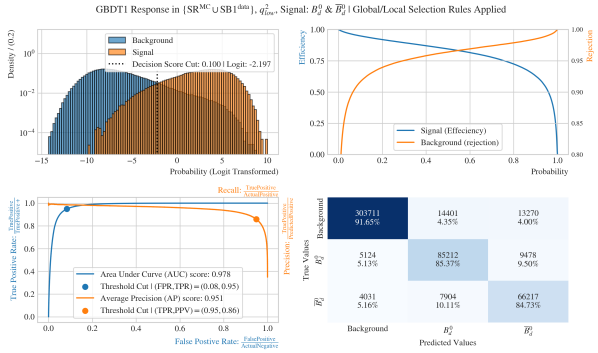


Figure 8.10: *Signal vs Background* Testing Suite with "2GNN to 2GBDT w extra features"-GBDT1 on non-train $SR^{MC} \cup SB1^{data}$. The figure in big format is seen in Fig. (A.60) in appendix A.7. The main takeaway is the classifying performance for GBDT1 of $AUC = 0.978$, $AP = 0.951$.

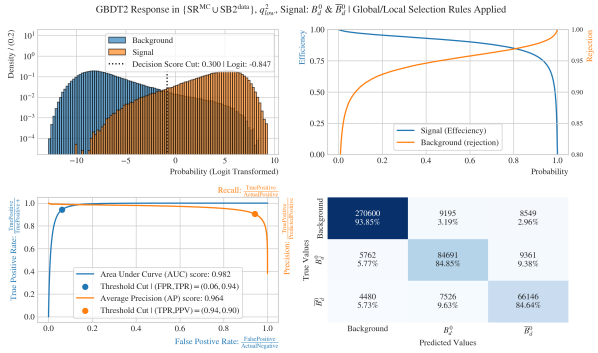


Figure 8.11: *Signal vs Background* Testing Suite with "2GNN to 2GBDT w extra features"-GBDT2 on non-train $SR^{MC} \cup SB2^{data}$. The figure in big format is seen in Fig. (A.61) in appendix A.7. The main takeaway is the classifying performance for GBDT2 of $AUC = 0.982$ and $AP = 0.964$.

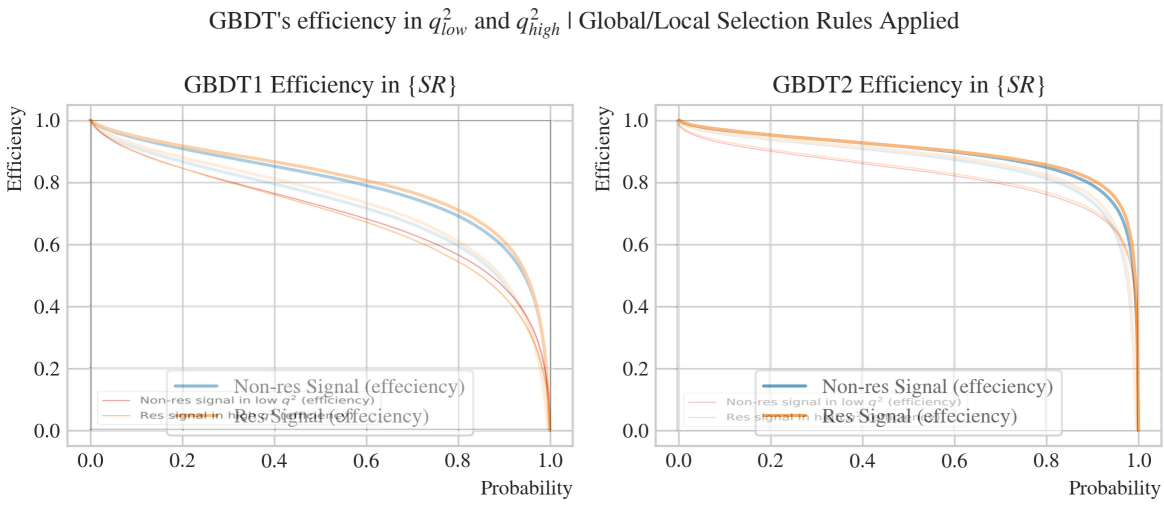


Figure 8.12: Efficiency comparison between the 2GNN (thin/blurred lines) vs the 2GBDT w extra features (upper lesser-blurred thick lines) and the original 2GBDT (middle blurred thick lines). The figure shows that the "2GNN vs the 2GBDT w extra features" approach has better signal performance for all probabilities and beats the original 2GNN vs the 2GBDT" approach.

The performance of "2GNN to 2GBDT w extra features" is seen in Fig. (8.12) where it both surpasses the RK*'s GNN approach and also beats the "2GNN to 2GBDT" approach in signal efficiency. All three are plotted on Fig. (8.12) (on top of each other) where the two (orange and blue) less blurred thick lines are the performance of "2GNN to 2GBDT w extra features" whereas the other thick lines

are the "2GNN to 2GBDT" with the thin lines the GNN approach. The "2GNN to 2GBDT w extra features" approach is also performing better in $[0.92 to 1.00]$ (which the "2GNN to 2GBDT" failed to do) such that now it has higher signal performance for all probabilities.

Steps (1)-(3) of the fitting routine are the same for "2GNN to 2GBDT w extra features" hence refer to section 7.0.2 for a detailed view of the first step of the fitting process. The Significance scan for the "2GNN to 2GBDT w extra features" gives the cuts: Significance$(\text{GBDT1} = 0.1, \text{GBDT2} = 0.3) = 24.2$ which are seen in Fig. (A.67) in appendix A.7.

Step (5) and *step (6)* is seen in Fig. (8.13) and Fig. (8.14) where with (GBDT1, GBDT2)=$(0.1, 0.7)$ in reduced signal range: $[4250, 5700]$. It is seen in Fig. (8.13) that again the best background PDF is $\text{Erf}(x) + \text{Exp}(x)$ with a p-value of 0.13. The shape of the signal after GBDT cuts: $(0.1, 0.3)$ has some distortion by a factor such that the double Gaussian + Bukin PDF (Eq. (7.7)) does not fit as well (p-value of 0.002) as before introducing cuts.

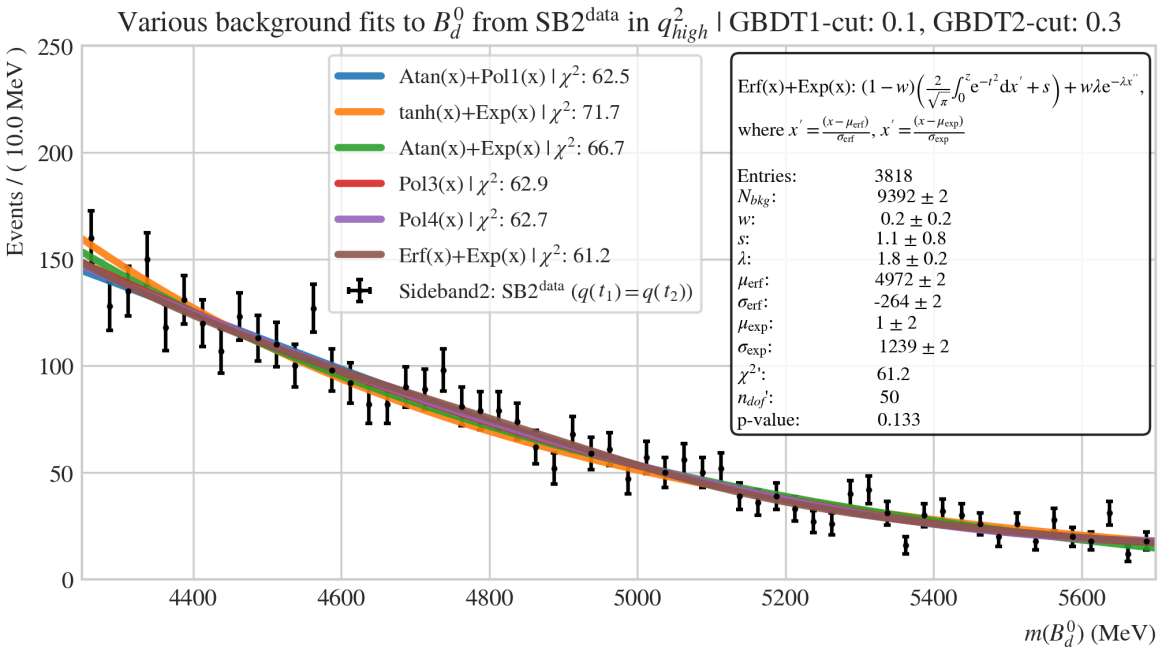


Figure 8.13: Background distribution fits on B_d^0 SB2$^{\text{data}}$, q^2_{high} with (GBDT1,GBDT2)=$(0.1, 0.3)$. The best fit is again the $\text{Erf}(x) + \text{Exp}(x)$ function. Note that SB2 is reduced into the range: $m(B^0) \in [4250, 5700]\text{MeV}/c^2$.

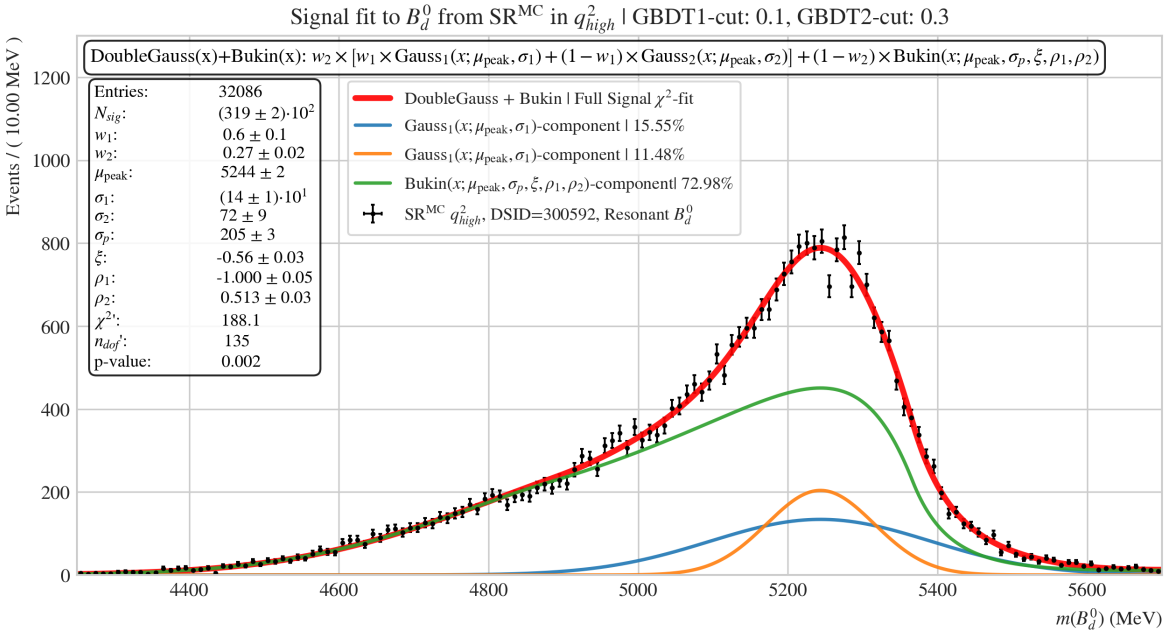


Figure 8.14: Signal fit: Double Gauss + Bukin PDF on B_d^0 SR^{*MC}, q^2_{high} with (GBDT1,GBDT2)=(0.1, 0.3).

The combined signal and background fit for GBDT cuts $(0.1, 0.3)$ is seen in Fig. (8.15). Again the double Gaussian + Bukin shape parameters are fixed to the MC signal fit in Fig. (8.14) where the common mean (μ_{peak}) and sigmas ($\sigma_1, \sigma_2, \sigma_p$) are floated with the ratio between the sigmas are fixed. The background PDF has all its shape parameters floated except the ratio between $\text{Erf}(x)$ and $\text{Exp}(x)$. The fit converges with GoF-parameters: $\text{p} - \text{value} = 1 - \chi^2_{CDF}(\chi^2 = 32.0, N_{dof} = 48) = 0.96$, which means the probability distribution fitted to the data represents the data to a very high degree. The final signal yield is $N_{Sig} = (1853 \pm 2)$.

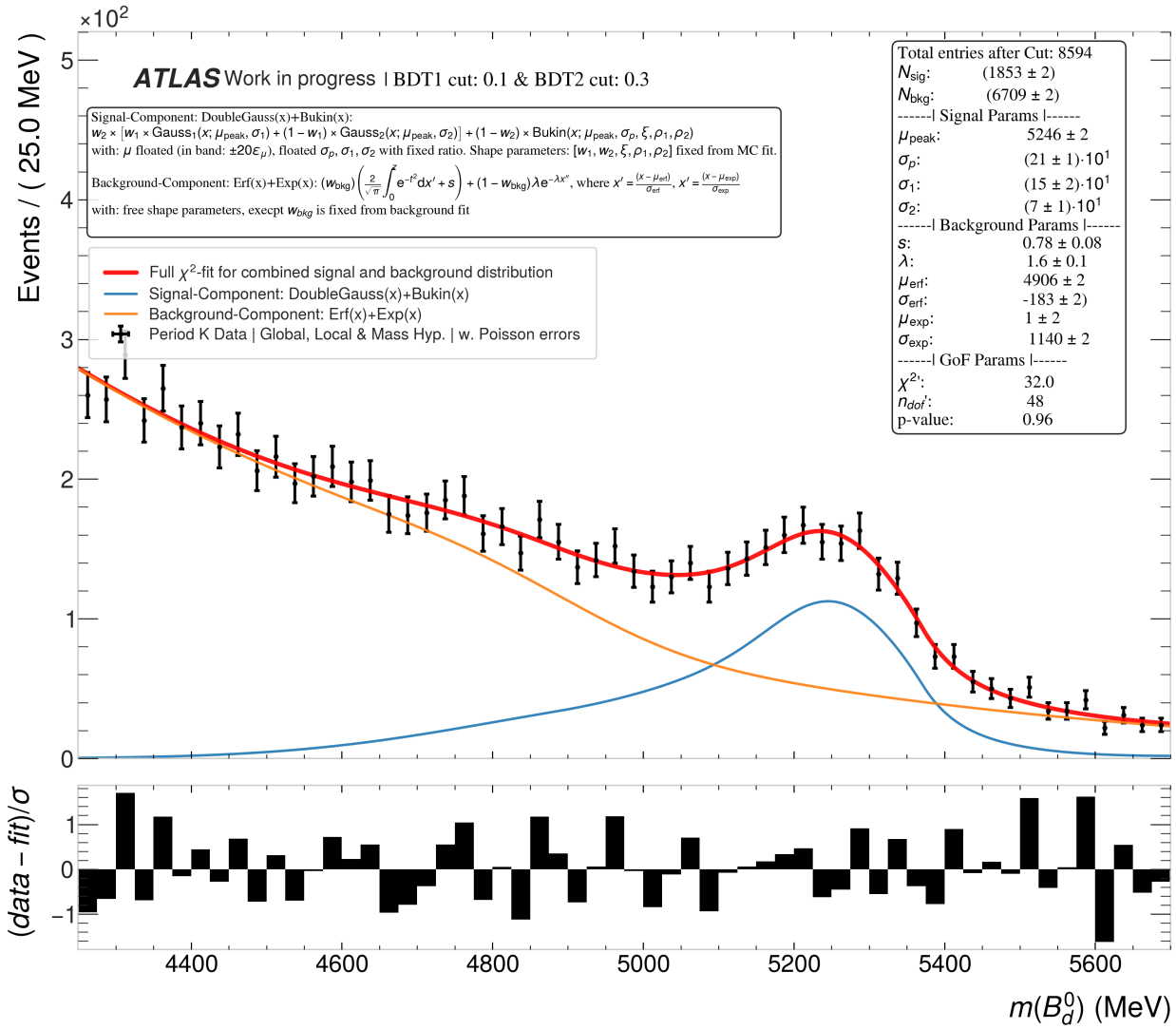


Figure 8.15: Full fit (Sig+Bkg) on B_d^0 period K, q^2_{high}. The threshold cuts for both GBDTs; (GBDT1,GBDT2)=(0.1, 0.3) comes from the maximum significance scan in Fig. (7.23). The combined signal and background PDF fit the data with GoF-parameter: p-value= 0.96 which means the fit agrees with the data. From this fit the signal yield can be extracted: $N_{Sig(B^0)} = 1852 \pm 2$.

The "2GNN to 2GBDT w extra features" has less strict GBDT cuts than the "2GNN to 2GBDT". Looking to figures: Fig. (8.10) and Fig. (8.11) the GBDT cuts: (GBDT1, GBDT2)=(0.1, 0.3) gives a classifier with (FPR, TRP, TPR) = $(0.08, 0.95, 0.86)$ which means GBDT1 correctly identifies 95% of the signals (TPR); however, it also incorrectly classifies 8% of the background events as signal (FPR) and 86% of the classified signals are true signals (PPV).
For GBDT2: with (FPR, TRP, TPR) = $(0.06, 0.94, 0.90)$ it identifies 94% of the signals (TPR), and it incorrectly classifies 6% of the background events as signal (FPR) and 90% of the classified signals are true signals (PPV).

Part III

Wrap Up

9 Discussion

The following section discusses the results and the general approach used to reach those results. Some subjects for discussion are the fitting routine, the Kaon Pion mass tails, and uncertainties.

9.1 The Results

The main result of the analysis is the results of the "2GNN to 2GBDT w extra features" approach, which has the highest signal efficiency. The signal yield is $N_{Sig(B^0)} = 1853 \pm 2$, which is from the combined signal and background fit seen in Fig. (8.15).
The estimated error on the yield: ± 2, which is a relative error of $\varepsilon^{\text{rel}}_{N_{Sig(B^0)}} = 0.11\%$ indicated that the uncertainty is underestimated.
The χ^2-fit matches the data with a p-value of 0.96, which might explain the underestimated error. It is not p-value$= 1$. However, it is still very high, as there might be too many free parameters.
As seen in both Fig. (8.13) and in Fig. (8.14), the sculpting of background is not present, and it drops off nicely off as the $m(B^0)$ increases. However, the signal gets distorted when the cuts are applied and are probably the bigger contributor to the error.

As this is a Poisson process, $N_{Sig(B^0)}$ must have at least errors of $\sqrt{N_{Sig(B^0)} = 1853}$ thus: $N_{Sig(B^0)} = 1853 \pm (\sqrt{1853} \pm 2 \pm \sigma_{\text{sys}})$ and omitting the systematic error notations; the result is $N_{Sig(B^0)} = 1853 \pm (\geq 45)$ (also reported in Eq. (10.1)) where the "$\geq$" refers to the fact that this is the smallest uncertainty for the signal yields and the uncertainty is without a doubt, bigger.

The significance for the signal and background fit of the "2GNN to 2GBDT w extra features" Fig. (8.15) approach is $\text{Significance}_{Sig(B^0)} = 20 \pm (\geq 0.4)$ calculated using the Poisson errors and the fit errors. The uncertainty is by standard error propagation with poison errors and fit errors. The result is also seen in Eq. (10.5).

The efficiency of the GBDTs is performing better than hoped. How-

ever, background rejection is also essential for fitting. Since a better performance in background rejection, the better the combined signal and background fit will be since the background will behave nicely. The significance scans indicate this since they only have maximums in the lower end instead of strict (high GBDTs) cuts. As seen in Fig. (9.1), the signal efficiency, as discussed earlier, performs excellently for all selection probabilities.

However, the background rejection is underperforming for all probabilities for both GBDTs. The gained signal efficiency boost from the "2GNN to 2GBDT w extra features"-approach compared with the GNN benchmark from [39] are approximate: $\sim +15\%$ for GBDT1 vs. GNN1 and $\sim +10\%$ for GBDT2 vs. GNN2.

However, as seen in Fig. (9.1), the background rejection is worse for the GBDTs with $\sim -5\%$ for GBDT1 vs. GNN1 and $\sim -5\%$ for GBDT2 vs. GNN2. This lack of background rejection indicates that the error on the yield is underestimated due to more background events leaking into the combined fit. Lack of background rejection will also pose a problem in the low q^2 bin since the statistics here are low. The RK* group estimates that 28 true signal events and 581 true backgrounds in the resonant channel (q^2_{low}) can be extracted for the 2018 Run2-data[39]. With these low statistics in signals, it is essential to reject as much as possible of the background.

On top of the signal yield, an estimate of the GBDTs total efficiency can be extracted from the confusion matrices for MC samples (Fig. (8.10) and Fig. (8.11)) where the errors on the signal efficiency; ε_{B^0} is binomial: $\Delta\varepsilon_{B^0} = \sqrt{\frac{\varepsilon_{B^0}(1-\varepsilon_{B^0})}{N}}$ where N is the total number of entries in the matrix. The efficiencies are calculated on $\{\mathrm{SR}^{\mathrm{MC}} \cup \mathrm{SB1}^{\mathrm{PeriodK}}_{\mathrm{Pre-sel}}\}$ and $\{\mathrm{SR}^{\mathrm{MC}} \cup \mathrm{SB2}^{\mathrm{PeriodK}}_{\mathrm{Pre-sel}}\}$ since the MC data is labeled[1] hence the possibility of estimating the efficiency.

The resulting efficiencies are $\varepsilon^{\{\mathrm{SR}^{\mathrm{MC}} \cup \mathrm{SB1}^{\mathrm{PeriodK}}_{\mathrm{Pre-sel}}\}}_{\mathrm{GBDT1}(B^0|\mathrm{cut:0.1})} = 0.854 \pm 0.001$ and $\varepsilon^{\{\mathrm{SR}^{\mathrm{MC}} \cup \mathrm{SB2}^{\mathrm{PeriodK}}_{\mathrm{Pre-sel}}\}}_{\mathrm{GBDT2}(B^0|\mathrm{cut:0.3})} = 0.848 \pm 0.001$ for GBDT1 and GBDT2 respectively. The total efficiency of the combined GBDT is $\varepsilon_{GBDT|tot} = 0.72 \pm 0.03$, calculated using standard error propagation. These results are also reported in Eq. (10.2), Eq. (10.3), and Eq. (10.4). This efficiency is only an indicator of the composite GBDTs' performance, and the actual efficiency studies are yet to be done in the RK* group. The main takeaway with the calculation of the combined efficiency of the two GBDTs is that the GBDT approach is a viable ML method for separating B^0, $\overline{B}^0$ and background, which is also seen in all the separation tests applied throughout the analysis.

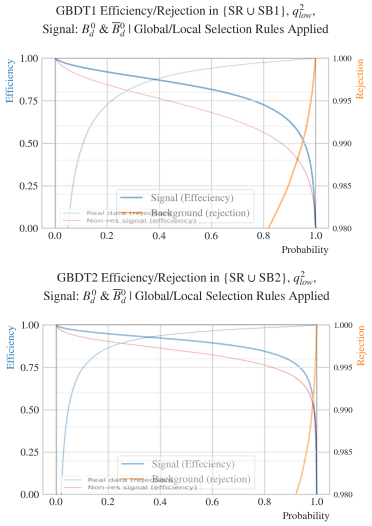


Figure 9.1: Signal efficiency and background rejection with the *Signal vs Background* Testing Suite with "2GNN to 2GBDT w extra features" GBDT1 on $\{\mathrm{SR}^{\mathrm{MC}} \cup \mathrm{SB1}^{\mathrm{data}}\}$ (top subfigure) and GBDT2 on $\{\mathrm{SR}^{\mathrm{MC}} \cup \mathrm{SB2}^{\mathrm{data}}\}$ (Lower subfigure). The thick lines are the GBDTs, and the thin lines are GNNS. Note that the thick blue GBDT line will be compared with the thin orange line for signal efficiency and vice versa for the background rejection.

[1] Note that $\mathrm{SB1}^{\mathrm{PeriodK}}_{\mathrm{Pre-sel}}$ just means that Period K data is used in the SB1-region after pre-selections.

9.1.1 *The Fitting Routine*

Even though the fitting is an essential part of extracting the signal yield, it has not been a priority of this thesis since the focus was on developing a machine learning framework for separating B^0, $\overline{B}^0$, and background. Other RK* group members have the fitting routine as their specialty, and hence a lot of the fitting is done using their recommended approach[2].

[2] such that the fits can be compared.

The use of the $\text{Erf}(x) + \text{Exp}(x)$ function for the background fitting is purely motivated by the lowest χ^2 value. The RK* fitting group also uses this background PDF; however, all the fitting functions were viable in capturing the background. As the RK* suggested, the two polynomials are not the best option for fitting since they can almost fit everything[3], hence it is better to use PDFs, which have some restrictions in their shape.

[3] They are like a Swiss-army knife for fitting.

The reason for restricting the signal range to $m(B^0) \in [4250, 5700]^{\text{MeV}/c^2}$ after GBDTs are applied in the fitting routine at step (5) is due to the distortion of the background shape. In the range $m(B^0) \in [4000, 4700]^{\text{MeV}/c^2}$ in Fig. (7.21) the background is increasing whereas in $m(B^0) \in [4700, 5700]^{\text{MeV}/c^2}$ the background shape is decreasing. It seems as if the GBDT background rejection is stronger for higher $m(B^0)$-values. This means after the cuts are applied, the background shape increases in $m(B^0) \in [4000, 4250]^{\text{MeV}/c^2}$ and then decreases in the rest: $m(B^0) \in\in [4250, 5700]^{\text{MeV}/c^2}$. Since $m(B^0) \in [4000, 4250]^{\text{MeV}/c^2}$ is far enough from the peaking of $m(B^0)$, it can be discarded in the fitting routine to achieve higher stability in the fits[4].

[4] Note that the RK* group restricts the fit range even further to $m(B^0) \in [4500, 5700]^{\text{MeV}/c^2}$.

The fitting routine used in this thesis is a χ^2 fit through the Python library: Iminuit[15]. The first fit routine applied to the selected data was binned maximum likelihood (MLLH) with bootstrapping for error estimation[5] however, the approach was abandoned since the χ^2-fit can directly estimate errors, thus making it faster. Typically the unbinned MLLH is a better choice since it fits directly using the likelihood function, whereas the χ^2-fit uses the residuals.

[5] An example of the MLLH fit is seen in Fig. (A.68) with the bootstrapped errors in Fig. (A.69) in appendix A.8.

Bin-width is also one thing to consider when using χ^2-fits; for the background and total fit[6] the bin-width is $25^{\text{MeV}/c^2}$ such that each bin had enough statistics for a reasonable fit. Since the MC samples are quite large, the bin-width was $10^{\text{MeV}/c^2}$ as were the background fit before GBDT cuts due to statistics. The discussion of bin width can be tricky. However, the key is consistency such that the p-value is not fitted to the bin width, and hence the results can be reliable (this is the idea with the blinded significance scan).

[6] after GBDT cuts.

9.1.2 *The ML Pipeline*

The analysis methodology used in this thesis is mainly adopted from the RK*s methodology for making the results comparable. Some of the elements in the approach that were changed from the GNN to GBDT were the substitution of the scaler in the ML pipeline. The GNN approach uses the *standard scaler*, and the GBDTs have used the *robust scaler*. The main differences are that the standard scaler assumes the existence of the first two moments. In contrast, the robust scaler does not rely on any distribution assumption other than the set is finite[7]. The robust scaler was chosen since no prior assumptions are needed on the feature distributions[8].

[7] Thus, the existence of quantiles.

[8] The standard scaler will not work on a Cauchy distribution where the moments are undefined.

The number of training and test samples and the percentages used for training, validation, and test-set is also up for discussion. The percentages used are not out of the ordinary and are compliant with standard ML train/test splitting[23]. The event numbers used for training in the analysis: 700000 background events and 20000 signal events are the same used in the RK* GNN analysis. The RK* local candidate selection which looks like: $\max \left\{ \begin{matrix} P_1(B^0) + P_1(\overline{B}^0) \, , \\ P_2(B^0) + P_2(\overline{B}^0) \end{matrix} \right\}$ and are found in Tab. (6.4) could potentially also have another form and as the same for the Mass hypothesis selection.[9]

[9] As shown, the "2GNN to 3GBDT" approach was an alternative Mass hypothesis selection strategy with the use of a GBDT for selecting the mass however with no success.

The Verstack[73] library also supports multiple metrics for optimization and multi-logloss[10] might not be the best even though this metric is used during the Optuna optimization of the LightGBM classifier in the training of all classifiers in the analysis. One has to be careful when selecting these metrics since the dataset is highly unbalanced. The advantage of multi-logloss[11] is that it penalizes the classifier if it confidently predicts wrongly - especially on the minority classes. However, other metrics could also be used: *f1*, *AP*, *balanced accuracy* etc., which all are suitable for imbalanced datasets. The number of trials in the optimization was set to 100 for all GBDT models. This could also be changed; however, as seen in all *LightGBM* Testing Suites, the loss functions converge to a minimum for all GBDT models.

[10] which has been used on all models.

[11] or cross-entropy.

9.1.3 *The Choice of the ML Testing Suite*

The metrics used in the four testing suites (+ the feature importance) are not the only ones that could be used, and the way of testing the GBDTs could also be changed. As the RK* group also uses some of these tests, they are also used in this analysis. These tests are the *Efficiency/Rejection* in the *Signal vs. background* testing

suite, the *2D Response Curve* in the *Sig(B^0) vs. Sig($\overline{B}^0$)* testing suite, and all the *mass shape* testing suites. The main idea is to open the "black box" of the ML to see how well they perform on different datasets and if they have learned some less-ideal tendencies, like the sculpting in the background and signal.

The Permutation Importance and Shapley values are calculated on 25% of the train and test set. The reason for not using the whole dataset is that the important trends are already encaptured in 25% of the data; hence using more would be redundant. The permutation importance is repeated ten times which should be enough to even out statistical fluctuations; however, one could always argue that it is insufficient, and the bottom line is that it is a precision/computing-time trade-off. The Permutation Importance also uses AUC as a score. As mentioned in the discussion of the ML pipeline, other metrics could be used here, like the ones mentioned. It all depends on the goal of the permutation test, and the AUC score serves that purpose. 6.8

9.1.4 *KaonPion and PionKaon Mass Tails*

The RK* group has weekly meetings, and in December, the preliminary results of this thesis were shown. At this meeting, it was pointed out that the shape of the Kaon-Pion/Pion-kaon mass; $m(K\pi)/m(\pi K)$ had a discontinuity, and it was decided that this discontinuity was not ideal since it does not represent the real physical distributions of $m(K\pi)$ and $m(\pi K)$. The cuts in the distribution are seen in Fig. (9.2) at $m(K\pi) = 1110\,\mathrm{MeV/c^2}$ or $m(\pi K) = 1110\,\mathrm{MeV/c^2}$. The reason for the discontinuity is the **OR** in equation Eq. (9.1), which are the RK* group pre-selection cuts from Tab. (6.3) for the $m(K\pi)$ and $m(\pi K)$.

$$\begin{gathered}\left(m(B^0) \in [3, 6.5]^{\mathrm{GeV/c^2}} \quad \text{AND} \quad m(K\pi) \in [690, 1110]^{\mathrm{MeV/c^2}}\right) \\ \text{OR} \\ \left(m(\overline{B^0}) \in [3, 6.5]^{\mathrm{GeV/c^2}} \quad \text{AND} \quad m(\pi K) \in [690, 1110]^{\mathrm{MeV/c^2}}\right)\end{gathered} \tag{9.1}$$

The reason for the OR is that the ATLAS detector can not distinguish between Kaons and pions; hence there are multiple cases, and those cases are solved with the pre-selection cut since an event can be either in one of the brackets or in both. However, at the December meeting, it was decided to search for an approach that takes the **OR** into account to smooth out the input $m(K\pi)$, $m(\pi K)$-distributions.

The reason for using GNNs for the analysis in the first place was

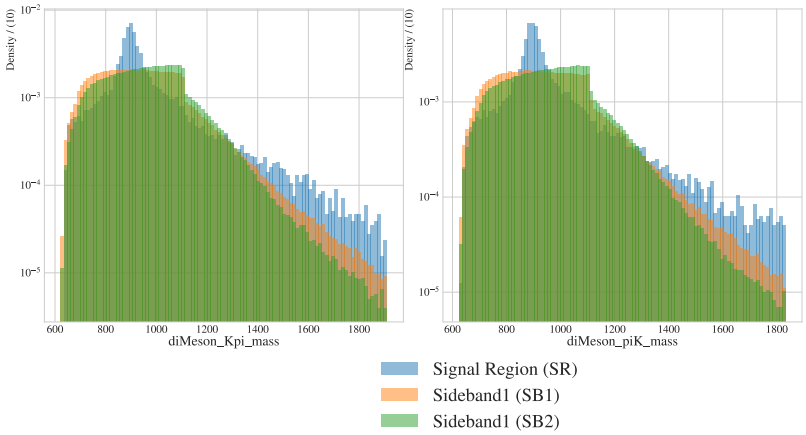


Figure 9.2: SR^{MC}, $SB1^{data}$, and $SB2^{data}$ distribution for $m(K\pi)$ and $m(\pi K)$ after pre-selection cuts are applied along with the mass-region cuts.

their ability to create an architecture where features could be linked into branches. This way, the branches can represent physically motivated correlations between features that mimic the B^0 decay topology. This is the extent of exploitation of the original RK* GNN architecture; however, the solution to the $m(K\pi)/m(\pi K)$ discontinuity was solved with a unique solution utilizing the GNN architecture. The original GNN architecture is seen in Fig. (5.2) in chapter 5; however, this architecture was changed such that it could handle the following three cases:

CASE1: $m(K\pi) \in [690, 1110]^{\text{MeV}/c^2}$ AND $m(\pi K) \in [690, 1110]^{\text{MeV}/c^2}$,
CASE2: $m(K\pi) \in [690, 1110]^{\text{MeV}/c^2}$ AND $m(\pi K) \notin [690, 1110]^{\text{MeV}/c^2}$,
and
CASE2: $m(K\pi) \notin [690, 1110]^{\text{MeV}/c^2}$ AND $m(\pi K) \in [690, 1110]^{\text{MeV}/c^2}$.

The new GNN architecture is seen in Fig. (9.4), where each case is handled by switching a branch on/off depending on the three cases above. If an event has the conditions of *CASE1*, the GNN graph looks like it does in Fig. (9.4).

If an event has conditions of *CASE2*, the branch/node: "$-1/2m(K\pi)m(\pi K)$" and all its edges are removed. For *CASE3*, it is the branch/node: "$+1/2m(K\pi)m(\pi K)$" and its edges that is removed, and this resulted in the smoothing of the training distributions as seen in Fig. (9.5). The implementation of the updated GNN architecture also improves the difference in signal efficiency for q^2_{low} and q^2_{high} such that it gets lower, see Fig. (9.3) where differences are almost vanishing for all probability thresholds.

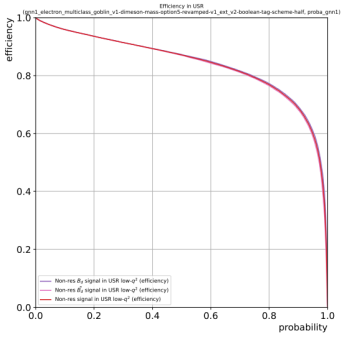


Figure 9.3: Signal efficiency for GNN1 in the SR region for the updated GNN architecture which handles the three Kaon-pion mass-cases. Source of figure: [41, p.11]

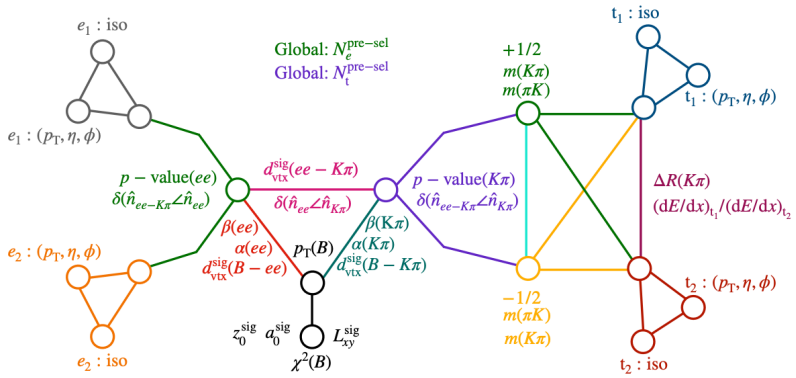


Figure 9.4: The RK* group GNN architecture which can handle the three cases: *CASE1*: $m(K\pi) \in [690, 1110]$MeV AND $m(\pi K) \in [690, 1110]$MeV, *CASE2*: $m(K\pi) \in [690, 1110]$MeV AND $m(\pi K) \notin [690, 1110]$MeV, and *CASE2*: $m(K\pi) \notin [690, 1110]$MeV AND $m(\pi K) \in [690, 1110]$MeV. Branch $+1/2 m(K\pi)m(\pi K)$ represents *CASE2*, branch $-1/2 m(K\pi)m(\pi K)$ represents *CASE3* and both branches together represents *CASE1*. Source of figure: [41, p.7]

The three case-handling of the $m(K\pi)$ and $m(\pi K)$ mass distributions are not as straightforward for GBDTS since the RK* groups approach is a special utilization of the very structure of how GNN works.

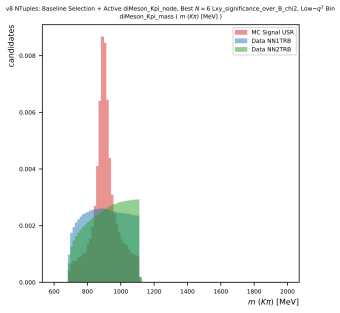
One approach is to replace the **OR** with an **AND** in the cut; however, the group abandoned this approach due to a significant statistical loss. A way to handle the three cases with the GBDS could be by mimicking the GNN approach by splitting data before the GBDTs by which case they belong to.

This would result in one GBDT for each of the three cases.[12] This means that GBDT1 and GBDT2 both consist of three GBDTs, each applied to one of the cases. This approach would still be valid with the training-time argument since $6\times \sim$ 10min are still less than $2 \times$ 4h.

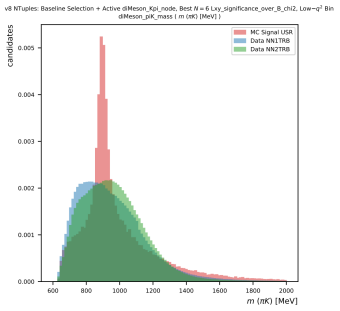
This 2×3 GBDT approach should be tested since GBDTS does not have equal signal efficiencies for q^2_{low} and q^2_{high} for both GBDTs. This is seen in 8.12, especially for GBDT1. This is now fixed for the GNN approach with the handling of the three: $m(K\pi)$,$m(\pi K)$ mass handling cases, and it also gave higher signal efficiency for the GNNs as seen in Fig. (9.3).

9.1.5 *Feature Engennering and GBTDs*

The question of "Which ML model is the best for classifying B^0, $\overline{B}^0$ or background" is not straightforward. As seen above, the RK*s GNN are currently utilizing some of the unique features a GNN model can have. One advantage of using a Neural Network (NN)[13] is that NN can capture non-linearity between the input and output. This means by design, a deep enough[14] NN can capture a lot more of the feature-correlated complexity than a more *shallow* model like a GBDT.



$m(K\pi)$ in low-q^2 bin for active diMeson_Kpi_node



$m(\pi K)$ in low-q^2 bin for active diMeson_Kpi_node

Figure 9.5: An example of how the updated GNN architecture handles the $m(\pi K)/m(K\pi)$ mass distributions in training. As seen, the $m(\pi K)$ are smoothed where the $m(K\pi)$ mass distribution is unsmoothed according to the *CASE2* scenario: $m(K\pi) \in [690, 1110]$MeV AND $m(\pi K) \notin [690, 1110]$MeV. Source of figure: [40, p.21]

[12] This was also a proposed solution with the GNNs however the downside would be the training of six GNNs with a training time of 4h hence abandoned.

[13] Which GNNs are a special case for.

[14] No one knows when a neural network is "Deep enough".

This leads to the topic: *feature engineering*. A deep and complex NN can capture the relation between raw input and output without feature engineering. In the end, feature engineering is another way of applying weighting functions to the original raw data. With enough computational power and a deep and well-crafted NN, the one who uses the NN only needs little to no domain[15] knowledge. The NN also are very robust to outliers and noise. Some NN disadvantages are the computational power required to train and hyperparameter optimization.

[15] Feature engineering needs knowledge on the subject to which the functions are applied.

Although GNNs have many advantages, the GBDTs also have an advantage. As shown, their training speed outperforms the GNNs significantly. They are also robust to outliers due to boosting and one of their main appealing features: interpretability. The tree(s) can be printed, and every decision process mapped out. One of their disadvantages is the inability to capture the non-linearity between input and output. This has to be done with feature engineering, and it puts quite a difficult task on the user of GBDTs. The user needs to have strong domain knowledge and understand how to encode all relationships[16] between features into new features. This means for GBDTs to get better performance, they need better features hence feature engineering. This is already seen from "2GNN to 2GBDT" to the "2GNN to 2GBDT w extra features", where the main difference is the added features/feature engineering.

[16] Both linear and non-linear.

Even though PCA might capture some of the feature relationships, it is still a linear dimensionality reduction method. The question becomes how to compute the non-linear relationship since these relations are not captured anywhere in the feature engineering applied in this thesis. This means further improvements to the GBDT model are possible. One way to capture the non-linearities is by combining NNs and GBDTs. The NN part could be an autoencoder. The idea behind an autoencoder is that it is an unsupervised ML approach made of two parts: An encode and a decoder. The encoder maps the input data into a lower dimension - a so-called bottleneck. The decoder maps it back to the original dimension. The encoder is used the say way as the PCA: dimensionality reduction. The advantage of the encoder is that it is based on a neural network[17], enabling it to capture non-linear relationships of the features. This usage of the autoencoder is called feature extraction. This approach might be better than PCA since it captures the non-linear, and the PCA algorithm assumes that the parts are linearly separable. Unfortunately, autoencoders have the same disadvantages as GNNs/NNs, making them computationally expensive and complicated to optimize hyperparameters.

[17] No domain knowledge is needed.

Another way could be using *Polynomial Features*[23]: which takes features and raises them to the power of a. This is a way to capture non-linearly. Another way could be using *Kernel PCA*[23], which maps the data to higher dimensionality and then takes the most important components like normal PCA. By using non-linear kernels like $RBF_{kernel}(X_1, X_2) = \exp(-\frac{\|X_1 - X_2\|^2}{2\sigma^2})$ the kernel PCA can capture the non-linearity between features.
This is to say that the GBDT's performance can still be improved, and in the end: the time used with GNNs goes to the construction of the network and hyperparameter tuning, whereas for the GBDTs, the time is spent on feature engineering.

9.2 *Uncertainties*

As mentioned in Tab. (2.2), the LHCb presented their newest measurements of $R_{K^{*0}}$ in december 2022[11] with values:
$R_{K^{*0}} = 0.927^{+0.093+0.036}_{-0.087-0.035}$ for $q^2 \in [0.10, 1.1]\text{GeV}/c^2$ and $R_{K^{*0}} = 1.027^{+0.072+0.027}_{-0.068-0.026}$ for $q \in [1.1, 6.0]\text{GeV}/c^2$[11]. A full overview of their analysis methodology can be seen in reference [44], and the different contributions to systematic uncertainties in their analysis are seen in Fig. (9.6).
As the RK* group focus' on the B^0 decay and not B^+, the main focus of Fig. (9.6) is the two rightmost subfigures. The calculation of the $R_{K^{*0}}$-ratio Eq. (2.6) is calculated using the measured yield: N for the signal/control channel and the efficiency; ε for the signal/-control channel. The uncertainties related to the estimated yield are denoted *(fit)*, and the uncertainties related to the efficiency are denoted *(ε)* in Fig. (9.6).

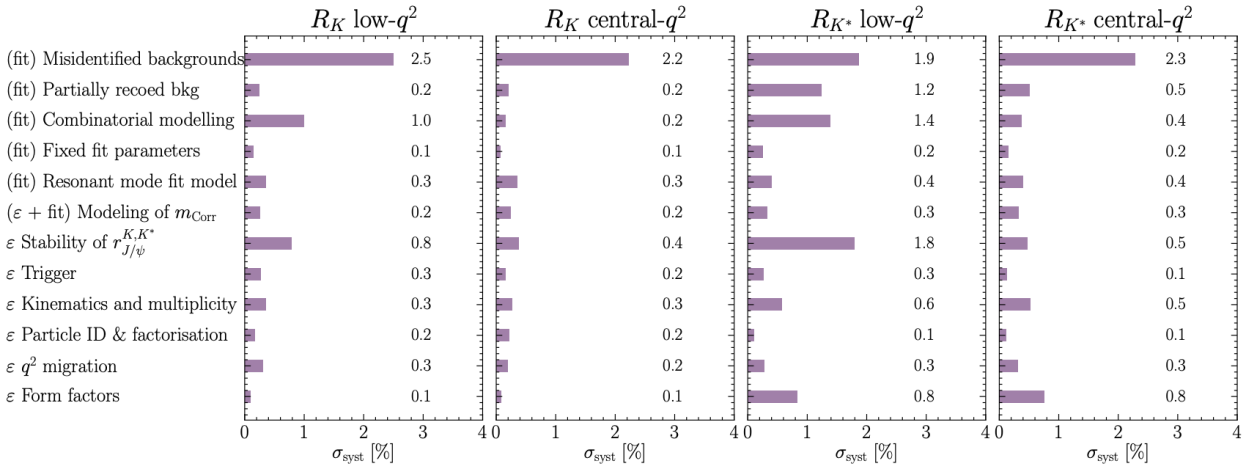


Figure 9.6: LHCb systematic uncertainties for the R_{K^+} and the $R_{K^{*0}}$ ratios in the q^2_{low} and q^2_{high} bin. The *(fit)* means the uncertainty is related to the yield of the fit, and the ε means the uncertainty is related to the efficiency. Source of figure: [57, p.48] and the numbers originate from [44, Table 9, p.54].

The uncertainties mention here, even though the figure and order originate from the LHCb measurements, are also valid for the RK* group measurement of the $R_{K^{*0}}$-ratio.

(fit) Misidentified backgrounds: This is a critical uncertainty to consider since this uncertainty comes directly from the ML efficiency in rejecting background while still keeping the signal. On top of this, the sculpting of the background and signal needs to be avoided, or this uncertainty will blow up. Since the ratio of signal and background events is highly unbalanced, this is a major part of the uncertainty for the RK* analysis, just as it is for the LHCb measurement.

(fit) Partially Reconstructed Backgrounds: The analysis is about measuring the result of the B^0 meson decay particles: K, π, e, and μ however, sometimes not all of these particles are reconstructed correctly (partially rec.) the event will still go through as a signal hence contaminating the signal. It is hard to estimate these uncertainties for ATLAS. However, the uncertainty is higher for ATLAS than LHCb since LHCb is designed for B-physics. ATLAS is an all-purpose detector and can not distinguish between Kaons and pions; hence not specialized for these experiments.

(fit) Combinatorial Modelling: The background consists of a combinatorial of different decays, and the background modeling can be tricky since the choice of background probability function leads to systematic uncertainties. In relation to this, the choice of sidebands also introduced uncertainties since this affects the shape of the background. This is one of the bigger uncertainty contributions for LHCb and the RK* experiment, which is even visible in the analysis of this thesis which is seen when introducing GBDT cuts in the background and the need for a smaller fit-range after the cuts.

(fit) Fixed fit paramters: LHCb, like the RK* group, uses fixed shape parameters for their fits. It is estimated that the errors from fixed parameters are not a significant contributor, which is the same for ATLAS.

(ε + fit) Modeling of m_{Corr}: The modeling of the correlated features to the B-mass are used in the separation of background and signal at LHCb - hence the relation to *(ε)*. This error also leaks into the fitting since the separation also comes into play for the fit hence a systematic error for the *(fit)*. This is a small uncertainty for LHCb and hence also for ATLAS.

(ε) Stability if $r_{J/\psi}^{K,K^*}$: This has to do with the control (J/ψ-part of the Eq. (2.6). The double ratio is used to correct discrepancies in

signal(ee) and signal($\mu\mu$). Given there are imperfections in the calibration of the $r_{J/\psi}^{K,K^*}$ due to discrepancy in data and simulation uncertainty, they are propagated to the final double ratio hence systematic errors. This is quite a large error for R_{K^*} in q^2_{low}; this is because the signal to background ratio is higher in q^2_{high} and hence the q^2_{low} is more sensitive to systematic errors. The relative importance of this uncertainty is the same for the ATLAS RK* analysis.

(ε) Trigger: The Trigger (low and high-level trigger) is used to filter events such that the ones with the highest potential for interesting physics passes. All events pass the trigger. Hence this source of uncertainty is essential to control and keep low since errors will propagate to all other parts of the analysis. The error here is relatively low for ATLAS as well; however bigger than LHCb since LHBC, as mentioned, is designed for B-physics.

(ε) Kinematics and multiplicity: The re-weighting of kinematic variables and dealing with multiplicity are also essential for the analysis. For LHCb, this is a somewhat important factor in total uncertainty. The probability of selecting the best candidates is an ongoing part of ATLAS RK* since it is related to the performance of the GNNs/GBDTs. For the relative size of the *kinematic re-weighting*, uncertainties are estimated to be the same for ATLAS RK*.

(ε) Particle ID & factorization: PID, just as the trigger is, so is the particle identification algorithm active before the analysis. Hence, the systematic uncertainties from this step are propagated through the analysis. Factorization is about the assumption that the components of the decay process are independent; however, in reality, they might be correlated, hence the introduction of systematic uncertainties.

(ε) q^2 migration: This has to do with events that are produced in one q^2-bin and, due to uncertainties in the reconstruction, ends up in another q^2-bin. This means the efficiency of a bin will have uncertainties.

(ε) Form factors: A Form factor is a function that describes the transition amplitudes between the initial state B^0 and the end state and takes the dynamics and the particle structures involved in the decay process into account. They are used to calculate the decay rate, and the uncertainty in form factors is propagated to the efficiency of selecting the signal. The uncertainty here is the same for ATLAS and LHCb.

As shown, multiple areas in the analysis introducing uncertainties; some are more significant than others.

The idea behind using the double ratio 2.6 is that it handles theoretical and systematic uncertainties such that they cancel out.

It is expected by the RK* group that the main uncertainties arise from the most dominant statistical uncertainties in the combined analysis. The statistical uncertainties come from the measured signal/background yield of the fits, which are fitted on limited statistics after the ML cuts are applied to the data.
The RK* groups estimate the uncertainty by generating toy Monte Carlo by randomly sampling from the background part and the signal part of the composite Signal+Background PDF, such they have three toy MC data histograms of Signal S, Background B, and S+B. Then the composite PDF is fitted to the three histograms. This is repeated with a slight change: no signal: S=0. The combined PDF is again fitted to the histograms. The amount of signal gained where there is no signal is called *spurious* signal, and it is possible from this to estimate the statistical uncertainty. Using the expected 28 true signal and 581 true background, it is estimated by the RK* group that the uncertainty at the moment is $\sim$ 50%[39] for now and can further be improved is a work in progress.

9.3 Outline of the Next Step of the RK* Analysis

As of the time of writing the thesis, the current status of the RK*-group analysis is as follows: The $B^0 \rightarrow K^{*0}\mu\mu$ analysis has begun with the separation of B^0, $\overline{B}^0$ and background with both GNNs and GBDTs. Preliminary fits are also done to extract the estimated yield on the muon side.

Furthermore, the estimation of efficiencies is begun along with the estimation of uncertainties. This is quite a task as the efficiency is a product of a long list of other efficiencies: $\varepsilon_{tot} = \varepsilon_{geo} \times (\varepsilon_{MVA} \times \varepsilon_{Pre-select} \times \varepsilon_{Trig} \times \varepsilon_{PID})$ (from Eq. (3.5)).

The work on the J/ψ-control channel is also a work in progress. In addition to calculating the $R_{K^{*0}}$ ratio, efforts are also being made to improve the current n-tuples, specifically enhancing the B-mass through improved feature engineering.

10
Conclusion and Outlook

With the analysis and discussion in mind, it can be concluded that GBDTs is a viable method in separating B^0, $\overline{B}^0$ from background in the $R_{K^{*0}}$-ratio analysis for the electron channel.

With the "2GNN to 2GBDT w extra features" approach yielding a superior signal efficiency in the two q^2-bins: q^2_{low} and q^2_{high} in SR^{MC} for all threshold probabilities. Not only a good signal efficiency, but the GBDT model is also $\sim$ 24 times faster at training and hyperparameter optimizing. With GBDT cuts at (GBDT1, GBDT2) = $(0.1, 0.3)$: The signal yield is $N_{Sig(B^0)} = 1853 \pm (\geq 45)$ in $\{\text{SR}^{\text{PeriodK}}\}$, q^2_{high} Period K data extracted from a χ^2-fit with GoF-parameter of p-value of 0.96 and significance at: $\text{Significance}_{Sig(B^0)} = 20 \pm (\geq 0.4)$ in the electron channel. The total estimated GBDT signal efficiency in $\{\text{SR}^{\text{MC}} \cup \text{SB1}^{\text{PeriodK}} \cup \text{SB2}^{\text{PeriodK}}\}$, q^2_{low} are $\varepsilon_{\text{GBDT}|tot} = 0.72 \pm 0.03$ for the (GBDT1, GBDT2) = $(0.1, 0.3)$ cut.

$$N_{Sig(B^0)} = 1853 \pm (\geq 45) \tag{10.1}$$

$$\varepsilon^{\{\text{SR}^{\text{MC}} \cup \text{SB1}^{\text{PeriodK}}_{\text{Pre-sel}}\}}_{\text{GBDT1}(B^0|\text{cut:0.1})} = 0.854 \pm 0.001 \tag{10.2}$$

$$\varepsilon^{\{\text{SR}^{\text{MC}} \cup \text{SB2}^{\text{PeriodK}}_{\text{Pre-sel}}\}}_{\text{GBDT2}(B^0|\text{cut:0.3})} = 0.848 \pm 0.001 \tag{10.3}$$

$$\varepsilon_{GBDT|tot} = 0.72 \pm 0.03 \tag{10.4}$$

$$\text{Significance}_{Sig(B^0)} = 20 \pm (\geq 0.4) \tag{10.5}$$

This means that GBDTs have the potential to be used in separating B^0, $\overline{B^0}$, and background and hence to be adopted by the RK* group for further analysis. With that said, the GNNs still have better background rejection even though the signal efficiency of the GBDTs is superior; this means that the GBDTs might not be the best solution as a complete substitution with the GNNs at their current stage. As the $^{\text{Signal}}/_{\text{Background}}$-ratio is small, just a few percentages in background rejection would have a large impact on the number of unfiltered backgrounds.

This suggests that the GBDT models do not capture the deeper facets of the background event properties, and improvements in this area need to be pursued.

Even though the GBDTs are inferior in background rejection, they could still be used as an analysis tool to discover new features since the training time is low. This means the time from feature engineering to a trained model is very short, and many iterations can be applied in a relatively small timeframe.

The main goal remains ahead of the RK* group, which is the calculation of the $R_{K^{*0}}$-ratio, and this includes the calculating of the signal yield from the J/ψ control channel for both electron and muons and the signal yield for B^0 decay on the muon side as well. Efficiency studies and estimation of uncertainties are also areas yet to be done[1].

As the thesis is concluded, the next step is to apply the B^0_d, $\overline{B}^0_d$, and background selection on the Muon channel. This work has already begun with other members of the RK* group applying the GBDT approach shown in this thesis on the muon channel. In addition to the contribution of the GBDT approach, codebase changes are suggested to the RK*-group, which entails the implementation of multiprocessing in the n-tuple to feather-file step along with faster pre-selection schemes[50]; these will be reviewed for the next iteration of the RK* codebase.

In addition to the muon channel, new n-tuples for the electron channel will provide better-calibrated features. This will result in even better GBDT models due to better features. On top of that, when Run2 data has been analyzed, the analysis can be applied to Run3, with has much higher luminosity which will provide the RK* group with more statistics and hence much better $R_{K^{*0}}$-ratio measurements. It will be an exciting paper to read the day it is published.

[1] Note that the mentioned approaches used by the RK* group are still a work in progress and may be subject to change in the future. For the current methodology used, please get in touch with the RK* group directly.

Bibliography

[1] Morad Aaboud et al. "Electron reconstruction and identification in the ATLAS experiment using the 2015 and 2016 LHC proton-proton collision data at $\sqrt{s}$ = 13 TeV". In: *Eur. Phys. J. C* 79.8 (2019), p. 639. DOI: 10.1140/epjc/s10052-019-7140-6. arXiv: 1902.04655 [physics.ins-det]. URL: https://doi.org/10.1140/epjc/s10052-019-7140-6.

[2] *Accelerators*. Url link and all sub-pages are about present and past accelerators at CERN, Last accessed 05/04/2023. URL: https://home.cern/science/accelerators.

[3] Takuya Akiba et al. "Optuna: A Next-Generation Hyperparameter Optimization Framework". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19. Anchorage, AK, USA: Association for Computing Machinery, 2019, pp. 2623–2631. ISBN: 9781450362016. DOI: 10.1145/3292500.3330701. URL: https://doi.org/10.1145/3292500.3330701.

[4] James Bergstra et al. "Algorithms for Hyper-Parameter Optimization". In: *Advances in Neural Information Processing Systems*. Ed. by J. Shawe-Taylor et al. Vol. 24. Curran Associates, Inc., 2011. URL: https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf.

[5] Sylvia Berryman. *Democritus*. Last accessed 07/03/2023. Jan. 2023. URL: https://plato.stanford.edu/entries/democritus/.

[6] Andrzej Buras and Robert Fleischer. "Quark Mixing, CP Violation and Rare Decays After the Top Quark Discovery". In: (May 1997). DOI: 10.1142/9789812812667_0002. URL: https://doi.org/10.1142/9789812812667_0002.

[7] James Catmore. *The ATLAS data processing chain: from collisions to papers*. Joint Oslo/Bergen/NBI ATLAS Software Tutorial. Last accessed 04/04/2023. Feb. 2016. URL: https://indico.cern.ch/event/472469/contributions/1982677/attachments/1220934/1785823/intro_slides.pdf.

[8] Tianqi Chen and Carlos Guestrin. "XGBoost". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Aug. 2016. DOI: 10.1145/2939672.2939785. URL: https://doi.org/10.1145/2939672.2939785.

[9] S. Choudhury et al. "Test of lepton flavor universality and search for lepton flavor violation in $B \to K\ell\ell$ decays". In: *Journal of High Energy Physics* 2021.3 (Mar. 2021), p. 105. DOI: 10.1007/JHEP03(2021)105. URL: https://doi.org/10.1007/JHEP03(2021)105.

[10] ATLAS Collaboration. *Athena*. Version 22.0.1. Last accessed 20/05/2023. Apr. 2019. DOI: 10.5281/zenodo.2641997. URL: https://doi.org/10.5281/zenodo.2641997.

[11] LHCb Collaboration. "Test of lepton universality in $b \to s\ell^+\ell^-$ decays". In: (Dec. 2022). DOI: 10.48550/ARXIV.2212.09152. URL: https://arxiv.org/abs/2212.09152.

[12] The ATLAS collaboration. "A study of the material in the ATLAS inner detector using secondary hadronic interactions". In: *Journal of Instrumentation* 7.01 (Jan. 2012). Last accessed 24/03/2023, P01013. DOI: 10.1088/1748-0221/7/01/P01013. URL: https://dx.doi.org/10.1088/1748-0221/7/01/P01013.

[13] Microsoft Corporation. *Light Gradient Boosting Machine - LightGBM*. https://github.com/microsoft/LightGBM. Last accessed 05/05/2023. Jan. 2023.

[14] *Data and Monte Carlo Datasets for Analysis*. Last accessed 04/04/2023, Internal source: access only for ATLAS members. Mar. 2023. URL: https://twiki.cern.ch/twiki/bin/view/AtlasProtected/DataMCForAnalysis#2018_rel22.

[15] Hans Dembinski et al. *scikit-hep/iminuit*. Last accessed 2/05/2023. URL: https://github.com/scikit-hep/iminuit.

[16] *Detector & Technology*. The url link are about the ATLAS detector in general and sub-pages include information regarding: The Inner Detector, The Magnet System, The Calorimeter, The Muon Spectrometer, The Trigger & Data Acquisition and lastly The Software & Computing. Last accessed 05/04/2023. URL: https://atlas.cern/Discover/Detector.

[17] Scikit-learn developers. *Sklearn Preprocessing RobustScaler*. Last accessed 10/04/2023. 2023. URL: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html.

[18] Luigi Di Lella and Carlo Rubbia. "The Discovery of the W and Z Particles". In: *Adv. Ser. Direct. High Energy Phys.* 23 (2015), pp. 137–163. DOI: 10.1142/9789814644150_0006.

[19] Wikimedia Commons - Public Domain. *RGB color model*. https://commons.wikimedia.org/. Last accessed 07/03/2023. Jan. 2018. URL: https://commons.wikimedia.org/wiki/File:RGB_color_model.svg.

[20] A. Einstein. "Die Grundlage der allgemeinen Relativitätstheorie". In: *Annalen der Physik* 354.7 (1916), pp. 769–822. DOI: https://doi.org/10.1002/andp.19163540702. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19163540702. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19163540702.

[21] Robert M Eisberg and Robert Resnick. *Quantum physics of atoms, molecules, solids, nuclei, and particles*. en. 2nd ed. Nashville, TN: John Wiley & Sons, Jan. 1985.

[22] F. Englert and R. Brout. "Broken Symmetry and the Mass of Gauge Vector Mesons". In: *Phys. Rev. Lett.* 13 (9 Aug. 1964), pp. 321–323. DOI: 10.1103/PhysRevLett.13.321. URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.321.

[23] Aurelien Geron. *Hands-on machine learning with scikit-learn, keras, and TensorFlow*. 2nd ed. Sebastopol, CA: O'Reilly Media, Oct. 2019.

[24] S. L. Glashow, J. Iliopoulos, and L. Maiani. "Weak Interactions with Lepton-Hadron Symmetry". In: *Phys. Rev. D* 2 (7 Oct. 1970), pp. 1285–1292. DOI: 10.1103/PhysRevD.2.1285. URL: https://link.aps.org/doi/10.1103/PhysRevD.2.1285.

[25] David Griffiths. *Introduction to elementary particles*. en. 2nd ed. Weinheim, Germany: Wiley-VCH Verlag, Aug. 2008.

[26] Mukund Gupta. *Calculation of radiation length in materials*. Tech. rep. Last accessed 24/03/2023. Geneva: CERN, 2010. URL: https://cds.cern.ch/record/1279627.

[27] Peter W. Higgs. "Broken symmetries, massless particles and gauge fields". In: *Phys. Lett.* 12 (1964), pp. 132–133. DOI: 10.1016/0031-9163(64)91136-9.

[28] *High-Luminosity LHC*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/high-luminosity-lhc.

[29] Gudrun Hiller and Frank Krüger. "More Model-Independent Analysis of b->s Processes". In: *Phys. Rev. D* 69 (7 Apr. 2004), p. 074020. DOI: 10.1103/PhysRevD.69.074020. URL: https://link.aps.org/doi/10.1103/PhysRevD.69.074020.

[30] Tomas Jakoubek. "Introduction, news". R(K*) analysis meeting, Last accessed 04/05/2023, Internal source: access only for ATLAS members. Mar. 21, 2022. URL: https://indico.cern.ch/event/1136406/contributions/4768309/attachments/2411372/4126332/jakoubek_rkstar_20220321.pdf.

[31] F. James and M. Roos. "Minuit - a system for function minimization and analysis of the parameter errors and correlations". In: *Computer Physics Communications* 10.6 (1975), pp. 343–367. ISSN: 0010-4655. DOI: https://doi.org/10.1016/0010-4655(75)90039-9. URL: https://www.sciencedirect.com/science/article/pii/0010465575900399.

[32] Guolin Ke et al. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

[33] 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.

[34] J. P. Lees et al. "Measurement of branching fractions and rate asymmetries in the rare decays $B \to K^{(*)}\ell^+\ell^-$". In: *Phys. Rev. D* 86 (3 Aug. 2012), p. 032012. DOI: 10.1103/PhysRevD.86.032012. URL: https://link.aps.org/doi/10.1103/PhysRevD.86.032012.

[35] *Linear accelerator 3*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/linear-accelerator-3.

[36] *Linear accelerator 4*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/proton-synchrotron.

[37] Ewa Lopienska. "The CERN accelerator complex, layout in 2022. Complexe des accélérateurs du CERN en janvier 2022". In: (2022). General Photo, Last accessed 20/03/2023. URL: https://cds.cern.ch/record/2800984.

[38] Scott Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *arXiv e-prints*, arXiv:1705.07874 (May 2017). Provided by the SAO/NASA Astrophysics Data System, arXiv:1705.07874. DOI: 10.48550/arXiv.1705.07874. arXiv: 1705.07874 [cs.AI].

[39] Dvij Chaitanya Mankad. "Introduction, news". Overview of the Current Status of the Electron Channel of the R(K*) Measurement, Last accessed 04/05/2023, Internal source: access only for ATLAS members. Aug. 10, 2022. URL: https://indico.cern.ch/event/1176850/contributions/4990561/attachments/2491433/4280700/analysis-review-talk-rkstar-ee-10Aug2022-updated.pdf.

[40] Dvij Chaitanya Mankad. "Proposed Roads to Dealing with the Dimeson Mass Tail". R(K*) Meeting | 11 January 2023. Last accessed 08/05/2023, Internal source: access only for ATLAS members. Jan. 11, 2023. URL: https://indico.cern.ch/event/1239058/contributions/5210233/attachments/2574154/4438532/dimeson-mass-tail.pdf.

[41] Dvij Chaitanya Mankad. "Update on Recent Work on GNNs". R(K*) Meeting | 15 February 2023. Last accessed 08/05/2023, Internal source: access only for ATLAS members. Feb. 15, 2023. URL: https://indico.cern.ch/event/1255183/contributions/5273009/attachments/2594215/4477614/gnn-update-16022023.pdf.

[42] Brian Martin and Graham P Shaw. *Particle Physics*. en. 3rd ed. Manchester Physics. Hoboken, NJ: Wiley-Blackwell, Oct. 2008.

[43] T. Massam et al. "Experimental observation of antideuteron production". In: *Il Nuovo Cimento A* 63.1 (Sept. 1965), pp. 10–14. DOI: 10.1007/BF02898804. URL: https://doi.org/10.1007/BF02898804.

[44] "Measurement of lepton universality parameters in $B^+ \to K^+\ell^+\ell^-$ and $B^0 \to K^{*0}\ell^+\ell^-$ decays". In: (Dec. 2022). arXiv: 2212.09153 [hep-ex].

[45] *Meet the Constants*. Last accessed 05/04/2023. Dec. 2019. URL: https://www.nist.gov/si-redefinition/meet-constants.

[46] Sascha Mehlhase. "ATLAS detector slice (and particle visualisations)". In: (2021). A slice of ATLAS as well as visualisations of how ATLAS detects different particles. Work based on earlier work/versions by Rebecca Pitt and Joao Pequenao. Last accessed 21/03/2023. URL: https://cds.cern.ch/record/2770815.

[47] Lingxin Meng. "ATLAS ITk Pixel Detector Overview". In: *International Workshop on Future Linear Colliders*. May 2021. arXiv: 2105.10367 [physics.ins-det].

[48] Microsoft. *LightGBM Parameters*. Last accessed 30/03/2023. URL: https://lightgbm.readthedocs.io/en/latest/Parameters.html.

[49] Cush MissMJ. *Standard Model of Elementary Particles*. This work has been released into the public domain by its author, Cush. This applies worldwide. In some countries this may not be legally possible; if so: Cush grants anyone the right to use this work for any purpose, without any conditions, unless such conditions are required by law. Last accessed 07/03/2023. Feb. 2023. URL: https://upload.wikimedia.org/wikipedia/commons/0/00/Standard_Model_of_Elementary_Particles.svg.

[50] Daniel Hans Munk. *RKstar_MasterThesis*. https://gitlab.cern.ch/dmunk/rkstar_masterthesis. Last accessed 20/05/2023. May 2023.

[51] Kevin P Murphy. *Probabilistic Machine Learning*. London, England: MIT Press, Mar. 2022.

[52] Izaak Neutelings. *B tagging jets*. This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/. Last accessed 27/03/2023. 2021. URL: https://tikz.net/jet_btag/.

[53] Izaak Neutelings. *CMS coordinate system*. This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/. Last accessed 27/03/2023. 2021. URL: https://tikz.net/axis3d_cms/.

[54] "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC". In: *Physics Letters B* 716.1 (2012), pp. 1–29. ISSN: 0370-2693. DOI: https://doi.org/10.1016/j.physletb.2012.08.020.

[55] Yoshihiko Ozaki et al. "Multiobjective Tree-Structured Parzen Estimator for Computationally Expensive Optimization Problems". In: *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*. GECCO '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 533–541. ISBN: 9781450371285. DOI: 10.1145/3377930.3389817.

[56] Joao Pequenao. "Computer generated image of the whole ATLAS detector". Figure Tag: CERN-GE-0803012-05, Last accessed 21/03/2023. 2008. URL: https://cds.cern.ch/record/1095924.

[57] Renato Quagliani. "Measurements of R_K and R_{K^*} with the full LHCb Run 1 and 2 data". LHC Seminar, CERN. Last accessed 15/05/2023. Dec. 20, 2022. URL: https://indico.

cern.ch/event/1187945/attachments/2569929/4431222/RX_LHC_Seminar_rquagliani.pdf.

[58] and R. Aaij et al. "Test of lepton universality in beauty-quark decays". In: *Nature Physics* 18.3 (Mar. 2022), pp. 277–282. DOI: 10.1038/s41567-021-01478-8. URL: https://doi.org/10.1038%5C%2Fs41567-021-01478-8.

[59] *SCIENCE HPC Centre*. Last accessed 07/04/2023. URL: https://science.ku.dk/english/research/research-e-infrastructure/science-hpc-centre/.

[60] "Search for lepton-universality violation in $B^+ \to K^+\ell^+\ell^-$ decays". Version 2. In: *Phys. Rev. Lett.* 122 (2019). All figures and tables, along with any supplementary material and additional information, are available at https://cern.ch/lhcbproject/Publications/p/LHCb-PAPER-2019-009.html, p. 191801. DOI: 10.1103/PhysRevLett.122.191801. arXiv: 1903.09252v2. URL: https://cds.cern.ch/record/2668514.

[61] Maksym Teklishyn. "Measurement of the η c (1S) production cross-section via the decay η c to proton-antiproton final state". In: (Sept. 2014). URL: https://www.researchgate.net/publication/280899986_Measurement_of_the_e_c_1S_production_cross-section_via_the_decay_e_c_to_proton-antiproton_final_state.

[62] "Test of lepton universality with $B^0 \to K^{*0}\ell^+\ell^-$ - decays". In: *Journal of High Energy Physics* 2017.8 (2017). DOI: 10.1007/jhep08(2017)055.

[63] "Tests of Lepton Universality Using $B^0 \to K^0_S\ell^+\ell^-$ and $B^+ \to K^{*+}\ell^+\ell^-$ Decays". In: *Phys. Rev. Lett.* 128 (19 May 2022), p. 191802. DOI: 10.1103/PhysRevLett.128.191802.

[64] *The history of CERN*. And sub-pages under this webpage. Last accessed 05/04/2023. URL: https://timeline.web.cern.ch/taxonomy/term/89?page=0.

[65] *The Large Hadron Collider*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/large-hadron-collider.

[66] *The Low Energy Ion Ring*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/low-energy-ion-ring.

[67] *The Proton Synchrotron*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/proton-synchrotron.

[68] *The Proton Synchrotron Booster*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/proton-synchrotron-booster.

[69] *The Super Proton Synchrotron*. Last accessed 05/04/2023. URL: https://home.cern/science/accelerators/super-proton-synchrotron.

[70] Makayla Vessella. *ATLAS Tracking Software Tutorial*. Subpages are also used. Last accessed 27/03/2023. July 2021. URL: https://atlassoftwaredocs.web.cern.ch/trackingTutorial/.

[71] S. Wehle et al. "Test of Lepton-Flavor Universality in $B \to K^* \ell^+ \ell^-$ Decays at Belle". In: *Phys. Rev. Lett.* 126 (16 Apr. 2021), p. 161801. DOI: 10.1103/PhysRevLett.126.161801. URL: https://link.aps.org/doi/10.1103/PhysRevLett.126.161801.

[72] R L Workman et al. "Review of Particle Physics". In: *Progress of Theoretical and Experimental Physics* 2022.8 (Aug. 2022). 083C01. ISSN: 2050-3911. DOI: 10.1093/ptep/ptac097. eprint: https://academic.oup.com/ptep/article-pdf/2022/8/083C01/49175539/ptac097.pdf. URL: https://doi.org/10.1093/ptep/ptac097.

[73] Danil Zherebtsov. *Verstack*. https://github.com/DanilZherebtsov/verstack. Version: 3.6.5. Last accessed 05/05/2023. May 2023.

Appendix

List of Figures

List of Tables

A.1 Feature Engineering

Here is a list of functions with their explanations above. Every function can be retraced to the n-tuples with the *BeeKst* at the beginning of the variables.

Four-momentum vector for electron X:

$$vtx\_eX\_p4 = \begin{bmatrix} pT = BeeKst\_electronX\_pT \\ \eta = BeeKst\_electronX\_eta \\ \phi = BeeKst\_electronX\_phi \\ m = 0.551\,\mathrm{MeV/c^2} \end{bmatrix} \text{ where } X \in [0,1] \quad \text{(A.1)}$$

Four-momentum vector for the K-meson at meson X:

$$vtx\_mX\_K\_p4 = \begin{bmatrix} pT = BeeKst\_mesonX\_pT \\ \eta = BeeKst\_mesonX\_eta \\ \phi = BeeKst\_mesonX\_phi \\ m = 493.677\,\mathrm{MeV/c^2} \end{bmatrix} \text{ where } X \in [0,1] \tag{A.2}$$

four-momentum vector for the π-meson at meson X:

$$vtx\_mX\_pi\_p4 = \begin{bmatrix} pT = BeeKst\_mesonX\_pT \\ \eta = BeeKst\_mesonX\_eta \\ \phi = BeeKst\_mesonX\_phi \\ m = 139.57\,\mathrm{MeV/c^2} \end{bmatrix} \text{ where } X \in [0,1] \tag{A.3}$$

Four-momentum vector for di-electron system:

$$vtx\_diLepton\_p4 = vtx\_e0\_p4 + vtx\_e1\_p4 \tag{A.4}$$

Four-momentum vector for the K-π system:

$$vtx\_Kpi\_p4 = vtx\_m0\_p4 + vtx\_m1\_p4 \tag{A.5}$$

Four-momentum vector for the B-meson

$$vtx\_Bd\_p4 = vtx\_diLepton\_p4 + vtx\_Kpi\_p4 \tag{A.6}$$

3D position vector of the primary vertex (PV) from which the B-meson is produced:

$$vtx\_Bd\_pv = \begin{bmatrix} x = BeeKst\_PV\_minA0\_x \\ y = BeeKst\_PV\_minA0\_y \\ z = BeeKst\_PV\_minA0\_z \end{bmatrix} \tag{A.7}$$

The 3D error position vector of the primary vertex (PV) from which the B-meson is produced:

$$vtx\_Bd\_pv\_err = \begin{bmatrix} x = BeeKst\_PV\_minA0\_x\_err \\ y = BeeKst\_PV\_minA0\_y\_err \\ z = BeeKst\_PV\_minA0\_z\_err \end{bmatrix} \tag{A.8}$$

The 3D position vector of the B-meson decay vertex:

$$vtx\_Bd\_vtx = \begin{bmatrix} x = BeeKst\_x \\ y = BeeKst\_y \\ z = BeeKst\_z \end{bmatrix} \tag{A.9}$$

The 3D error position vector of the B-meson decay vertex:

$$vtx\_Bd\_vtx\_err = \begin{bmatrix} x = BeeKst\_x\_err \\ y = BeeKst\_y\_err \\ z = BeeKst\_z\_err \end{bmatrix} \tag{A.10}$$

The 3D position vector of the two mesons decay vertex:

$$vtx\_diMeson\_vtx = \begin{bmatrix} x = BeeKst\_diMeson\_vtx\_x \\ y = BeeKst\_diMeson\_vtx\_y \\ z = BeeKst\_diMeson\_vtx\_z \end{bmatrix} \tag{A.11}$$

The 3D error position vector of the two mesons decay vertex:

$$vtx\_diMeson\_vtx\_err = \begin{bmatrix} x = BeeKst\_diMeson\_vtx\_x\_err \\ y = BeeKst\_diMeson\_vtx\_y\_err \\ z = BeeKst\_diMeson\_vtx\_z\_err \end{bmatrix} \tag{A.12}$$

The 3D position vector of the two-electron decay vertex:

$$vtx\_diLepton\_vtx = \begin{bmatrix} x = BeeKst\_diElectron\_vtx\_x \\ y = BeeKst\_diElectron\_vtx\_y \\ z = BeeKst\_diElectron\_vtx\_z \end{bmatrix} \tag{A.13}$$

The 3D error position vector of the two-electron decay vertex:

$$vtx\_diLepton\_vtx\_err = \begin{bmatrix} x = BeeKst\_diElectron\_vtx\_x\_err \\ y = BeeKst\_diElectron\_vtx\_y\_err \\ z = BeeKst\_diElectron\_vtx\_z\_err \end{bmatrix} \tag{A.14}$$

The displacement vector between the primary vertex (PV) and the two meson vertex:

$$vtx\_B2diMeson = vtx\_diMeson\_vtx - vtx\_Bd\_vtx \tag{A.15}$$

The displacement vector between the primary vertex (PV) and the two lepton vertex:

$$vtx\_B2diLepton = vtx\_diLepton\_vtx - vtx\_Bd\_vtx \tag{A.16}$$

The direction of the plane (normal vector) that contains the trajectories of the two leptons and two mesons produced in the B meson decay:

$$vtx\_n1 = vtx\_B2diLepton \times vtx\_B2diMeson \tag{A.17}$$

The direction of the plane (normal vector) that contains the trajecto-

ries of the two leptons and the K-π pair:

$$vtx\_n1\_pp = vtx\_diLepton\_p4 \times vtx\_Kpi\_p4 \tag{A.18}$$

The direction of the plane (normal vector) that contains the trajectories of the two leptons and is orthogonal to the plane described by vtx_n1.:

$$vtx\_n2\_pee = vtx\_n1 \times vtx\_diLepton\_p4 \tag{A.19}$$

The direction of the plane (normal vector) that contains the trajectories of the two electrons:

$$vtx\_n2\_ee = vtx\_e0\_p4 \times vtx\_e1\_p4 \tag{A.20}$$

The direction of the plane (normal vector) that contains the trajectories of the K-π pair and is orthogonal to the plane described by vtx_n1.:

$$vtx\_n2\_pmm = vtx\_n1 \times vtx\_Kpi\_p4) \tag{A.21}$$

The direction of the plane (normal vector) that contains the trajectories of the K and π:

$$vtx\_n2\_mm = vtx\_m0\_K\_p4 \times vtx\_m1\_pi\_p4 \tag{A.22}$$

The direction (normal vector) that is orthogonal to both the plane containing the trajectories of the two leptons and the plane described by vtx_n1:

$$vtx\_n3\_pee = vtx\_n2\_pee \times vtx\_n1) \tag{A.23}$$

The direction (normal vector) that is orthogonal to both the plane containing the trajectories of the K-π pair and the plane described by vtx_n1:

$$vtx\_n3\_pmm = vtx\_n2\_pmm \times vtx\_n1) \tag{A.24}$$

A.2 2GNN to 2GBDT

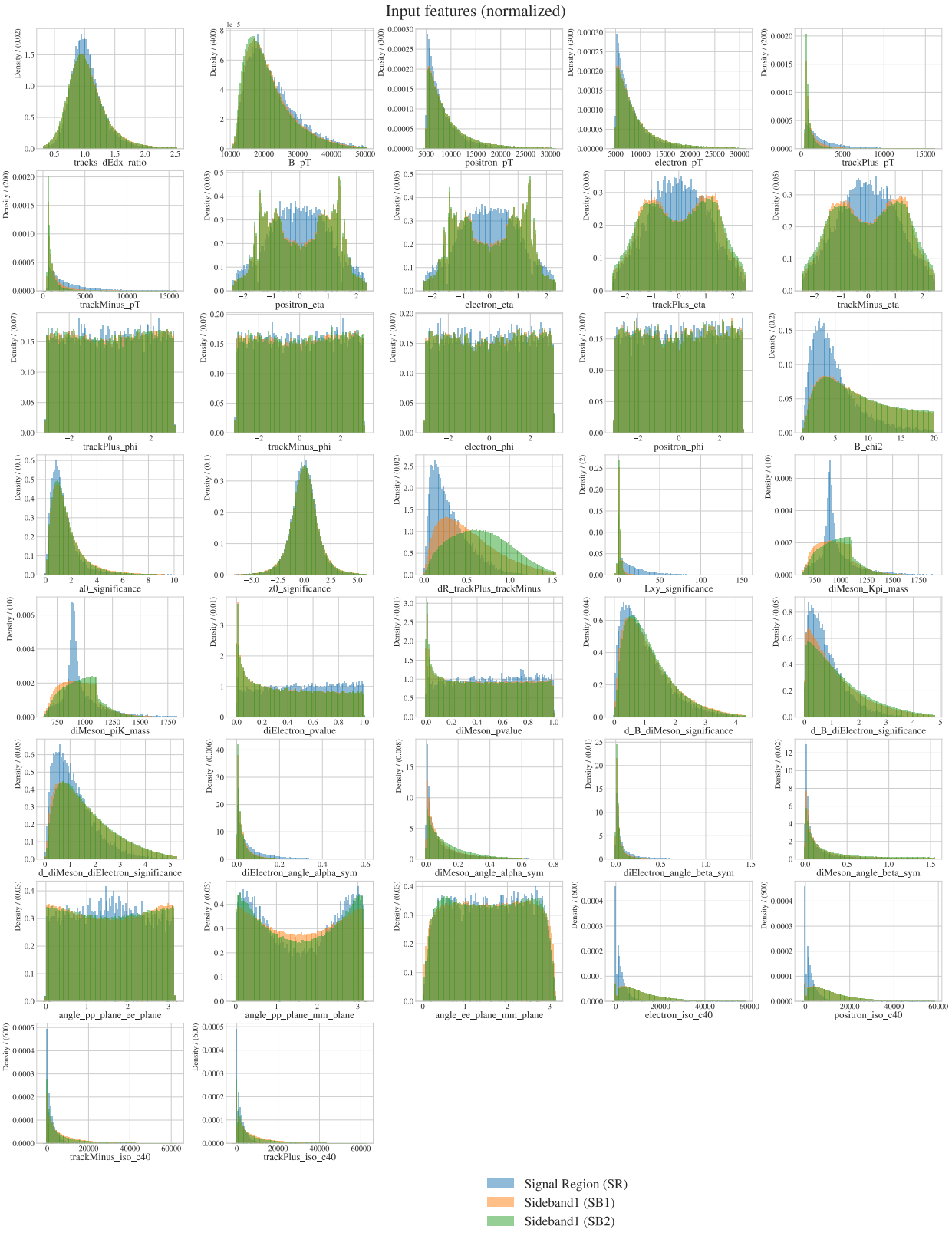


Figure A.1: Normalized features used in "2GNN to 2GBDT". The plotted features are all separated into Signal Region (SR), Sideband1 (SB1), and Sideband2.

	GBDT1	GBDT2
Traning time	8min, 10.2sec	9min 20,4sec
task	train	train
learning_rate	0.05	0.05
num_leaves	73	190
colsample_bytree	0.856116	0.798722
subsample	0.849546	0.668812
bagging_freq	1	1
max_depth	-1	-1
verbosity	-1	-1
reg_alpha	0.000003	0.000001
reg_lambda	0.001103	0.000019
min_split_gain	0.0	0.0
zero_as_missing	False	False
max_bin	255	255
min_data_in_bin	3	3
random_state	42	42
device_type	cpu	cpu
num_classes	3	3
objective	multiclass	multiclass
metric	multi_logloss	multi_logloss
num_threads	42	42
min_sum_hessian_in_leaf	1.862499	2.497306
n_estimators	246	104

Table A.1: "GNN to GBDT" Best values for the LightGBM models

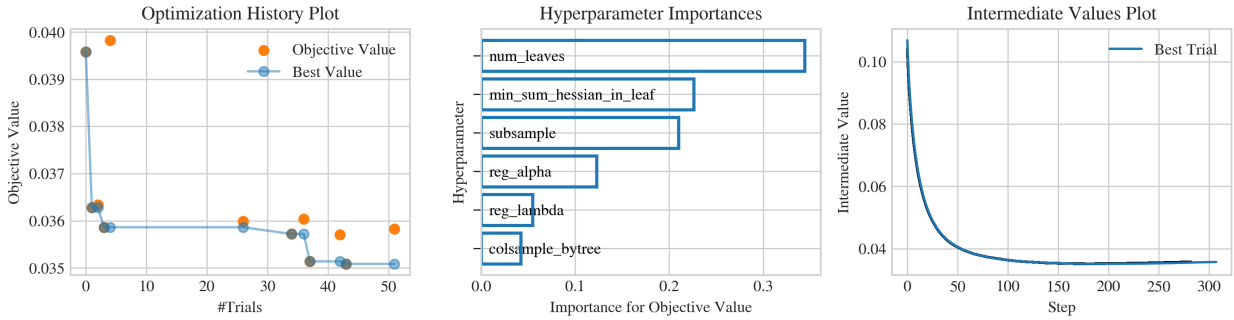


Figure A.2: *LightGBM* Testing Suite on "2GNN to 2GBDT"-GBDT1. Showing that GBDT1s training has converged.

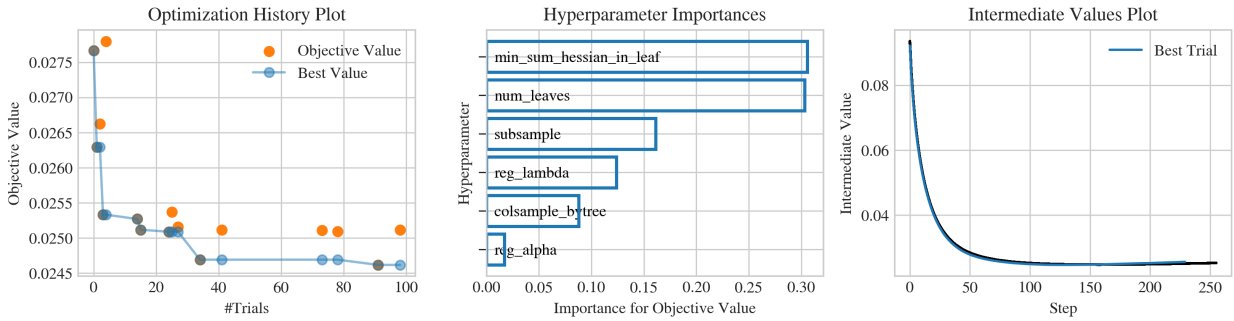


Figure A.3: *LightGBM* Testing Suite on "2GNN to 2GBDT"-GBDT2. Showing that GBDT2s training has converged.

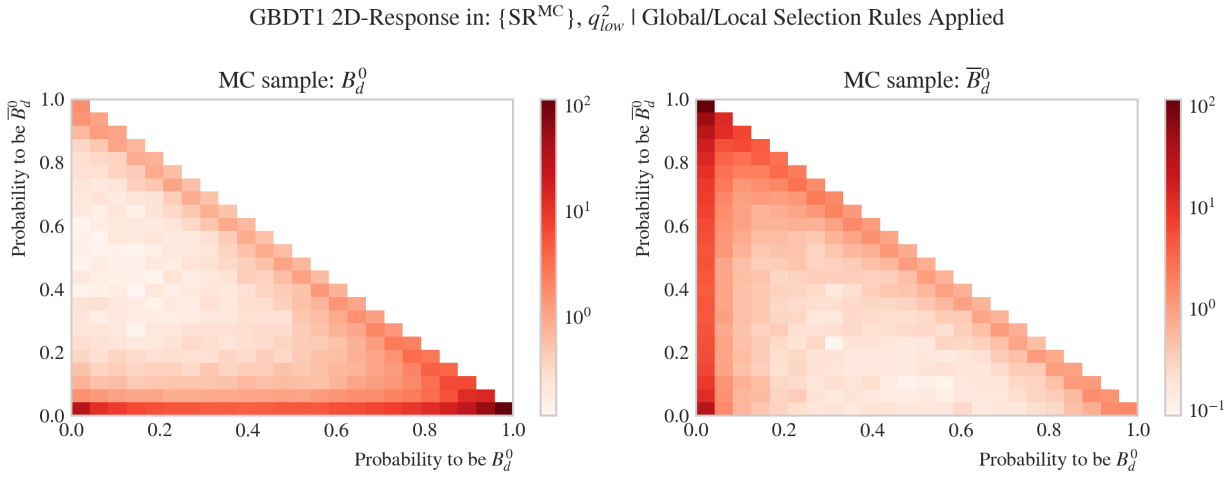


Figure A.4: *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT"-GBDT1 with density on log-scale showing good classification.

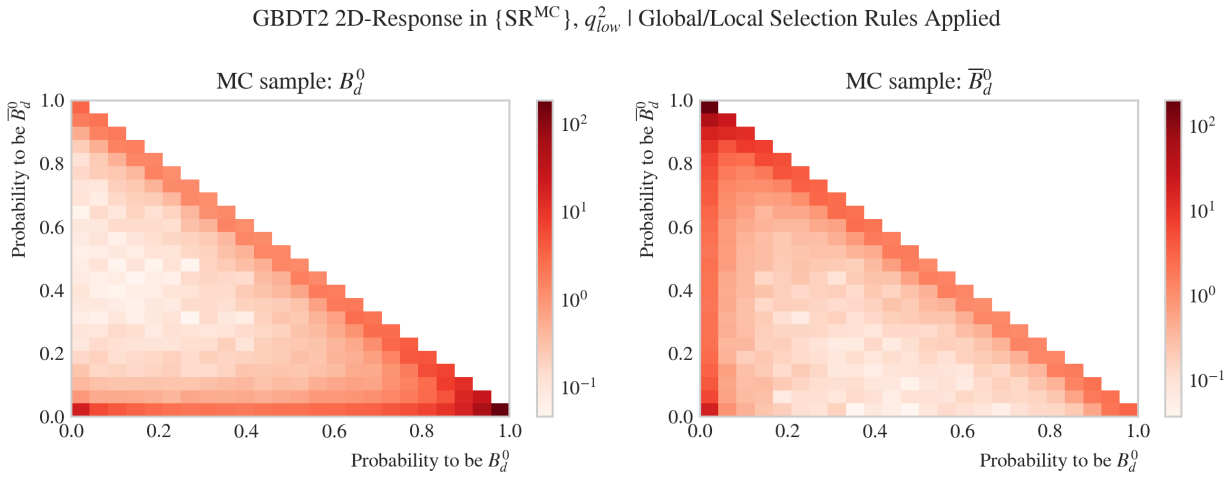


Figure A.5: *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT"-GBDT2 with density on log-scale showing good classification.

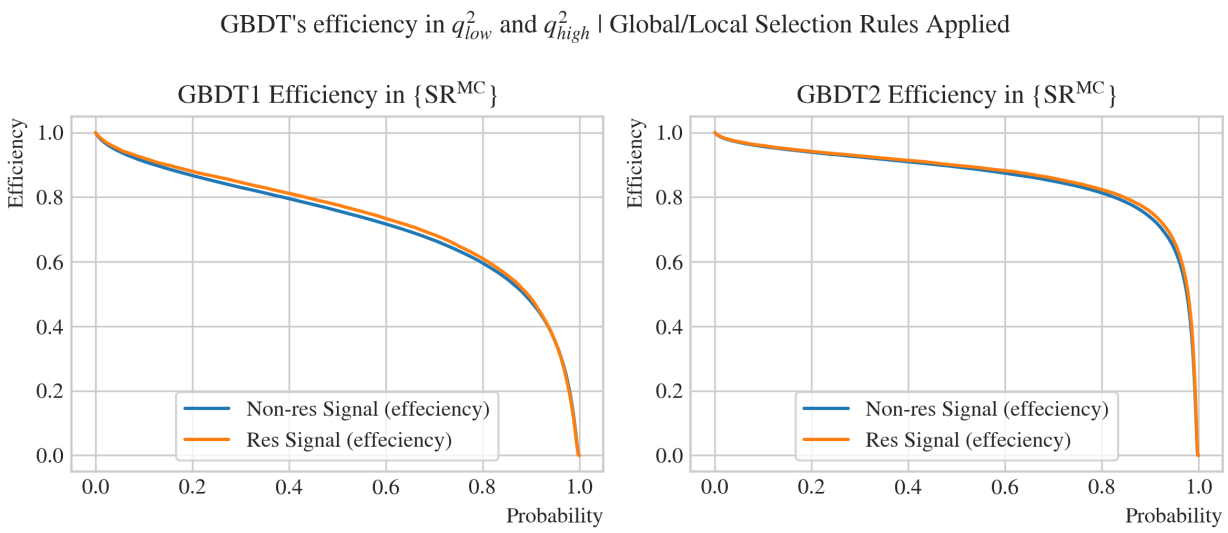


Figure A.6: Resonant vs. Non-Resonant signal efficiency plots for both the q^2_{low} and q^2_{high} bin for both GBDTs in "2GNN to 2GBDT". The plot shows the resonant and non-resonant has similar performance.

A.3 2GNN to 3GBDT

	GBDT1	GBDT2	GBDT3
Training time	7min 18.5sec	5min 36.4sec	2min 32.4sec
task	train	train	train
learning_rate	0.05	0.05	0.03
num_leaves	245	224	91
colsample_bytree	0.980572	0.802162	0.96863
subsample	0.663678	0.818206	0.50196
bagging_freq	1	1	1
max_depth	-1	-1	-1
verbosity	-1	-1	-1
reg_alpha	0.01227	0.000003	0.00224
reg_lambda	0.000014	0.000519	0.039626
min_split_gain	0.0	0.0	0.0
zero_as_missing	False	False	False
max_bin	255	255	255
min_data_in_bin	3	3	3
random_state	42	42	42
device_type	cpu	cpu	cpu
num_classes	1	1	1
objective	binary	binary	binary
metric	binary_logloss	binary_logloss	binary_logloss
num_threads	30	30	30
min_sum_hessian_in_leaf	5.93633	3.673439	0.040701
n_estimators	237	163	237

Table A.2: "2GNN to 3GBDT" best values for the LightGBM models.

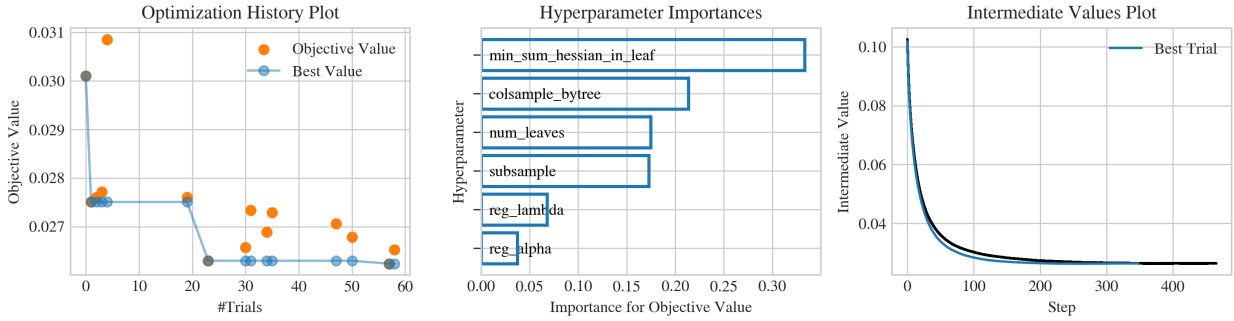


Figure A.7: *LightGBM* Testing Suite on "2GNN to 3GBDT"-GBDT1. Showing that GBDT1s training has converged.

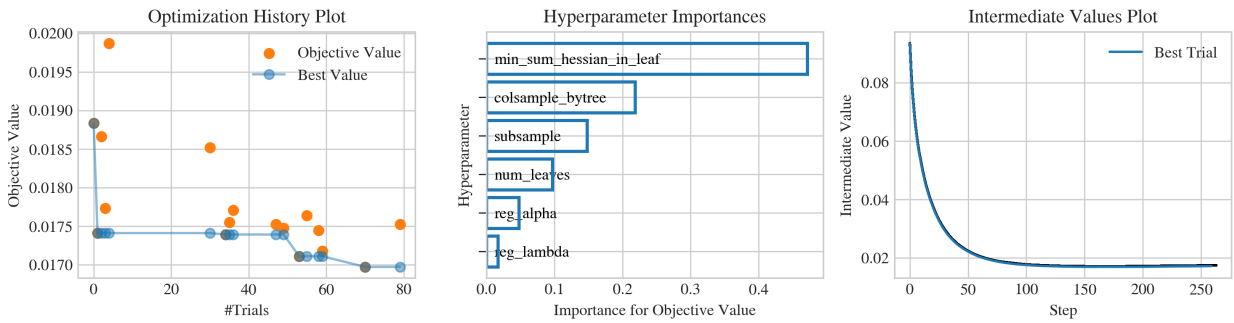


Figure A.8: *LightGBM* Testing Suite on "2GNN to 3GBDT"-GBDT2. Showing that GBDT2s training has converged.

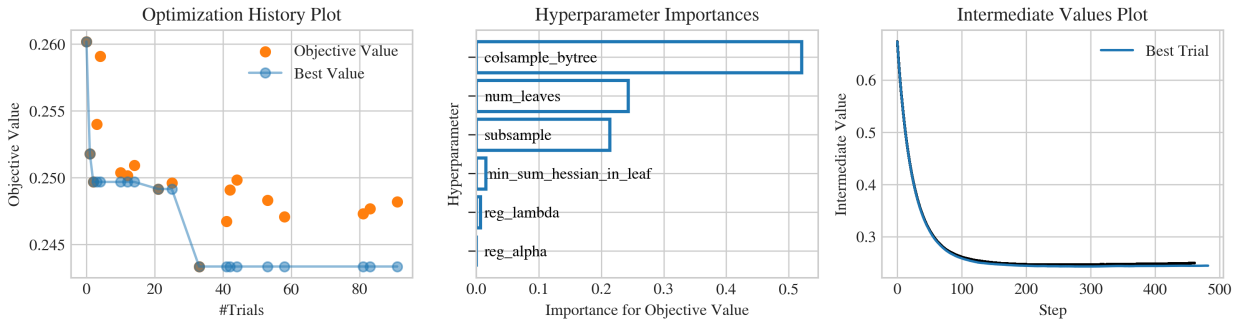


Figure A.9: *LightGBM* Testing Suite on "2GNN to 3GBDT"-GBDT3. Showing that GBDT2s training has converged.

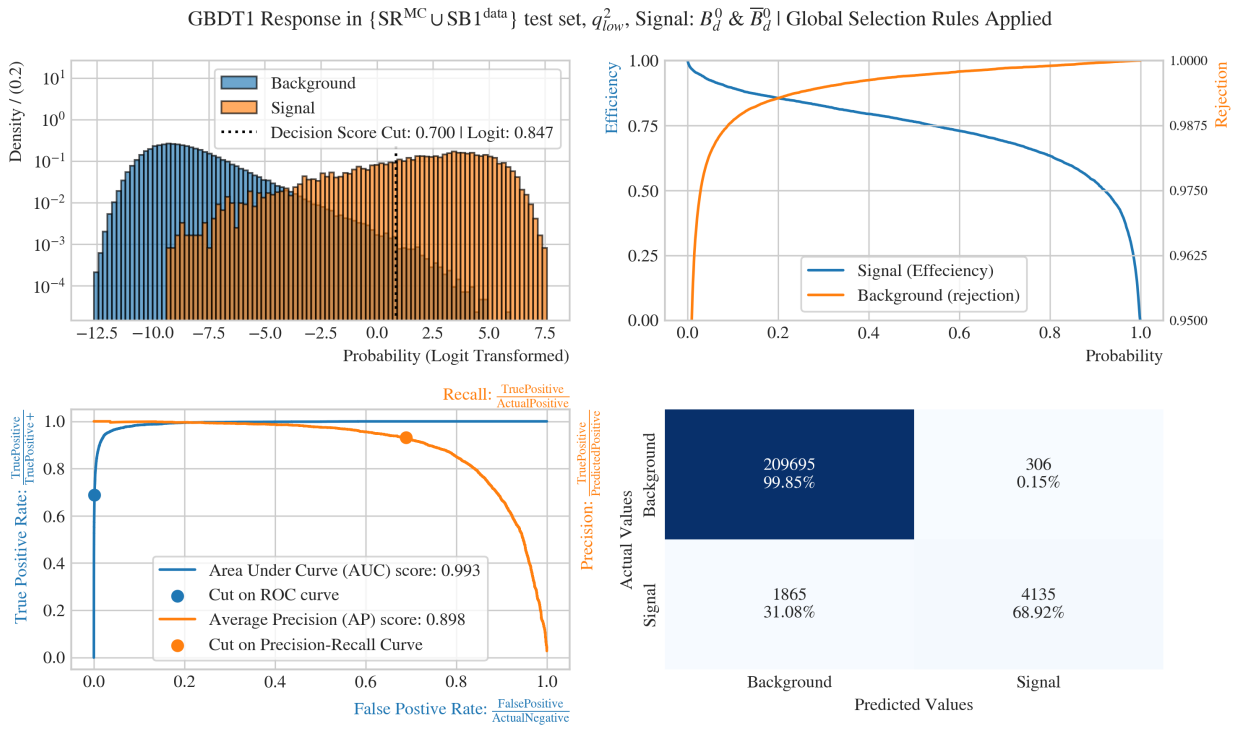


Figure A.10: *Signal vs. Background* Testing Suite on "2GNN to 3GBDT"-GBDT1 test-set with an overall good classification performance.

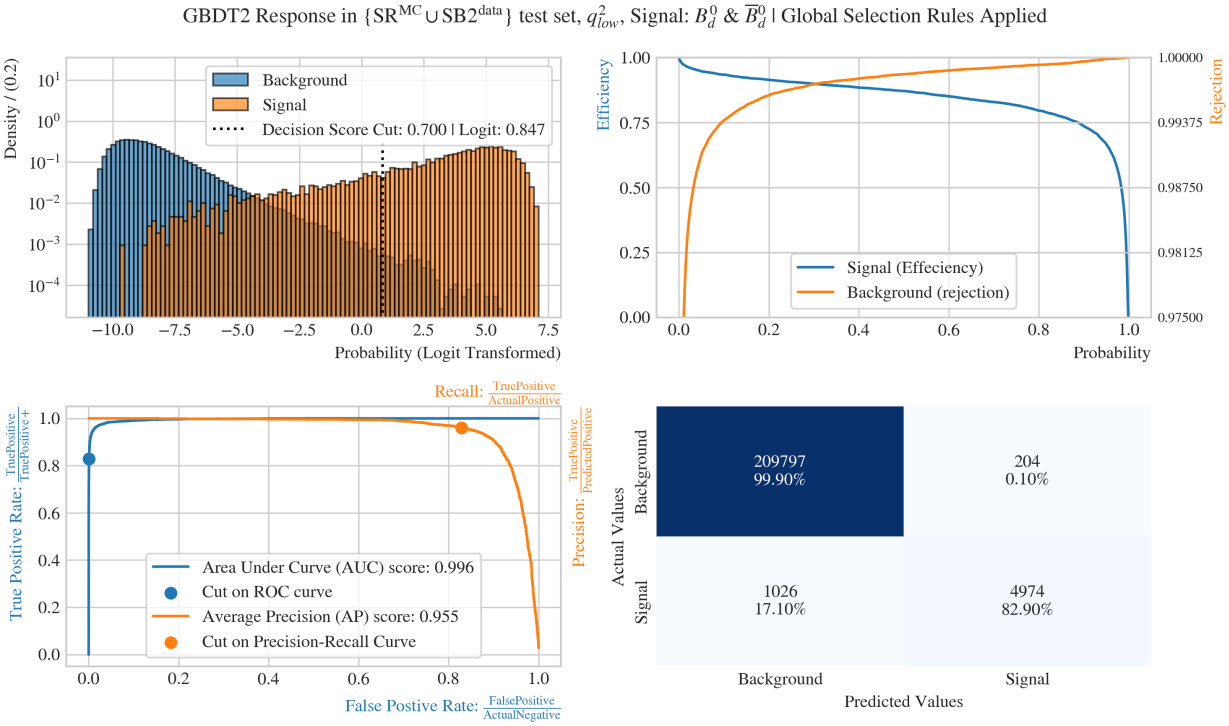


Figure A.11: *Signal vs. Background* Testing Suite on "2GNN to 3GBDT"-GBDT2 test-set with an overall good classification performance.

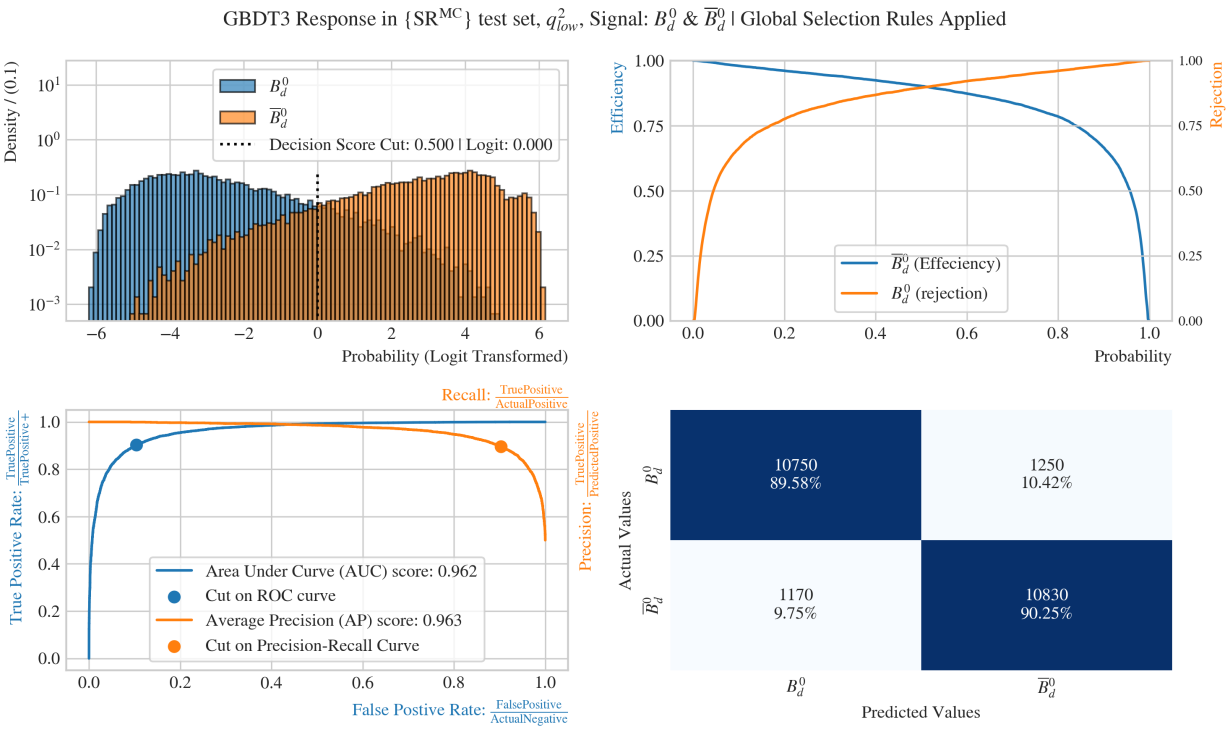


Figure A.12: *Signal vs. Background* Testing Suite on "2GNN to 3GBDT"-GBDT3 test-set with an overall good classification performance in separating B^0 from $\overline{B}^0$.

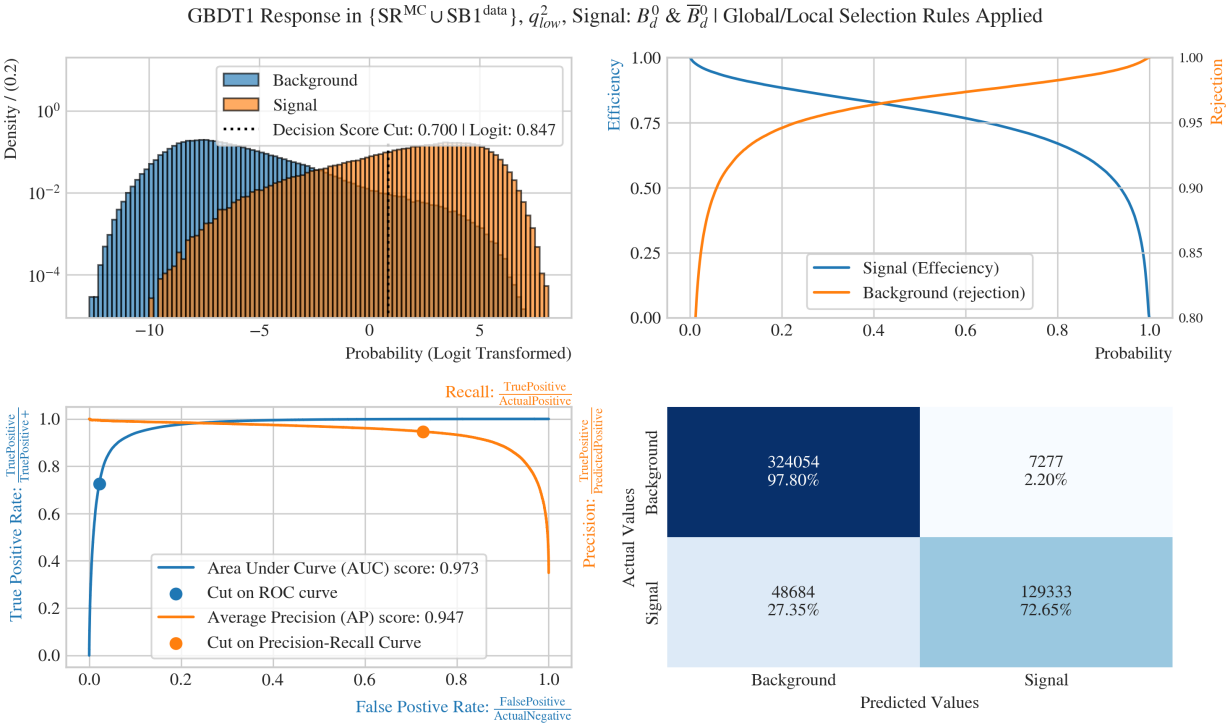


Figure A.13: *Signal vs. Background* Testing Suite with "2GNN to 3GBDT"-GBDT1 on non-train SR, SB1.

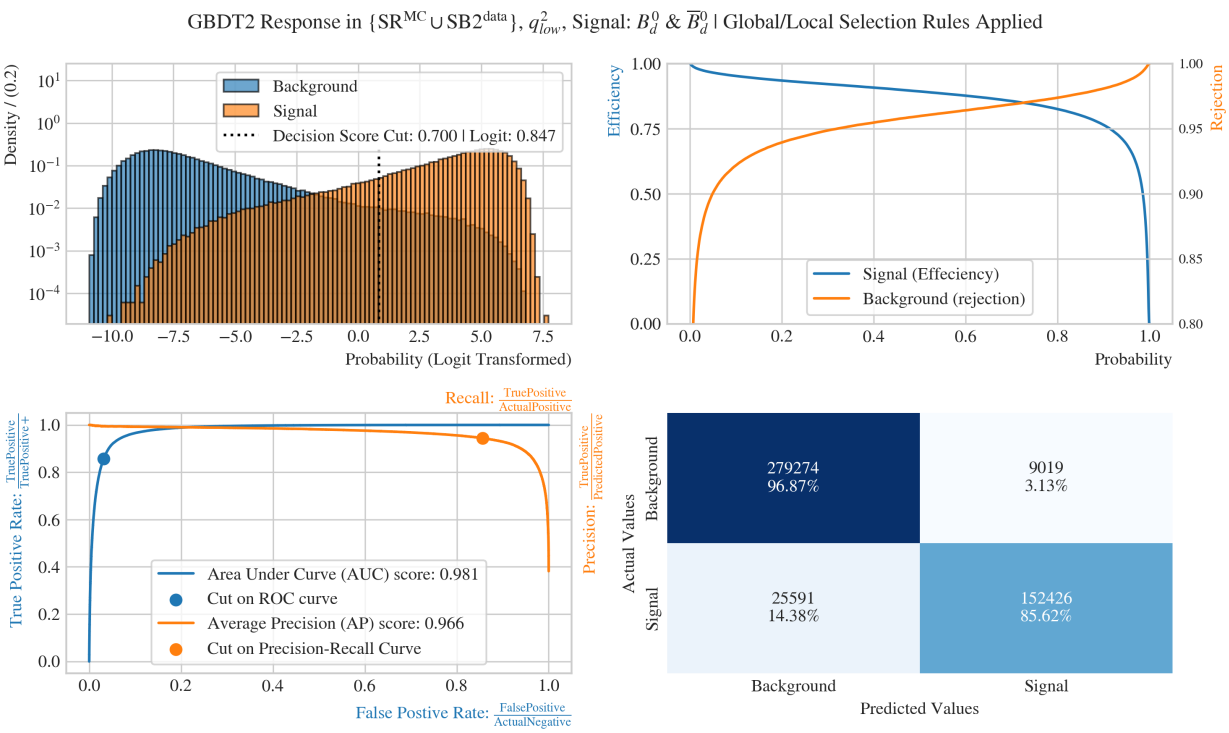


Figure A.14: *Signal vs. Background* Testing Suite with "2GNN to 3GBDT"-GBDT2 on non-train SR, SB2.

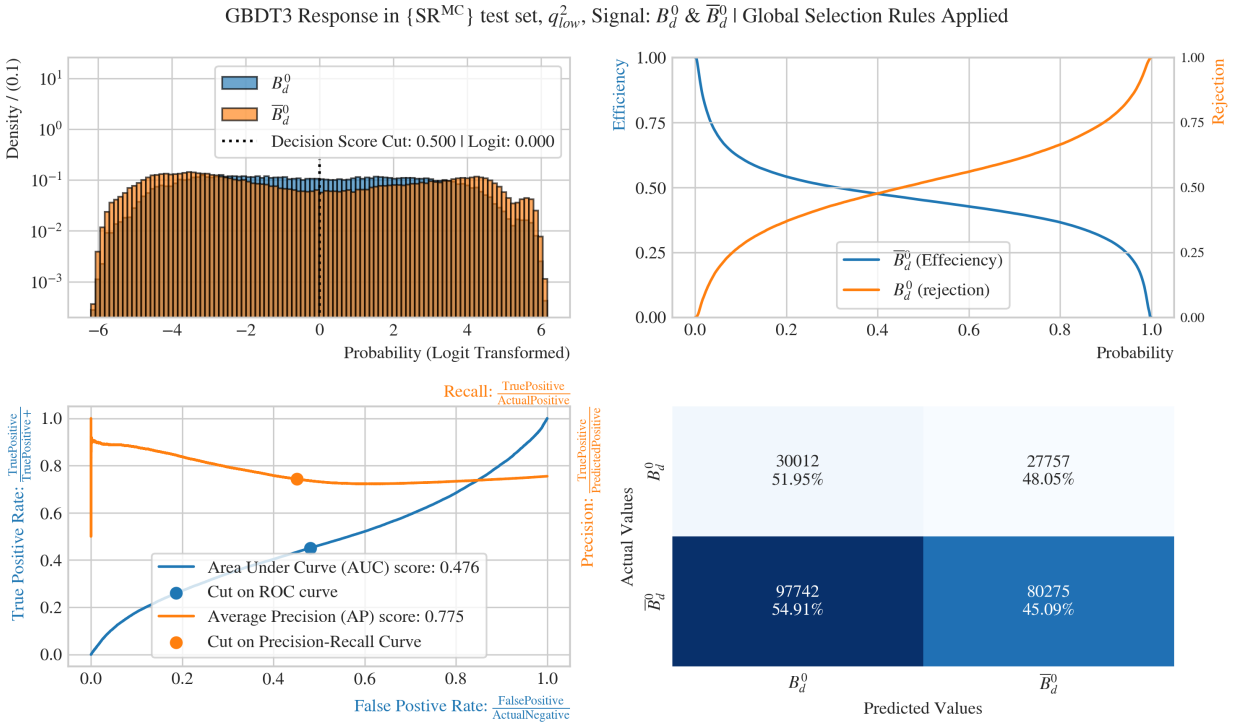


Figure A.15: *Signal vs. Background* Testing Suite with "2GNN to 3GBDT"-GBDT3 on non-train SR, SB2. This plot shows that the "2GNN to 3GBDT" is not viable, since GBDT3 can not distinguish between B^0 and $\overline{B}^0$. This is seen both in the plot and with $AUC < 0.5$.

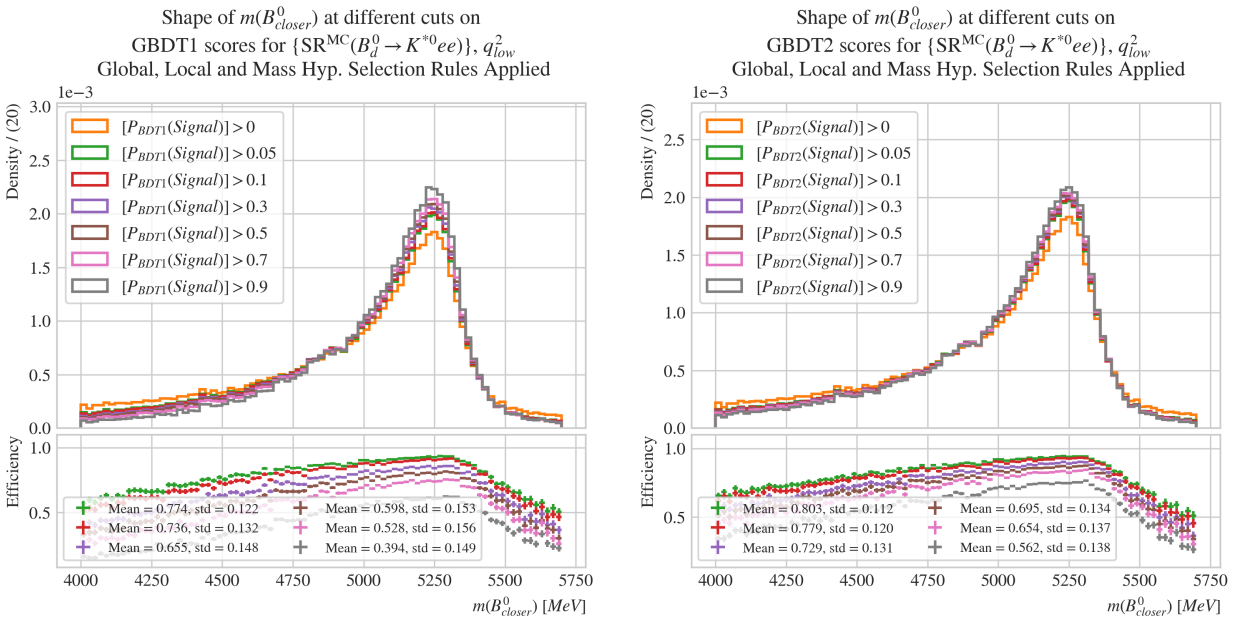


Figure A.16: *Mass Shape* Testing Suite for Signal on "2GNN to 3GBDT".

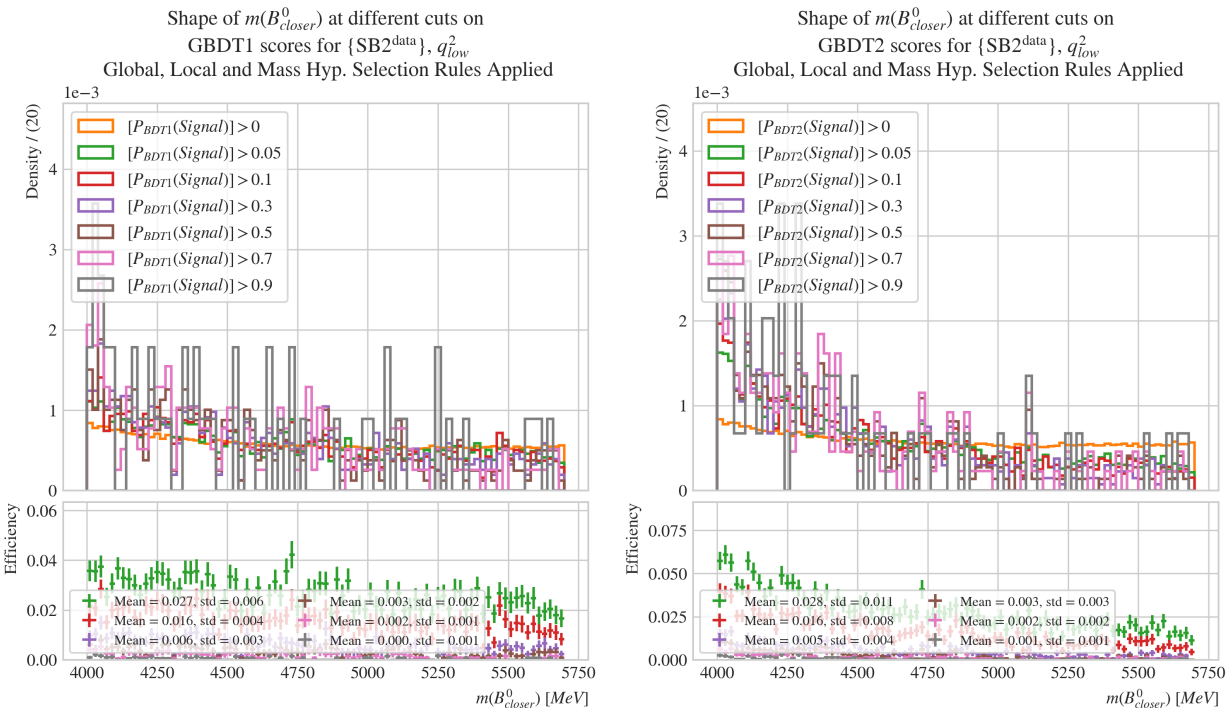


Figure A.17: *Mass Shape* Testing Suite for background on "2GNN to 3GBDT".

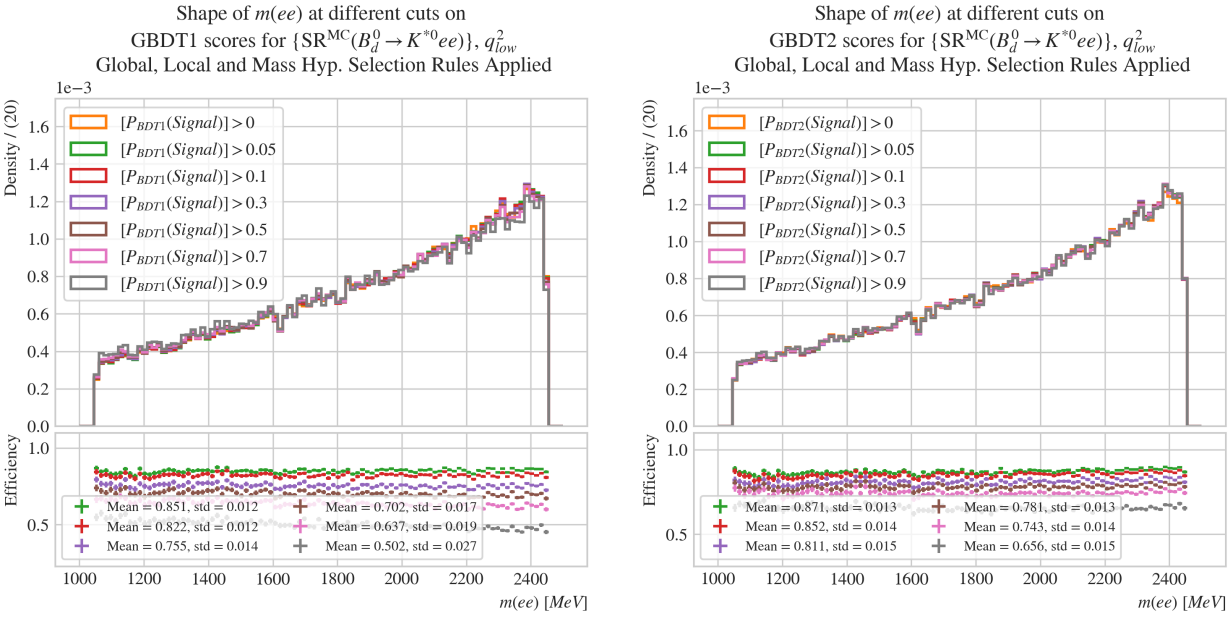


Figure A.18: *Mass Shape* Testing Suite for $m(ee)$ on "2GNN to 3GBDT".

A.4 2GNN to 2GBDT w. Enriched MC Background

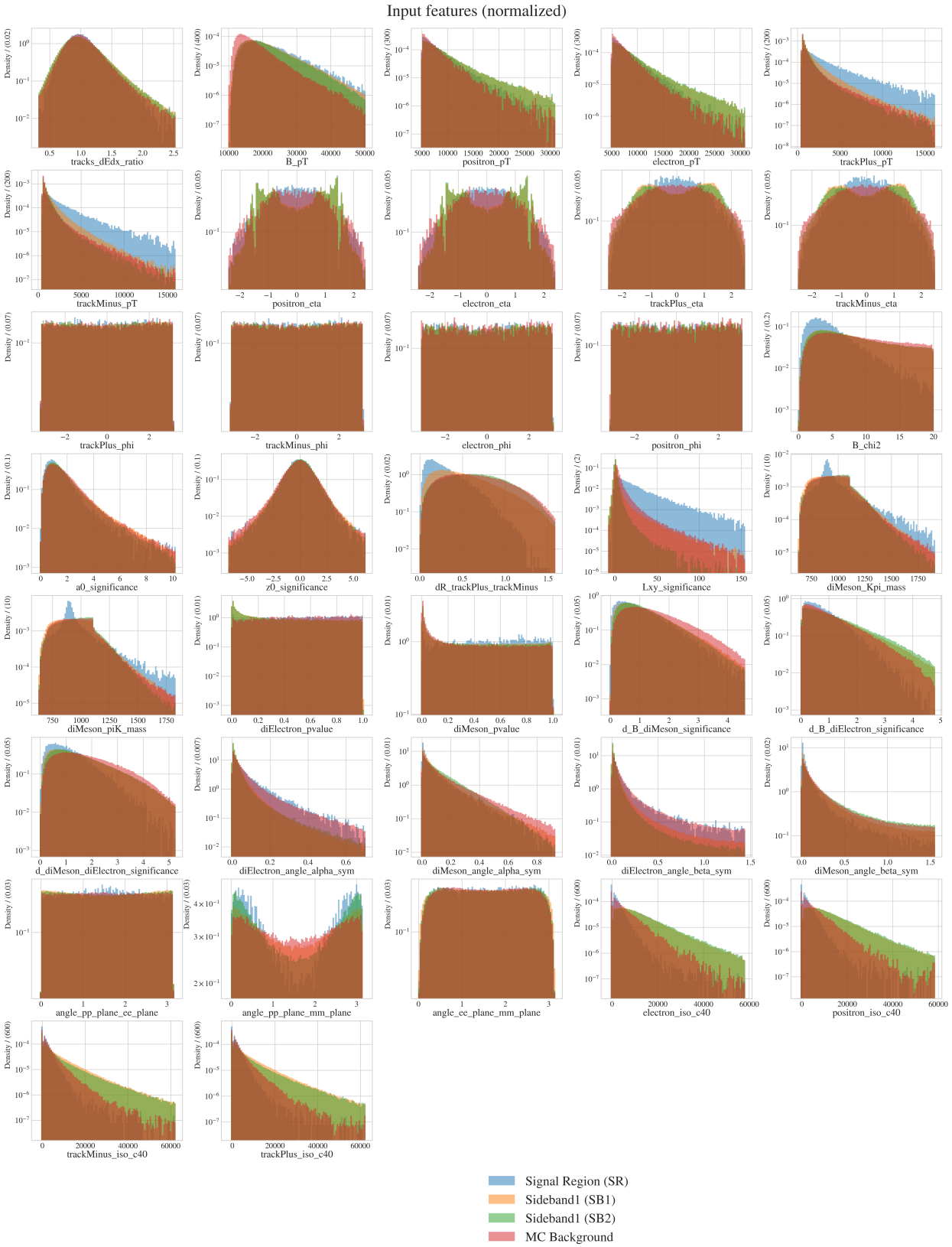


Figure A.19: Normalized features used in "Enriched 2GNN to 2GBDT". The plotted features are all separated into Signal Region SR^{MC}, $SB1^{data}$, $SB2^{data}$, and Bkg^{MC}}

	GBDT1	GBDT2
Training time	12min 25.1sec	14min 7.6sec
task	train	train
learning_rate	0.05	0.05
num_leaves	114	131
colsample_bytree	0.914813	0.894935
subsample	0.738661	0.599724
bagging_freq	1	1
max_depth	-1	-1
verbosity	-1	-1
reg_alpha	0.000002	0.00013
reg_lambda	1.504215	0.000842
min_split_gain	0.0	0.0
zero_as_missing	False	False
max_bin	255	255
min_data_in_bin	3	3
random_state	42	42
device_type	cpu	cpu
num_classes	3	3
objective	multiclass	multiclass
metric	multi_logloss	multi_logloss
num_threads	30	30
min_sum_hessian_in_leaf	9.931649	9.198197
n_estimators	215	185

Table A.3: "Enriched 2GNN to 3GBDT" best values for the LightGBM models.

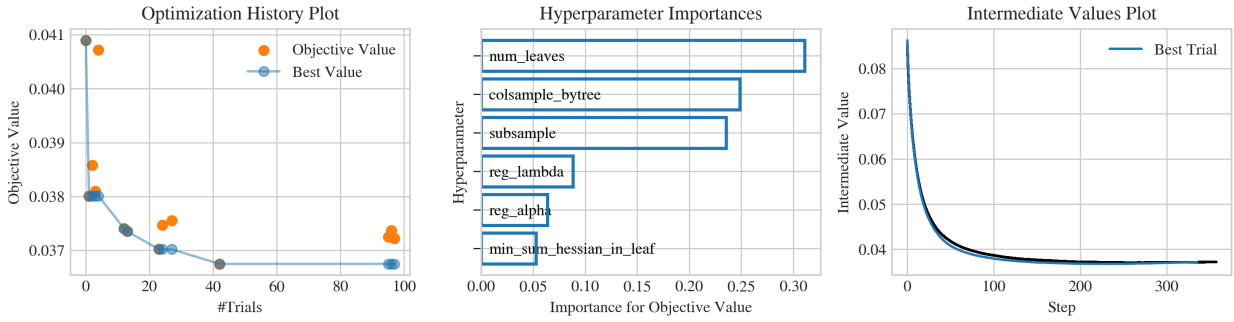


Figure A.20: *LightGBM* Testing Suite on "Enriched 2GNN to 2GBDT"-GBDT1. Showing that GBDT1s training has converged.

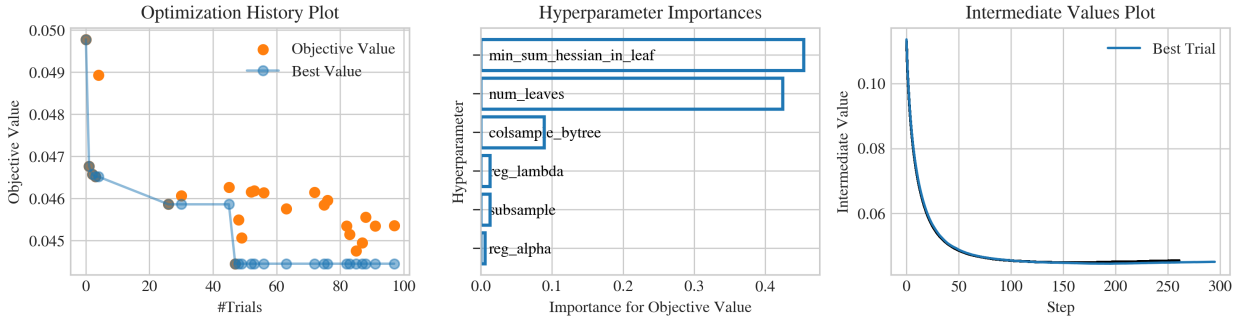


Figure A.21: *LightGBM* Testing Suite on "Enriched 2GNN to 2GBDT"-GBDT2. Showing that GBDT2s training has converged.

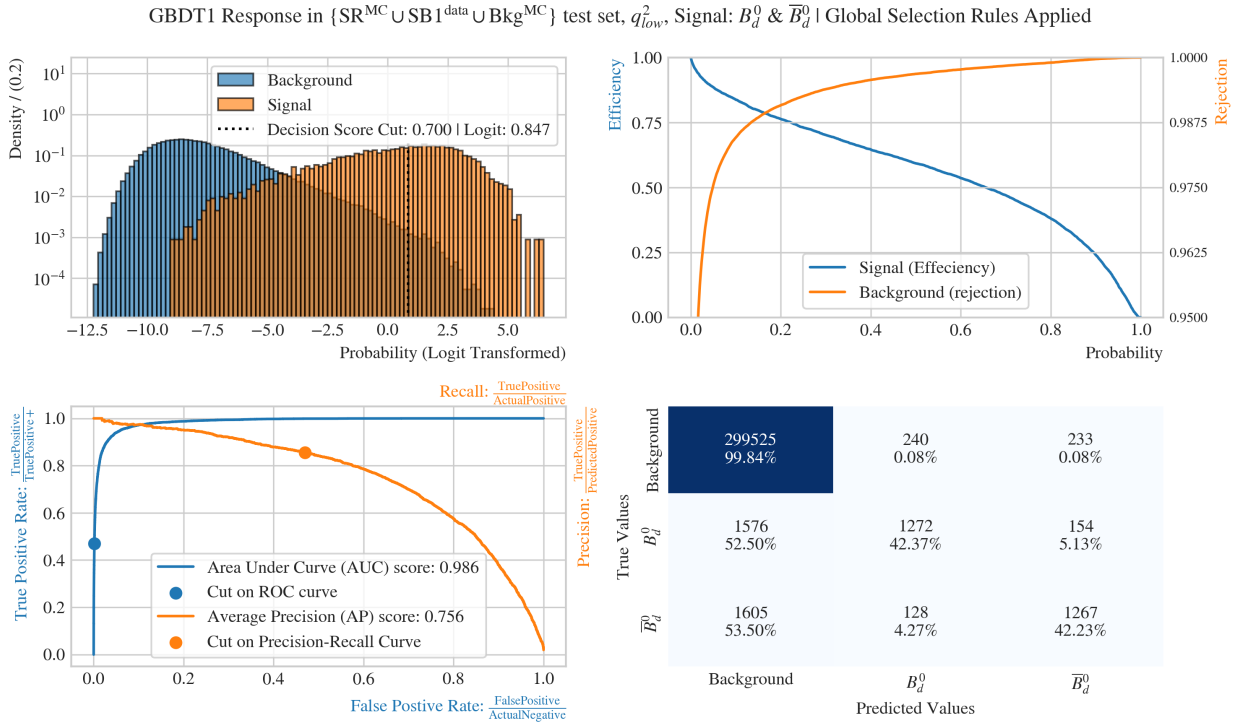


Figure A.22: *Signal vs. Background* Testing Suite on "Enriched 2GNN to 2GBDT"-GBDT1 test-set with suboptimal signal efficiency.

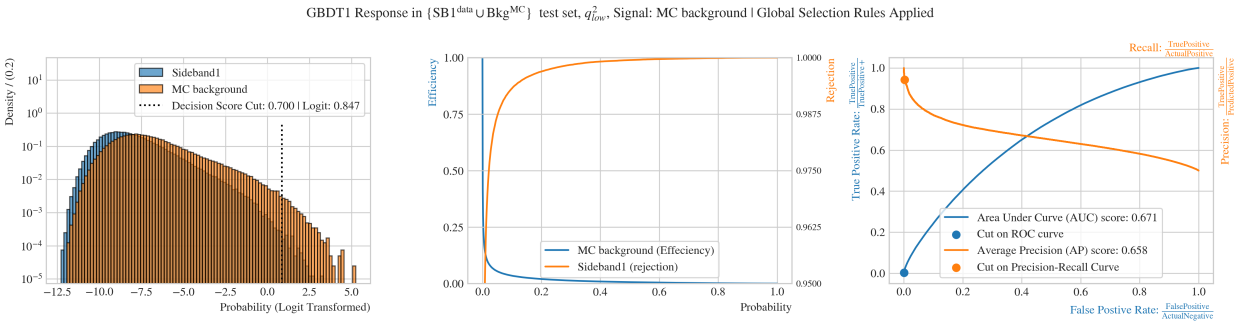


Figure A.23: Modified *Signal vs. Background* Testing Suite (MC vs SB1) on "Enriched 2GNN to 2GBDT"-GBDT1. A good indication that the GBDT2 model does not see the difference in SB1$^{\text{data}}$} and Bkg$^{\text{MC}}$} is seen in the AUC score.

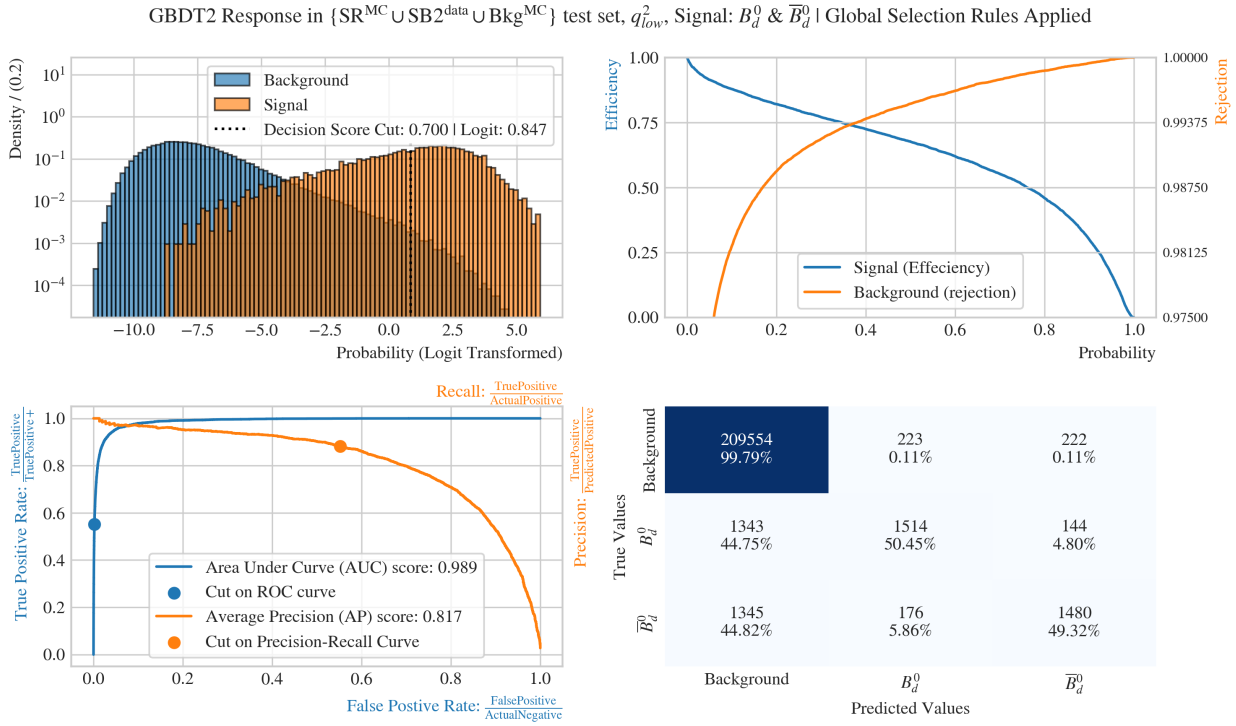


Figure A.24: *Signal vs. Background* Testing Suite on "Enriched 2GNN to 2GBDT"-GBDT2 test-set with suboptimal signal efficiency.

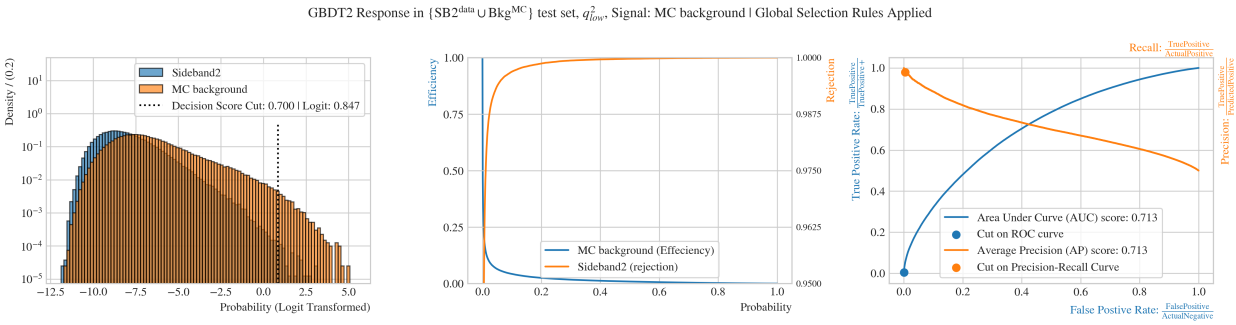


Figure A.25: Modified *Signal vs. Background* Testing Suite (MC vs SB2) on "Enriched 2GNN to 2GBDT"-GBDT2. A good indication that the GBDT2 model does not see difference inSB2$^{\text{data}}$} and Bkg$^{\text{MC}}$} is seen in the AUC score.

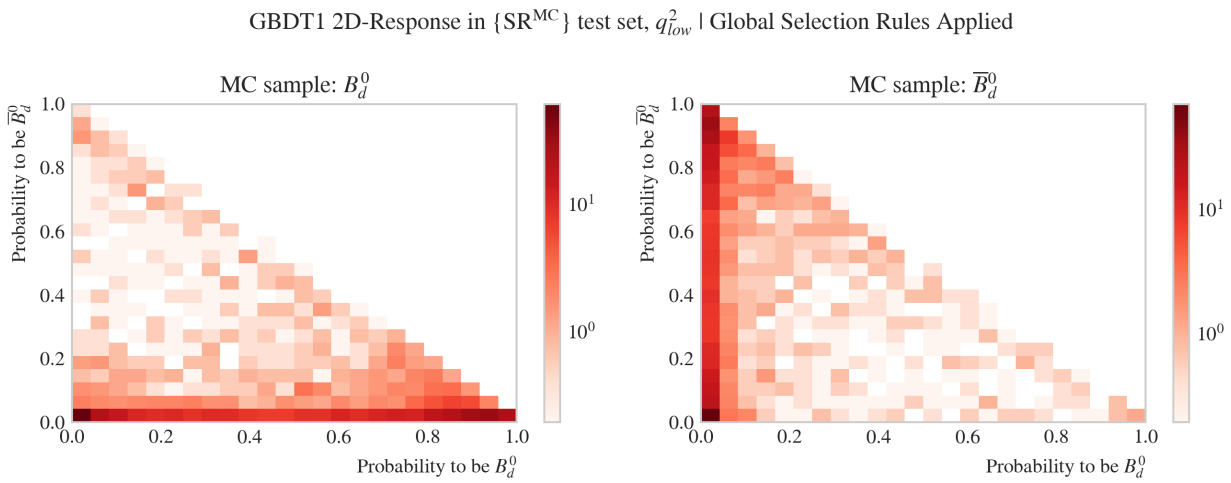


Figure A.26: *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite on "Enriched 2GNN to 2GBDT"-GBDT1 test-set with density on log-scale.

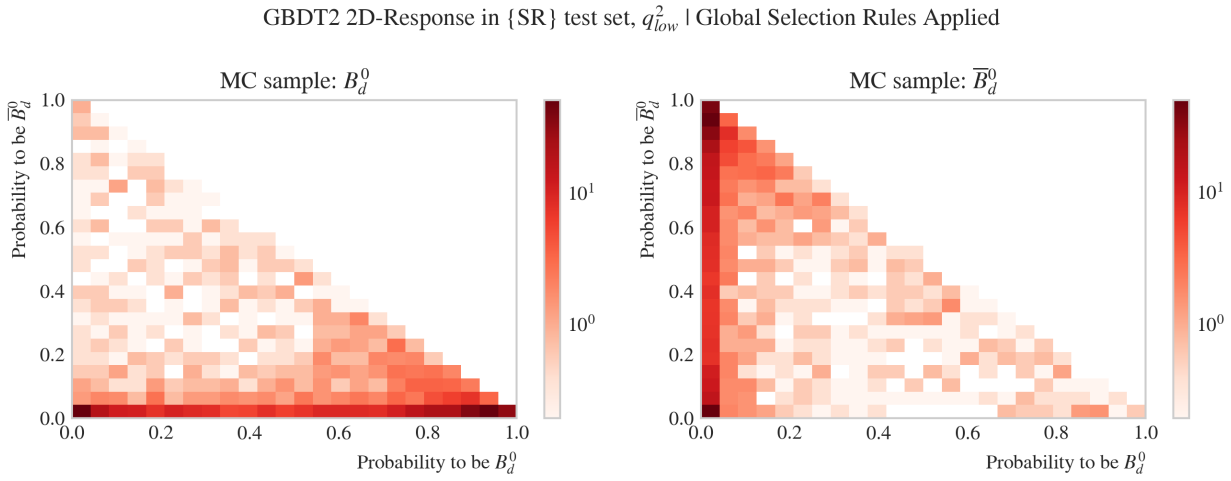


Figure A.27: *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite on "Enriched 2GNN to 2GBDT"-GBDT2 test-set with density on log-scale.

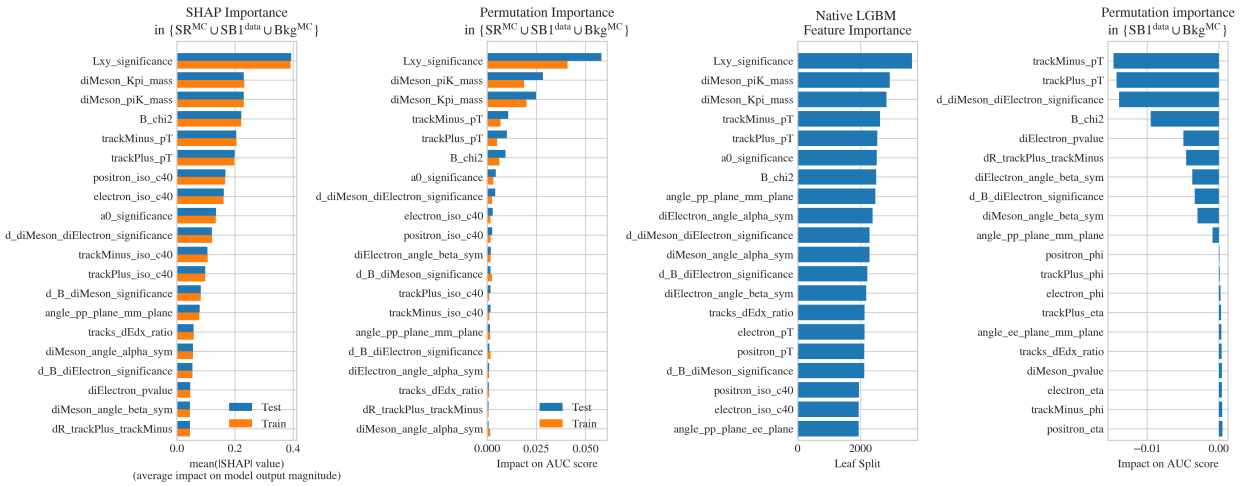


Figure A.28: 20 of the highest scoring feature importances on "Enriched 2GNN to 2GBDT"-GBDT1 for both train- and test-set. The rightmost subfigure: Permutation importance on separating SB1 and MC background. The worse the AUC score, the better feature.

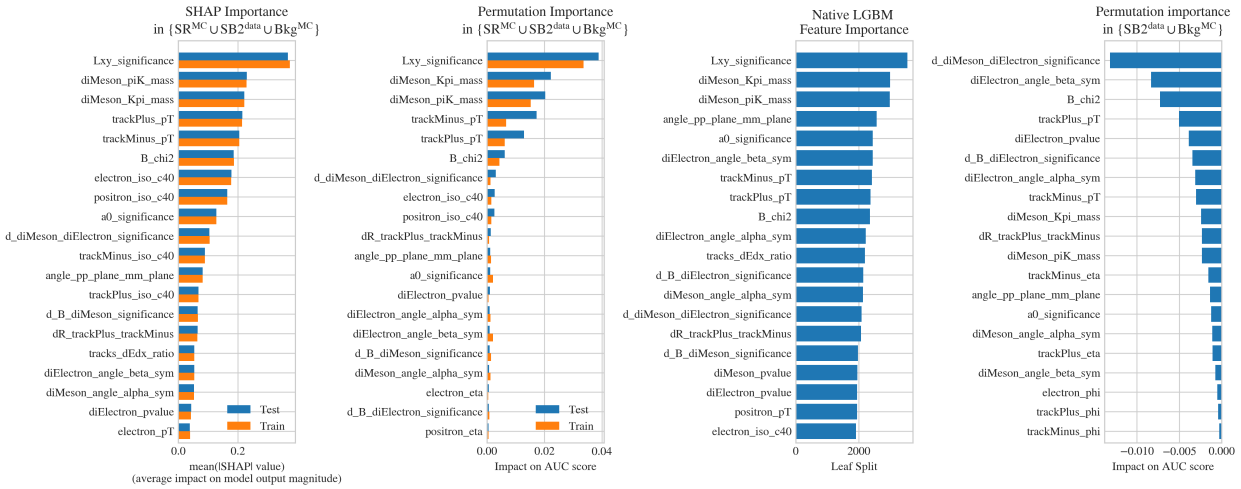


Figure A.29: 20 of the highest scoring feature importances on "Enriched 2GNN to 2GBDT"-GBDT1 for both train- and test-set. The rightmost subfigure: Permutation importance on separating SB1 and MC background. The worse the AUC score, the better feature.

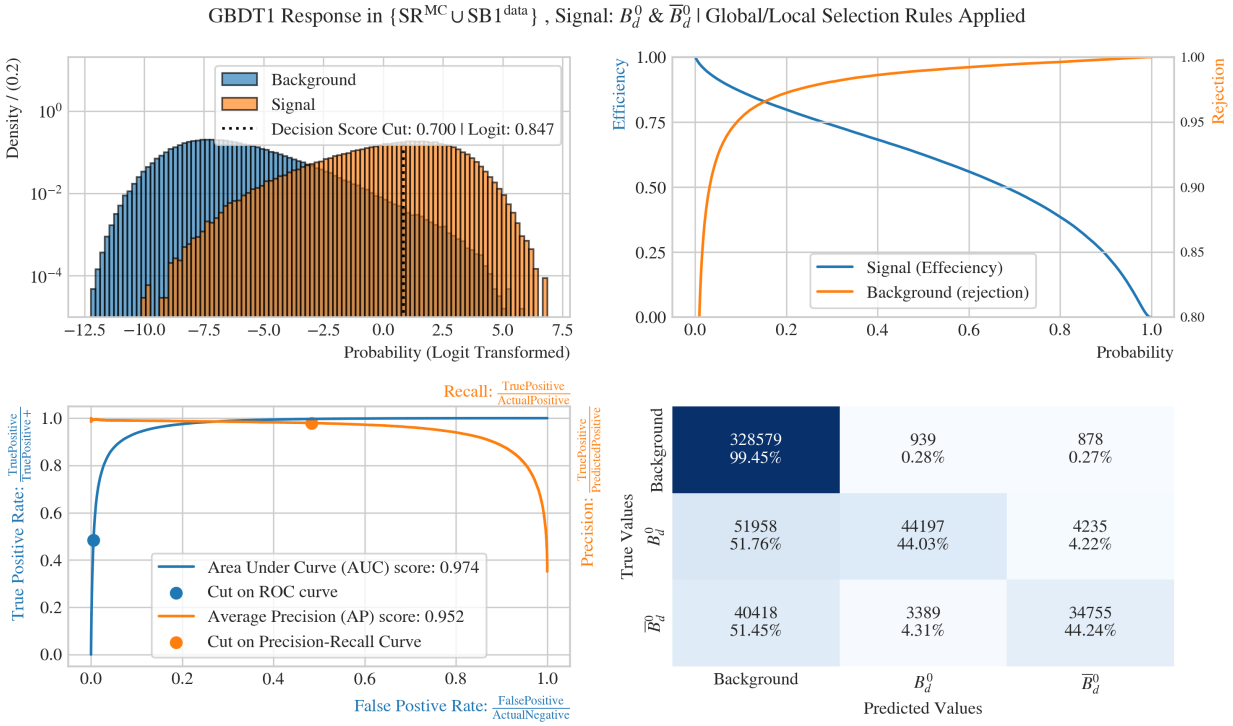


Figure A.30: *Signal vs Background* Testing Suite with "Enriched 2GNN to 2GBDT"-GBDT1 on non-train $\{SR^{MC} \cup SB1^{data} \cup Bkg^{MC}\}$ with suboptimal signal efficiency.

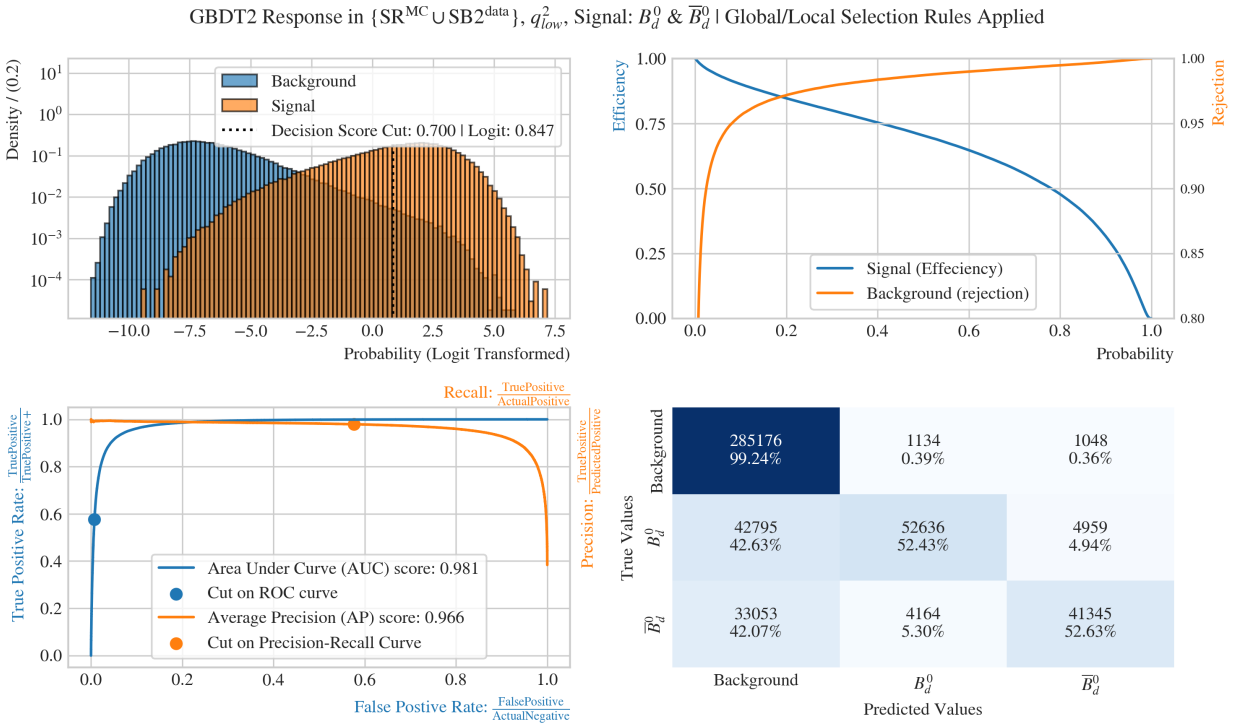


Figure A.31: *Signal vs Background* Testing Suite with "Enriched 2GNN to 2GBDT"-GBDT2 on non-train $\{SR^{MC} \cup SB2^{data} \cup Bkg^{MC}\}$ with suboptimal signal efficiency.

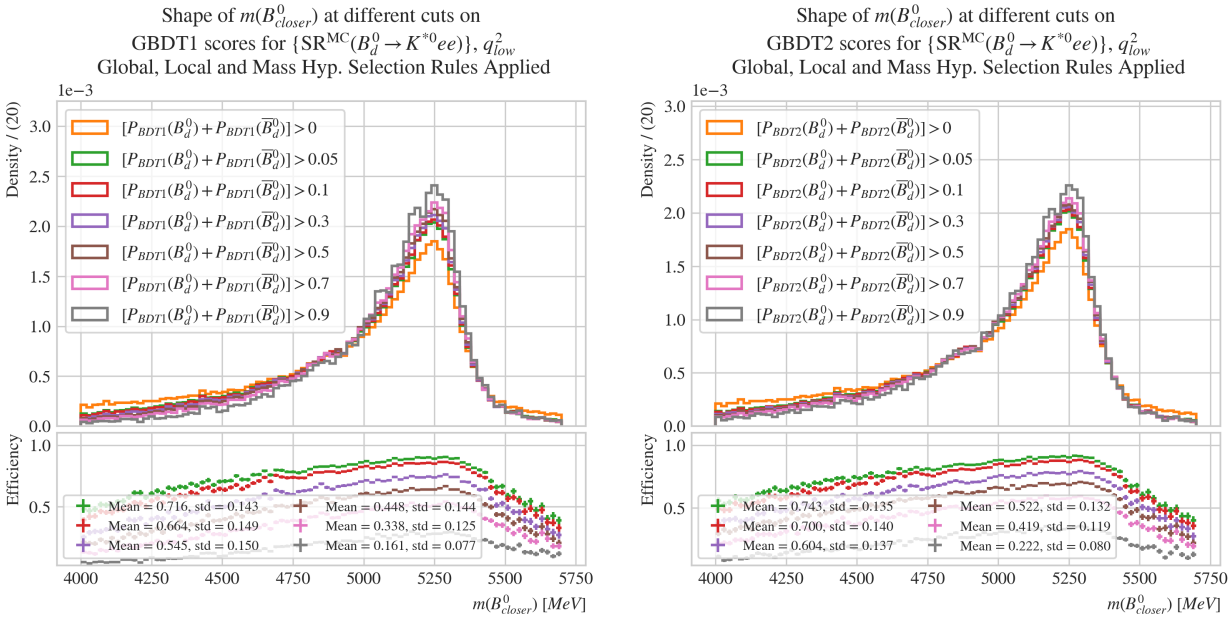


Figure A.32: *Mass Shape* Testing Suite for Signal on "Enriched 2GNN to 2GBDT".

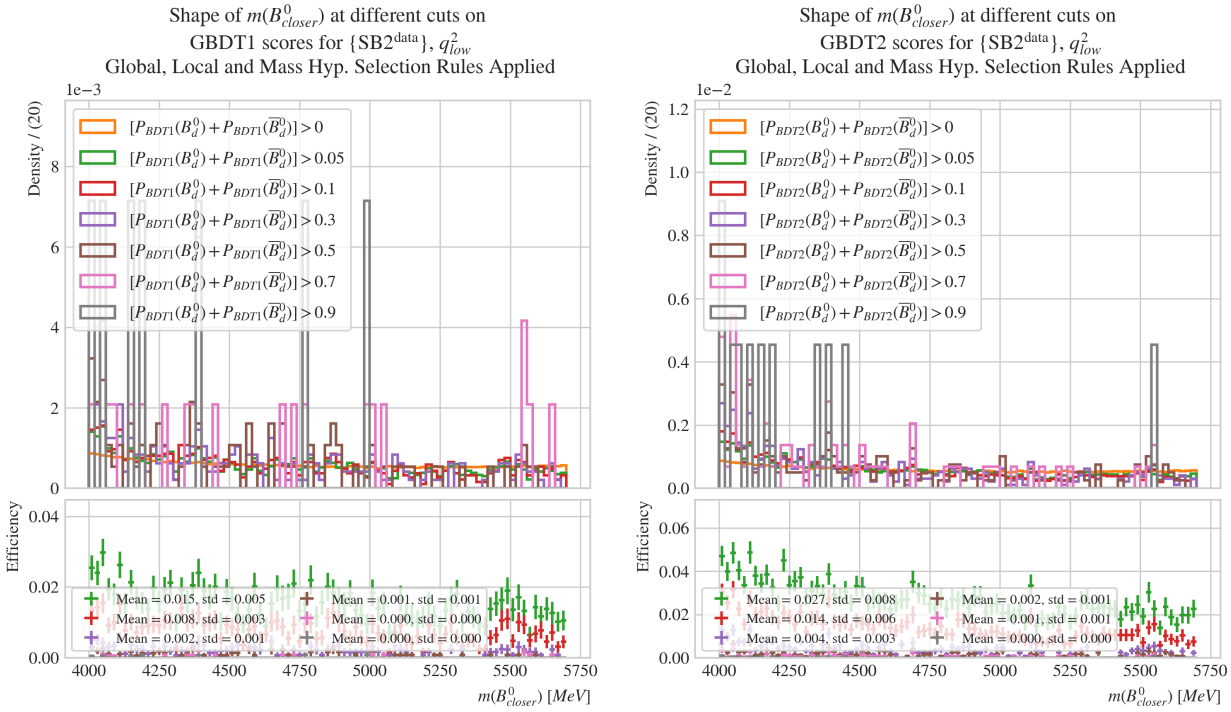


Figure A.33: *Mass Shape* Testing Suite for background on "Enriched 2GNN to 2GBDT".

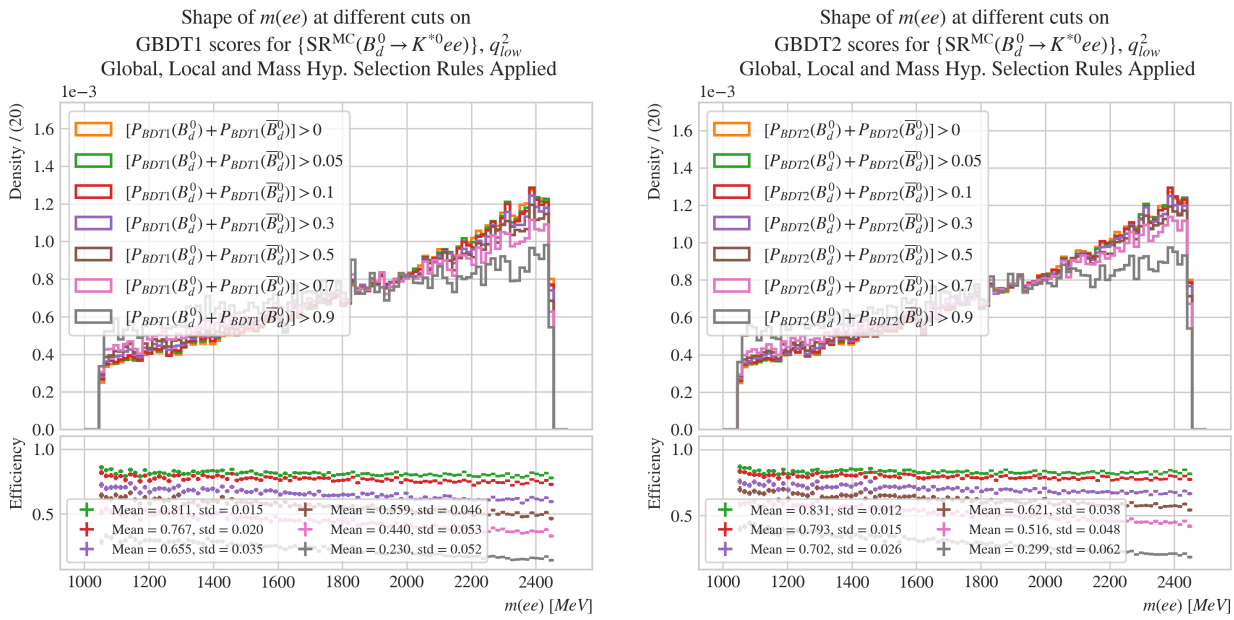


Figure A.34: *Mass Shape* Testing Suite for $m(ee)$ on "Enriched 2GNN to 2GBDT". The sculpting in $m(ee)$ is present or high cuts in the GBDTs (see grey lines in top and bottom subfigure).

A.5 $m(B^0_{closer})$-correlation

	GBDT(SR)	GBDT(SB1)	GBDT(SB2)
Training Time	29min 41sec	1h 38min 48sec	38min 42sec
RMSE	31.098	30.970	23.340
	95%CI [31.112, 31.085]	95%CI [30.983, 30.956]	95%CI [23.350, 23.330]
learning_rate	0.03	0.03	0.03
num_leaves	66	255	66
colsample_bytree	0.916221	0.65591	0.916221
subsample	0.590912	0.954889	0.590912
verbosity	-1	-1	-1
random_state	42	42	42
device_type	cpu	cpu	cpu
objective	regression	regression	regression
metric	rmse	rmse	rmse
num_threads	25	25	25
reg_alpha	0.000005	0.000001	0.000005
min_sum_hessian_in_leaf	0.005415	1.04371	0.005415
reg_lambda	0.000528	0.0	0.000528
n_estimators	10000	10000	10000

Table A.4: "$m(B^0_{closer})$-correlation" best values for the LightGBM models.

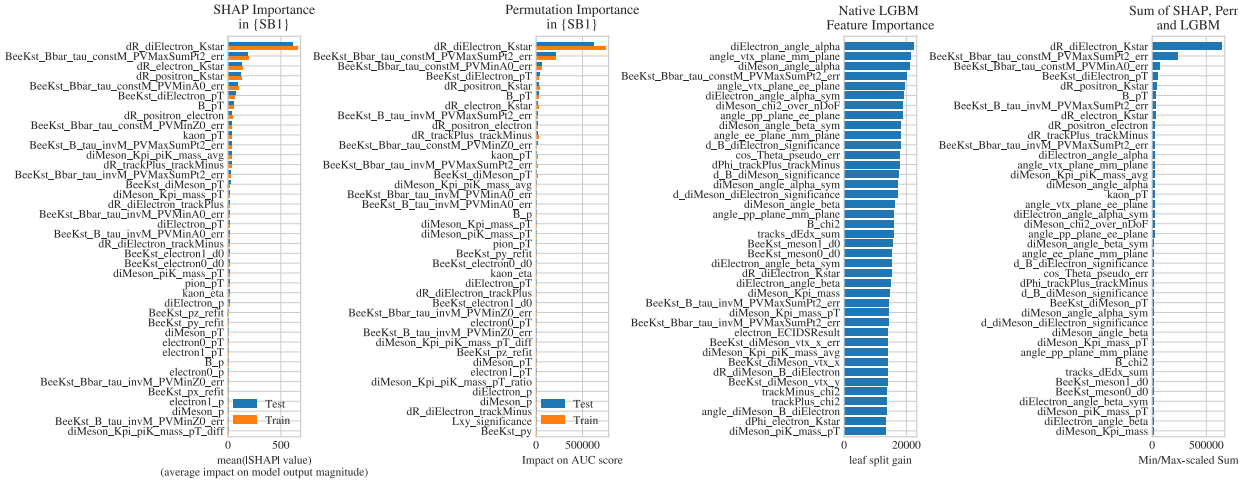


Figure A.35: Feature Importance for GBDT-regressor model on SR against $m(B^0_{closer})$

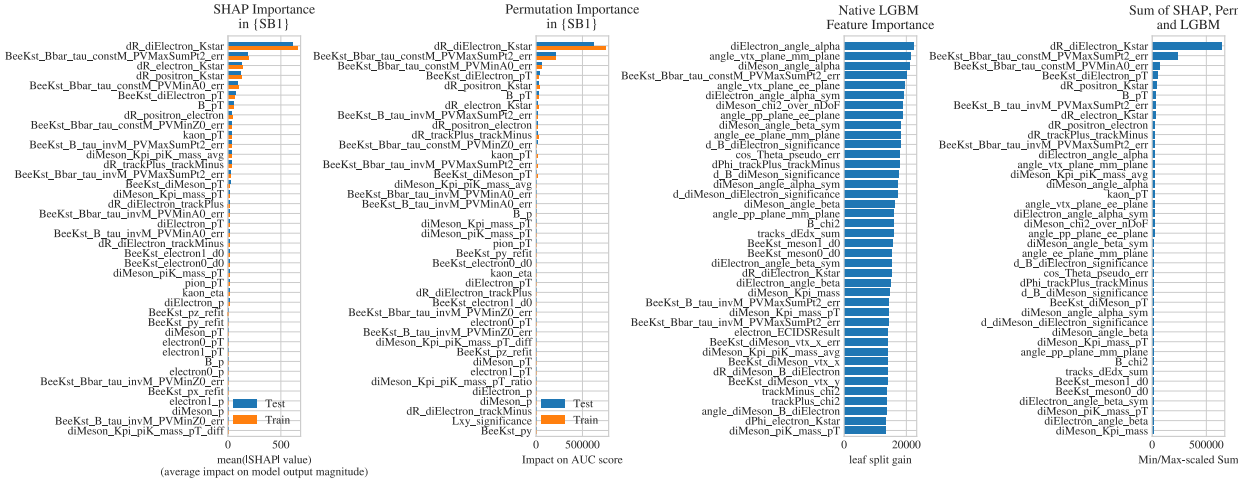


Figure A.36: Feature Importance for GBDT-regressor model on SB1 against $m(B^0_{closer})$

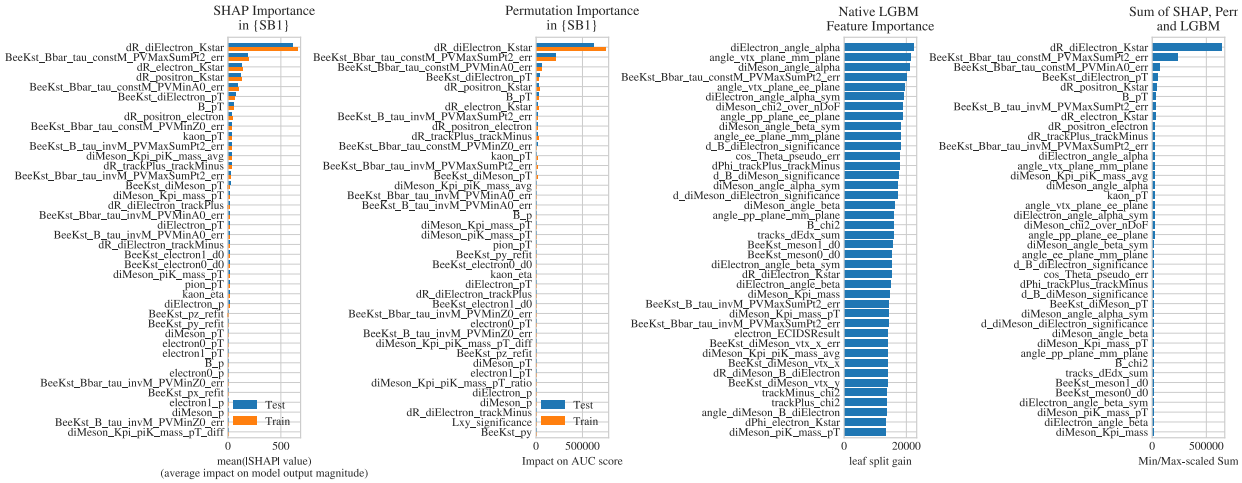


Figure A.37: Feature Importance for GBDT-regressor model on SB2 against $m(B^0_{closer})$

A.6 Full n-tuple Feature Search

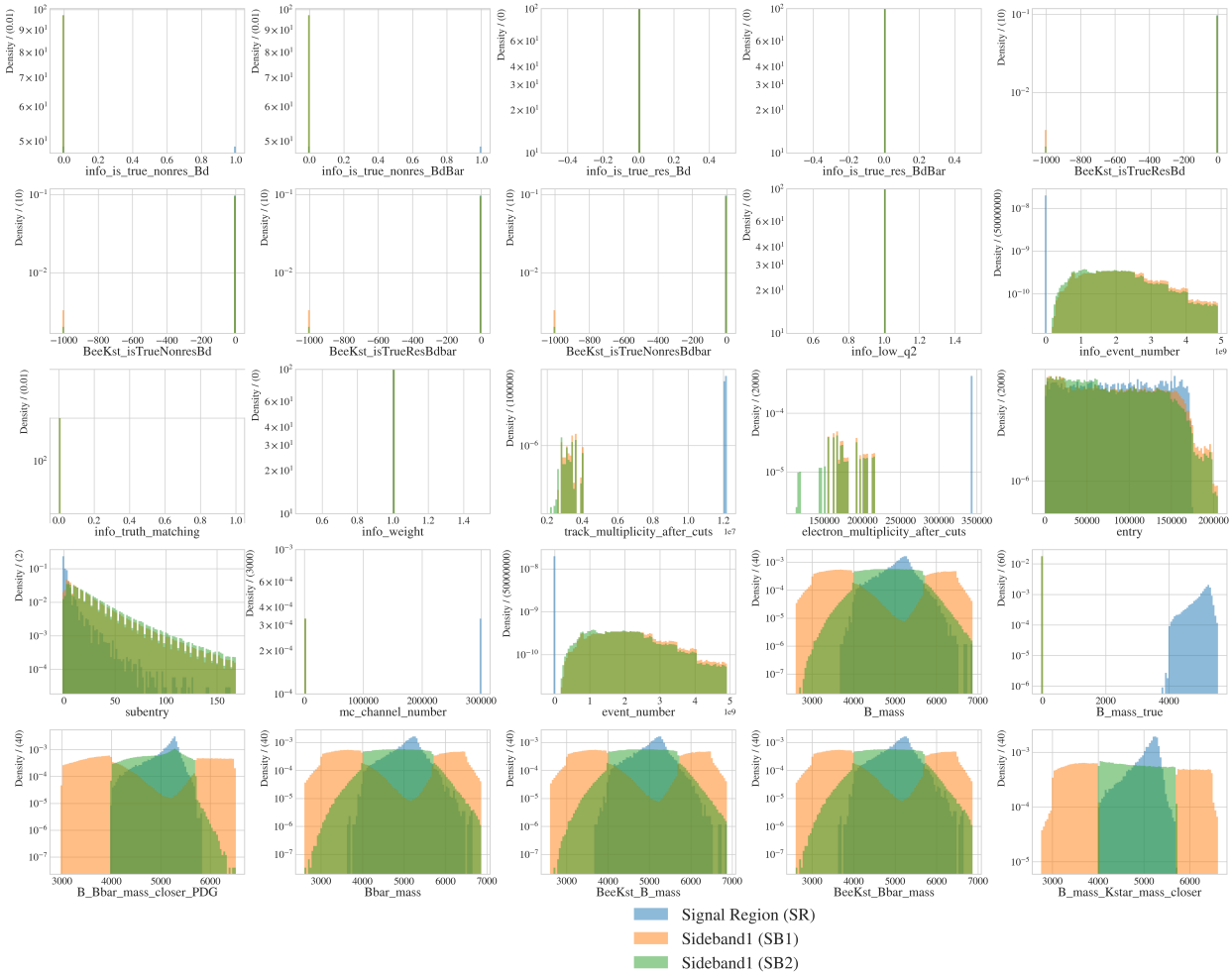


Figure A.38: Features removed from iteration 1 of the full n-tuple feature search.

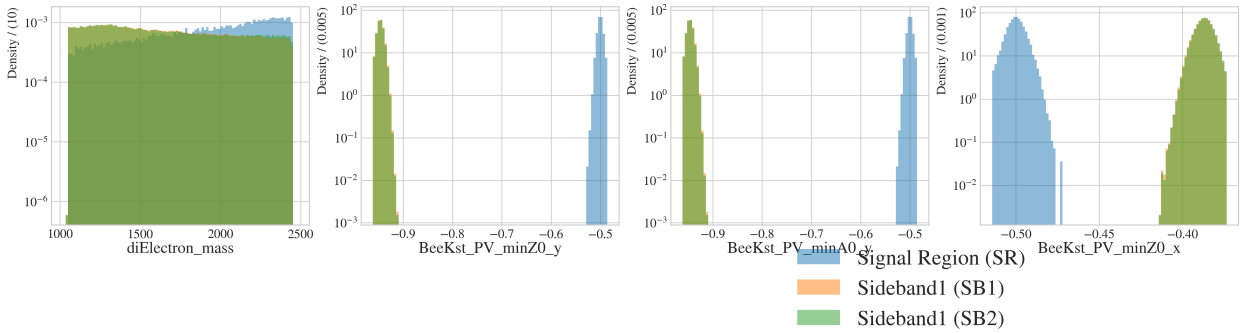


Figure A.39: Features removed from iteration 2 of the full n-tuple feature search.

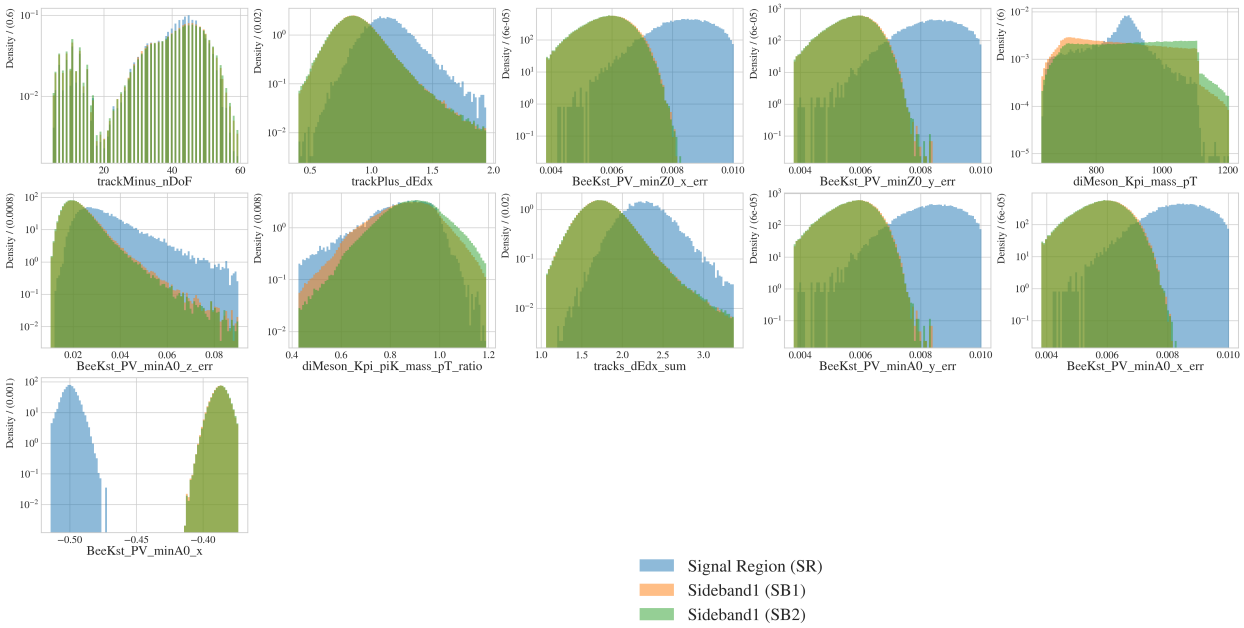


Figure A.40: Features removed from iteration 3 of the full n-tuple feature search.

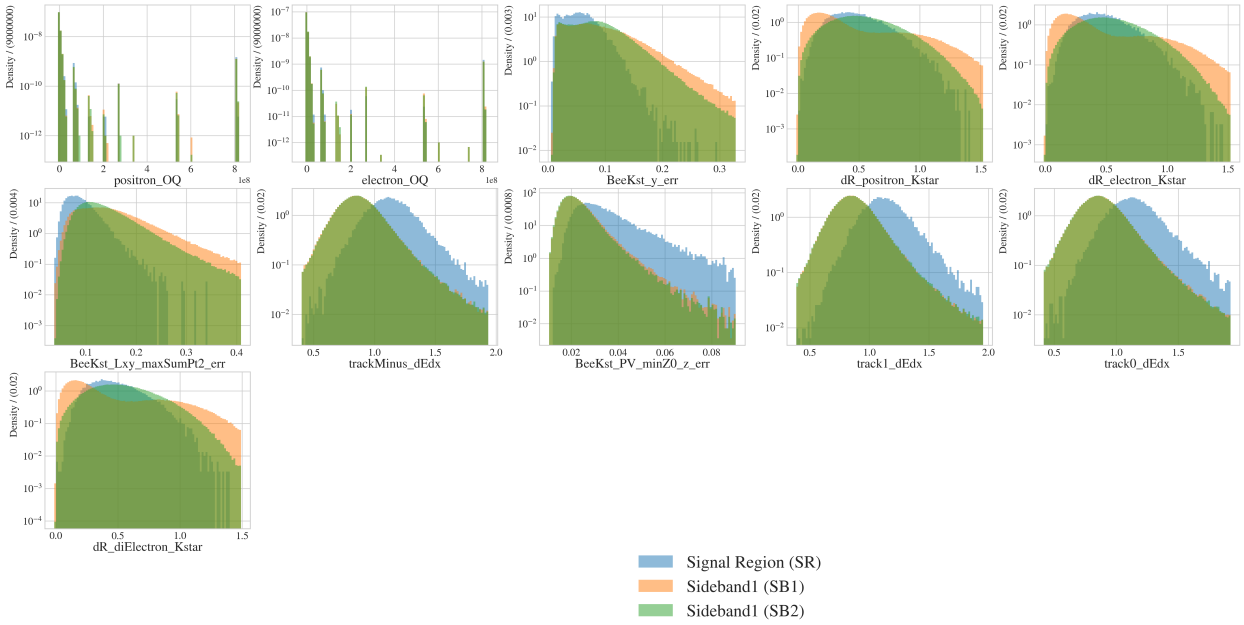


Figure A.41: Features removed from iteration 4 of the full n-tuple feature search.

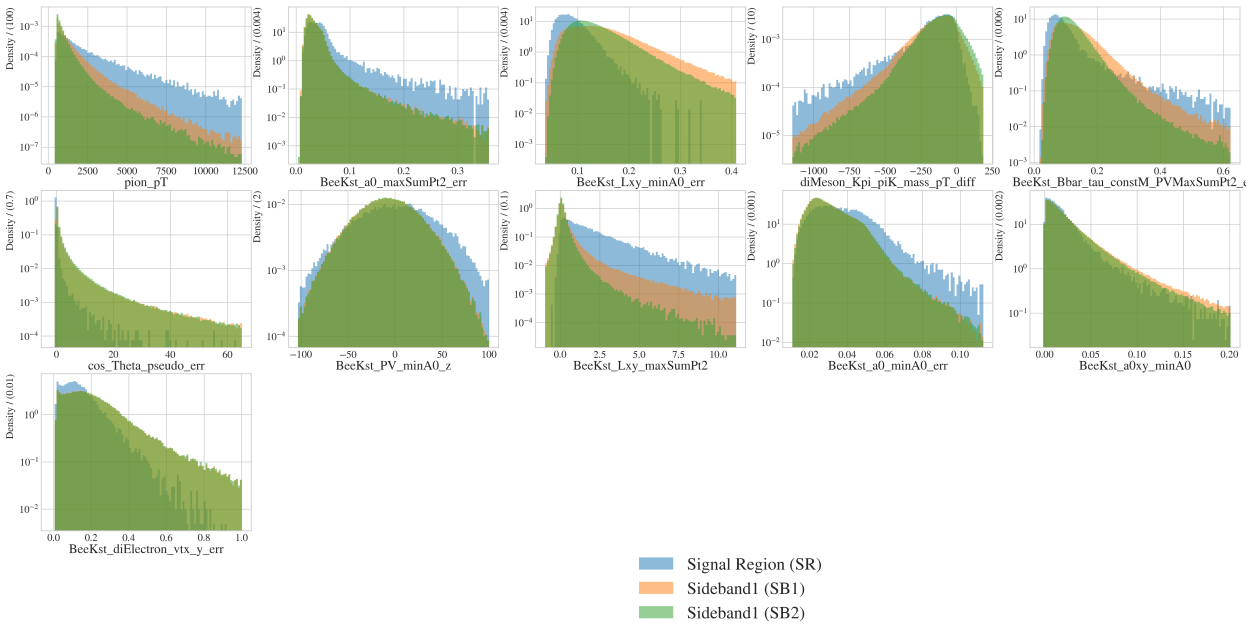


Figure A.42: Features removed from iteration 5 of the full n-tuple feature search.

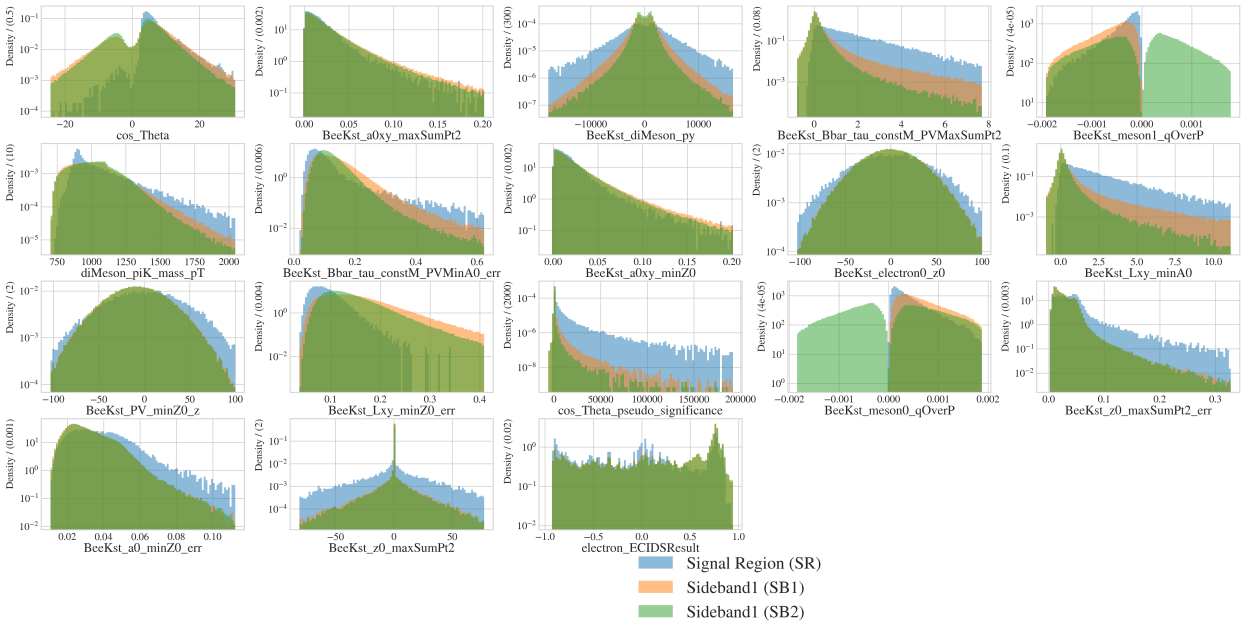


Figure A.43: Features removed from iteration 6 of the full n-tuple feature search.

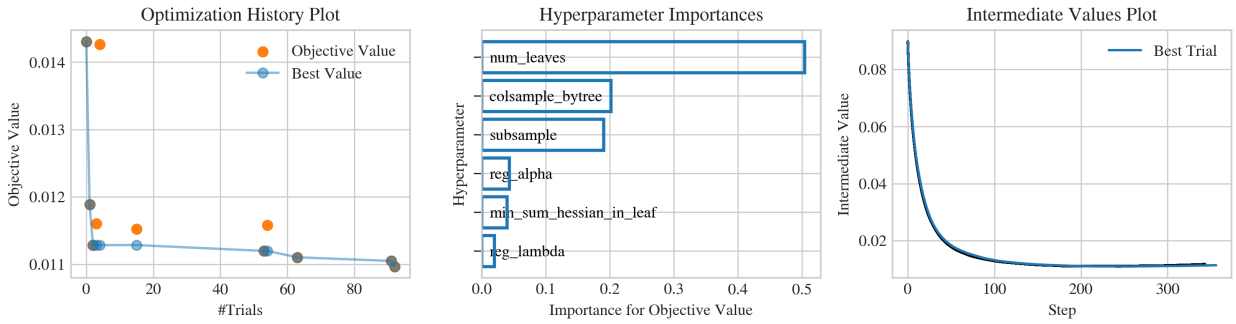


Figure A.44: *LightGBM* Testing Suite on "2GNN to 2GBDT full search"-GBDT1 test set for iteration 6 of the full n-tuple feature search. Showing that GBDT1s training has converged.

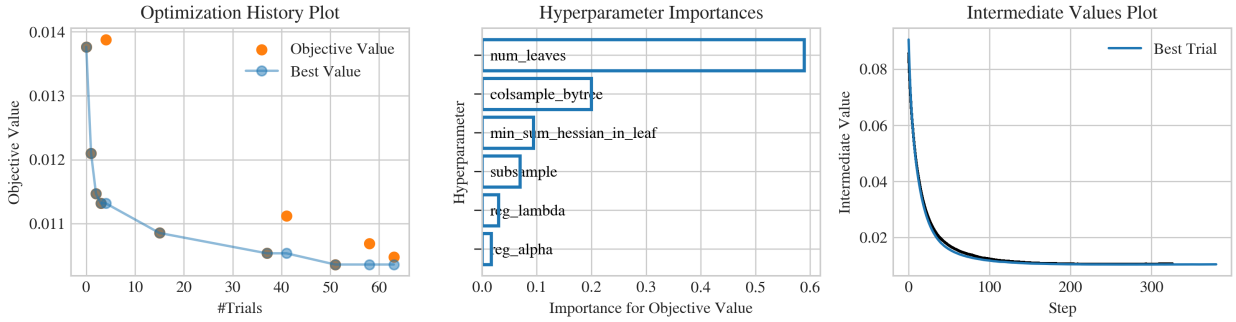


Figure A.45: *LightGBM* Testing Suite on "2GNN to 2GBDT full search"-GBDT2 test-set for iteration 6 of the full n-tuple feature search. Showing that GBDT2s training has converged.

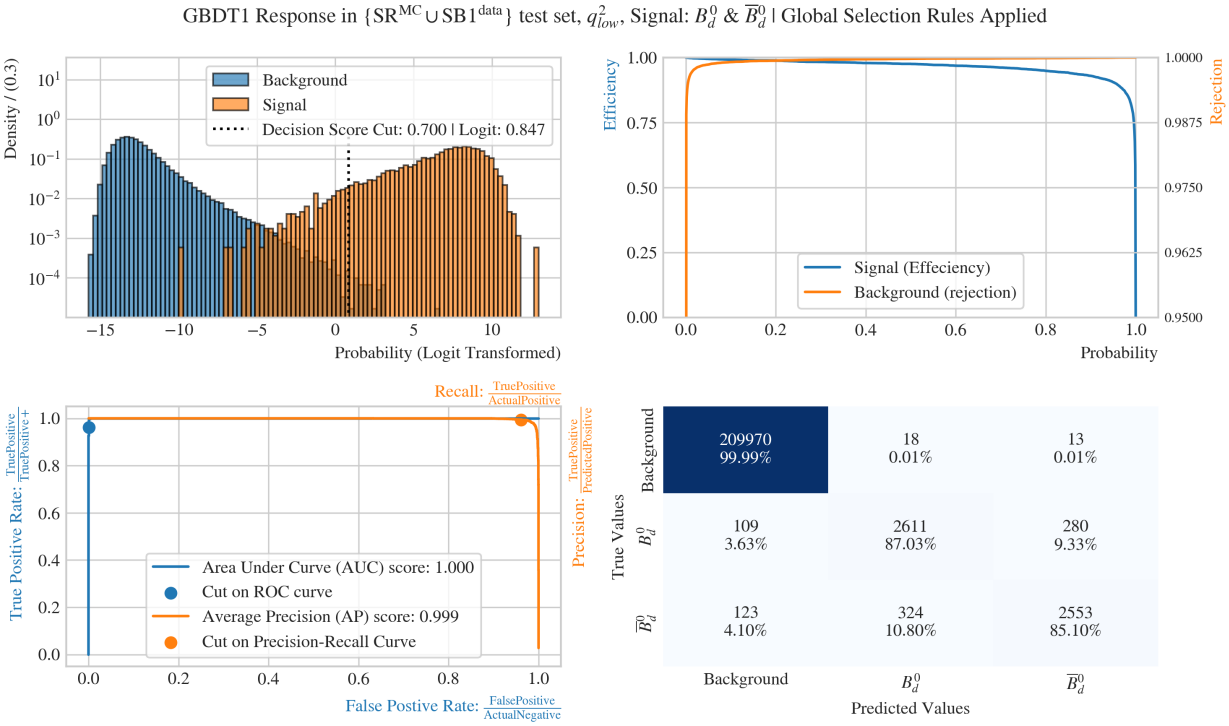


Figure A.46: *Signal vs. Background* Testing Suite on "2GNN to 2GBDT full search"-GBDT1 test-set for iteration 6 of the full n-tuple feature search. The AUC and AP score shows that there are still some leaking features.

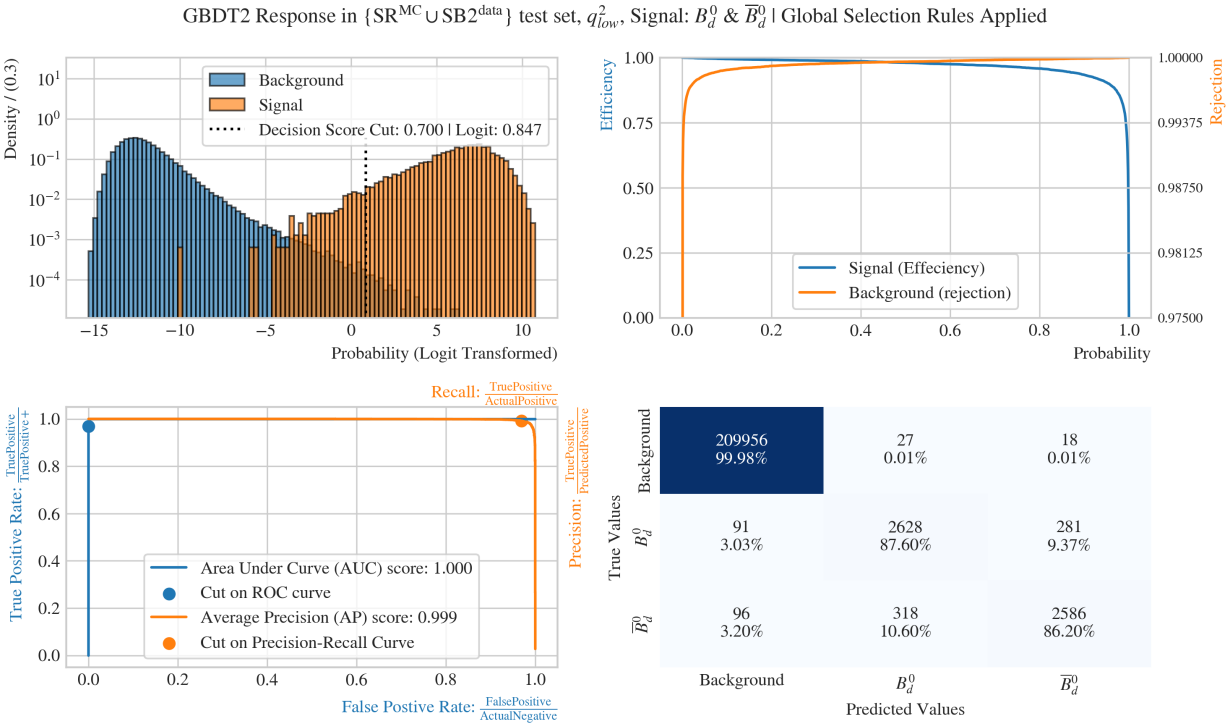


Figure A.47: *Signal vs. Background* Testing Suite on "2GNN to 2GBDT full search"-GBDT2 test-set for iteration 6 of the full n-tuple feature search. The AUC and AP score shows that there are still some leaking features.

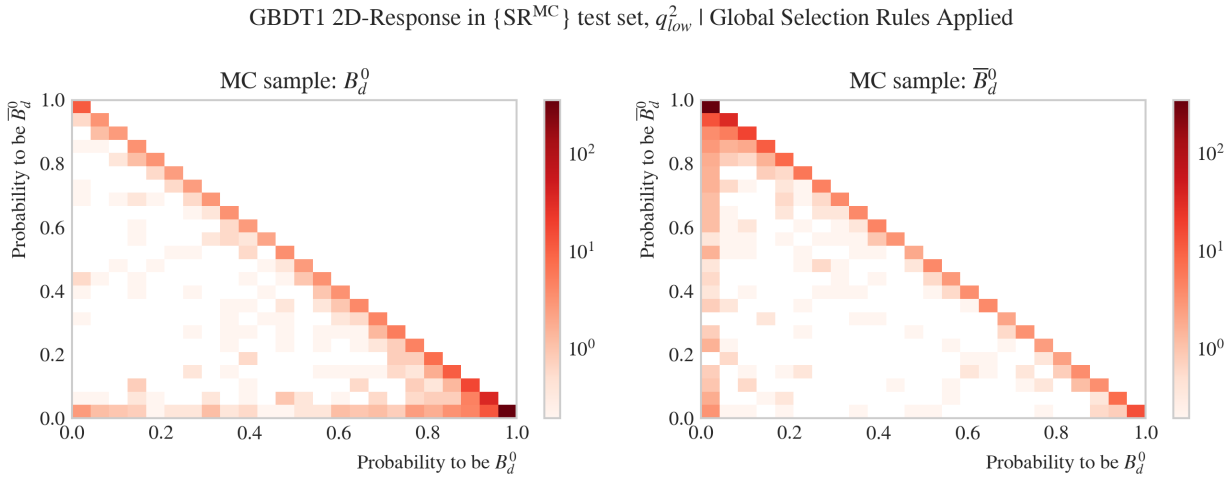


Figure A.48: *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT full search"-GBDT1 test-set with density on log-scale for iteration 6 of the full n-tuple feature search.

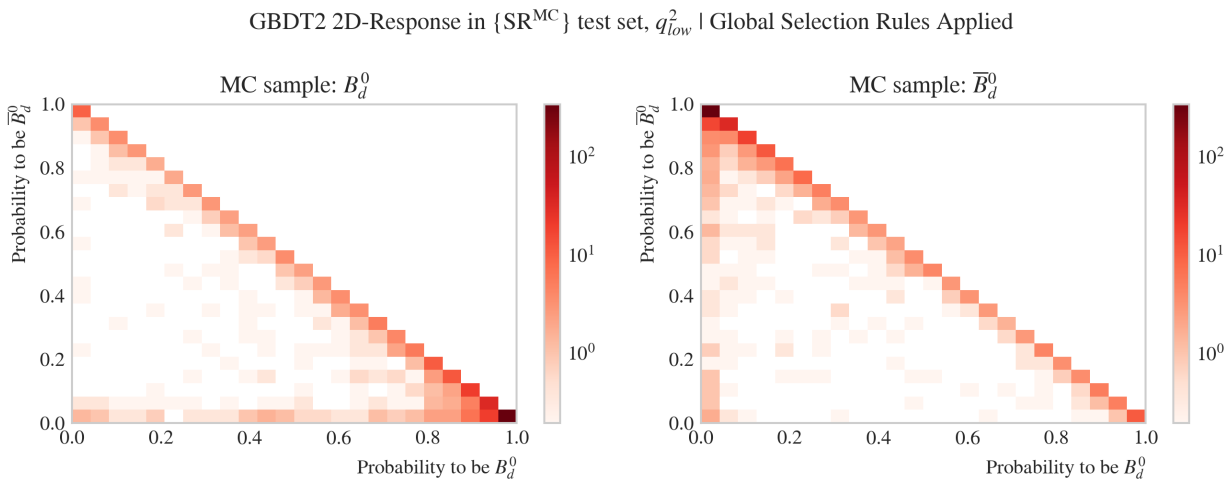


Figure A.49: *Sig(B^0) vs. Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT full search"-GBDT2 test-set with density on log-scale for iteration 6 of the full n-tuple feature search.

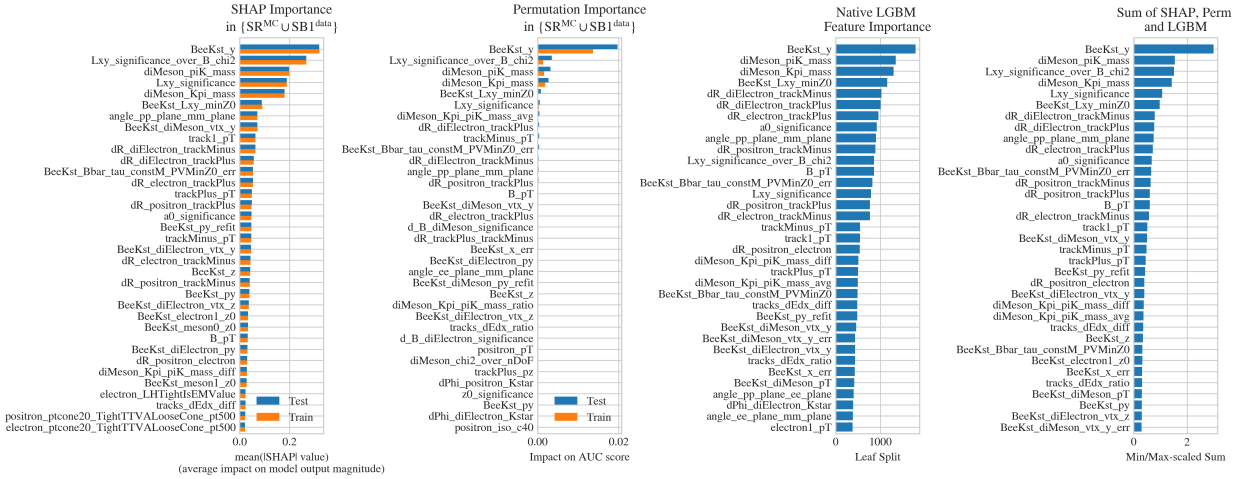


Figure A.50: 35 of the highest scoring feature importance's on "2GNN to 2GBDT full search"-GBDT1 for both train- and test-set for iteration 6 of the full n-tuple feature search.

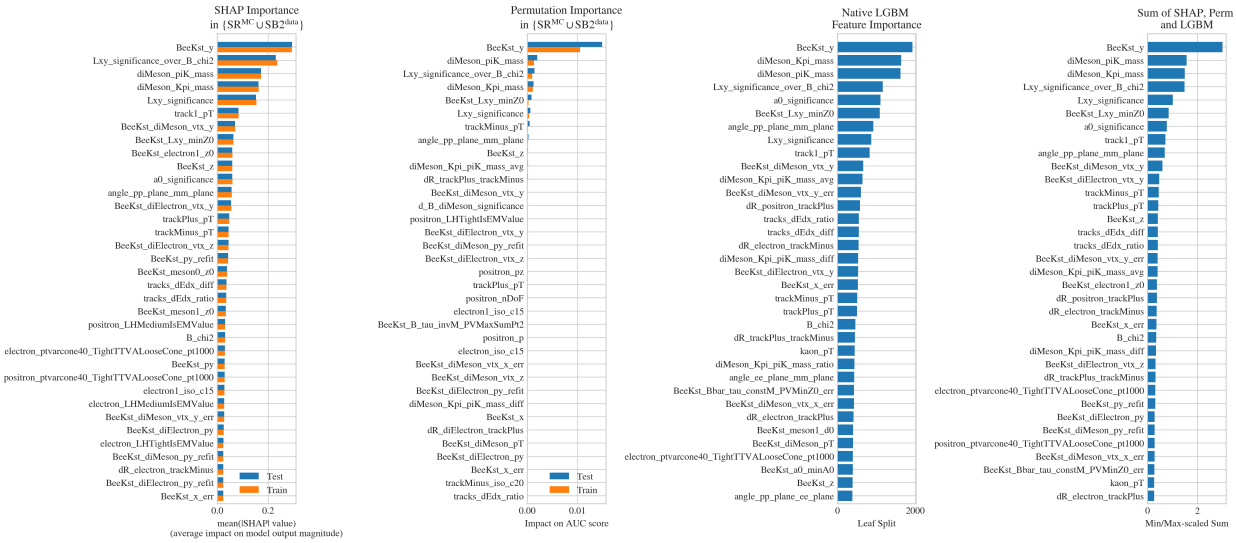


Figure A.51: 35 of the highest scoring feature importance's on "2GNN to 2GBDT full search"-GBDT2 for both train- and test-set for iteration 6 of the full n-tuple feature search.

A.7 2GNN to 2GBDT w Extra Features

Training Time	10min 4.119sec	8min 43.533sec
task	train	train
learning_rate	0.05	0.05
num_leaves	133	85
colsample_bytree	0.915007	0.715973
subsample	0.85014	0.805926
bagging_freq	1	1
max_depth	-1	-1
verbosity	-1	-1
reg_alpha	0.00026	0.000004
reg_lambda	3.801479	0.00002
min_split_gain	0.0	0.0
zero_as_missing	False	False
max_bin	255	255
min_data_in_bin	3	3
random_state	42	42
device_type	cpu	cpu
num_classes	3	3
objective	multiclass	multiclass
metric	multi_logloss	multi_logloss
num_threads	35	35
min_sum_hessian_in_leaf	2.637486	0.003614
n_estimators	235	234

Table A.5: "2GNN to 2GBDT w extra features" best values for the LightGBM models.

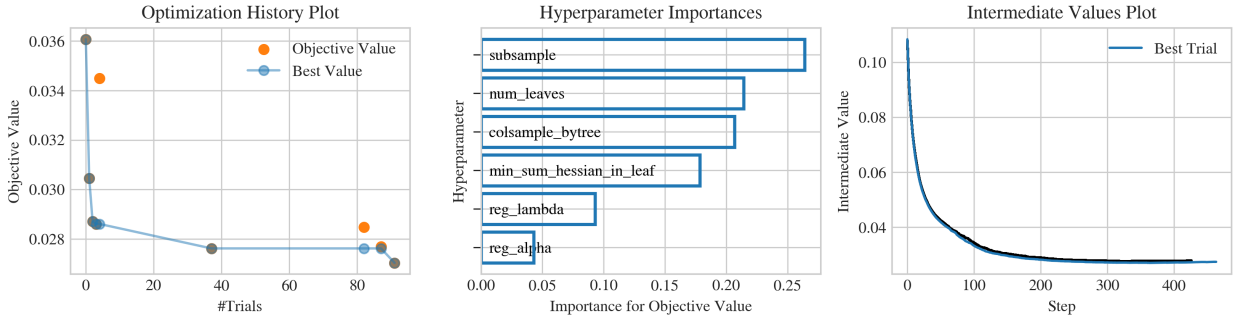


Figure A.52: *LightGBM* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT1 test-set. Showing that GBDT1s training has converged.

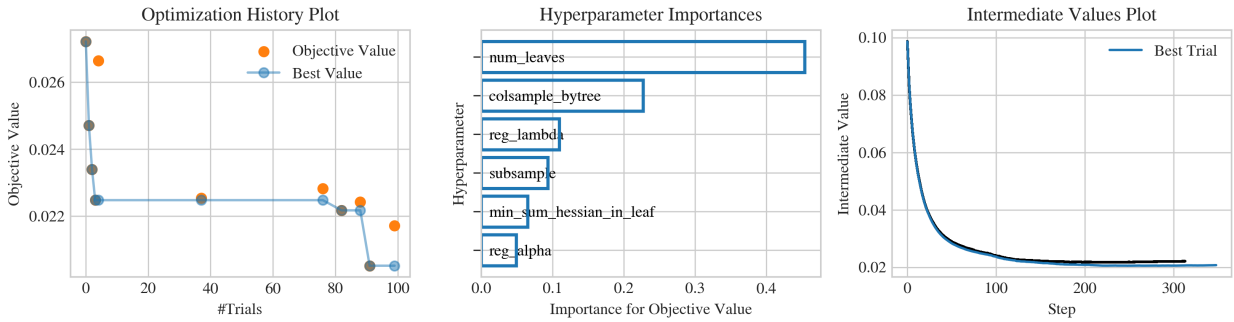


Figure A.53: *LightGBM* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT2 test-set. Showing that GBDT2s training has converged.

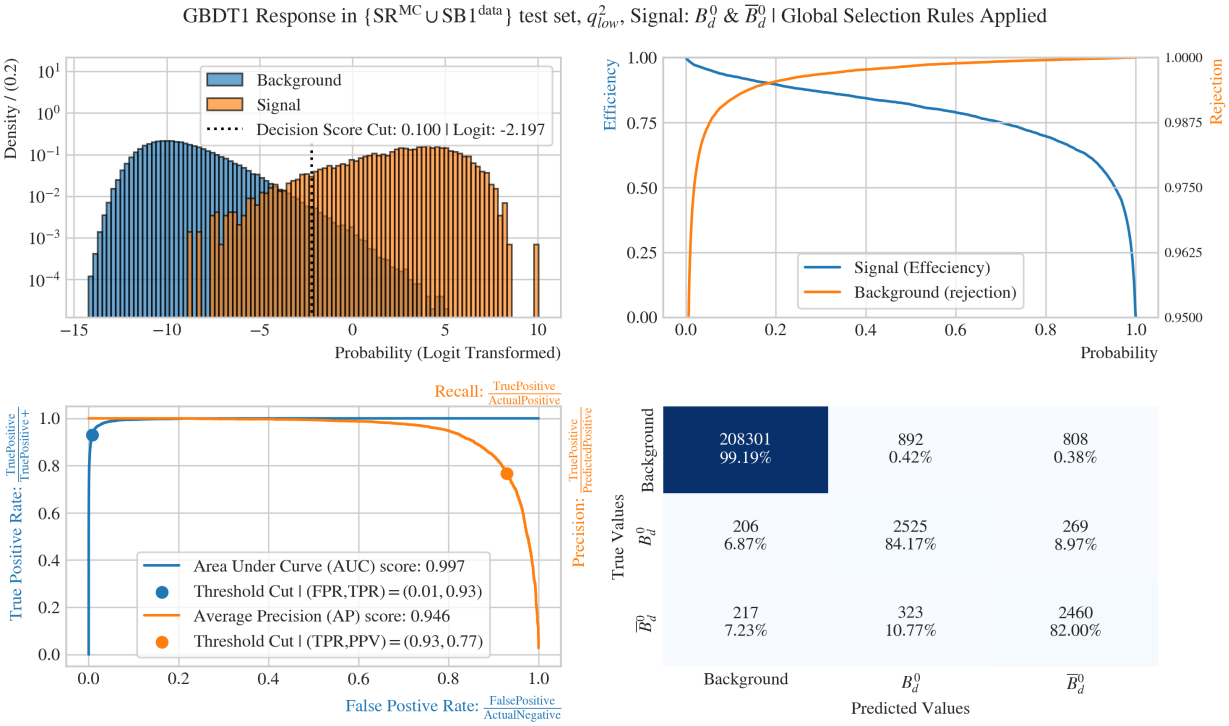


Figure A.54: *Signal vs Background* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT1 test-set which shows good classification performance overall.

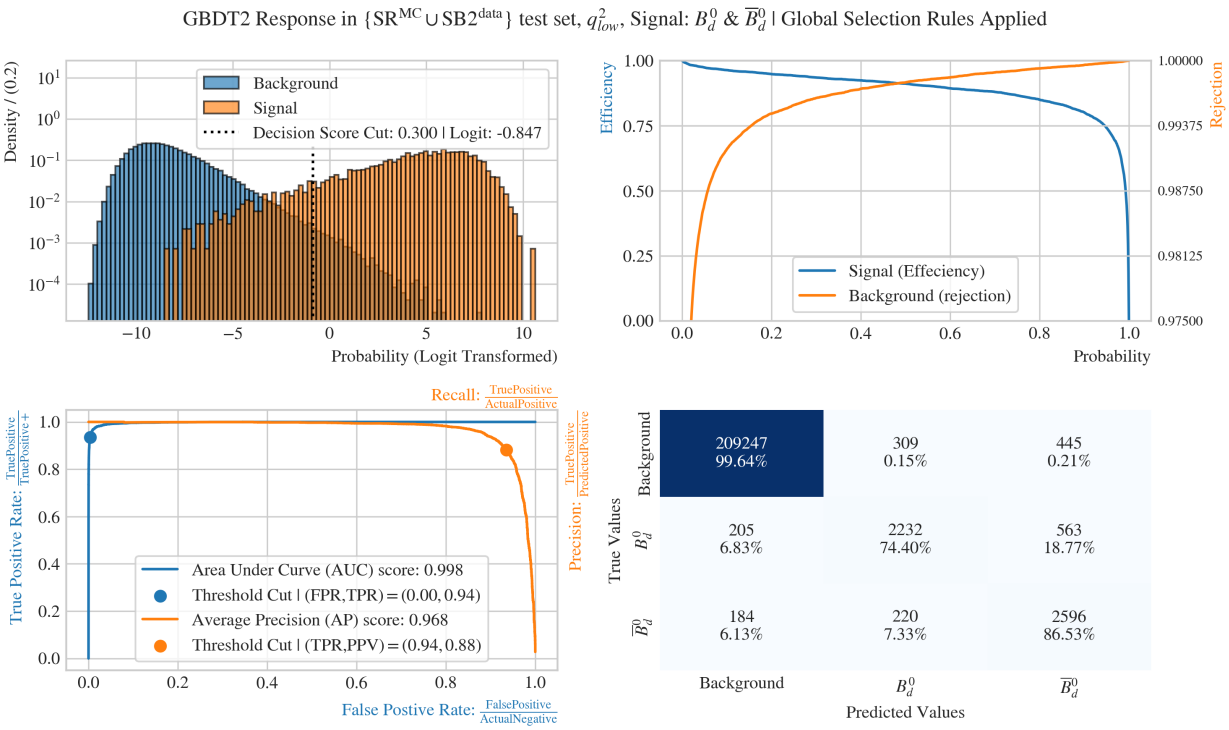


Figure A.55: *Signal vs Background* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT2 test-set which shows good classification performance overall.

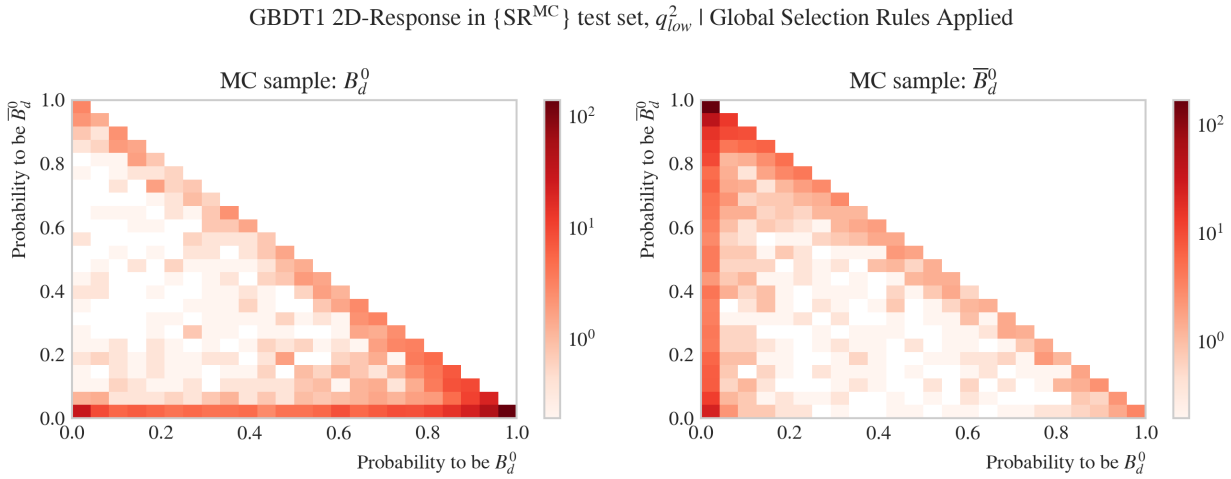


Figure A.56: *Sig(B^0) vs Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT1 test-set with density on log-scale which shows good classification performance on the different signal species.

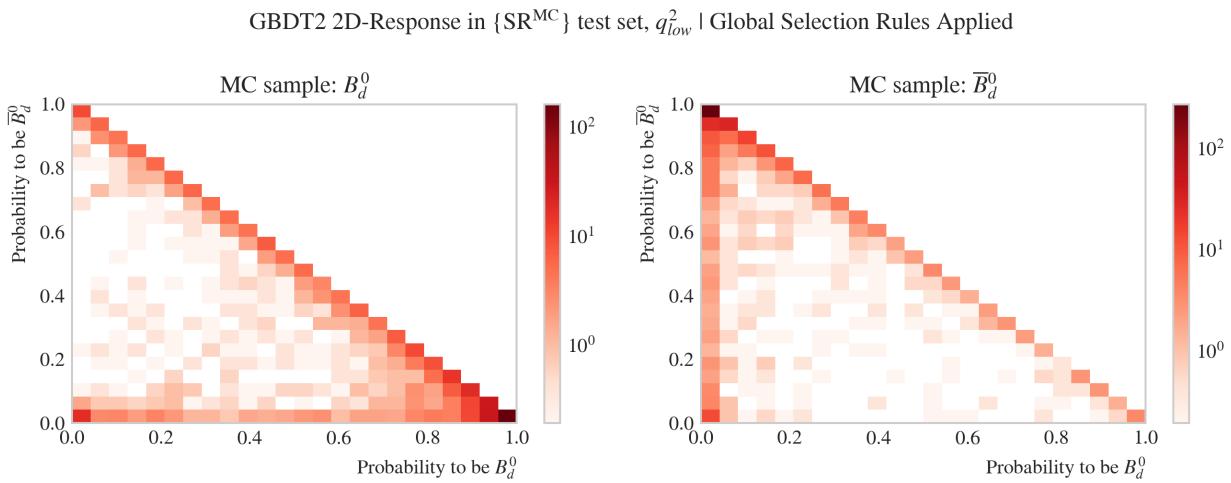


Figure A.57: *Sig(B^0) vs Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT2 test-set with density on log-scale which shows good classification performance on the different signal species.

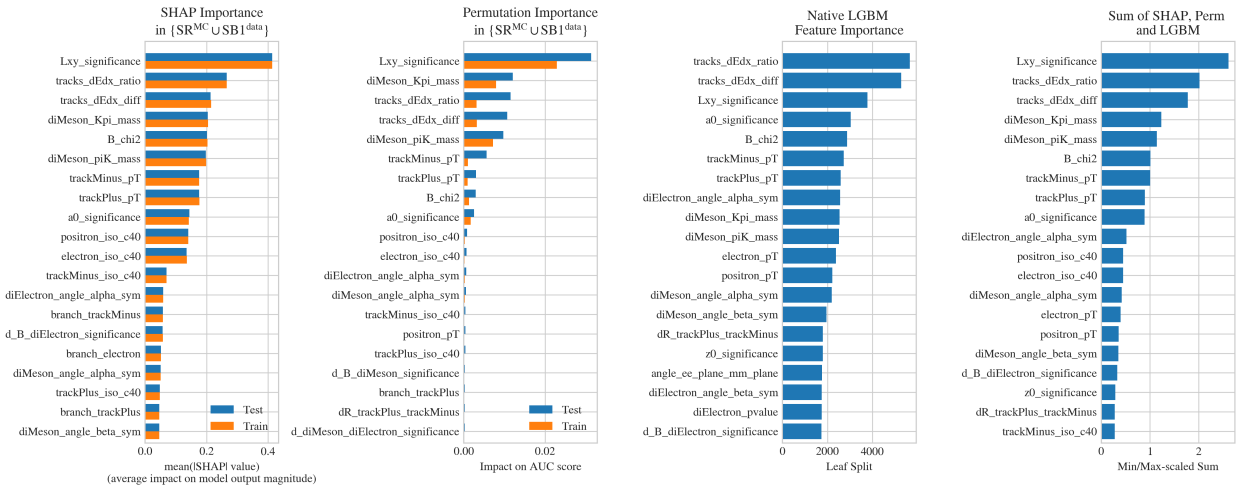


Figure A.58: 20 of the highest scoring feature importances on "2GNN to 2GBDT w extra features"-GBDT1 for both train- and test-set.

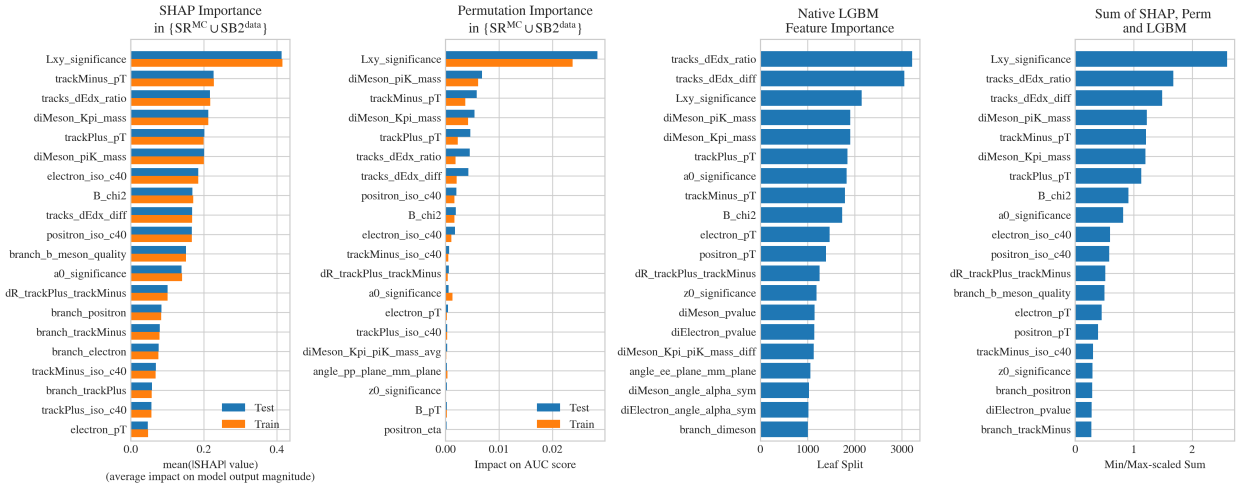


Figure A.59: 20 of the highest scoring feature importances on "2GNN to 2GBDT w extra features"-GBDT2 for both train- and test-set.

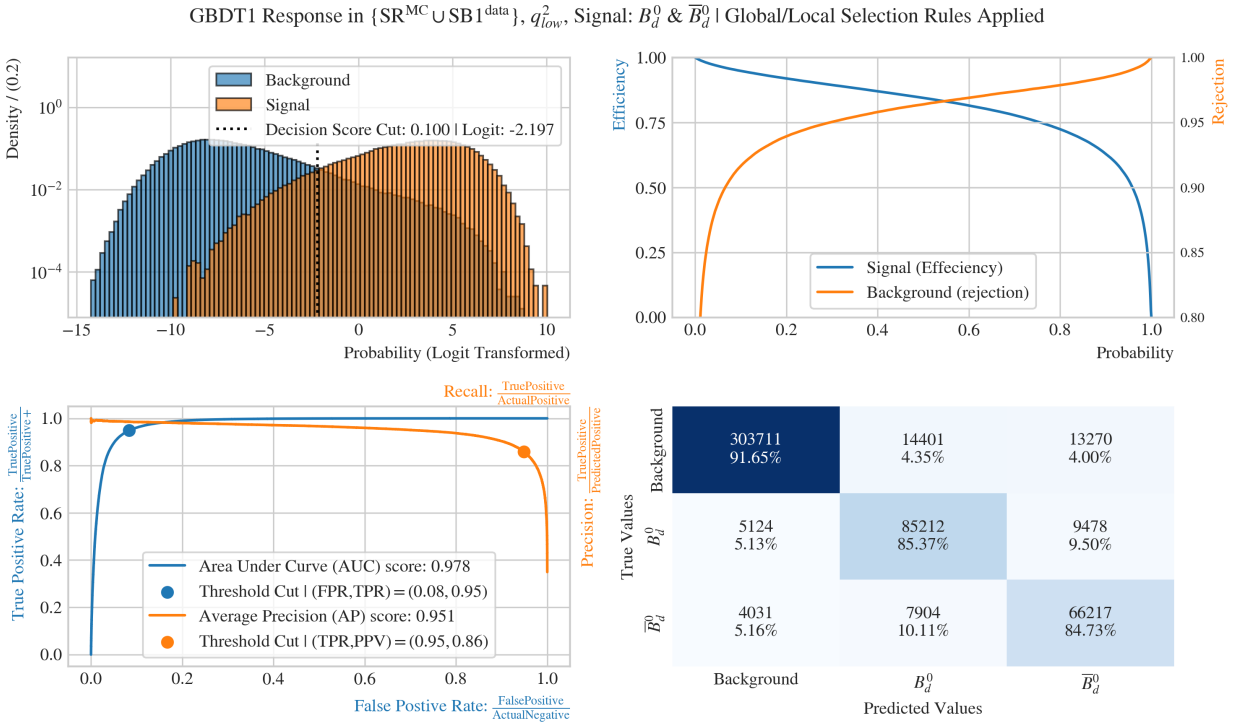


Figure A.60: *Signal vs Background* Testing Suite with "2GNN to 2GBDT w extra features"-GBDT1 on non-train SR, SB1 which shows good classification performance overall when the global and local selection rules are applied.

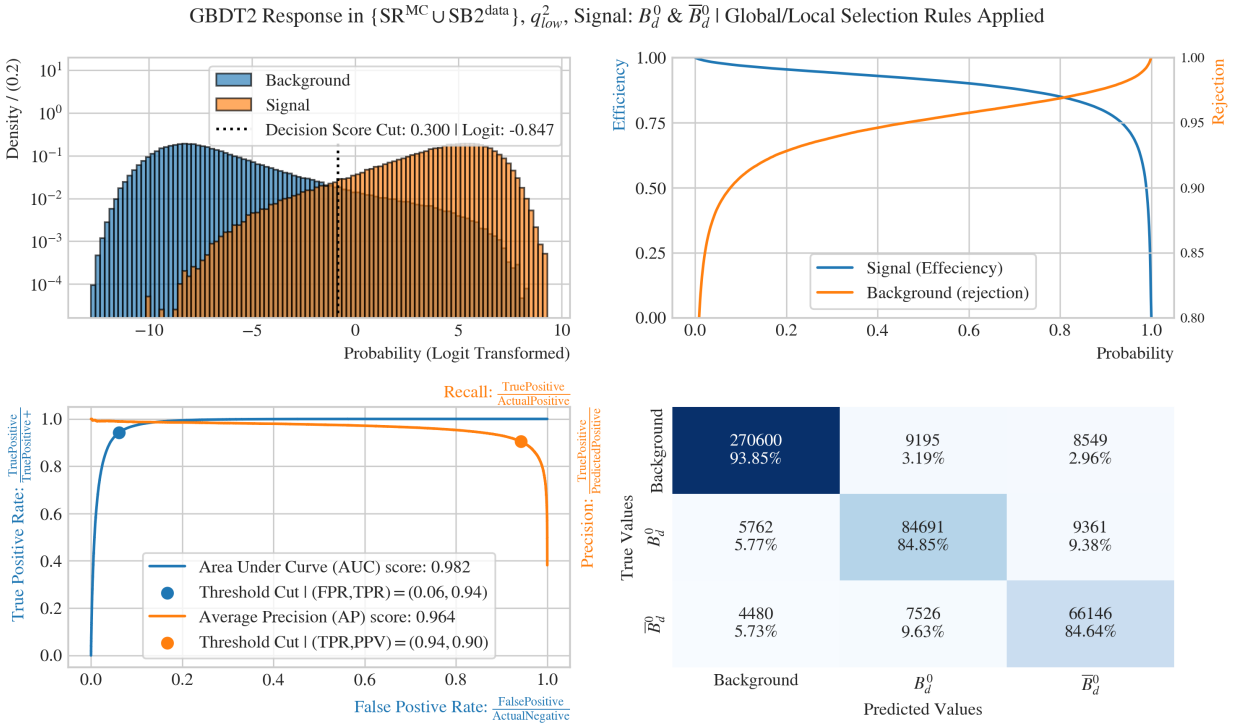


Figure A.61: *Signal vs Background* Testing Suite with "2GNN to 2GBDT w extra features"-GBDT2 on non-train SR, SB2 which shows good classification performance overall when the global and local selection rules are applied.

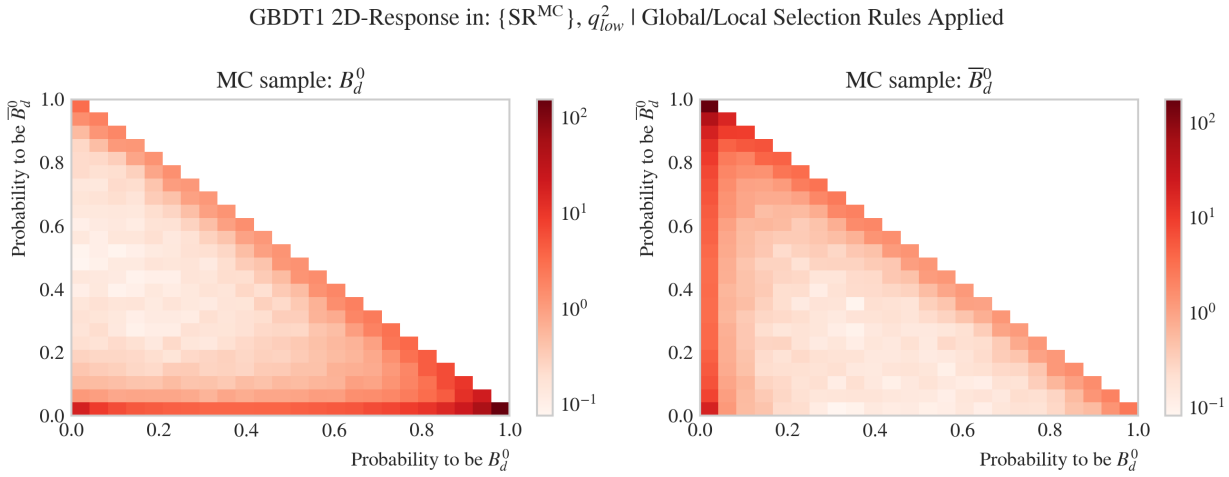


Figure A.62: *Sig(B^0) vs Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT1 with density on log-scale.

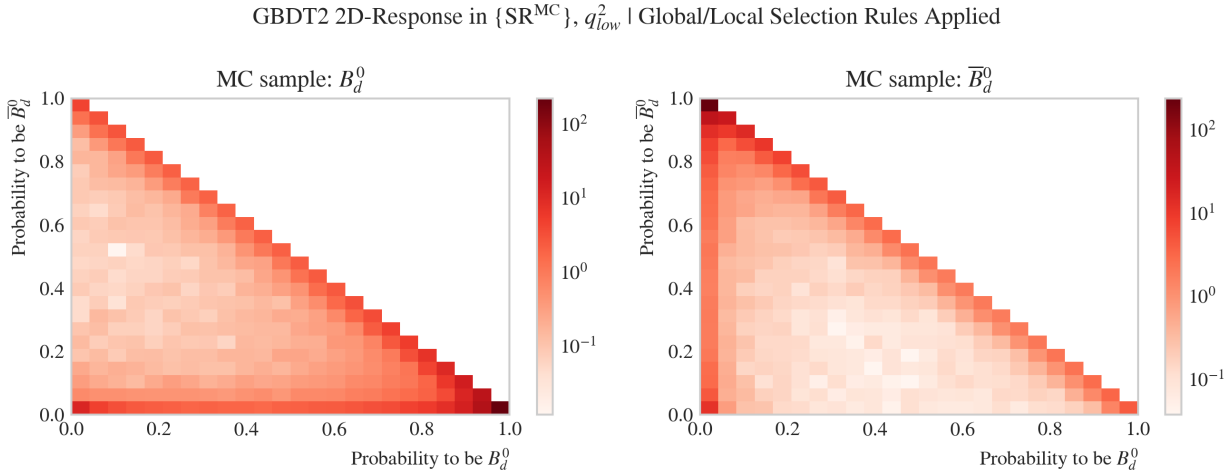


Figure A.63: *Sig(B^0) vs Sig($\overline{B}^0$)* Testing Suite on "2GNN to 2GBDT w extra features"-GBDT2 with density on log-scale.

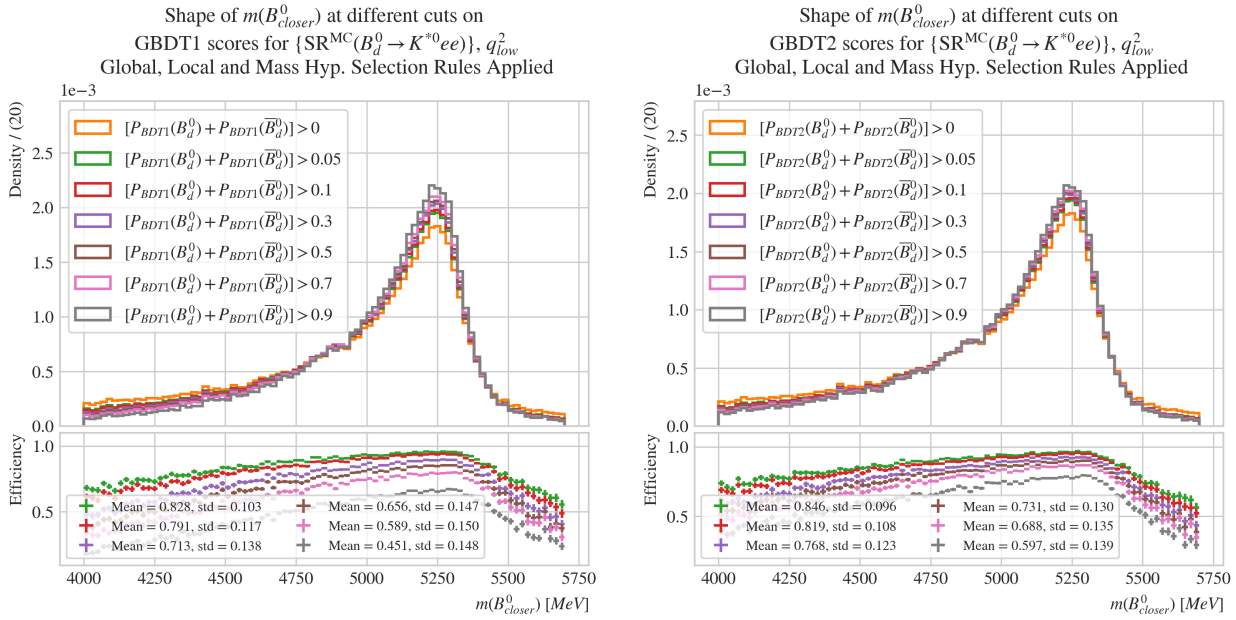


Figure A.64: *Mass Shape* Testing Suite for Signal on "2GNN to 2GBDT w extra features". Just as for the "2GNN to 2GBDT" approach: no distortion around the signal peak, however, the distortion increases as $m(B^0)$ increases and decreases around the peak.

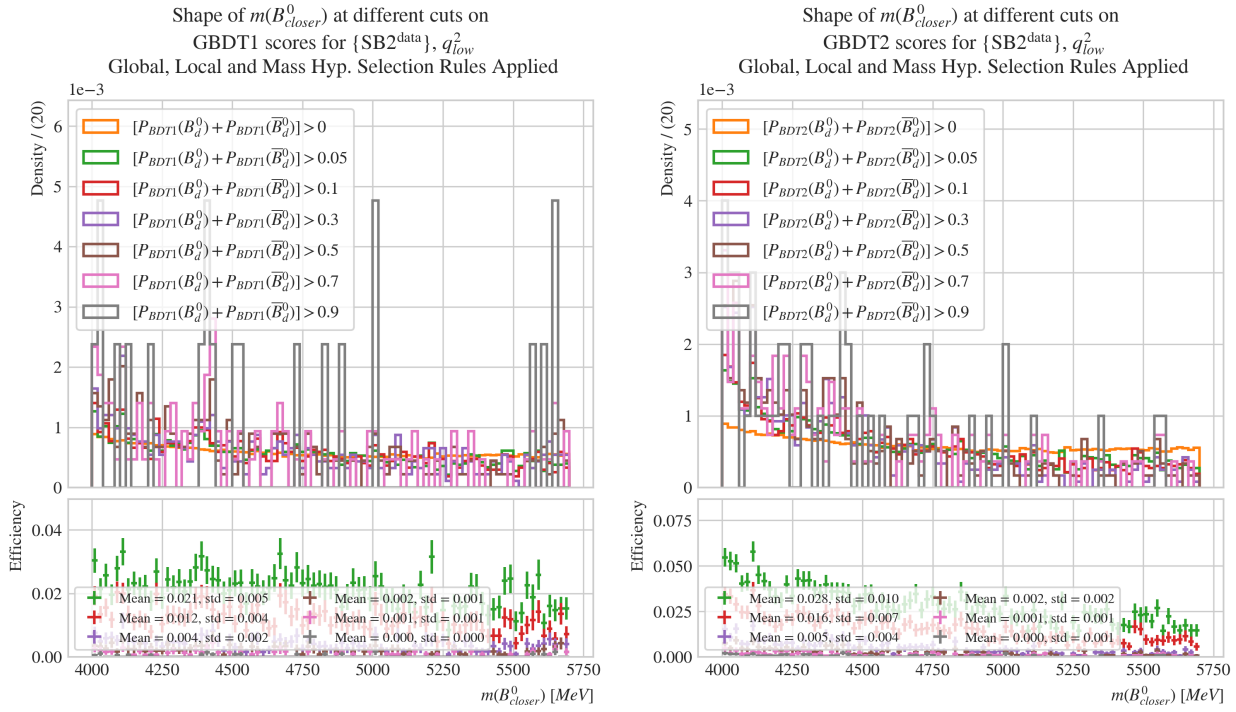


Figure A.65: *Mass Shape* Testing Suite for background on "2GNN to 2GBDT w extra features". The distortion of the background is close to the "2GNN to 2GBDT" approach meaning there i little to no sculpting in the background.

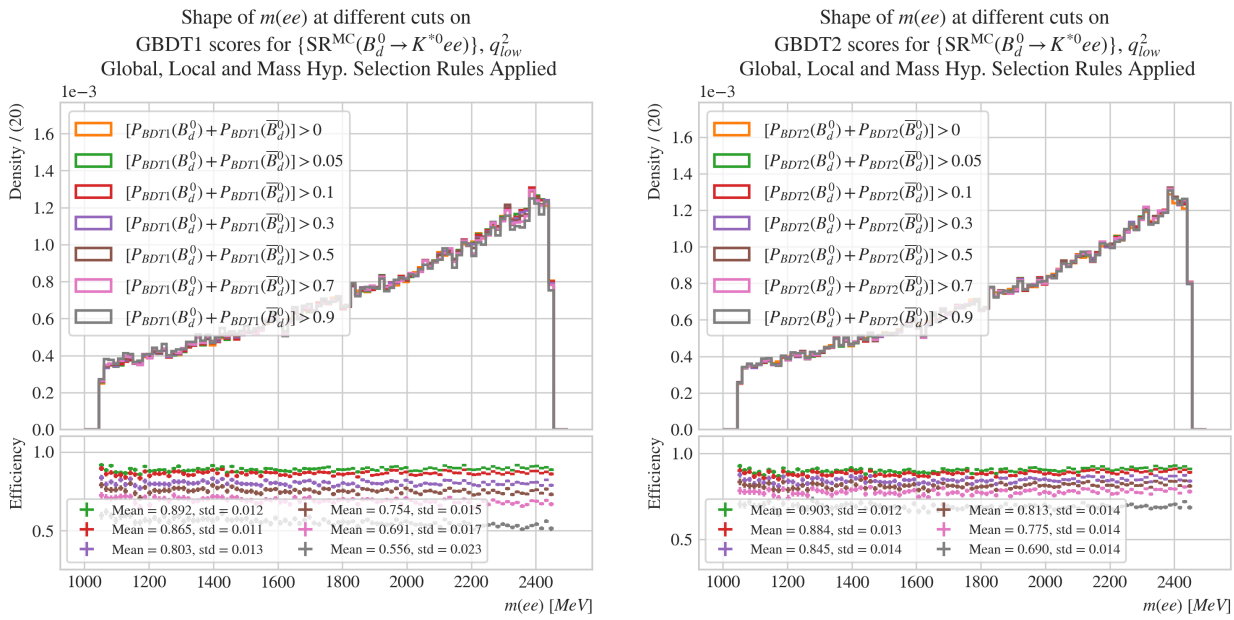


Figure A.66: *Mass Shape* Testing Suite for $m(ee)$ on "2GNN to 2GBDT w extra features". No significant distortion is seen in $m(ee)$ for any GBDT cuts.

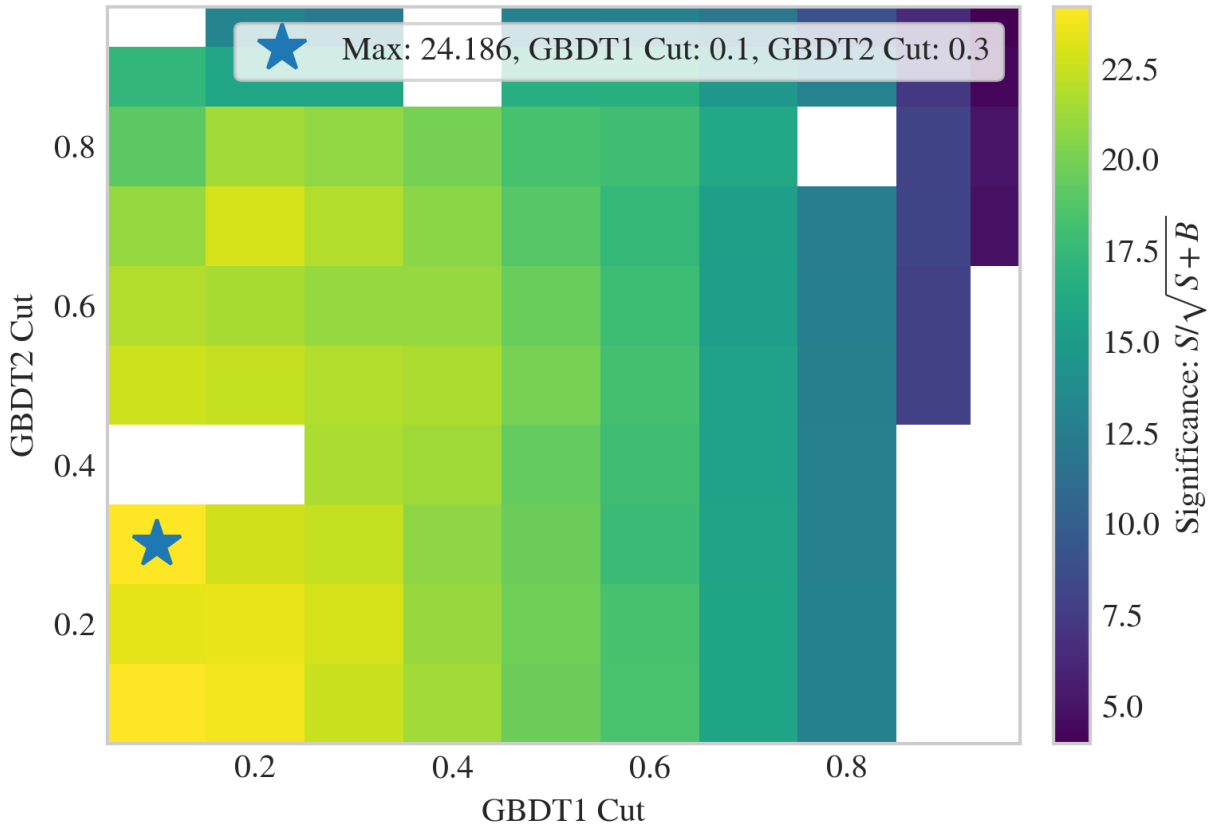


Figure A.67: Blinded Significance on B_d^0 q^2_{high} MC signal with the "2GNN to 2GBDT w extra features". The signal PDF shape parameters are fixed, and background PDF shape parameters are free. The Grid used are $\mathcal{M} \times \mathcal{M}$ where $\mathcal{M}$ =[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95].

A.8 MLLH Fits

An example of the fits done with the binned MLLH-fit routine before transitioning to the χ^2-fit routine.

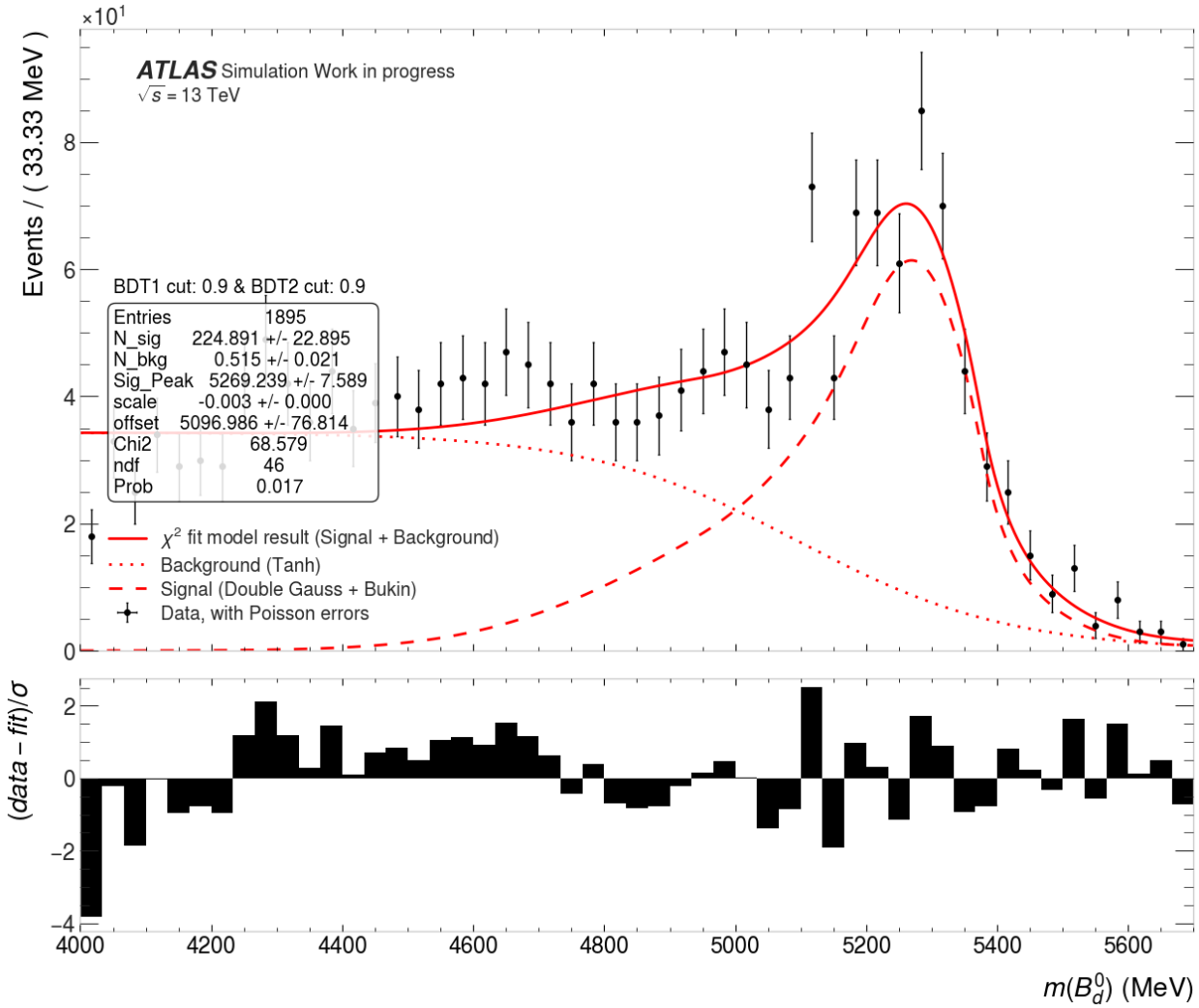


Figure A.68: MLLH test fit (sig+Bkg) on $m(B_d^0)$, q^2_{high} period K data

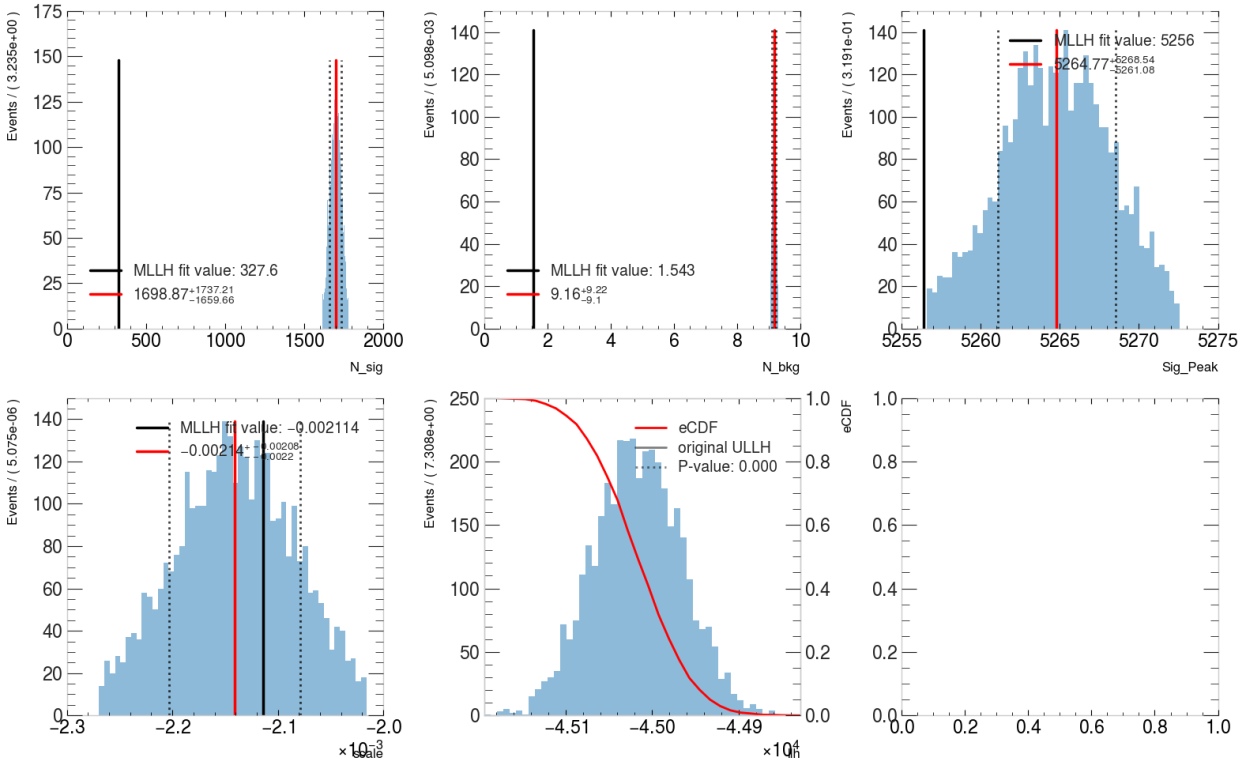


Figure A.69: Finding errors for the MLLH fit in Fig. (A.68) with bootstrapping.