Karlsruhe Institute of Technology

ETP-KA/2022-17

Anomaly Detection in Searches for Inelastic Dark Matter with a Dark Higgs at Belle II

Jonas Eppelt

Masterthesis

14th November 2022

Institute of Experimental Particle Physics (ETP)

Advisor: Prof. Dr. Torben Ferber
Coadvisor: Prof. Dr. Günter Quast

Editing time: 12th November 2021 – 11th November 2022

Karlsruhe Institute of Technology

ETP-KA/2022-17

Anomalie Detektion in Suchen nach inelastischer Dunkler Materie mit einem Dunklen Higgs bei Belle II

Jonas Eppelt

Masterarbeit

14. November 2022

Institut für Experimentelle Teilchenphysik (ETP)

Referent: Prof. Dr. Torben Ferber
Korreferent: Prof. Dr. Günter Quast

Bearbeitungszeit: 12. November 2021 – 11. November 2022

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Karlsruhe, 14. November 2022

. . . . . .

(**Jonas Eppelt**)

Abstract

In recent years a complementary search paradigm to classical searches has gained traction in the high energy physics community. While classical searches select regions of interest in the data based on a simulation of potential signal models, anomaly detection seeks to identify data regions, that are anomalous with respect to the data. Such searches offer the benefit of independence from concrete models. This thesis explores three autoencoder architectures as a machine-learning-driven tool for model-independent searches at the Belle II experiment. Autoencoders are types of neural networks that compress information in a lower dimensional latent space and reconstruct the information from it. The three architectures explored in this thesis use unregularised latent spaces, latent spaces with a Gaussian prior and with a Dirichlet prior. With the process of encoding and decoding optimized for certain samples, a higher error in the reconstruction is expected for rare samples or samples not represented during the training. This thesis studies the sensitivities of this error for two free parameters of the inelastic Dark Matter model with Dark Higgs. In a comparison between the three architectures unregularized latent spaces provide the highest sensitivities. Studies on the dimensionality of the latent spaces show varying sensitivities for different mass configurations. The studies yield the highest sensitivities for small-mass configurations with an 8-dimensional latent space and for large-mass configurations with a 9-dimensional latent space. Further studies on using the latent space to identify anomalies and efforts to validate the autoencoders on Belle II data are presented.

Disclaimer

This work builds on Patrick Ecker's (KIT, ETP) work done for a search for Inelastic Dark Matter with a Dark Higgs with displaced vertices. His programs were partly changed or extended. This applies to the workflow management, event simulation, generator simulation, and all histogram plots. The background samples were produced and centrally reconstructed by the Belle II collaboration. The simulation for the signal samples was written by Patrick Ecker and run by me with adjusted parameters. The content of all plots is created by me unless stated otherwise. All autoencoders trained and analyzed in this work are implemented by me. The implementation of the Dirichlet Variational Autoencoder is inspired by Barry M. Dillon's implementation from [1]. The training algorithm was developed with the support of Mahnoor Tanveer from Helmholtz AI and implemented by me using *pytorch* and *scikit-learn* [2]. All analyses and results presented in this thesis were done by me.

Contents

1. Introduction

Throughout years of research, the standard model of particle physics has proven to be a very accurate description of the smallest particles. However, some observations still can not be described by it. One of these is what is called Dark Matter. Observations of stars and galaxies point to the existence of a form of matter undetectable by our current methods. Many models describing such matter have been proposed and many experiments are collecting vast amounts of data in search for it.

To falsify these models, regions of interest in the data are selected and the features of the models are checked. This process however must be repeated for every model. Therefore, a complementary search paradigm has been proposed: Instead of looking for specific features of each model, unusual data points are identified and investigated. Such a search is agnostic towards specific models and can provide new hints for physics beyond the standard model. This ansatz of detecting anomalies is already established in other areas like IT security or banking [3] and is gaining traction within High Energy Physics (HEP).

To explore new ways how such searches could be conducted in the context of HEP, the LHC-Olympics [4] were held in 2020. The promising results are inspiring this work to apply anomaly detection in the context of the Belle II experiment. As an e^+e^- collision experiment, Belle II is sensitive to different signatures of Dark Matter. One of these signatures searched for is from the Inelastic Dark Matter model. This model boasts up to seven free parameters, making simulations for all its different configurations time- and resource-intensive. These configurations also show a wide range of features, complicating the selection of regions of interest. Here, methods of Anomaly Detection (AD) can simplify this search and offer a model-parameter-independent selection.

Its features and the parameter ranges considered in this work are described in Chapter 3, as well as the standard model background processes considered. Selection and reconstruction of events are described in Chapter 4 including a study of the background processes used in the training. For anomaly detection, three different architectures of autoencoders (Chapter 5) are tested and their sensitivities to different parameter configurations are studied in Chapter 7.

2. The Belle II Experiment

The Belle II experiment allows for precise tests of the Standard Model (SM) and to investigate its shortcomings using byproducts of electron-positron collisions. In this position, it presents itself as an application for anomaly detection and model-independent searches.

2.1. SuperKEKB

Located at KEK in Tsukuba, Japan, the SuperKEKB accelerator is currently the world's most luminous electron-positron collider. As described in its technical report [5] and in [6], it is an upgrade to the previous accelerator KEKB to achieve a target luminosity of $650\,\mathrm{fb}^{-1}\,\mathrm{s}^{-1}$. It mostly operates at the energy of the $\Upsilon(4S)$ with electrons accelerated to 7 GeV in the high energy ring (HER), and positrons to 4 GeV.

Since at the resulting center of mass energy of $\sqrt{s} = 10.58\,\mathrm{GeV}$ approximately 96% of the produced $\Upsilon(4S)$ mesons decay into B meson pairs, SuperKEKB is classified as a B factory. However, operations at the $\Upsilon(1S)$ up to $\Upsilon(6S)$ resonances are possible.

Compared to its predecessor, KEKB, SuperKEKB has increased the beam currents and implements the nano-beam scheme, in which superconducting quadrupole magnets squeeze the beams in the vertical direction and increase the crossing angle. A scheme of the accelerator, including the pre-accelerators and the four experimental halls is shown in Fig. 2.1.

2.2. Belle II

The Belle II detector is the upgrade of the Belle detector to, among other goals, extend the possible reach of searches for new physics. Some of the main improvements over the predecessor are the new Pixel Detectors (PXD) and the Electromagnetic Calorimeter (ECL) electronics. It is designed as a general-purpose, 4π-detector using several tracking layers, an ECL, systems for Particle Identification (PID), and a dedicated K_L and Muon Detector (KLM). A detailed description of these components can be found in [7]. Here, only a short overview based on this technical report should suffice.

The detector's design is layered (see Fig. 2.2), starting with the vertex detectors right after the beam pipe. Innermost, the PXD is located, built of two layers of silicon pixel sensors in Depleted Field Effect Transistor (DEPFET) technology. Next is the Silicon Vertex

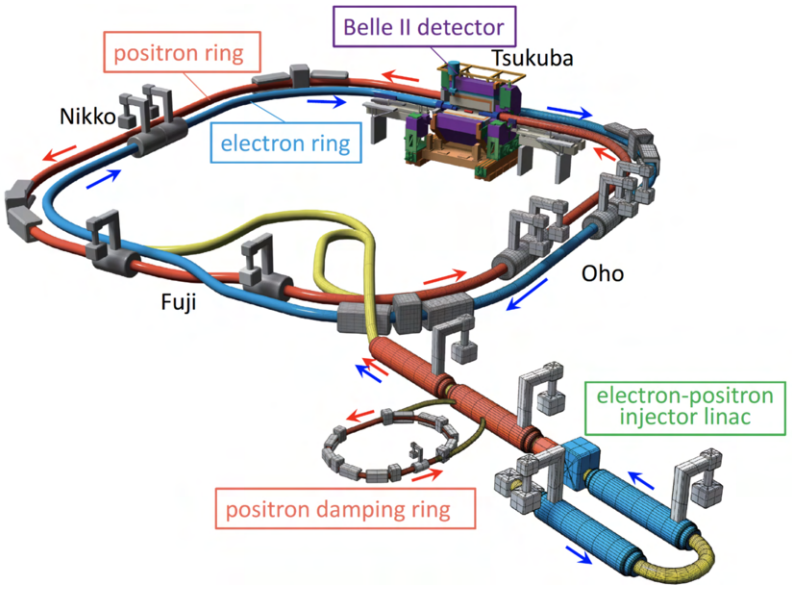

Figure 2.1.: Scheme of the SuperKEKB accelerator complex including the Belle II detector. Blue (Red) marks the beamlines and linear pre-accelerators for the electrons (positrons). The four experimental halls are also marked with Tsukuba hall containing the Belle II detector at the only Interaction Point (IP). Taken from [5].

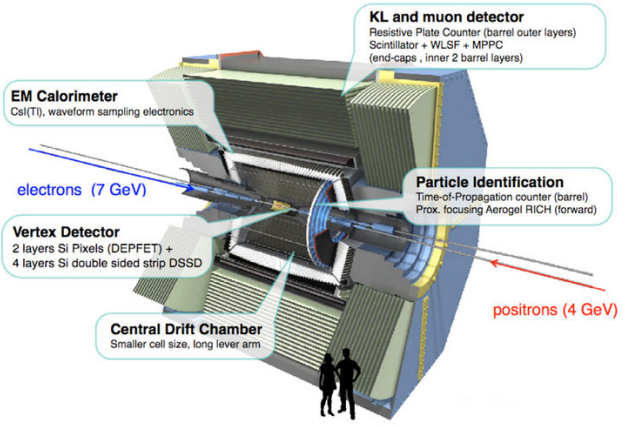

Figure 2.2.: Schematic overview of the Belle II detector with notes to the subsystems. Additionally, the directions and energies of the electrons and positrons at a typical run at $\Upsilon(4S)$ are shown. Due to the asymmetric energies, the detector itself is built asymmetrically.

Detectors (SVD) with four layers of double-sided silicon strip sensors. The last part of the tracking detectors is the Central Drift Chamber (CDC), made of wires and filled with Helium-Methane gas. Besides its main use for precise measurements of charged particles' tracks, the energy loss of particles within the gas gives information on the type of particle.

This information is combined with measurements of the Time-of-Propagation (TOP) and Aerogel Ring-imaging Cherenkov (ARICH), to give the PID. The TOP uses the Cherenkov effect in quartz radiators and ARICH measures the Cherenkov cone in an aerogel with photon detectors.

The surrounding ECL is built out of CsI(Tl) crystals covering the polar angle from 12.4° to 155.1°. In it, most particles are stopped as they cause cascading particle showers in the crystals. Since they often reach neighboring crystals, a single particle is detected by a cluster of crystals. Via the scintillation of these showers, the energy of a particle is measured.

Particles that are not stopped by the ECL, usually K_L^0 and μ, reach the KLM system. It is located outside of the superconducting solenoid and consists of steel plates (for the magnetic reflux) and glass-electrode resistive plate chambers. On the endcaps and the first two layers of the barrel region, scintillators are used to accommodate the high beam backgrounds produced by SuperKEKB.

Since the luminosity provided by SuperKEKB is too high to record every single event, a fast decision logic called trigger is used [8]. Multiple of these triggers are used in two steps: In the first step, the L1 trigger, the decision to keep or discard an event is made using low-level data from the detector. After that, a reconstruction of tracks, clusters, and particles is performed to base the High Level Trigger (HLT) decisions on.

2.3. basf2

To handle the data generated by the detector, the Belle II collaboration has created an analysis software framework called Belle II Analysis Software Framework (`basf2`) [9] [10]. It contains tools to generate Monte Carlo (MC) particles, the detector's response and to handle raw data of the sub-detectors. Most importantly it also contains algorithms for tracking, clustering, and particle identification, which form the basis of any further analysis. This work uses its modules for decay reconstruction and combinatorics, candidate selection, and vertex fitting.

3. Physics Theory

3.1. Physics Beyond the Standard Model

While the SM has proven itself with very precise predictions, it is also a consensus among the physics community that some observed phenomena can not be explained by the SM. The rotation speed of galaxies around their center is one of these. Stars far away from a galaxy's center rotate much faster around the center than the gravitation of visible matter would allow. The prevalent explanation for this is non-visible matter, called Dark Matter (DM).

While the first idea of non-visible matter can be found in ancient Greece, the modern search gained momentum with Vera Rubin's and Kent Ford's research in the 60s and 70s [11]. By now DM several phenomena like the Comsmic Microwave Background (CMB), the dynamics of galaxy clusters, and the strong CP problem are considered to be linked with DM. Over time, a myriad of different models on the nature of DM has been proposed.

Inelastic Dark Matter with a Dark Higgs

Increasingly stronger bounds on the mass(es) of potential DM particles have shifted the focus to light (GeV − MeV) candidates. Strong bounds imposed by the CMB do not apply to inelastic couplings to SM particles. A detailed description of this model can be found in [12]. A short overview of it with a focus on some simplifications taken for this work is given below.

The inelastic Dark Matter with a Dark Higgs (IDMDH) model not only introduces two Dark Matter particles but also bosonic particles, that can interact with the Dark Matter. Together, Dark Matter and the exchange particles make up the Dark Sector. In this setup, the dark sector can be reached by kinetic mixing of a Dark Photon A' and the SM photon. The lightest matter particle of the dark sector is called χ_1. By coupling to the A' it can be excited into a χ_2. In analogy to the SM Higgs mechanism, a Dark Higgs h' is introduced, creating a second, independent portal to the SM sector and therefore creating a rich phenomenology. Like in the SM this Higgs sector is needed, to give the DM particles mass. This leads to seven free parameters:

- The mass of the A', $m_{A'}$
- The mixing angle of the SM photon to the A', ϵ

- The mass of the h', $m_{h'}$
- The mixing angle of the SM Higgs to the h', θ.
- The mass of the χ_1, m_{χ_1}
- The coupling of the χ_1 and χ_2 to the h', f
- The coupling of the χ_1 and χ_2 to the A', g_X

Importantly, not all possible combinations of these parameters correspond to the perturbative regime, e.g. all couplings are required to be smaller than $\sqrt{4\pi}$. This implies, that the h' can not be much heavier than the A'

$$m_{h'}^2 \lesssim \frac{\sqrt{\pi}}{4g_X} m_{A'}^2. \tag{3.1}$$

Further, from considerations of DM annihilation and the CMB the masses have the constraints

$$\frac{f^4}{64\pi^2} m_{\chi_1} < m_{h'} \lesssim m_{\chi_1} < m_{A'}. \tag{3.2}$$

Simplifications

As shown in [12] there are multiple processes and final states, that could be used to search for IDMDH. In this work, only the process shown in Fig. 3.1 is considered. In this process,

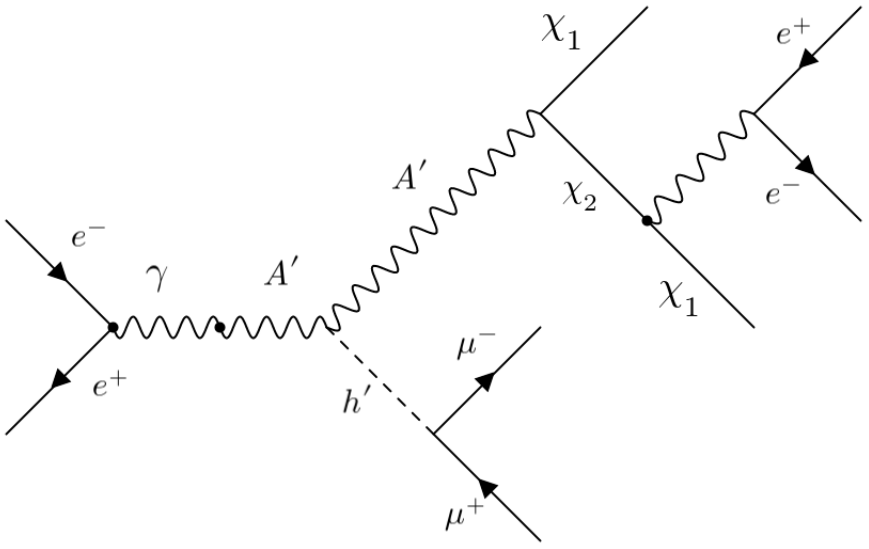

Figure 3.1.: Feynman diagram of the DM production process considered in this work.

a SM photon mixes with a Dark Photon, which radiates a Dark Higgs. This Dark Higgs decays into a muon-anti-muon pair. The Dark Photon decays into the two Dark Matter particles χ_1 and χ_2. The heavier χ_2 further decays into a χ_1 and, via a Dark Photon, into an electron-positron pair. This leaves a final state in the detector of two lepton pairs $\mu^+\mu^-$

and e^+e^- and missing Energy. As the two χ_1 produced can not be detected, the sum of all particles' energies should not add up to the total energy of the collision.

For further simplification, the Dark Photons mass is set to $m_{A'} = 4m_{\chi_1}$. From the theory, the mass of the χ_2 is

$$m_{\chi_2} = m_{\chi_1} + \frac{f \cdot m_{A'}}{g_X} \tag{3.3}$$

The coupling constants are fixed to

$$f = \sqrt{4\pi\alpha_f} \approx 0.2476 \tag{3.4}$$

and

$$g_X = \sqrt{4\pi\alpha_D} \approx 1.12 \tag{3.5}$$

with $\alpha_f = 0.006$ and $\alpha_d = 0.1$. This means that

$$\frac{f}{g_X} \cdot 4 \cdot m_{\chi_1} \approx m_{\chi_1}. \tag{3.6}$$

Since this work uses Belle II as an experimental setup, the parameter combinations to consider are restricted by the SuperKEKB accelerator's usual energy 10.58 GeV. This leads to the condition

$$m_{\chi_1} + m_{\chi_2} + m_{h'} < 10.58\,\text{GeV}. \tag{3.7}$$

From constraints discussed in [12] follows

$$m_{\chi_1} > m_{h'} \tag{3.8}$$

and

$$\Delta m = m_{\chi_2} - m_{\chi_1} > 2 \cdot m_\mu. \tag{3.9}$$

As discussed in [12], the search for non-prompt decays in this model benefits greatly from fewer SM backgrounds. Such a search is currently conducted by Patrick Ecker using a classical approach with selections and bump hunts. Such an approach, however, can not be easily conducted for prompt decays due to a much higher SM background. This motivates the use of machine learning tools since they rely on huge amounts of data. Therefore, this work focuses on prompt decays of both, the h' and the χ_2. This is ensured by fixing the mixing angles ϵ and θ to the sufficiently high value of 10^{-2}, since the mixing angles control the lifetimes of the A' and h'.

3.2. Standard Model

3.2.1. Possible Background Processes

A DM process, as displayed in Fig. 3.1 is expected to have the following signature in the detector:

- a muon-anti-muon pair from the decay of the h',
- an electron-positron pair from the decay of the χ_2,

- missing energy from the undetectable χ_1 particles.

There are multiple possible SM processes that can mimic such a final state. The process $e^+e^- \to e^+e^-\mu^+\mu^-$ would directly produce the lepton pairs and missing energy could be introduced by errors in measurement or reconstruction. Since tauon decays include an undetectable neutrino, the process $e^+e^- \to \tau^+\tau^-$ could look like the signal's final state. While the processes $e^+e^- \to e^+e^-$ and $e^+e^- \to \mu^+\mu^-$ lack one lepton pair and missing energy, the combination with beam background and/or errors in detection and reconstruction can fake the signature of the signal. Since muons and pions have similar masses and therefore can be easily confused, the $e^+e^- \to e^+e^-\pi^+\pi^-$ process can also contribute to the background.

The SuperKEKB accelerator is mostly run on the $\Upsilon(4S)$ resonance. This resonance mostly decays into B mesons, which have multiple decay channels like $B^\pm \to l^\pm\nu_l X$ or $B^0 \to K^0 l^+ l^-$. While usually, such events contain much more than four particles, the high production rate of B-Mesons still gives a background caused by reconstruction errors. Besides the resonant production of a b-quark and anti-b-quark, the non-resonant productions $e^+e^- \to u\bar{u}$, $e^+e^- \to d\bar{d}$, $e^+e^- \to s\bar{s}$, and $e^+e^- \to c\bar{c}$ can also contribute to the background. These processes are also collectively called "continuum background".

3.2.2. Beam Background

Beam background describes several processes that are caused by the operation of the e^- and e^+ beams independent from their collisions. The rate of these processes has increased with the high luminosity of SuperKEKB. Beam background measurements for the current Belle II configuration (Phase 3) are not yet finished, so the measurements for the previous configuration (Phase 2) [13] form the basis of this section. The main processes causing these backgrounds are:

- Touschek background: Coulomb interactions between particles in the same bunch. Such particles collide with the beam pipe wall and produce showers.
- Interactions with residual free gas molecules. Additionally to a baseline of gas present in the beam pipe, outgassing of the beam pipe material increases during operation.
- Synchrotron radiation: Photons with energy in the 10 - 100 keV range may be produced by synchrotron radiation.
- Injection background: Beams are constantly injected, due to the short lifetime of circulating beams. These continuous injections introduce perturbations, which cause a higher background rate.

Together, these processes produce particles, that have nothing to do with the collision process itself. They can produce additional tracks in the tracking detectors or clusters in the ECL. As such, they can complicate analyses and searches by faking signatures of particles.

4. Dataset Generation and Analysis

To train the autoencoders in this work, MC samples of the processes discussed in Section 3.2 are used. Sensitivity studies are performed on MC samples of IDMDH processes with various parameter configurations. Section 4.1 describes the methods used to generate the MC signal samples. The details of reconstructing Final State Particles (FSPs) and intermediate particles are described in Section 4.2, as well as further selections and their effects. Section 4.3 follows with an analysis of the MC samples and the physics processes involved.

4.1. Simulation

It is common praxis in the High Energy Physics (HEP) community to develop and test new analysis methods and algorithms on simulated samples. To do so the Belle II collaboration provides general purpose MC samples. This work uses the MC samples with simulated beam background from the 2021 production campaign.

The background processes that are used in this work are listed in Table 4.1. Samples with a different simulated luminosity than $100\,\mathrm{fb}^{-1}$ are reweighted later accordingly. For the samples with four leptons, $e^+e^- \to e^+e^-\mu^+\mu^-$ and $e^+e^- \to e^+e^-e^+e^-$, the parameters of the simulation are restricted [14][1]: When the invariant mass of one of the lepton pairs is smaller than $0.5\,\mathrm{GeV}\,c^{-2}$, the process is not simulated in order to reduce the effective cross-section of the process. The production of signal samples is done privately and uses *MadGraph5* [15] for event generation. The detector response is simulated by `basf2`, which has a model of the detector implemented using *Geant4* [16]. After the detector simulations, the reconstruction of tracks, clusters, and calculation of PID likelihoods is performed. The model parameters given in Table 4.2 are combined according to the constraints discussed in Section 3.1. For each possible combination, 25,000 events are simulated. Beam background events are simulated separately and overlayed during the event simulation. In the further discussion, three exemplary signal samples are used to represent the three extreme cases:

- small masses DM: $m_{h'} = 0.5\,\mathrm{GeV}\,c^{-2}$, $m_{\chi_1} = 0.5\,\mathrm{GeV}\,c^{-2}$
- large masses DM: $m_{h'} = 2.5\,\mathrm{GeV}\,c^{-2}$, $m_{\chi_1} = 2.5\,\mathrm{GeV}\,c^{-2}$
- high mass splitting: $m_{h'} = 0.5\,\mathrm{GeV}\,c^{-2}$, $m_{\chi_1} = 3\,\mathrm{GeV}\,c^{-2}$

[1] While the internal note on the used generators cited here also mentions this limitation for the $e^+e^- \to \mu^+\mu^-\mu^+\mu^-$ sample, this is not reflected in the `basf2` source code.

Table 4.1.: Summary of MC samples with the produced luminosity and number of events simulated

process	simulated luminosity in fb^{-1}	number of events ($\cdot 10^6$)
$e^+e^- \to e^+e^-\mu^+\mu^-$	100	188.3
$e^+e^- \to \tau^+\tau^-$	100	91.9
$e^+e^- \to e^+e^-\pi^+\pi^-$	100	189.5
$e^+e^- \to e^+e^-e^+e^-$	100	3955
$e^+e^- \to \mu^+\mu^-$	100	114.8
$e^+e^- \to e^+e^-$	100	2958
$e^+e^- \to B_0\bar{B}_0$	100	54
$e^+e^- \to B^+B^-$	100	51
$e^+e^- \to u\bar{u}$	100	160.5
$e^+e^- \to d\bar{d}$	100	40.1
$e^+e^- \to s\bar{s}$	100	38.3
$e^+e^- \to c\bar{c}$	100	132.9
$e^+e^- \to \mu^+\mu^-\mu^+\mu^-$	2000	0.35120
$e^+e^- \to K^0\overline{K}^0(\gamma)$	1000	0.886400

Table 4.2.: Summary of the model parameter values simulated.

model parameter	values
m_{χ_1} in $\mathrm{GeV}\,c^{-2}$	[0.25, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0]
$m_{h'}$ in $\mathrm{GeV}\,c^{-2}$	[0.25, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0]
$m_{A'}$ in $\mathrm{GeV}\,c^{-2}$	$4 \cdot m_{\chi_1}$
f	2.746×10^{-1}
g_X	1.12

4.2. Reconstruction and Selection

Typically, searches for specific model configurations select data based on the expected signal features. These selections are optimized in order to reduce background events as much as possible while keeping the signal ones. With the signal's features depending on the chosen parameter configuration, such selections are optimized to reach high signal-to-background ratios. Since the goal of this work is to replace such selections geared towards specific model parameter configurations, the approach to event selection differs from a typical analysis. Instead of having a high signal-to-background ratio, this selection aims for samples well suited for training and representative of what processes are happening within the detector. While one might be inclined to use all backgrounds and let the selection be fully done by the autoencoders, this approach is not practical. Firstly, the huge amount of data would pose a computing challenge and training times would probably increase strongly. Secondly, using such a wide range of data would also make it harder to actually reflect the physics. Such samples would include a huge amount of events, like beam background and mis-reconstructed events. As such events do not represent collision physics, they are considered 'unphysical'. As such, they can mimic any given signal just by chance. Typically these events can be rejected using detector information like the number of CDC hits or high-level variables like the PID. Such variables often take discrete values or have sharp cut-offs, which make them not well suited for usage in neural networks. As a consequence, in this work, a preselection of events is performed using this information. These selections search to strike a balance between keeping as many signal events as possible, while also reducing the number of background and 'unphysical' events in the training samples. Therefore, this selection focuses on some basic requirements all signal mass configurations share.

4.2.1. Reconstruction

Final State Particles

The first basic requirement an event should have are a pair of opposite-charged muons and a pair of electron and positron in its final state. As such FSP leptons, only good tracks are considered, which are defined as:

- The track has at least 20 hits in the CDC.
- The track originates from the IP ($|z_0| < 2\,\text{cm}$ and $d_0 < 0.5\,\text{cm}$).
- The track is in the CDC acceptance $17^\circ < \theta < 150^\circ$.

Here, $|z_0|$ describes the longitudinal and d_0 the longitudinal projection of the distance of closest approach of the track with respect to the IP The polar angle θ is defined with respect to the detector's cylindrical axis in the direction of the electron beam. To reduce the number of misidentified particles, the PID is used. It uses information across all subdetectors to calculate the likelihood of multiple particle hypotheses. Here, a binary version of it is used, only considering the electron and muon hypotheses. This variable is one if the particle is likely to be an electron and zero if it is likely to be a muon:

$$\text{PID}(e, \mu) = \frac{\mathcal{L}_e}{\mathcal{L}_e + \mathcal{L}_\mu} \tag{4.1}$$

The selection on it is:

- The binary PID (e, μ) must be greater than 0.1 for electron candidates and smaller than 0.9 for muon candidates.

Intermediate Particles

From these FSPs, candidates for the intermediate particles h' and χ_2 are constructed by combining their respective daughter candidates as shown in Fig. 3.1. As described in Section 3.1, the h' is reconstructed from two muons with opposite charge and the χ_2 from an electron-positron pair. When both particles originate from the same parent particle, it should be possible to find this decay vertex by extrapolating their track. This also allows to restrict the particles' path further by fitting the tracks to exactly intersect. A vertex fit, as this is called, then yields a χ_{prob} on how likely its result is and is performed for both pairs. The selection criteria imposed on them are:

- The decay vertex must originate from the IP by requiring that the radial distance of the decay vertex with respect to the IP $dr < 0.2\,\text{cm}$.
- All candidates with failed fits are rejected.
- At least one of the vertex fits must fulfill $\chi_{\text{prob}} > 0.01$

Veto for π^0

Since the signal's final state does not include any photons, another requirement is the absence of these. However, photons can be produced by beam background, so vetoing the existence of photons directly would also remove signal events. Instead, the production of photons in collision processes is targeted by vetoing the decay $\pi^0 \to \gamma\gamma$. A π^0 occurs often in τ decays like $\tau \to \pi + \pi^0 + \nu_\tau$. To apply this veto, two photons need to be in the event and a π^0 needs to be reconstructed from them. The conditions for selecting for photons are:

- The number of cluster hits in the ECL is greater than 1.5.
- The cluster is located between 17° and 150° in the ECL.
- The reconstructed energy is smaller than $0.25\,\text{GeV}$
- The absolute time difference between the collisions and measurment of the photon in the ECL must be smaller 200 ns.

By combining two of these photons, π^0 candidates are constructed and only candidates with an invariant mass $0\,\text{GeV}\,c^{-2} < m_{\gamma\gamma} < 0.3\,\text{GeV}\,c^{-2}$ are kept. Later, these candidates are used to discard the events.

Rest of the Event

Since the signal is expected to only have four tracks, events with more tracks fulfilling the criteria for FSPs are discarded. Additionally, the energy of all remaining ECL clusters is summed up. All events with more energy than $0.05\,\text{GeV}$ in these clusters are discared.

Additionally, the missing energy four-vector is calculated using the beam energy $\sqrt{s} = 10.58\,\text{GeV}$:

$$E_{\text{miss}} = P_{\text{beam}} - \sum_{i \in \text{FSP}} P_i. \tag{4.2}$$

with the beam four vector $P_{\text{beam}} = (E_{\text{beam}}, 0, 0, 0)$ and the FSPs' four vectors P_i.

4.2.2. Event Selection

For a better understanding of the effect the reconstruction and following selections have on the sensitivity towards the signals, the Punzi Figure of Merit (PFOM) [17] is used as a metric for sensitivity:

$$\text{PFOM} = \frac{\epsilon}{\frac{a}{2} + \sqrt{B}}, \tag{4.3}$$

with the signal efficiency $\epsilon = N_{\text{after selection}}/N_{\text{produced events}}$ and the number of remaining background events B after application of the selection criteria. The parameter a corresponds to the significance level expressed in terms of σ corresponding to one-sided Gaussian tests at a given significance. Its value is chosen as $a = 1$.

After the reconstruction step, signals are selected with a total efficiency between 0.58 and 0.48, depending on the simulated model configurations described in Table 4.2. As Fig. 4.1 shows, the efficiency of the selection is highest for large $m_{h'}$ and m_{χ_2}. There is a drop in

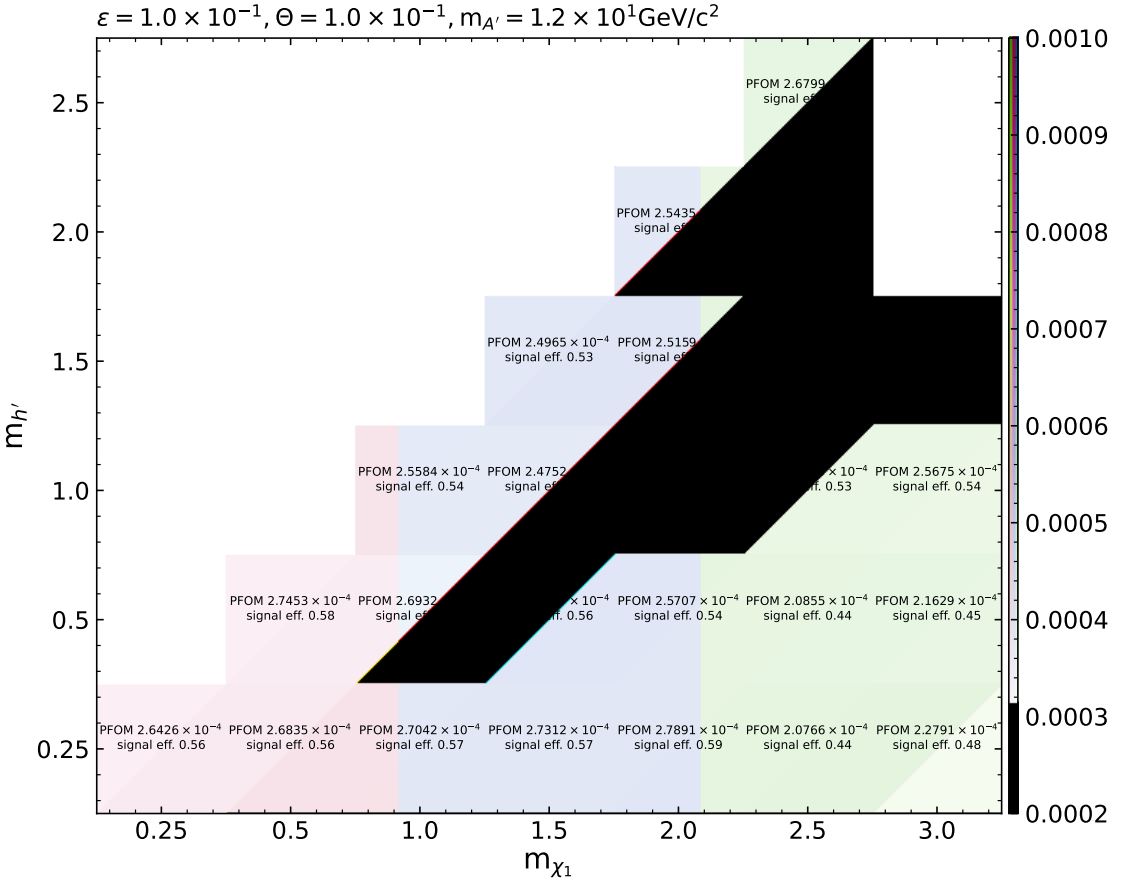

Figure 4.1.: Signal efficiency and PFOM for the signal model parameter configurations after reconstruction and selections applied.

the efficiency for model configurations with a large difference between the masses of the h' and the χ_2.

Vetoing π^0

For the π_0 veto, all the event candidates with $0.1\,\mathrm{GeV}\,c^{-2} < m_{\gamma\gamma} < 1.17\,\mathrm{GeV}\,c^{-2}$ are discarded. Fig. 4.2 demonstrates this selection for background samples and the three exemplary signals. Since only π_0 candidates with a mass close to the nominal π_0 mass

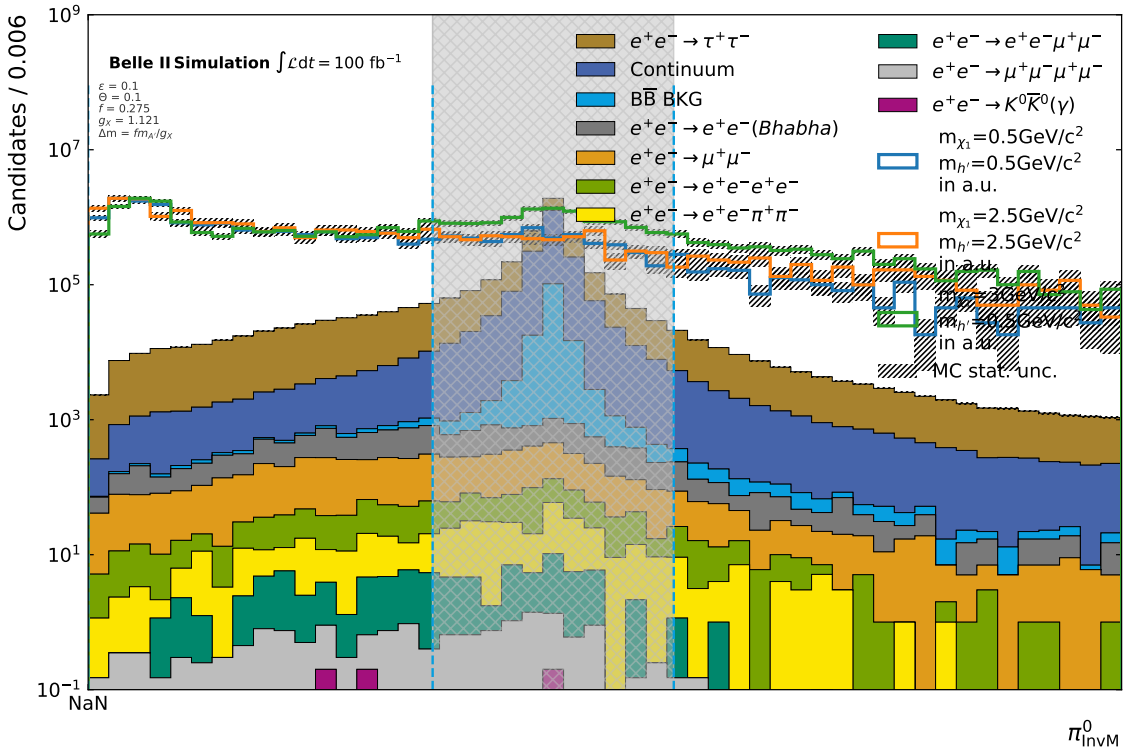

Figure 4.2.: Visualization of the π_0 veto. Signals are scaled to the bin with the highest background. The grayed-out area marks the events discarded. Not all event candidates are in this histogram, as some of them do not have a valid π_0 candidate.

are used, there is a bias towards such candidates, even in the cases where photons come from the beam background. This causes reduced signal efficiency. The loss of efficiency, compared to after the reconstruction, is low for equal m_{χ_1} and $m_{h'}$ ($\approx 1\%$). Towards high mass splittings, it increases. However, this selection also reduces the background and therefore enhances the sensitivity, as the increase of the PFOM in Fig. 4.3 shows.

Missing Energy

A selection on the missing energy is also applied. This selection is motivated by a few events with a very high missing energy in the background samples as shown in Fig. 4.4. These events stem from reconstruction errors and would heavily influence the preprocessing and training process. As can easily be seen from Eq. (5.2), the mean $\mu(x_{\mathrm{unscaled}})$ and standard deviation $\sigma(x_{\mathrm{unscaled}})$ would be heavily influenced by large outliers, which would in turn heavily influence the standardization. Therefore, all event candidates with unrealistic

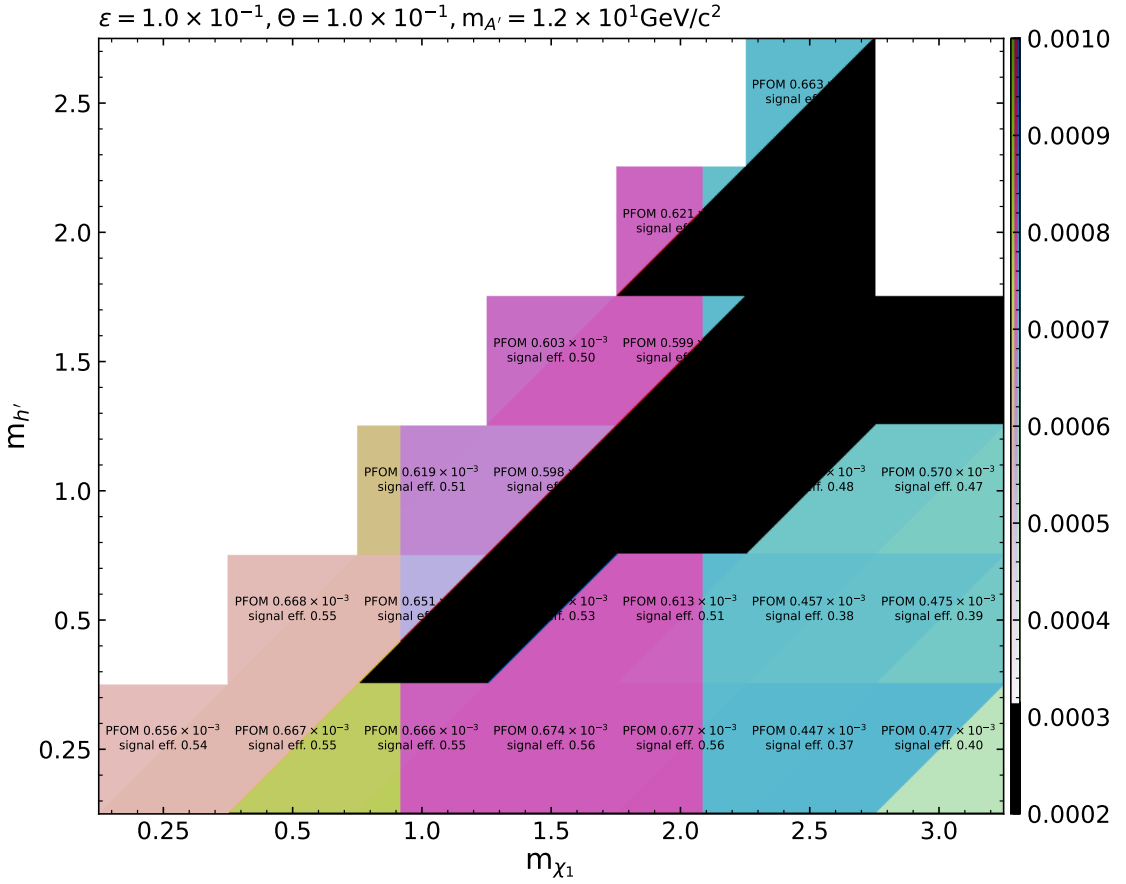

Figure 4.3.: PFOM and efficiency for different masses at fixed angles π_0 veto.

missing energy, $E_{\mathrm{miss}} < 0\,\mathrm{GeV}$ or $E_{\mathrm{miss}} > 10.58\,\mathrm{GeV}$ are removed. Since this selection targets only very few events, efficiency and PFOM barely change.

Best Candidate Selection

The remaining events still can have multiple candidates, when tracks qualify for multiple FSPs. As Fig. 4.6 shows, this affects only a few events: In some, two tracks can be interchanged (i.e. be taken as a muon or an electron) leading to a multiplicity of two. In the other case, two track pairs can be interchanged, giving a multiplicity of 4. Higher multiplicity events are already removed by previous selections. A multiplicity of 3 does not occur, as this would require that the charge of two particles is ambiguous. The candidate with the lowest missing energy is taken as the best candidate. This choice is arbitrary and may be revised. However, as Fig. 4.6 shows, only a few events have multiple candidates and therefore the impact of the selection is small.

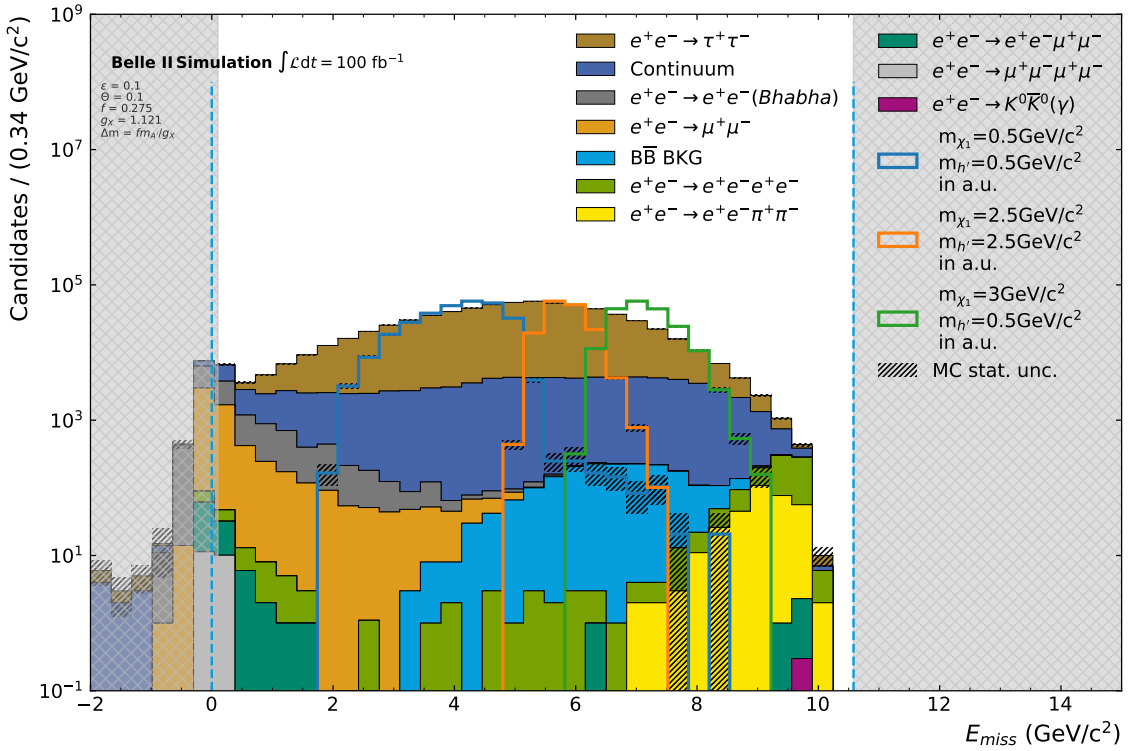

Figure 4.4.: Visualization of the selection on E_{miss}. The grayed-out area marks the events discarded. While the upper limit has no effect, the lower one removes a few events.

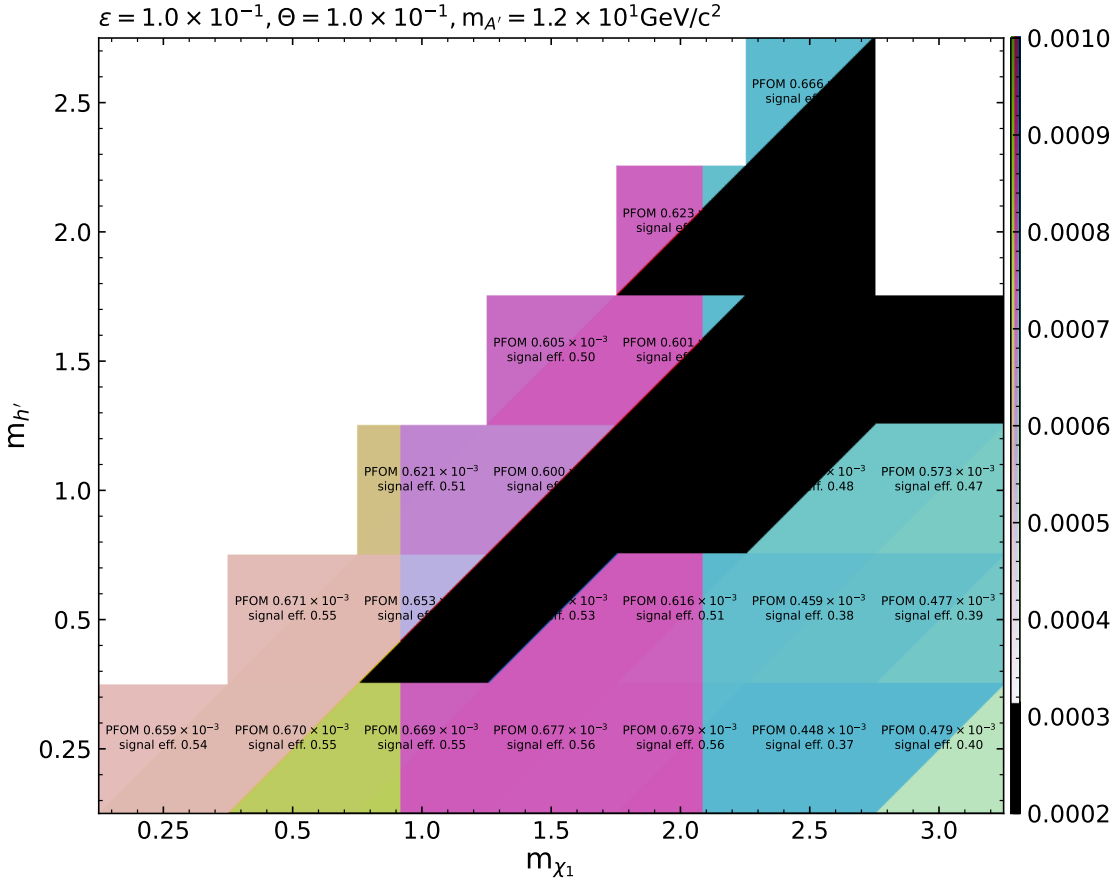

Figure 4.5.: PFOM and efficiency for different masses for the E_{miss} selection.

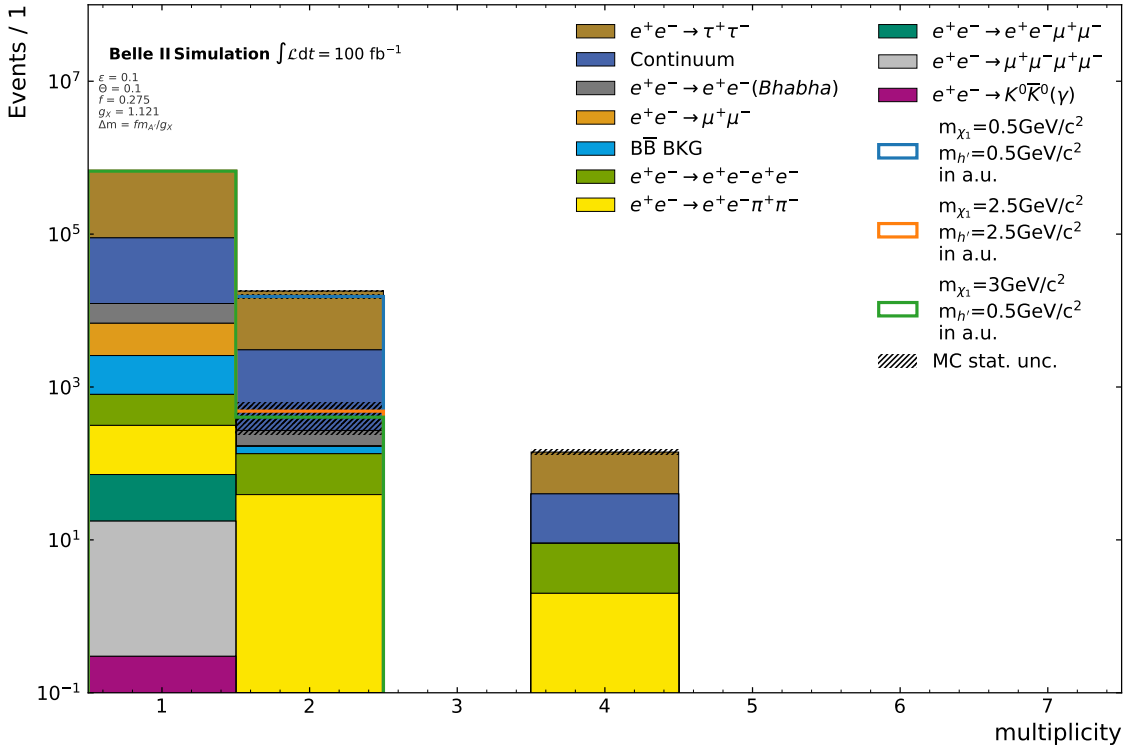

Figure 4.6.: Distributions of multiplicities of event candidates for the background samples and example signals.

Table 4.3.: Event counts and their percentage of the total amount of the background processes. All numbers are normed for $100\,\mathrm{fb}^{-1}$.

process	events	percentage
$e^+e^- \to \tau^+\tau^-$	590702	97%
Continuum	79813	1.2%
$e^+e^- \to \mu^+\mu^-$	4243	0.6%
$e^+e^- \to B\bar{B}$	1811	2.7×10^{-1}%
$e^+e^- \to e^+e^-e^+e^-$	588	0.9×10^{-1}%
$e^+e^- \to e^+e^-$	565	0.8×10^{-1}%
$e^+e^- \to e^+e^-\pi^+\pi^-$	284	0.4×10^{-1}%
$e^+e^- \to e^+e^-\mu^+\mu^-$	54	0.8×10^{-1}%
$e^+e^- \to \mu^+\mu^-\mu^+\mu^-$	17.3	2.6×10^{-3}%
$e^+e^- \to K^0\overline{K}^0(\gamma)$	0.4	0.6×10^{-4}%
$\sum$	678077.7	100 %

4.3. Background Studies

To gain an understanding, the autoencoders need to learn, it is helpful to gain an understanding of which particles and processes make up the background. As an overview, Table 4.3 shows the number of events for each background process with all the previous selections applied. The main contribution stems from $e^+e^- \to \tau^+\tau^-$ events. Small contributions come from continuum events, with the remaining background sources being negligible.

It is possible, that other particles are mistaken for electrons or muons since the qualifications for muons and electrons are not very strict. In MC one can look up, which true particles are causing the track and clusters. This truth-matching process allows assigning a true particle to the particles used as the electron and muon FSP. In some cases, no particle can be assigned. Here the tracks are originating from the beam background or hits from multiple particles are used when reconstructing the tracks.

Fig. 4.7 shows the rates of the different combinations of true particles used to reconstruct the χ_2 and h' decays. For the sake of simplicity, all hadrons are grouped together. This figure shows that most of the background's FSP are pions. This misidentification is driven by the fact, that only the PID information with respect to muons and electrons is used and the likelihoods for pions, and other particles, do not factor in. Using only the binary likelihood between electrons and muons makes no use of likelihoods with other particle hypotheses. This certainly could be used to improve the selection, testing this however was out of the time scope of this thesis.

In consequence, Fig. 4.7 demonstrates, that the background of the IDMDH signals not only includes processes with the same final state but other misidentified particles.

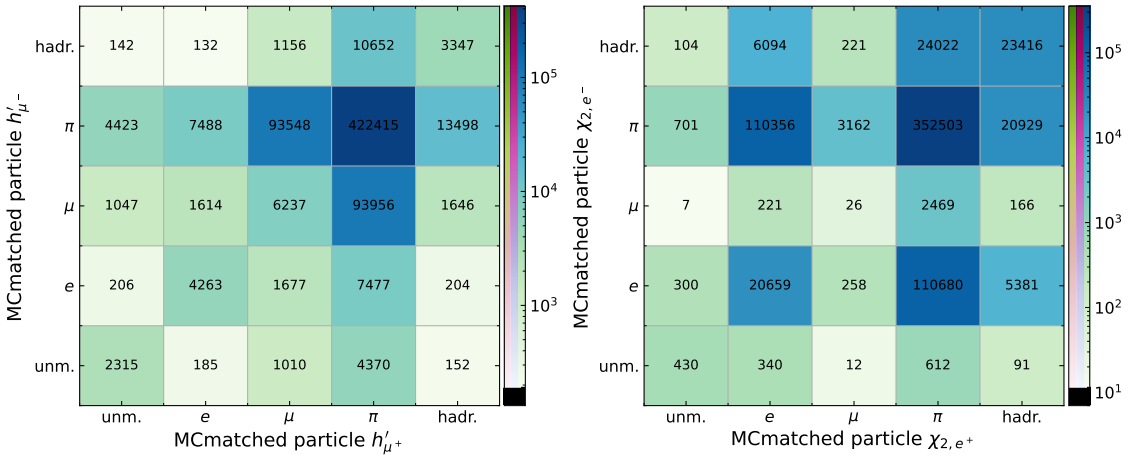

Figure 4.7.: True FSP used to reconstruct h' and χ_2.

5. Machine Learning and Anomaly Detection

In general, anomalies are defined as substantial variations from the norm [3]. This can be understood in two ways: Firstly, anomalies can be data points with properties, the majority of the data does not have. Such anomalies are called out of distribution. The other understanding, and the one used in the context of this thesis, is that of over-densities: While anomalies show properties that can be part of the normal data points, certain properties occur more often than in the rest of the data. As HEP is statistical by the nature of quantum mechanics, this work follows the second understanding of anomalies.

Independent of the understanding of anomalies, there is no generally accepted metric to define anomalies. Different, so-called Anomaly Scores (ASs), can be developed depending on the methods used to analyze the data points.

In recent years the search for anomalies, also called Anomaly Detection (AD), has gained traction as a complementary paradigm in searches for physics beyond the Standard Model (BSM). The rising number of increasingly complex theoretical models for possible BSM scenarios is impossible to cover with dedicated searches for each one of them. Therefore, model-independent searches can identify data, which deviates from SM based predictions. In these regions, more precise and dedicated searches can then be conducted to set limits or extract information on the BSM models.

For these problems, established methods of AD are considered as not well suited. This sparked the LHC Olympics 2020 [4], a challenge held to develop new ansatzes for model-independent searches in HEP. In its course, several methods for AD were developed. the challenge presented a search in dijet events with three DM particles $Z' \to X(\to q\bar{q})Y(\to q\bar{q})$ as a benchmark and test case. For development, there was one data set consisting of background and signal data with corresponding labels. The evaluation was done on three different kinds of Black Boxes; one with a signal similar to the development sample but with different masses, one without any signal, and one with a signal with additional decay modes into three jets. While several submitted models could identify the signal in the first Black Box, they could not make statements about the absence of a signal. Additionally, none could identify the structurally different signal of Black Box 3. The following lessons are taken from the results of the LHC-Olympics:

- A model should indicate the absence of an anomaly.
- The complexity of the signal has a huge impact on the detecting abilities and a model tuned towards one expected kind of signal might be insensitive to a structural different

model.

As a consequence, this work will only attempt to be model-parameter-independent.

While the LHC Olympics focused on newly created algorithms, the use of autoencoders has gained some popularity within the HEP community [1], [18]. Autoencoders are a common tool in AD in several fields as they are rather easy to train. In the following sections their base principles are introduced, several varieties are explained and the training algorithm used in this work is explained.

5.1. Machine Learning and Autoencoders

The broad term Machine Learning (ML) describes a wide range of algorithms, which behaviors are determined by tuning parameters based on example inputs. During this process called training, these examples are passed through the algorithm and a value based on its output is calculated. Then the algorithm's parameters are adjusted to minimize this value using, among others, gradient descent.

5.1.1. Neural Networks

As one kind of ML algorithms Neural Networks (NNs) are inspired by the function of a biological brain. Their smallest building blocks, the neurons, were first described as Rosenblatt-percteptrons [19]. These neurons are simple functions taking inputs and returning a 0 or 1 based on them. In this information processing, each input is multiplied with a weight w_i. All weighted inputs are then summed up and passed through an activation function f mapping the output to 0 or 1.

$$y = f(\sum_i w_i \cdot x_i + b). \tag{5.1}$$

Usually, a bias b is added to the sum. In praxis, a continuous mapping is chosen as an activation function, which must not necessarily be restricted between 0 and 1.

Neurons are ordered in so-called layers. In fully connected layers, the only kind used in this work, each neuron in a layer receives the output of all neurons in previous layers as input. Together these layers make up a NN. These networks are trained to minimize a given metric based on their output. The choice of the metric, also called the loss function, depends on the task the NN is expected to do. To train the NN a process called backpropagation is used. During it, the loss function's gradient for the last layer's weights and biases is calculated. Via gradient descent, they are adjusted and this adjustment is propagated through all previous layers. For computational efficiency, training samples are usually not shown one after the other, but in batches with the loss function being averaged over all batches. There exist several additional algorithms and variations to this concept, some of which are used in this work and described in Section 5.2.

In this work, the `pytorch` [20] library is used for the technical implementation

5.1.2. Prescaling

It is common, to scale all input features to be distributed in the same range of values, usually centered around 0. One simple way to do this is the standardization:

$$x_{\text{scaled}} = \frac{x_{\text{unscaled}} - \mu(x_{\text{unscaled}})}{\sigma(x_{\text{unscaled}})}. \tag{5.2}$$

Here $\mu(x_{\text{unscaled}})$ is the mean of the unscaled feature x over all simulated background samples, and $\sigma(x_{\text{unscaled}})$ is the standard deviation.

5.1.3. Autoencoder Architectures

Basic Autoencoders

The first Autoencoder (AE) were proposed as a tool for dimension reduction [21]. As its core concept, a NN takes input features and puts out a usual lower number of features. This first NN is called the encoder. Taking the lower dimensional representation, a second NN (decoder) produces an output of the same form as the input. The loss function must then reflect the difference between the input to the encoder and the output of the decoder. This loss is also called reconstruction error and different metrics exist to quantify it. In this work the Mean Squared Error (MSE) per event is used:

$$MSE = \sum_{i}^{N} \frac{(x_i - \hat{x}_i)}{N}, \tag{5.3}$$

with the input features x_i, the reconstructed features $\hat{x}_i$, and the number of features N. This is a common metric used to train autoencoders in the context of HEP [1]. A schematic overview of an AE can be seen in Section 5.1.3.

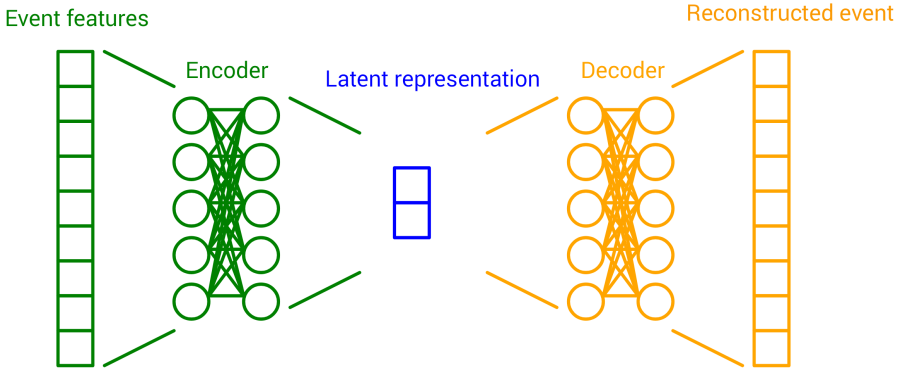

Figure 5.1.: Scheme of an Autoencoder. The input features are compressed into a bottleneck (latent representation, in blue) with an encoder (green). From it, the features are then reconstructed via the decoder (yellow).

With using this setup two possibilities exist to identify anomalies: One way is to use the additional variables from the lower dimensional representation. This reduces the number

of dimensions to detect anomalies in, but AD remains a multi-dimensional problem. The second way to use AEs for AD is to use the reconstruction error. The autoencoder is expected to reconstruct the samples from its training well. Rare, anomalous samples are expected to be reconstructed badly and therefore have a higher reconstruction error. This reduces AD to a one-dimensional problem.

There are several variations of this concept, some of which will be described in the following. To distinguish the basic AE model from the general class of models, the model will be capitalized or abbreviated with AE.

Variational Autoencoder

Basic AEs map each input sample to one point in latent space. However, it does not enforce, that close-by points in latent space also represent similar data. This lack of generalization was targeted for improvement when Variational Autoencoders (VAEs) was introduced [22]. Instead of learning deterministic representations, a Gaussian distribution for each sample is learned. Therefore, for each sample a mean $\mu = (\mu_1, \mu_2, ...\mu_N)$ and a variance $\sigma^2 = (\sigma_1^2, \sigma_2^2, ...\sigma_N^2)$ is learned. Both define a Gaussian distribution

$$g(z) = \frac{1}{\sqrt{\sigma^2(2\pi)^N}} e^{-\frac{1}{2}\frac{z-\mu}{\sigma^2}}, \tag{5.4}$$

from which a sample z is pulled. This sample is then used as a latent representation and to reconstruct the input. Section 5.1.3 shows a schematic overview. Since the variance is always

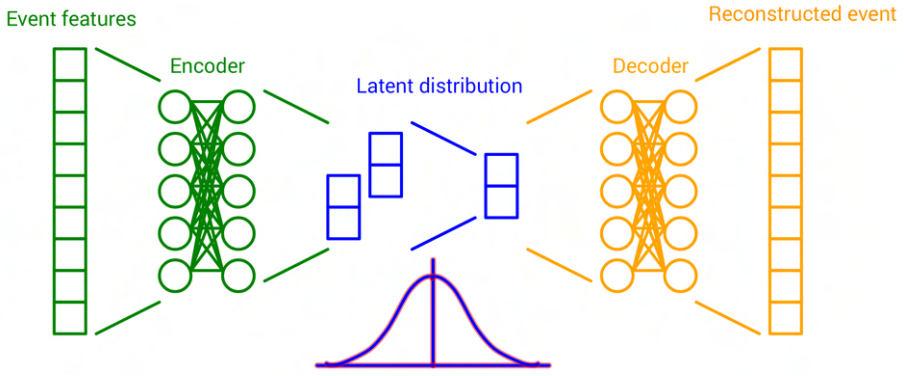

Figure 5.2.: Scheme of a Variational Autoencoder. The decoder now has a $2 * L$ dimensional output from which half is interpreted as a mean and the other as a variance.

positive, in praxis its logarithm logvar $= \log\left(\sigma^2\right)$ is learned by the encoder. Therefore, negative values are interpreted as small variances. Additionally, the overall distribution is regularised to be a standard gaussian ($\mu = 0, \sigma = 1$), via the Kullback-Leibler Divergence (KLD) [22]:

$$D_{KL}((N)(\mu, \sigma)|(N)(0, 1)) = -0.5 \sum_{l}^{L} (1 + \log\left(\sigma_l^2\right) - \mu_l^2 - \sigma_l^2), \tag{5.5}$$

with the gaussian distribution (N), the number of latent dimensions L, the latent mean μ_l and the latent variance σ_l. The total loss function that has to be minimized per event becomes

$$\mathcal{L} = MSE + \beta D_{KL} = \sum_{i}^{N} \frac{(x_i - \hat{x_i})}{N} - \beta \cdot 0.5 \sum_{l}^{l} (1 + \log\left(\sigma_l^2\right) - \mu_l^2 - \sigma_l^2). \tag{5.6}$$

The hyperparameter β gives weight to the regularising term and is set to 0.1 in this work.

Dirichlet Variational Autoencoder

Dirichlet Variational Autoencoder (DVAE) are an extension of VAE where instead of a gaussian distribution a Dirichlet distribution is used. The implementation in this work follows [1], where a Dirichlet distribution is approximated by applying a softmax function in the latent space. Section 5.1.3 shows the schematics of an DVAE. A Dirichlet distributions

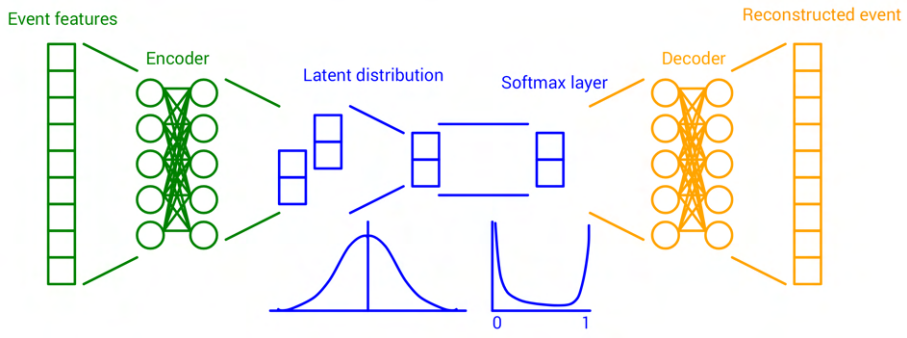

Figure 5.3.: Scheme of a Dirichlet Variational Autoencoder. After sampling from a gaussian distribution, a softmax function is applied. The results are then passed to the decoder.

probability function has the form

$$\mathcal{D}_\alpha(z) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i^L \Gamma(\alpha_i)} \prod_i^L r_i^{\alpha_i - 1}, \tag{5.7}$$

with the hyperparameters $\alpha_i > 0$, the number of latent dimensions L, and the Γ function as the extension of the factorial function to non-integer values. The distribution can be interpreted as a distribution over different possibilities. For example, consider a particle that could be either a e, μ, or a π: one can assign probabilities to each hypothesis, e.g. 30% to be an e, 40% to be a μ, and 30% to be a μ. For each of these triplets, the Dirichlet distribution assigns a probability for this combination being true. From this example, it is also clear that only triplets which sum up to 100% are allowed. So the distribution is only defined on an L-dimensional simplex.

For the approximation via a softmax function on a gaussian distribution

$$\mathcal{D}_\alpha(r) \approx \text{softmax}(\mathcal{N}(z, \tilde{\mu}, \tilde{\sigma})) \tag{5.8}$$

with the softmax function

$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_j^N e^{x_j}} \tag{5.9}$$

the parameters become

$$\tilde{\mu}_i = \log \alpha_i - \frac{1}{L} \sum_i^L \alpha_i \tag{5.10}$$

and

$$\tilde{\sigma}_i = \frac{1}{\alpha_i}(1 - \frac{2}{L}) + \frac{1}{R^2} \sum_i^L \frac{1}{\alpha_i}. \tag{5.11}$$

With that, the regularisation term in the loss is the KLD of the learned Gaussians to the imposed prior defined by the hyperparameter α_i:

$$D_{KL}(\mathcal{N}(\mu_l, \sigma_l)|\mathcal{D}_\alpha(r)) = \frac{1}{2} \sum_i^L (\frac{\tilde{\sigma}_i^2}{\sigma_i^2} + \frac{(\tilde{\mu}_i - \mu_i)^2}{\sigma_i^2} - L - \log \frac{\sigma_i^2}{\tilde{\sigma}_i^2}). \tag{5.12}$$

With the choice of the hyperparameter α, a hierarchical structure can be given to the latent space. Following the reasoning in [1], anomalous samples should be pushed towards variables with a lower weight. Since it is not clear what categories will be encoded in the latent space, a naive choice for these hyperparameters is done:

$$\alpha_i = \begin{cases} 0.1, & \text{for } i = L - 1 \\ 0.9, & \text{else} \end{cases}. \tag{5.13}$$

This gives $L - 1$ categories the same weight and creates on rare category ($\alpha_L = 0.1$). Of course, other choices can be made, making this a possibility for further exploration.

5.2. Training

Training the various versions of autoencoders brings several challenges in terms of ensuring convergence and training time. Namely, convergence can fluctuate heavily between epochs and might not happen at all. To tackle these issues, cross-validation, weight averaging, gradient clipping, and learning rate scheduling are employed. As some of these measures also slow down convergence and prolong training time, early stopping and a best epoch selection are implemented.

Cross-validation

The first measure used to stabilize the training is cross-validation. For this, in each epoch, the data set is split into five parts, called folds. Four of these are used to perform the backpropagation, while the other one is used for validation. Within an epoch, each of the folds is used for validation once, while the others are used for backpropagation, so each data point is used four times for backpropagation within an epoch. For each epoch, the data is shuffled and new folds are defined. This procedure prevents biases in the training data, which could arise from a fixed split between training and validation data sets. While it might seem unlikely in a case with a rather big data set of around 68 thousand points, tests showed, that cross-validation still stabilized convergence. This is likely due to the nature of the data, which contains a high proportion of nonphysical events caused by beam background and errors in the reconstruction algorithm. In this work, the `scikit-learn` [2] `KFold` class is used.

Weight Averaging

To further ensure smooth convergence, stochastic weight averaging is used. In this, backpropagation does not change the model's weights and biases within an epoch. Instead, the weights and biases of each backpropagation are accumulated, and only at the end of the epoch, they are averaged and applied to the model. By doing that the stability of the training is further improved by averaging out the adjustments on the weights and biases. However, this also slows down the convergence and more epochs are needed until convergence is reached. While this method was vital during test runs on smaller data sets (10^6 events) during the design phase of the model, the final data set might not need this measure. The effect of removing weight averaging on training time and convergence was not studied due to time constraints. For the technical implementation, the `torchcontrib` [20] `SWA` class was used.

Gradient Clipping

Another measure taken to avoid instabilities in training is gradient clipping. During each backpropagation, the norm of the gradient is limited to 0.001. This also contributes to longer training times as it limits the changes made per epoch. The implementation uses the *pytorch* [20] *clip_grad_norm_* function.

Early Stopping and Best Epoch Selection

To ensure no longer training than needed and get optimal results a best epoch selection in combination with early stopping is employed. As the best epoch, the epoch with the lowest validation loss is considered. If there is no new best epoch for more than 15 epochs the training is stopped.

Learning Rate Scheduling

Learning rate scheduling is adapted to yield better training results and stabilize the training in its later stages on the one hand, but also to reduce the number of epochs needed to reach convergence. For this *pytorchs* [20] *ReduceLROnPlateau* scheduler is used. It monitors the validation loss and reduces the loss by a factor of 0.1 when there is no relevant change for a set number of 10 epochs. As an irrelevant change, a relative difference smaller than 10^{-4} to the best epoch is considered.

6. Training Analysis

For the training, all background samples described in Section 4.3 are used.

Additionally, weights in the loss calculation are needed to reflect the discrepancy between the actual number of events of a background process and its expected fraction in the experiment.

The input features are each FSP's four-vector and the missing momentum and missing energy of the event. In total, these are 20 input features. Across all autoencoder architectures, the hyperparameters described in Table 6.1 are used. These hyperparameters were chosen based on experiments using only a small subset of the training samples. For the AE and VAE architectures models with latent space dimensionality 1 to 10 are trained. Since for the DVAE, the one-dimensional case is not defined, architectures with latent space dimensions 2 to 10 are trained.

During the training, each epoch's average total loss, MSE, and regularising terms for VAE and DVAE are reported. Additionally, each epoch's learning rate is tracked to monitor the learning rate scheduler.

6.1. Basic Autoencoders

A summary of the training results is provided in Table 6.2. With an increasing number of latent dimensions the reconstruction improved. This improvement is expected since more dimensions allow for an easier encoding of event features. As a downside, higher numbers

Table 6.1.: Overview of the hyperparameters fixed across all autoencoders.

Hyperparameter	Value
Encoder architecture	3 layers, 100 neurons each
Decoder architecture	3 layers, 100 neurons each
starting learning rate	10^{-5}
Batch size	256
maximum epochs	500

Table 6.2.: Summary of training results for latent dimensions 1-10 for the AE.

Latent dimensions	Training time in h	avg. MSE of best epoch
1	1.6	0.7
2	17	0.3
3	22	0.22
4	26	0.14
5	23	0.1
6	27	0.08
7	28	0.06
8	25	0.04
9	21	0.028
10	27	0.013

of latent space dimensions tend to have longer training times. However, since the training was performed on different GPUs, the training times in Table 6.2 only give tendencies.

Fig. 6.1 shows the trainings progress for the 1-dimensional AE. Until epoch 9 the loss function decreases. After that, an increase is observed which triggers the learning rate scheduler. This then stalls the training until its termination conditions are reached. Training higher dimensions qualitatively shows the same behavior but takes longer. As Fig. 6.2 shows, the learning rate scheduling shows an effect in these cases: The drop in the learning rate before epoch 400 causes a slight drop in the loss function. This follows from the smaller steps in adjusting the weights and biases the smaller learning rate causes. With the higher learning rate, the training oversteps the minimum in the loss function, while smaller steps can get closer to the minimum.

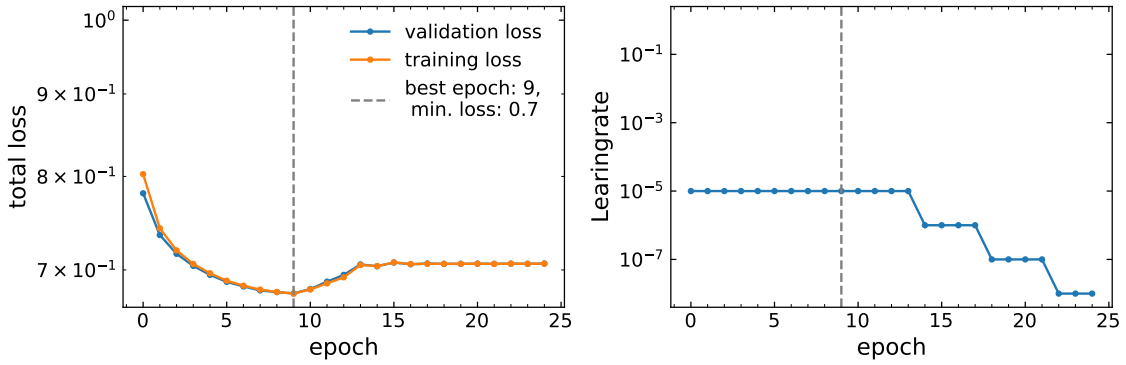

Figure 6.1.: Training details for the training of an AE with one-dimensional latent space. The total loss is equal to the MSE.

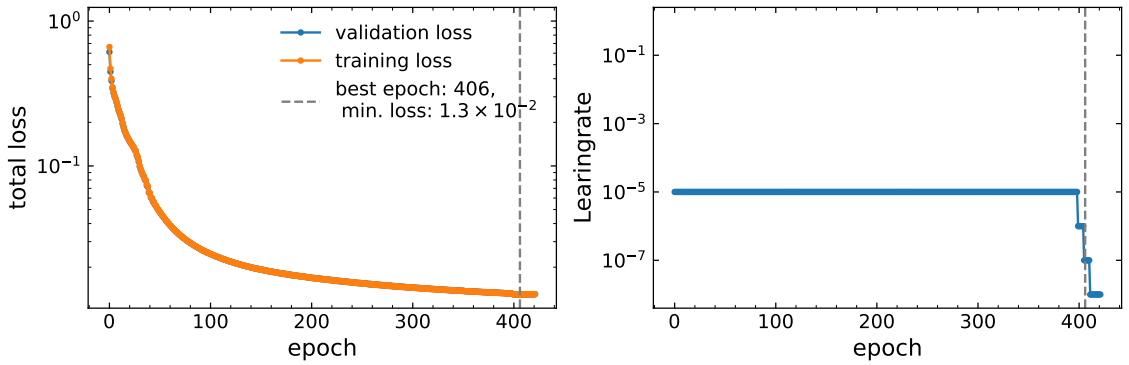

Figure 6.2.: Training details for the training of an AE with 10-dimensional latent space. The total loss is equal to the MSE.

Table 6.3.: Summary of training results for latent dimensions 1-10 for the VAE.

Latent dimensions	Training time in h	avg. loss of best epoch
1	2.1	0.9
2	8	0.8
3	9	0.7
4	7	0.7
5	9	0.7
6	7	0.7
7	9	0.7
8	8	0.7
9	8	0.8
10	8	0.7

6.2. Variational Autoencoder

In the same manner as the AE, the training of the VAE can be analyzed. A summary of the training results is provided in Table 6.3. Like for the AE, the 1-dimensional model converges fast and shows the loss on a plateau after some epochs. Again this is not changed by the learning rate scheduler. This can be seen in Fig. 6.3.

The 8-dimensional VAE (Fig. 6.4) shows fluctuations of the loss functions during training but still converges. Something similar can be observed in the 9-dimensional case, but no other training.

As Table 6.3 shows, all models with more than one latent dimension, end around the same loss values. The suspected reason for this is the choice of $\beta = 0.1$. This choice gives the KLD term a higher weight and thereby incentivizes the VAE to structure the latent space before minimizing the MSE. As a consequence, the encoder structures the latent space into a Gaussian distribution without encoding information useful for the reconstruction. This effect can be observed in Fig. 6.5. This plot shows the latent variables of a 4-dimensional VAE with their correlations.

Testing other values for β could not be done in the scope of this thesis.

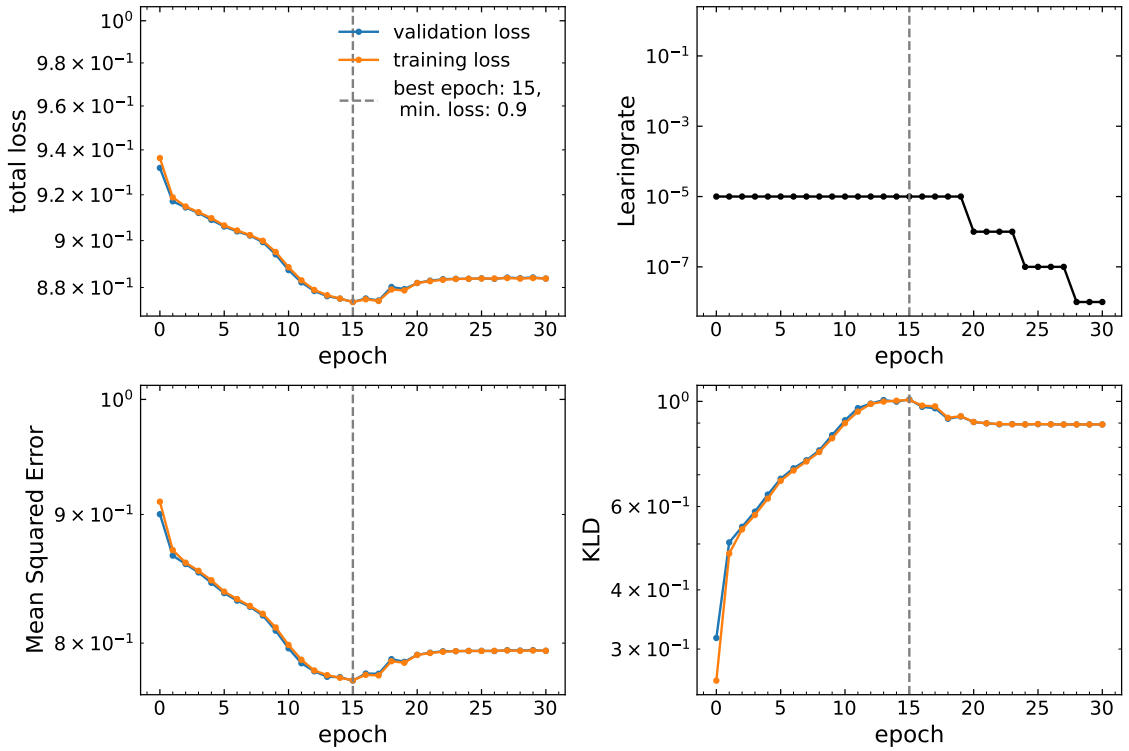

Figure 6.3.: Training details for the training of an VAE with one-dimensional latent space (top). The total loss is equal to the MSE + 0.1 × KLD. The two constituents of the loss are shown in the lower row of the plot (bottom).

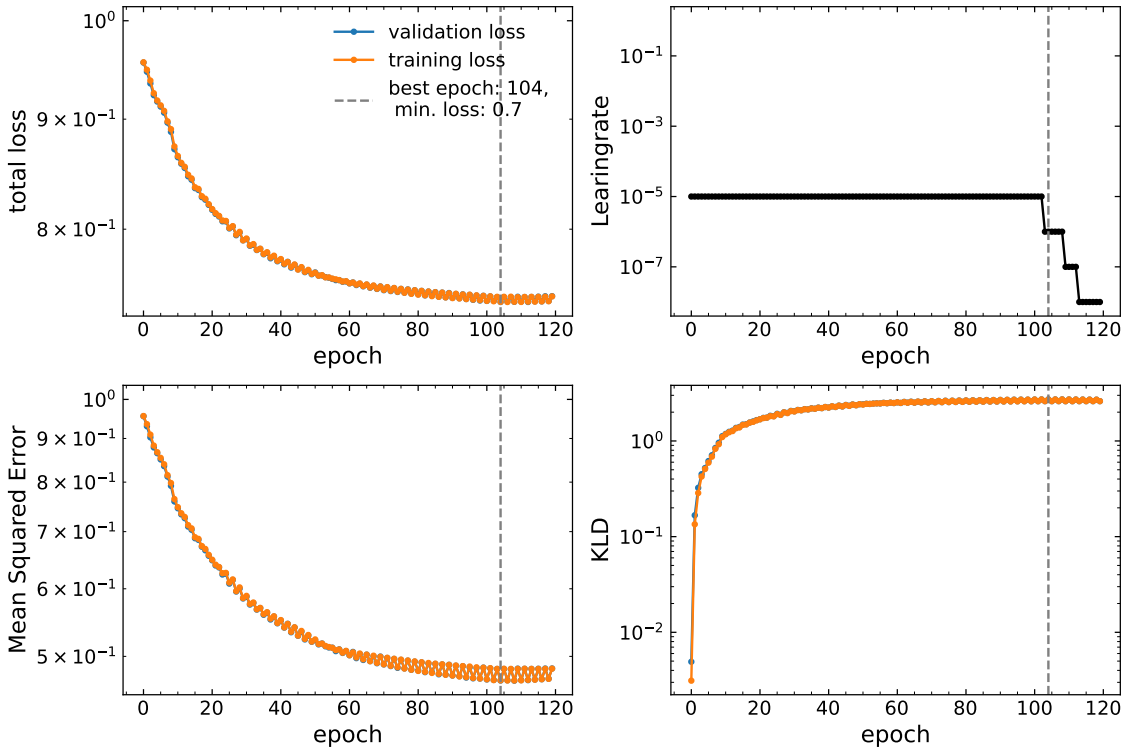

Figure 6.4.: Training details for the training of an VAE with 8-dimensional latent space (top). The total loss is equal to the MSE + 0.1 × KLD. The two constituents of the loss are shown in the lower row of the plot (bottom.

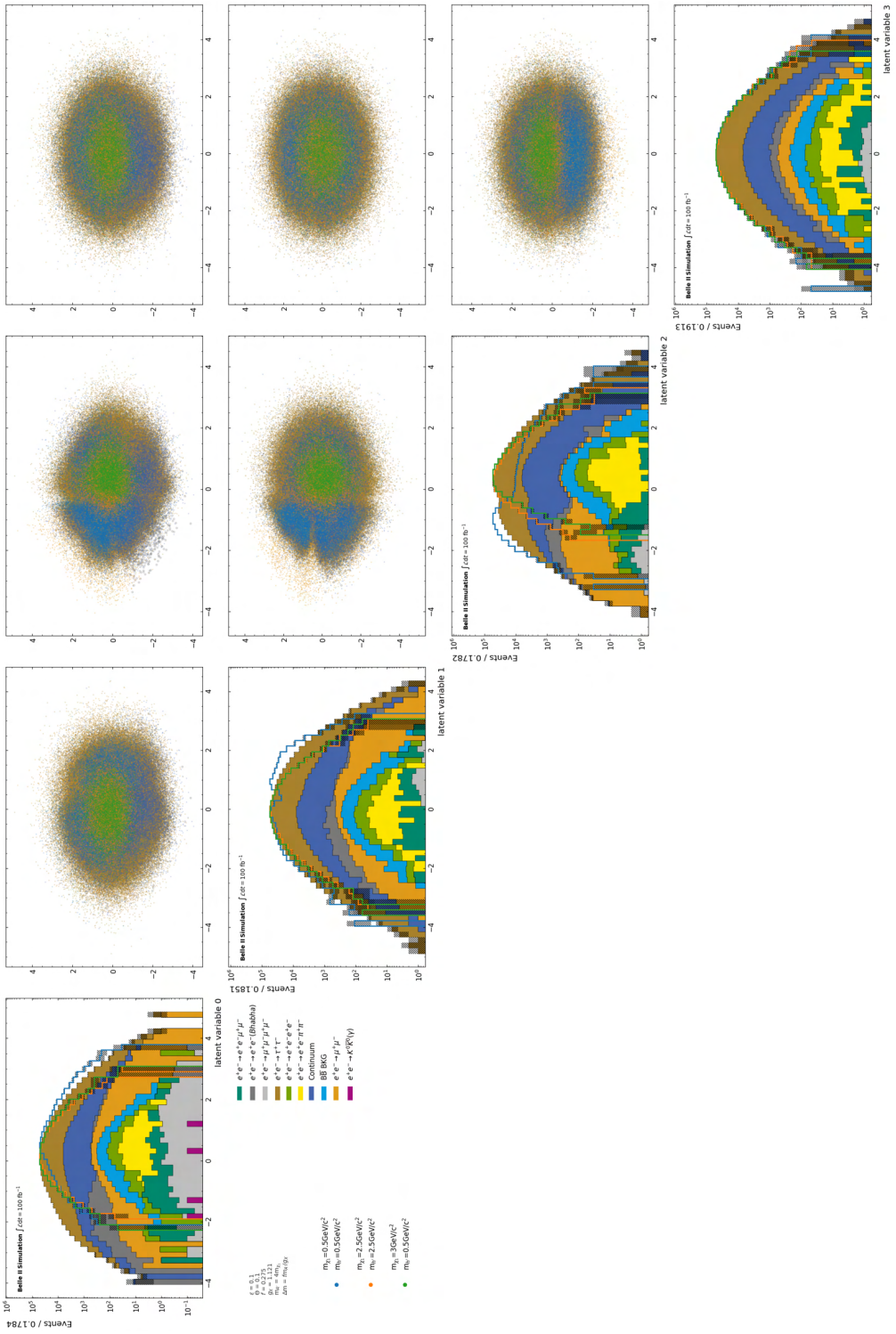

Figure 6.5.: Latent variables and their correlation of the 4-dimensional VAE.

Table 6.4.: Summary of training results for latent dimensions 2-10 for the DVAE.

Latent dimensions	Training time in h	avg. loss of best epoch
2	8	0.7
3	18	0.4
4	21	0.29
5	20	0.28
6	30	0.2
7	30	0.15
8	30	0.12
9	28	0.09
10	40	0.08

6.3. Dirichlet Variational Autoencoder (DVAE)

Training evaluation

Like in the case of the AEs (Section 6.1), the DVAEs show an increase in training time and a decrease in the loss of the best epoch with an increasing number of latent dimensions (Table 6.4). As can be seen in Fig. 6.6, training for 10 latent dimensions already approaches the maximum number of epochs. Here, even training for more epochs might be possible, as

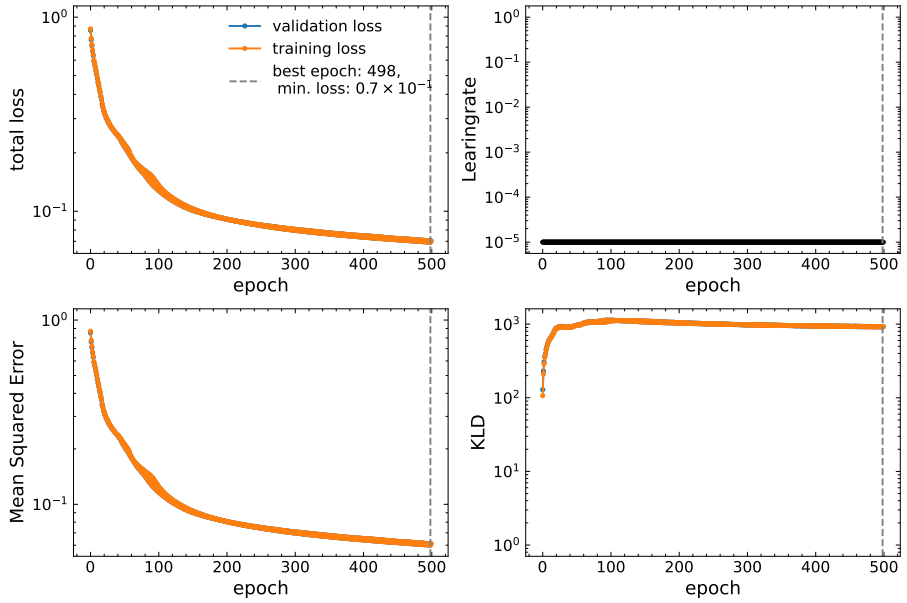

Figure 6.6.: Training details for the training of an DVAE with 10-dimensional latent space. The total loss is equal to the MSE.

indicated by the fact that the learning rate scheduler did not take any action. Testing this, however, was not possible within the time frame of this thesis.

7. Detecting Anomalies

As discussed in Chapter 5, the MSE is expected to be higher for samples the autoencoder was not trained on. So, it should be possible to extract signal events by selecting events above a given MSE threshold. But not all anomalies are equally anomalous. Autoencoders learn to replicate an event by compressing them into lower dimensions. This compression means, that some information is lost and only some features are encoded. Signals sharing these features therefore might be equally well reconstructed than the background. The features that are encoded heavily depend on the number of latent dimensions. This is supported by the lower MSE for more latent dimensions shown in Chapter 6. So, it is expected, that the MSE will vary for different model parameter configurations and the number of latent dimensions. In the following three sections, this connection is explored for the three autoencoder architectures AE, VAE, DVAE.

7.1. Basic Autoencoder

As discussed in Section 5.1.3, the reconstruction error MSE between the input of an autoencoder and its output is an intuitive choice as AS. Since all models were trained on only background samples, it is expected, that the signal samples show higher MSE values when passed through the AEs. Fig. 7.1 shows these distributions for the simple case of a 1-dimensional AE for the example signals defined in Section 4.1. The figure shows, that some example signals, namely the one representing high masses and high mass differences, have a similar low MSE as the background. Only in the case of light masses, a distinctive peak is formed. As a consequence, using the MSE as AS will not show the same sensitivity toward all model parameter configurations.

In comparison, the MSE distribution for a 10-dimensional AE (Fig. 7.2) shows fewer distinctive differences for the signals. All example signals show lower MSE than in the 1-dimensional case. Also, the example with small $m_{h'}$ and m_{χ_2} does not have a distinctive peak.

To evaluate the potential sensitivity of a model towards the signals, the PFOM from Eq. (4.3) is used. The efficiencies in these calculations are with respect to the number of events after the selections described in Section 4.2. Therefore, only the sensitivity of the autoencoder is evaluated. The PFOM is calculated for selections using different MSE values as selection criteria. The granularity in which it is evaluated is calculated using the 1%-tile and 99%-tile of the MSE distribution. The values between both ends are split into 100

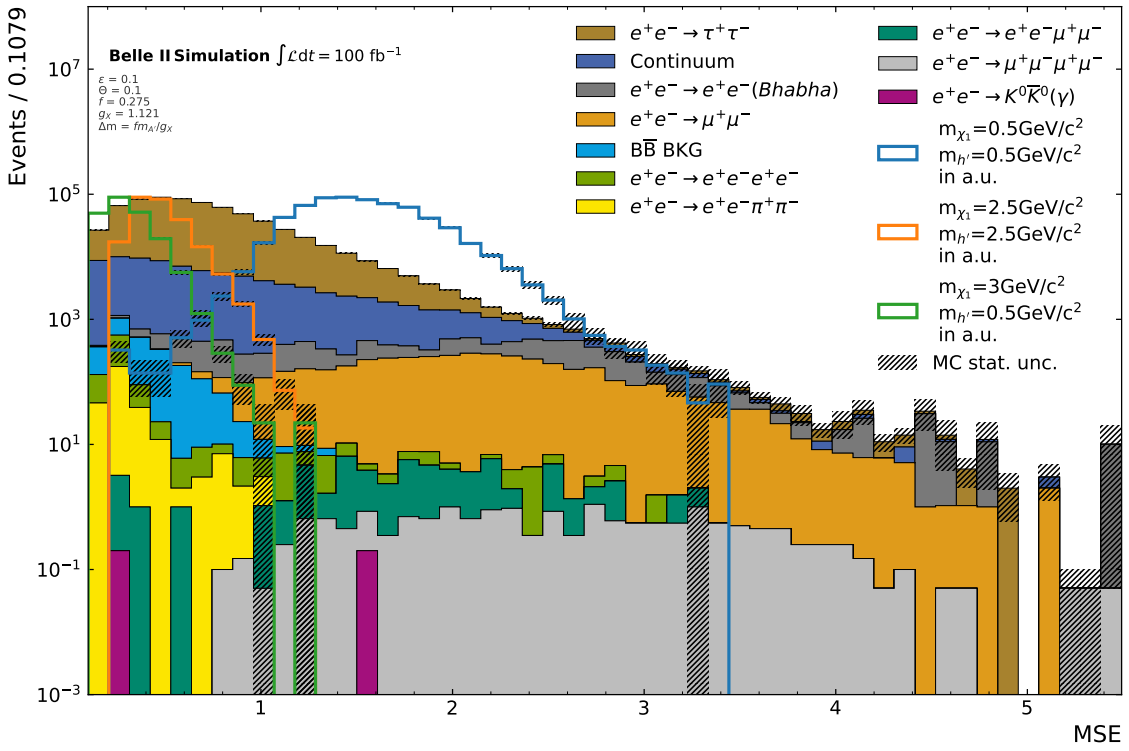

Figure 7.1.: Distribution of the MSE of the background samples and the example signals for a 1-dimensional AE.

evenly-sized steps. Using this stepsize, the values between the minimum and maximum MSE found in the background samples are probed. This procedure is chosen to be robust towards outliers while still having fine enough bins to capture the changes in the sensitivity. While this is not as relevant with the one-dimensional AE, it becomes more relevant in models with more latent dimensions. As can be seen in Fig. 7.2, the region of low MSE values have a high event density, while outliers are still present. Simply using the maximal and minimal values and splitting these at equidistant values might not capture differences of the signals.

For each of the signal examples, the PFOM is evaluated, leading to the graphs in Fig. 7.3 in the case of the one-dimensional AE. Clearly, the example signal with high mass splitting does not gain sensitivity from the use of this model. However, the sensitivity for the other examples can be improved with it.

To get the optimal sensitivity for each of the signals, the selection can be chosen at the point of maximal PFOM for each signal. This approach does in a sense drop the premise of a model parameter-independent search. However, the problem of searching for multiple possible parameters is reduced to optimize the selection for just one variable. Performing this PFOM optimization for all simulated signals yields Fig. 7.4. This plot confirms the trend of higher sensitivity towards low masses of this model. This tendency, however, should be taken with some caution, since some training samples have a cut-off for low masses as discussed in Section 4.1. For all signals with $m_{\chi_1} > 2\,\mathrm{GeV}\,c^{-2}$, this model does not show

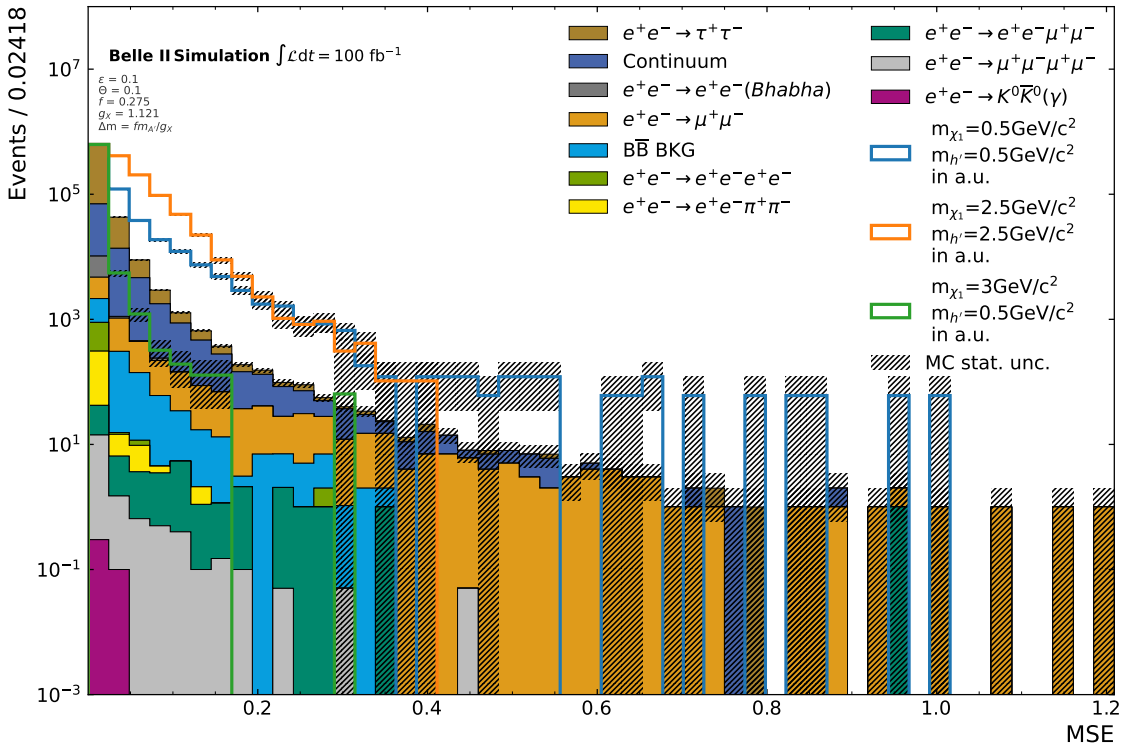

Figure 7.2.: Distribution of the MSE of the background samples and the example signals for a 10-dimensional AE.

any significant sensitivities. The selection efficiencies drop under 1%, lowering the number of remaining signal events below the threshold for statistical significance.

This optimization is repeated for every AE. For the AE with 10 latent dimensions, this leads to Fig. 7.5. This model shows a sensitivity towards signals with high masses, while there is no sensitivity for lower masses. With that in mind, it becomes clear that different-sized latent spaces show different capabilities to detect signals. To get an overview of where different AE have their sensitivities, the example signals are used. For each of these the maximal PFOM value is plotted over the number of latent dimensions in the model in Fig. 7.6. This shows, that no model has any sensitivity for signals with a high difference between the m_{χ_1} and $m_{h'}$. For the high mass examples, the sensitivity rises until 8 latent dimensions. It does not lose as much sensitivity as the example with small masses. Training models with even higher dimensions might reveal even higher sensitivities in this area. The example for small masses shows the highest sensitivity for all models, except the 10-dimensional one. Based on this figure, the 8-dimensional AE seems to be the optimal choice. The optimization results for all signal samples are in Fig. 7.7. This figure shows the very high sensitivity for low-mass configurations. Configurations with high mass differences are undetectable for this model, as with any other one. Comparing high mass configurations to Fig. 7.5, some configurations, like $m_{h'} = 1.5\,\mathrm{eV}\,c^{-2}$, $m_{\chi_1} = 1.5\,\mathrm{eV}\,c^{-2}$ show less sensitivity than the 10-dimensional AE.

To visualize the effect, Fig. 7.8 shows the h' mass distribution for the background samples

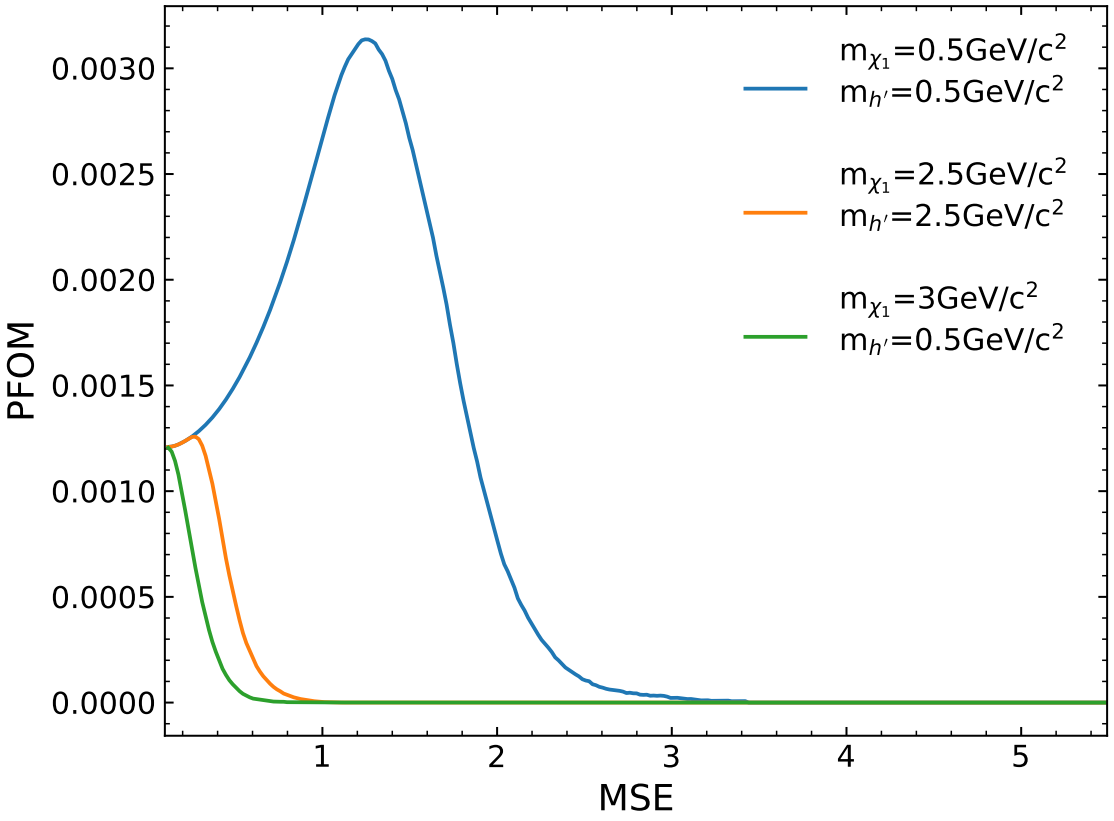

Figure 7.3.: PFOM over MSE of the 1-dimensional AE for the three example signals.

and the three example signals. The left figure shows the distributions after the selections discussed in Section 4.2. For the right figure, the 8-dimensional AE is used. The selection on the MSE is optimized towards the example signal for small masses.

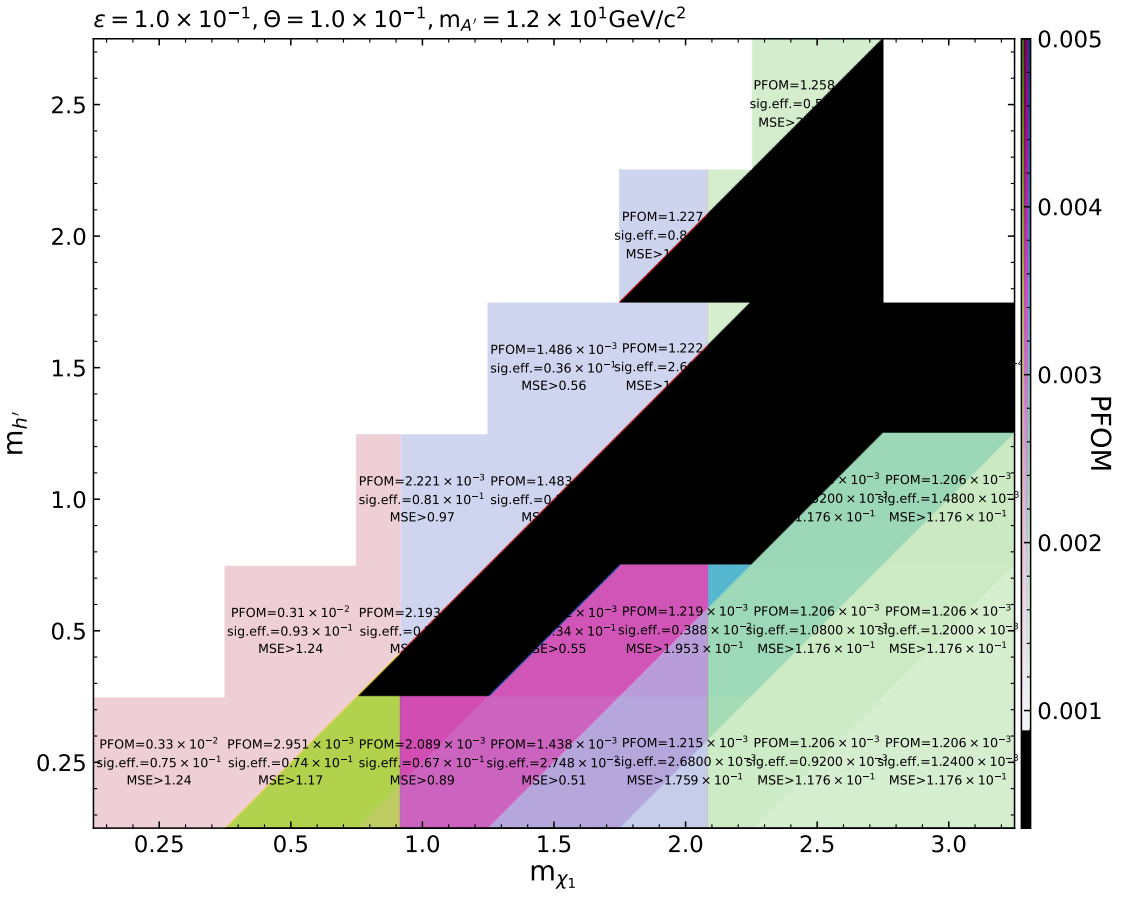

Figure 7.4.: PFOM, signal efficiency and MSE value of the selection for each simulated signal after the PFOM optimization for a 1-dimensional AE.

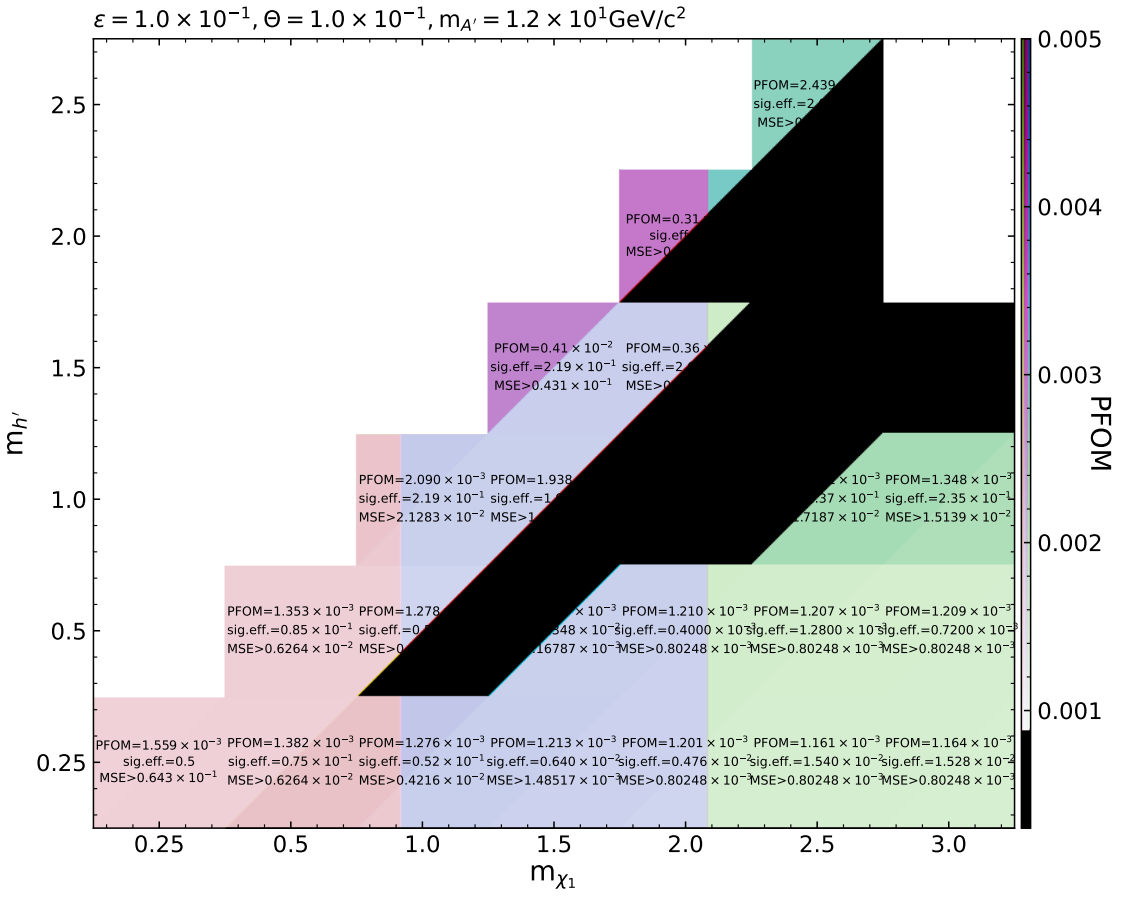

Figure 7.5.: PFOM, signal efficiency and MSE value of the selection for each simulated signal after the PFOM optimization for a 1-dimensional AE.

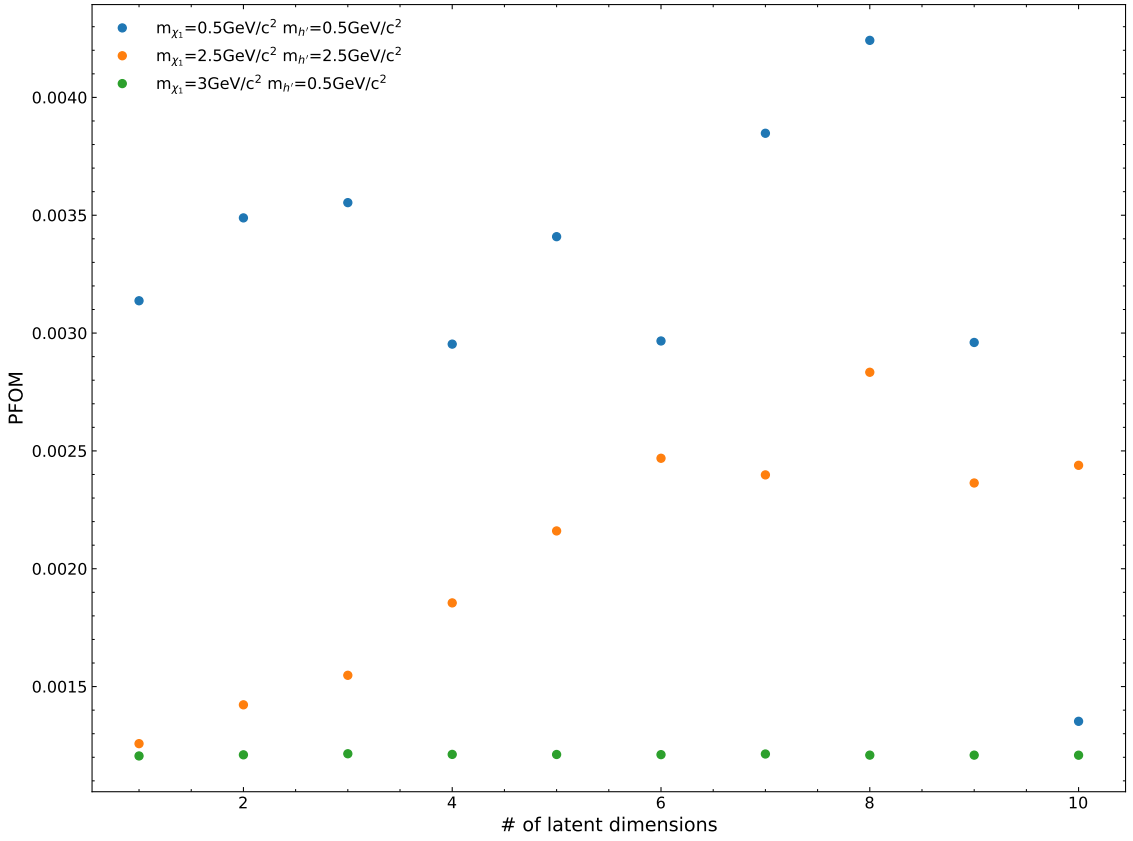

Figure 7.6.: Maximal PFOM of the example signals over the number of latent dimensions for AE.

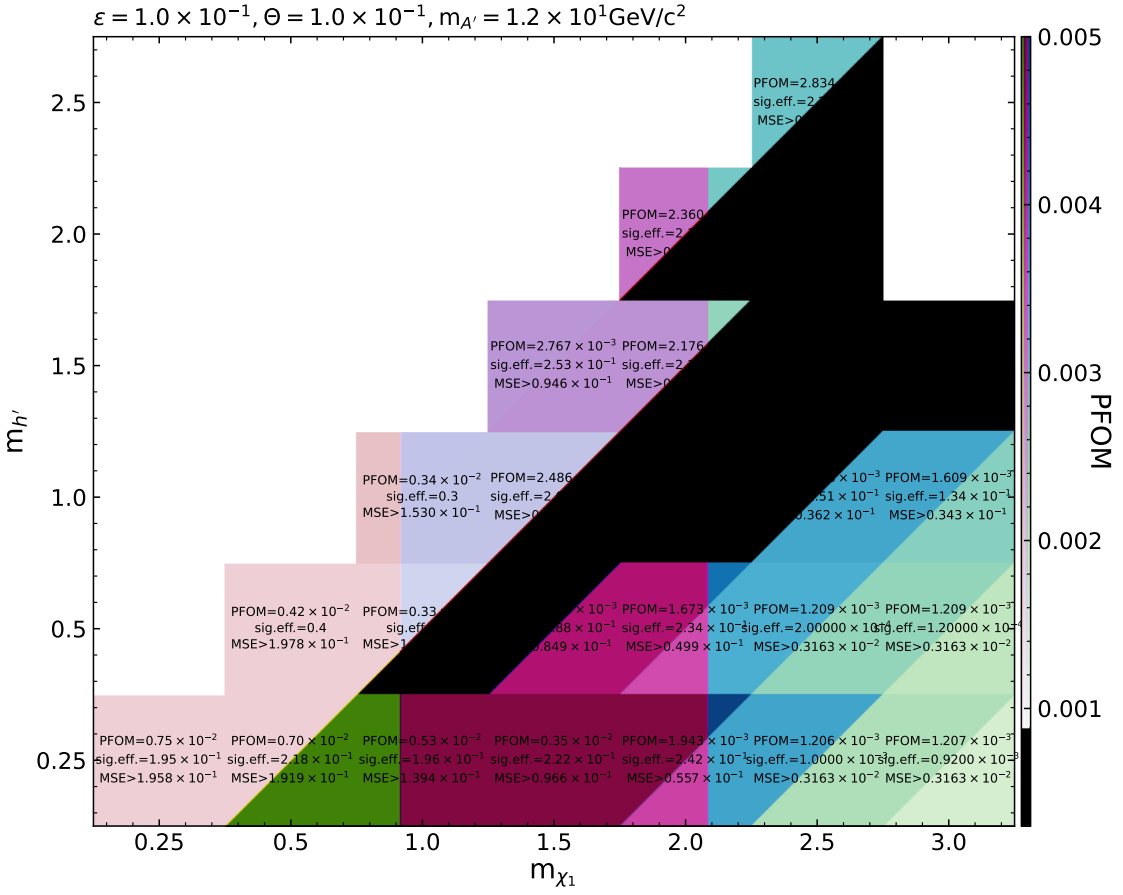

Figure 7.7.: PFOM, signal efficiency and MSE value of the selection for each simulated signal after the PFOM optimization for a 8-dimensional AE.

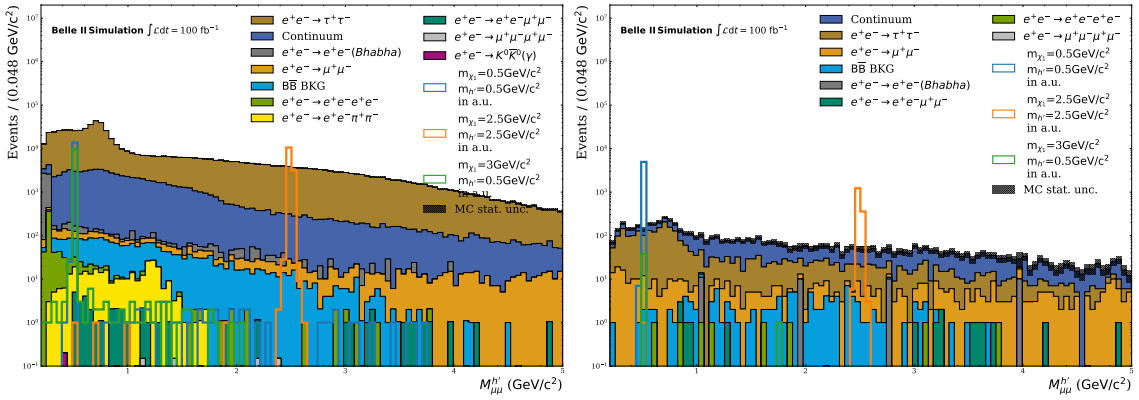

Figure 7.8.: Comparison of the invariant mass of the h' candidates before and after the selection $MSE > 0.1978$.

7.2. Variational Autoencoder (VAE)

Starting with the comparison of the maximal PFOM values for the example signals in Fig. 7.9, it becomes clear, that the VAE architecture shows little promise. Again, the low

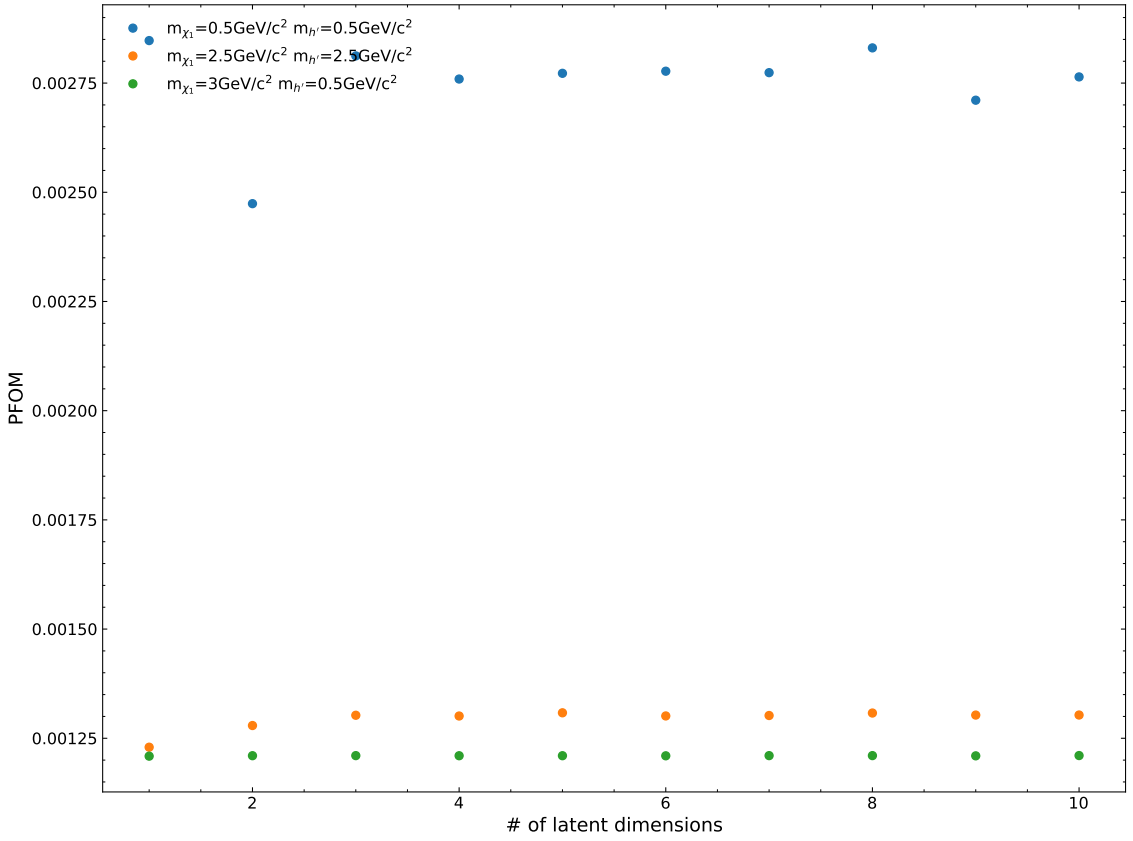

Figure 7.9.: Maximal PFOM of the example signals over the number of latent dimensions for VAE.

mass example shows the highest sensitivity. However, since the MSE changes little with the latent dimensions, the sensitivities also do not change. With the sensitivity for the heavy example rising until 3 latent dimensions, this could be considered as the optimal VAE. As Fig. 7.10 shows, there is only significant sensitivity in the low-mass regions. Even there, the sensitivity does not exceed that of the AE models.

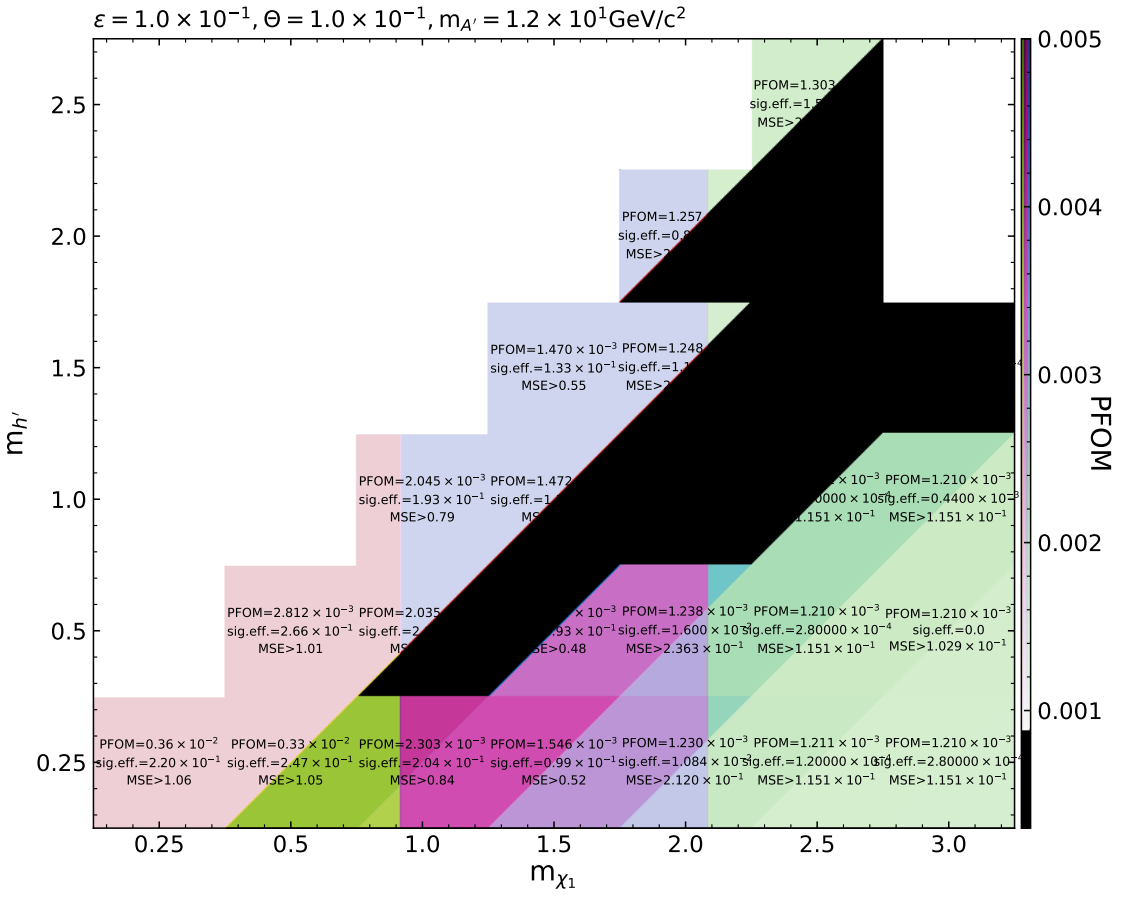

Figure 7.10.: PFOM, signal efficiency and MSE value of the selection for each simulated signal after the PFOM optimization for a 3-dimensional VAE.

7.3. Dirichlet Variational Autoencoder (DVAE)

Again, the maximal PFOM of the example signals can be used as a starting point to analyze the models. As Fig. 7.11 shows, with higher numbers of latent dimensions the sensitivity for the high mass example increases. This suggests the 9-dimensional DVAE as the optimal

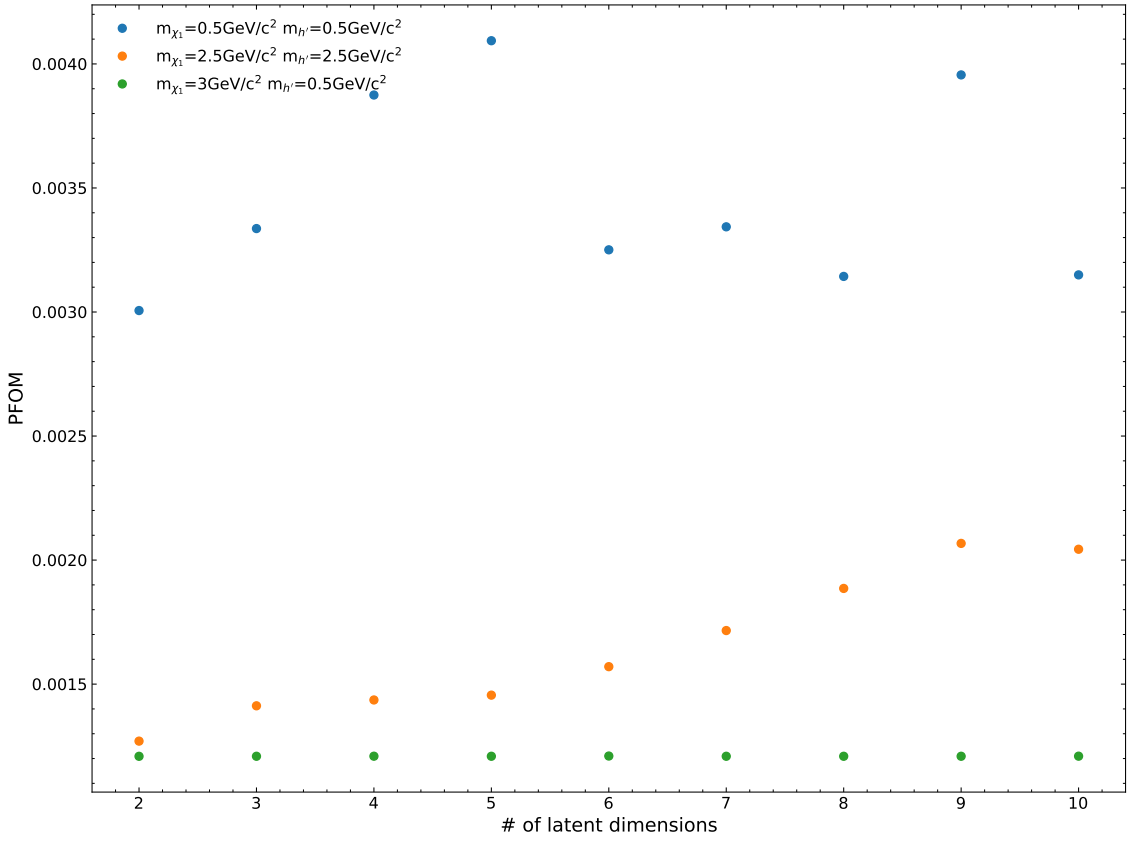

Figure 7.11.: Maximal PFOM of the example signals over the number of latent dimensions for DVAE.

model. Fig. 7.12 shows the results of the PFOM optimization for all signal. It shows a higher sensitivity towards low masses and now towards high mass differences. Also, some sensitivity for high-mass configurations can be observed. Qualitatively, this picture does not change across the different-sized latent spaces. When comparing these results, with the AE, especially the 8- and 10-dimensional one, the DVAE is not more sensitive for any configuration.

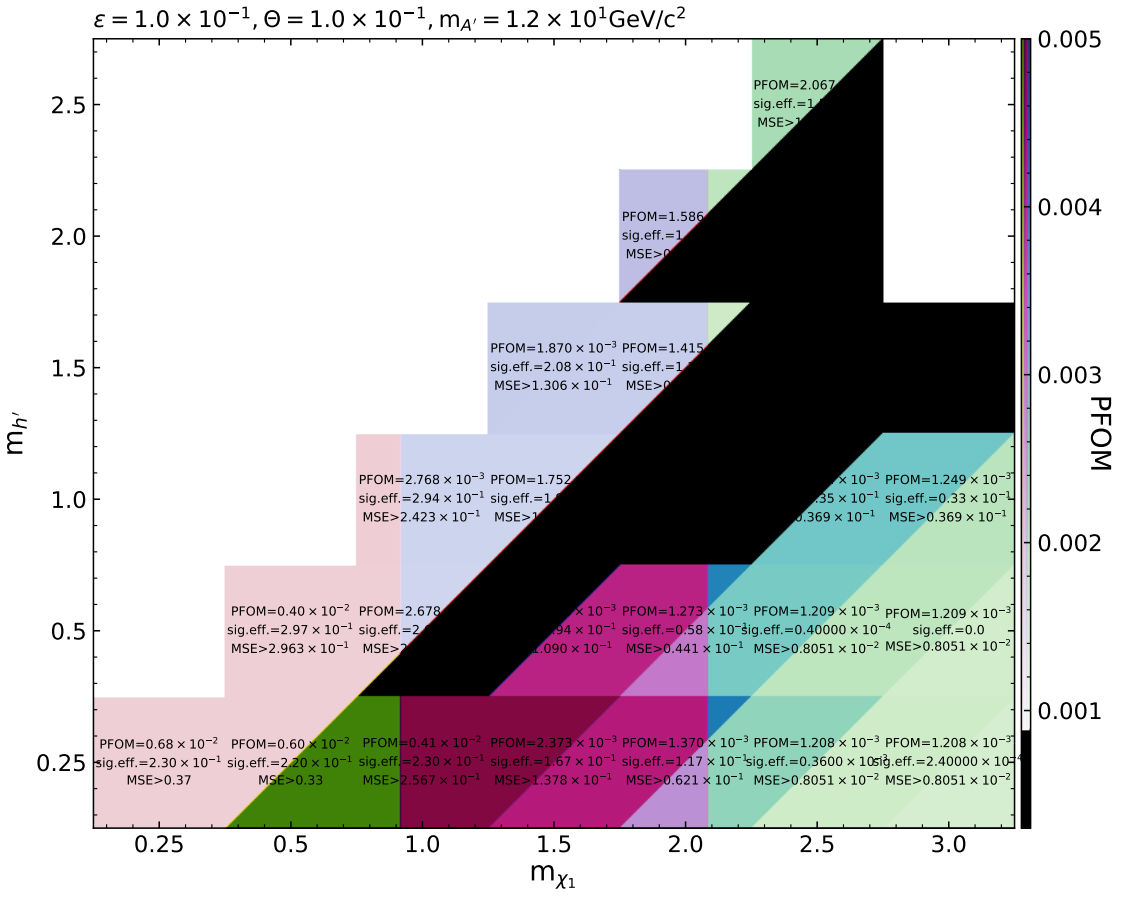

Figure 7.12.: PFOM, signal efficiency and MSE value of the selection for each simulated signal after the PFOM optimization for a 9-dimensional DVAE.

7.4. Anomaly Detection in the Latent Space

The fact, that neither the DVAEs nor the VAEs are worse at detecting anomalies via the MSE is not surprising. While the AE is only optimized for this metric, the others additionally give a structure to the latent space. As such it is also interesting to use the latent space to identify anomalies. Compared to the classical selection-based search this would reduce such a search to fewer variables. The tradeoff for that is, that these variables do not have a physical interpretation. Conducting such a search for multiple models is beyond the scope of this thesis. Therefore, only a few interesting cases are studied.

Gaussian Latent Spaces

The latent spaces of the VAEs are regularised to be a Gaussian distribution. As discussed in Section 6.2 all VAEs with more than 1 latent dimension behave equivalently. Therefore, the 3-dimensional case is chosen arbitrarily. Its latent space is visualized in Fig. 7.13.

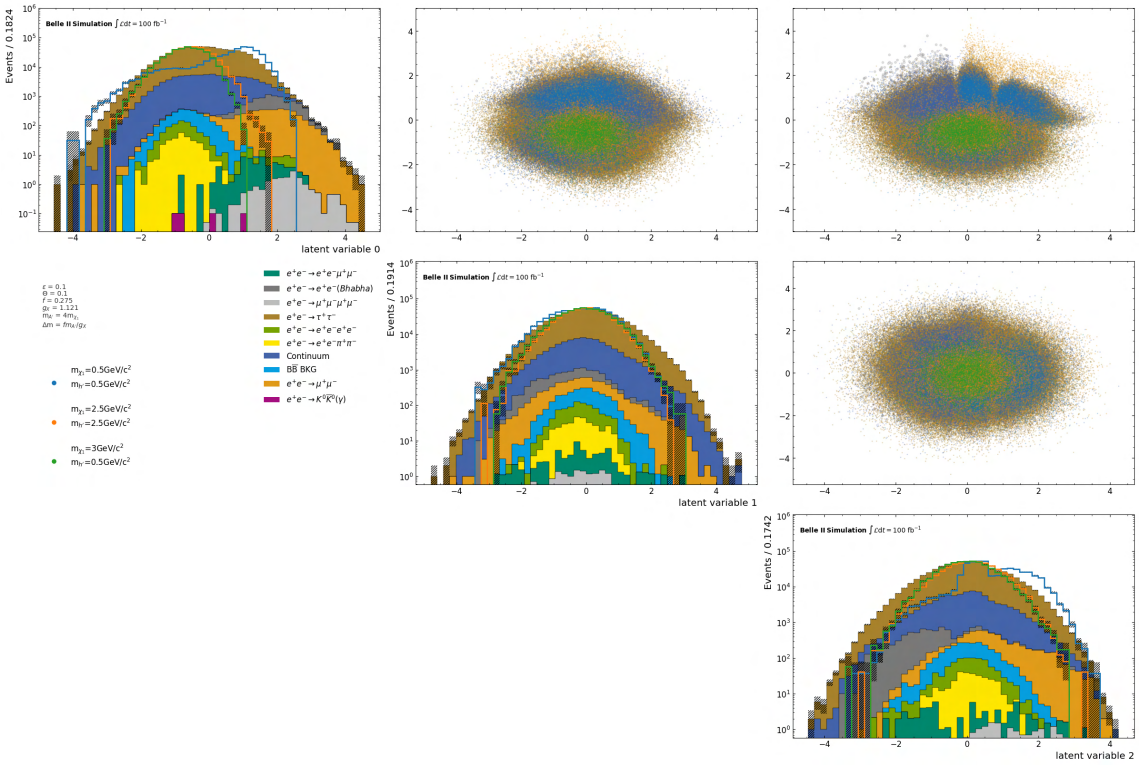

Figure 7.13.: Latent space of the 3-dimensional VAE. On the diagonal, the distributions of the latent variables are shown. In the scatter plots, the correlations between two variables are shown.

In only two of the three variables, a structure of the background samples can be observed. E.g. latent variable 2 shows a separation between the $e^+e^- \to \mu^+\mu^-$ and $e^+e^- \to e^+e^-$ samples. Still, a concrete interpretation of these variables is not possible. Looking at the signal, only a little difference to the background can be seen. One reason for this is the choice of β as discussed in Section 6.2. A too-high value of β gives precedence to organizing

the latent space. This prioritizes the construction of a Gaussian distribution in the latent space over encoding information for the reconstruction. Therefore, the VAE do not learn features of the training data but place inputs into a gaussian distribution. As a consequence, anomalies are also placed into a gaussian distribution without regard to their features.

Dirichlet Latent Spaces

Unlike the AEs and VAEs, the DVAEs give some kind of interpretation. As discussed in Section 5.1.3, the prior hyperparameter α is chosen such, that the last latent variable can represent the rare events. This makes its interpretation as a 'degree of anomaly' somewhat intuitive. In the simple case of just two latent variables, the latent space looks like presented in Fig. 7.14.

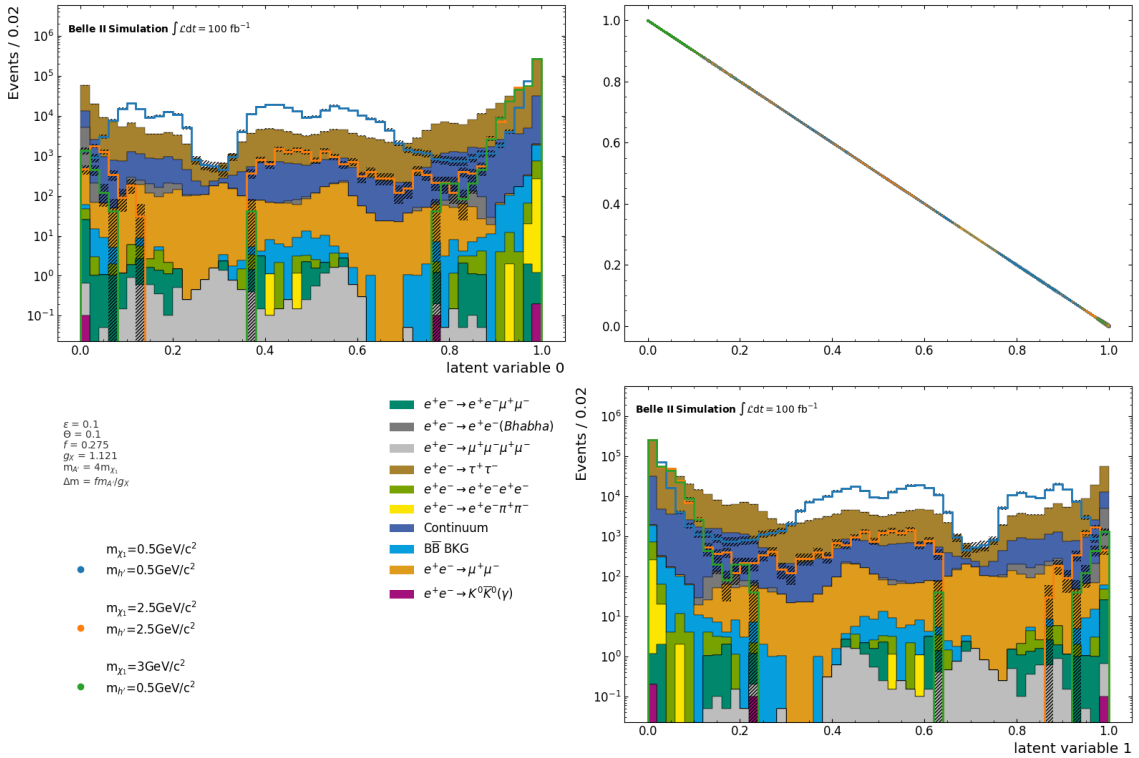

Figure 7.14.: Latent space of the 2-dimensional DVAE. On the diagonal, the distributions of the latent variables are shown. The scatter plots show the correlations between two variables. For the 2-dimensional DVAE, this is per definition a line.

Even though a strong hierarchy ($\alpha = (0.9, 0.1)$) is used, anomalous events are not pushed towards one edge. Moving to higher dimensional latent spaces, more interesting cases arise. With the categorical interpretation of the latent space, the PFOM optimization method used for the MSE in Section 7.1 can be adapted by simply switching the MSE for the latent variable. With this method, every single latent variable can be tested as an AS. Fig. 7.15 shows the results for the example signals for all latent variables of the 10-dimensional DVAE.

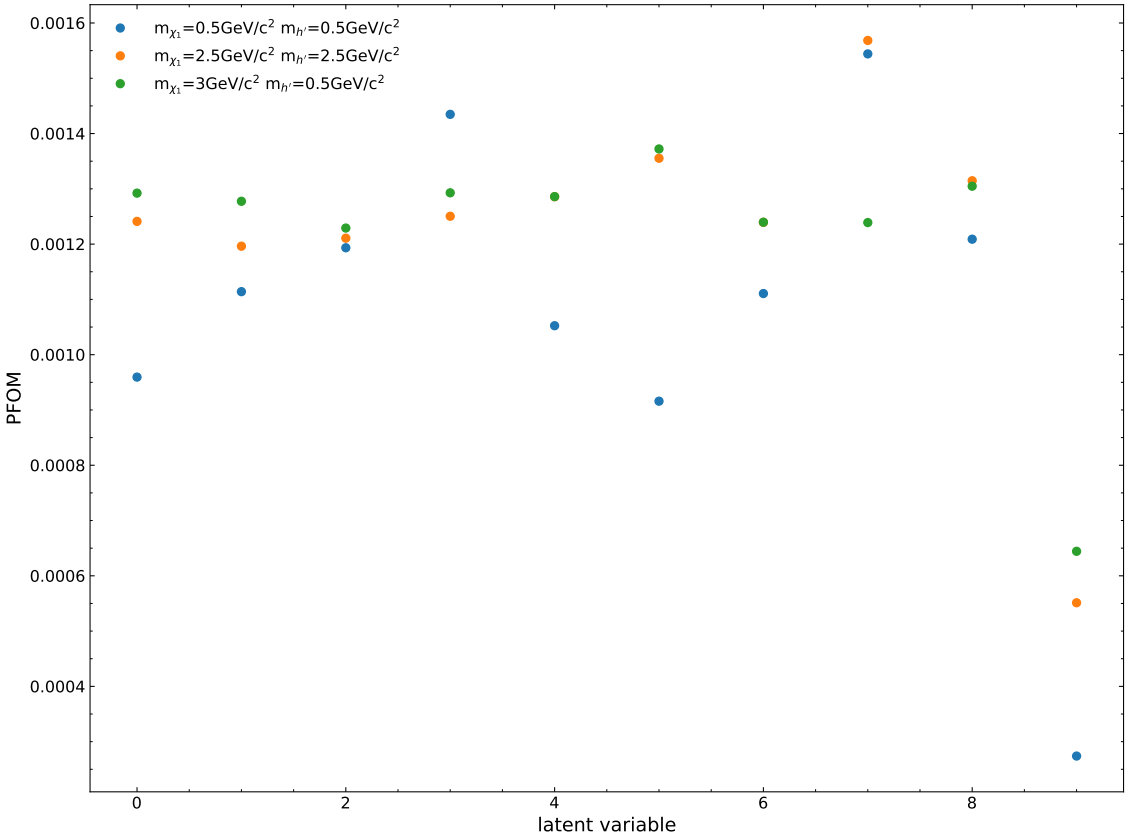

Figure 7.15.: Results of the PFOM optimization for the example signals for the latent space of the 10-dimensional DVAE.

The latent space shows only small sensitivities across all example signals. This demonstrates the complexity of such high-dimensional latent spaces, even if they are structured. Using different priors α might improve the performance, though it is not clear what values for α would accomplish that.

8. Validation

Until now, all training and analyses were based on simulated samples. But simulations only approximate the real data and the behavior of the autoencoders could differ when applied to real data. Mismodeling aspects of the data could bias the autoencoder: Since it has not seen these aspects, it can not encode or decode them. In turn, this will lead to higher MSEs values for data samples, which then wrongfully suggests anomalies. There are two possible ways to handle this problem:

Training on Data

One could train directly on data. This would remove all possible mismodeling of the simulation, but then potential anomalies would also be included in the training samples. This would change the premise of AD with autoencoders. Where before, unseen samples were expected to have a higher MSE, now rare samples are expected to have a higher MSE. This still is a reasonable assumption but does require some validation. Training the autoencoders with increasing numbers of signals injected would give a limit on how rare a signal could be until the autoencoder learns to reconstruct it. Since the sensitivity studies presented in Chapter 7 show dependencies of the MSE on the model parameters, this test might have to be repeated with multiple model parameter configurations. Another, more model-parameter-independent way, would be to select side-band samples. Such samples would be rejected by the selection described in Section 4.2. But since the selection of the training samples mostly aims to reject misreconstructed events and beam background, there are no side bands left.

Applying to Data

The second option for validation is, to use the autoencoders trained on MC samples and apply them on data. This requires that data and simulation are in good enough agreement, that the autoencoders behave the same for MC and data. The latent space delivers appropriate variables for this test, as the latent variables capture, what the encoder learns to encode. As a test sample, data with an integrated luminosity of $\int \mathcal{L} = 1.03\,\mathrm{fb}^{-1}$ is processed as described in Section 4.2 and passed through the AE with 8 latent dimensions. Additionally, a correction for the particles track momenta is applied [23]. Up until now, no measures were taken, to ensure that any of the simulated events used for training are actually triggered. For the following test, all data and simulated events are required to be triggered by the HLT and the L1-Trigger for 3 full tracks (fff).

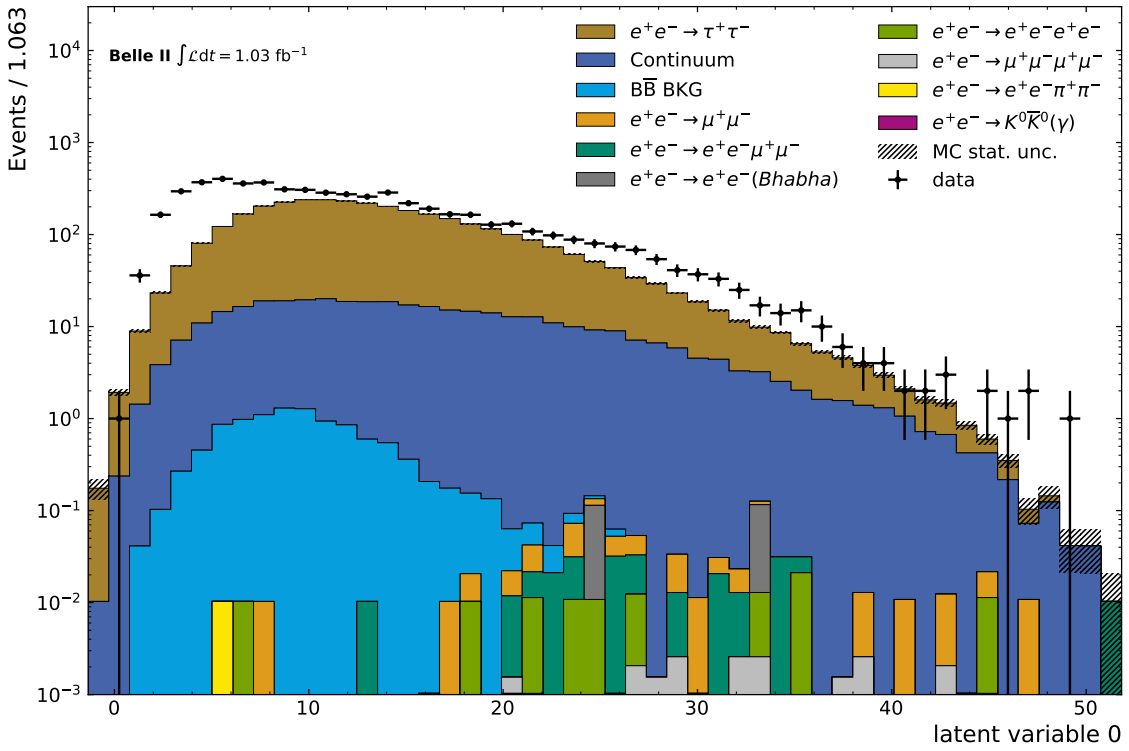

Figure 8.1.: Comparison of the distribution of latent variable 0 of the 8-dimensional AE for MC and data.

These additional corrections allow comparing data and MC as shown in Fig. 8.1, Fig. 8.2. While the basic structure of the simulation is also present in the data, some clear differences can be seen. First, the simulations underestimate the number of events for most bins. Both variables also show a higher excess of data events in the area of the $e^+e^- \to B\bar{B}$ sample. In variable 0 there also is a higher excess in the area of the four lepton processes.

There could be several causes for these discrepancies. One known is that the current event generators used do not simulate initial or final state radiation [14]. In these cases, a photon is radiated by the electron or positron shortly before they collide (Initial State Radiation (ISR)) or by the FSP after the collision process (Final State Radiation (FSR)). In both cases, this leads to more missing Energy as the radiated photons often do not leave the beam pipe. Other possible discrepancies might arise from the trigger efficiency. This comparison, therefore, requires a much more detailed investigation, before any possible claims on the existence of anomalies could be made.

The knowledge gained from this investigation can also be applied to the selection of training samples. Excluding events, that are excluded in the data-MC comparison could also be excluded from the training. This could reduce the range of different inputs the autoencoder needs to replicate and therefore may improve sensitivities.

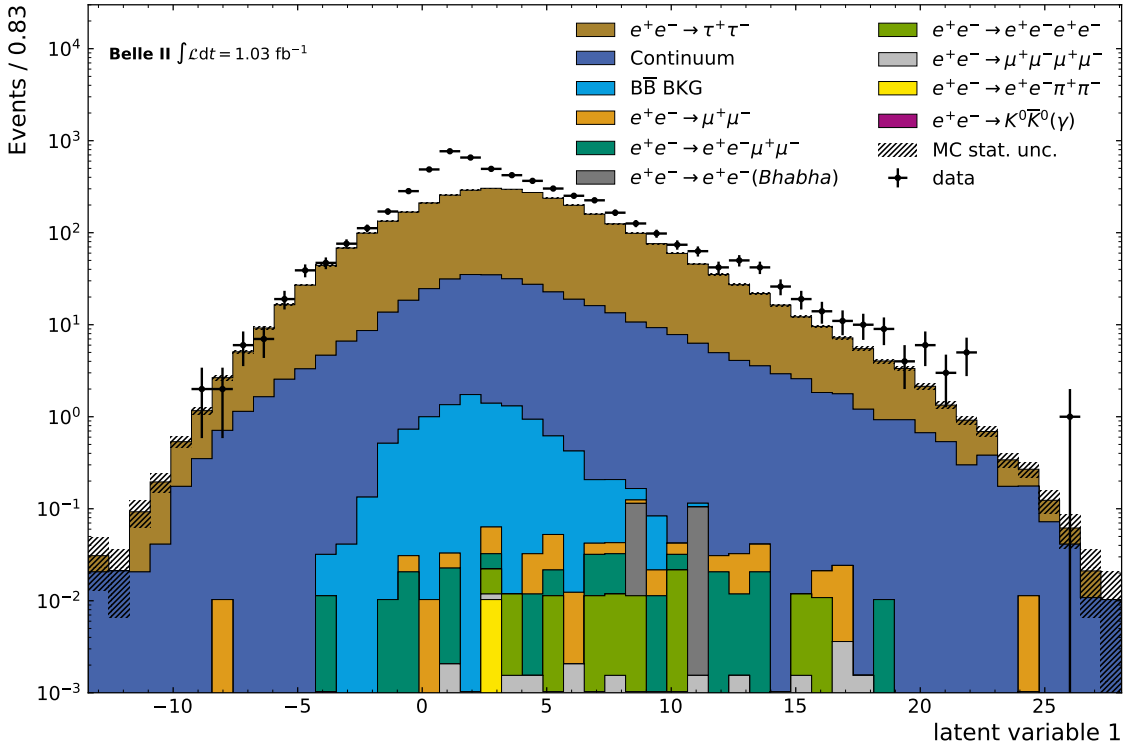

Figure 8.2.: Comparison of the distribution of latent variable 1 of the 8-dimensional AE for MC and data.

9. Conclusions

In this thesis, three different autoencoder architectures, Autoencoder (AE), Variational Autoencoder (VAE), and Dirichlet Variational Autoencoder (DVAE) are explored as a tool for model-parameter-independent searches for inelastic Dark Matter with a Dark Higgs (IDMDH). For all, errors in the event reconstruction from a latent space measured by the Mean Squared Error (MSE) are analyzed as Anomaly Score (AS). The sensitivity of this metric is studied for different mass configurations of the IDMDH model.

Mean Squared Error as Anomaly Score

The sensitivity of the MSE varied with the number of latent dimensions and the mass configurations. No single autoencoder showed sensitivity towards all regions in the model parameter space. For configurations with small ($<1.5\,\mathrm{GeV}/c^2$) $m_{h'}$ and m_{χ_2}, the highest sensitivity is reached with an 8-dimensional AE. A 9-dimensional AE has the highest senstivity for configurations with large ($>1.5\,\mathrm{GeV}/c^2$) $m_{h'}$ and m_{χ_2}. For configurations with high mass differences between the χ_2 and h', none of the autoencoders showed sensitivity. Sensitivities for low mass configurations are possibly biased by the lack of simulation for the $e^+e^- \to e^+e^-\mu^+\mu^-$ and $e^+e^- \to e^+e^-e^+e^-$ background process below an invariant mass of $0.5\,\mathrm{GeV}/c^2$. Including such samples in the training samples is planned in further efforts. The studies can also be extended to more latent dimensions as current studies suggest a possible improvement in sensitivity for large mass configurations.

The VAEs are restricted by the strong regularisation of their latent spaces. This limits their ability to reconstruct resulting in no lower average MSE values with a higher number of latent dimensions. Further improvement could be reached with less regularised latent spaces. Autoencoders with unstructured latent spaces proved more effective than structured ones, though the DVAE did not perform much worse. Further experiments with more latent dimensions and less strict regularisations could improve their viability.

DVAEs showed a similar behaviour than the unregularised AEs, but with slightly lower sensitivities overall. The studies suggest further sensitivities could be reached by longer training times and more latent dimensions.

Searching Anomalies in Latent Space

Bringing efforts in latent tagging presented in [1] proved difficult. Only the latent spaces of DVAEs showed some sensitivities but showed no improvement over the MSE. The naive

choice of priors could be one of the reasons and further studies with more elaborate choices could improve latent tagging. One starting point could be to reflect on the hierarchy of the background processes in the choice of priors. The results of the VAEs could certainly be improved with a less strict regularisation.

Additional methods for multivariate analysis like clustering, density estimation, or decision trees could be tools to define anomalies.

Prospects of Autoencoders Anomaly Detection

Autoencoders, as trained and used in this thesis, proved not to be able to detect all anomalies equally. While a comparison with a purely selection-based approach is not made, it is likely the currently reached sensitivities with autoencoders are not significantly higher. However, additional information, like non-binary Particle Identification (PID) likelihoods, used in the selection of the training sample can lead to higher sensitivities. The usage of advanced autoencoder architectures like Normalized Autoencoders [18] or graph-based architectures can bring improvements too. As demonstrated by Chapter 8, autoencoders show similar behavior on data as for simulation. But some aspects of this comparison like trigger efficiencies and limitations in the Monte Carlo (MC) event generators require further investigation.

Outlook

Besides improvements, the current work offers multiple ways to extend its scope. Additional model parameters like the mixing angles θ and ϵ could be varied and the sensitivities studied. More final states of the signal process could be included or the search widened for non-prompt decays. On the Machine Learning (ML) side, Anomaly Detection (AD) in High Energy Physics (HEP) is a very active field of development. With the CMS and ATLAS collaborations at the forefront of this effort, many methods based on jet images have been developed. New autoencoder architectures like the Normalized Autoencoder [18] have emerged in the last year, which tackle some of the issues with the architectures described here. Also, ansatzes based on density estimations like Classification without Labels (CwoLa) [24] could be brought to the context of this work.

While this work focused on using AD in offline analysis, the usage on the trigger level is another discussed topic. With the high luminosities of Belle II and future experiments have to deal with, it becomes increasingly important to quickly identify interesting events to not miss any hint for physics beyond the standard model. This could be another future application of this work.

Bibliography

[1] B. Dillon, T. Plehn, C. Sauer, and P. Sorrenson, "Better latent spaces for better autoencoders," *SciPost Physics* **11** no. 3, (Sep, 2021) . https://doi.org/10.21468%2Fscipostphys.11.3.061.

[2] Pedregosa, F. et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research* **12** (2011) 2825–2830. http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html.

[3] K. G. Mehrotra, "Anomaly detection principles and algorithms," 2017. https://doi.org/10.1007/978-3-319-67526-8.

[4] G. Kasieczka et al., "The LHC Olympics 2020: A community challenge for anomaly detection in high energy physics," *Reports on Progress in Physics* **84** no. 12, (Dec, 2021) 124201. https://doi.org/10.1088%2F1361-6633%2Fac36b9.

[5] **SuperKEKB accelerator team**, K. Akai, K. Furukawa, and H. Koiso, "SuperKEKB collider," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **907** (Nov, 2018) 188–199. https://doi.org/10.1016%2Fj.nima.2018.08.017.

[6] **The Belle II Collaboration**, F. Forti, "Snowmass Whitepaper: The Belle II Detector Upgrade Program," 2022. https://arxiv.org/abs/2203.11349.

[7] **The Belle II Collaboration**, T. Abe et al., "Belle II Technical Design Report," Nov, 2010. https://arxiv.org/abs/1011.0352.

[8] Y. Iwasaki, B. Cheon, E. Won, X. Gao, L. Macchiarulo, K. Nishimura, and G. Varner, "Level 1 trigger system for the Belle II experiment," *IEEE Trans. Nucl. Sci.* **58** (2011) 1807–1815.

[9] T. Kuhr, C. Pulvermacher, M. Ritter, and T. H. N. Braun, "The Belle II Core Software," *Computing and Software for Big Science* **3** no. 1, (Nov, 2018) 1. https://doi.org/10.1007/s41781-018-0017-9.

[10] **The Belle II Collaboration**, "Belle II Analysis Software Framework (basf2)." https://doi.org/10.5281/zenodo.5574115.

[11] G. Bertone and D. Hooper, "History of dark matter," *Reviews of Modern Physics* **90** no. 4, (Oct, 2018) . https://doi.org/10.1103%2Frevmodphys.90.045002.

[12] M. Duerr, T. Ferber, C. Garcia-Cely, C. Hearty, and K. Schmidt-Hoberg, "Long-lived dark Higgs and inelastic dark matter at Belle II," *Journal of High Energy Physics* **2021** no. 4, (Apr, 2021) . https://doi.org/10.1007%2Fjhep04%282021%29146.

[13] Z. Liptak et al., "Measurements of beam backgrounds in SuperKEKB Phase 2," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **1040** (Oct, 2022) 167168. https://doi.org/10.1016%2Fj.nima.2022.167168.

[14] P. Urquijo and T. Ferber, "Overview of the Belle II Physics Generators," 2016. https://docs.belle2.org/record/282/files/BELLE2-NOTE-PH-2015-006.pdf.

[15] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H.-S. Shao, T. Stelzer, P. Torrielli, and M. Zaro, "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations," *Journal of High Energy Physics* **2014** no. 7, (Jul, 2014) . https://doi.org/10.1007%2Fjhep07%282014%29079.

[16] S. Agostinelli et al., "Geant4 — a simulation toolkit," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **506** no. 3, (2003) 250–303. https://www.sciencedirect.com/science/article/pii/S0168900203013688.

[17] G. Punzi, "Sensitivity of searches for new signals and its optimization," 2003. https://arxiv.org/abs/physics/0308063.

[18] B. Dillon, L. Favaro, T. Plehn, P. Sorrenson, and M. Krämer, "A Normalized Autoencoder for LHC Triggers," 2022. https://arxiv.org/abs/2206.14225.

[19] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychological Review, American Psychological Association* **65** (1958) 386–408.

[20] A. Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," pp. 8024–8035. Curran Associates, Inc., 2019. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[21] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science* **313** no. 5786, (2006) 504–507. https://www.science.org/doi/abs/10.1126/science.1127647.

[22] D. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv e-prints* (Dec, 2013) . https://arxiv.org/abs/1312.6114.

[23] The Belle II Tracking Group, "Tracking Performance in Early Belle II Data," *Belle II internal Paper draft* (Dec, 2021) . https://docs.belle2.org/record/2801.

[24] E. M. Metodiev, B. Nachman, and J. Thaler, "Classification without labels: learning from mixed samples in high energy physics," *Journal of High Energy Physics* **2017** no. 10, (Oct, 2017) . https://doi.org/10.1007%2Fjhep10%282017%29174.

A. Appendix

A.1. Background Studies

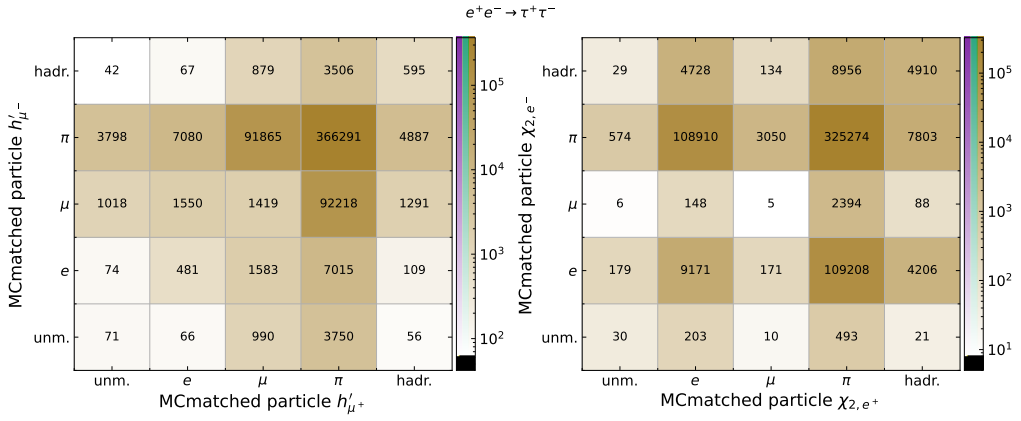

Figure A.1.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to \tau^+\tau^-$ sample.

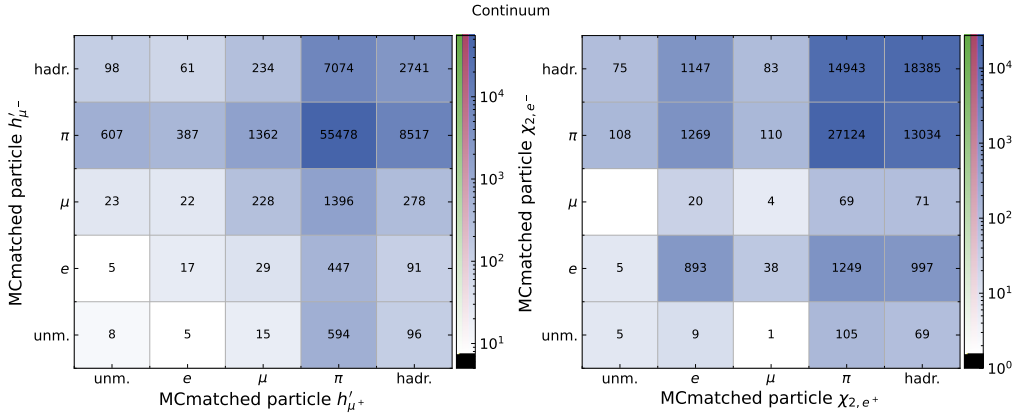

Figure A.2.: True particles used to reconstruct the h' and χ_2 candidates for the Continuum sample.

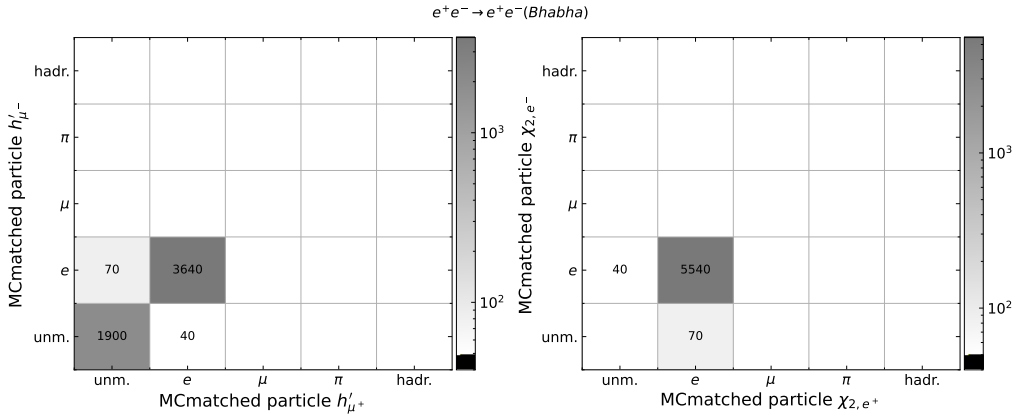

Figure A.3.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to e^+e^-$ sample.

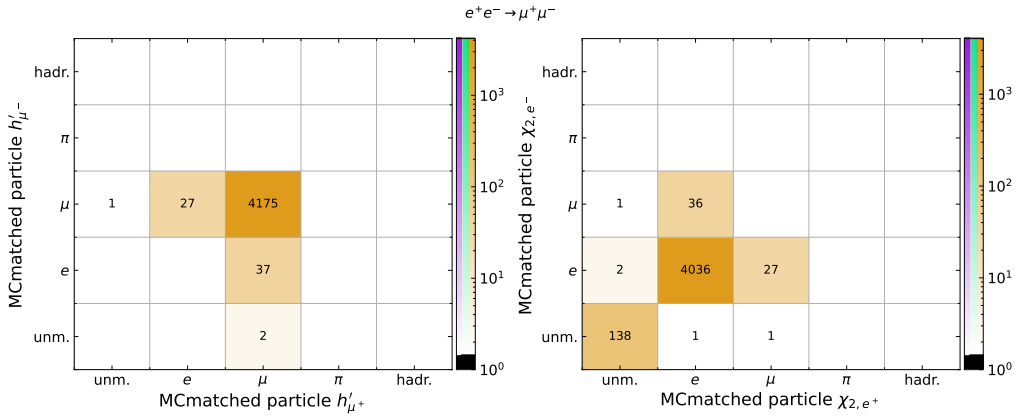

Figure A.4.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to \mu^+\mu^-$ sample.

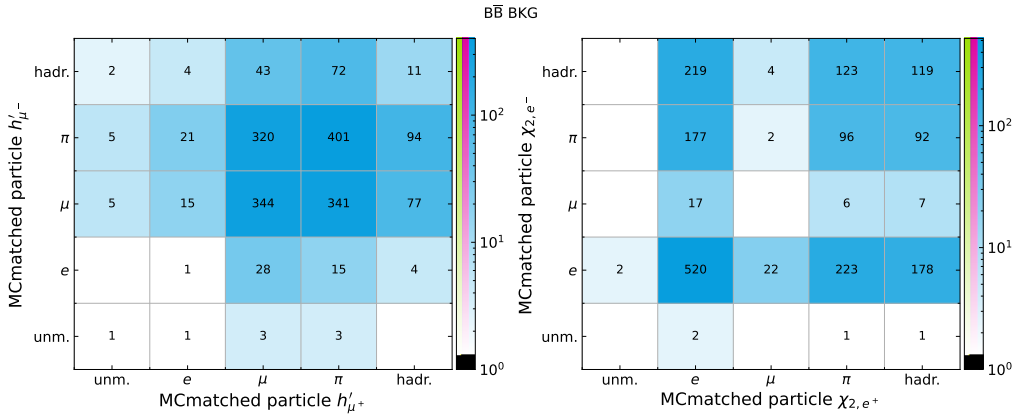

Figure A.5.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to B\bar{B}$ sample.

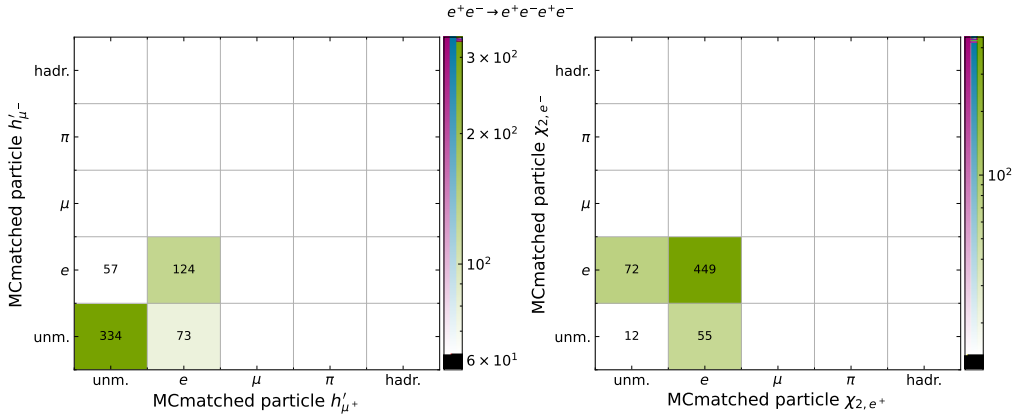

Figure A.6.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to e^+e^-e^+e^-$ sample.

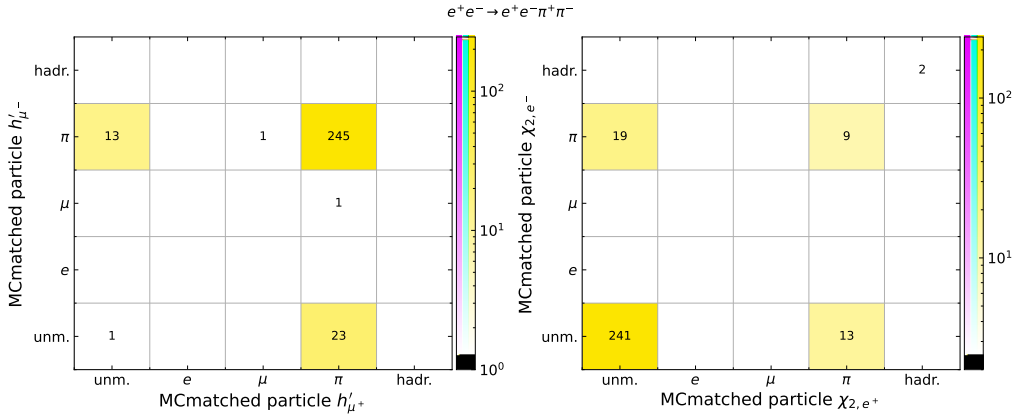

Figure A.7.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to e^+e^-\pi^+\pi^-$ sample.

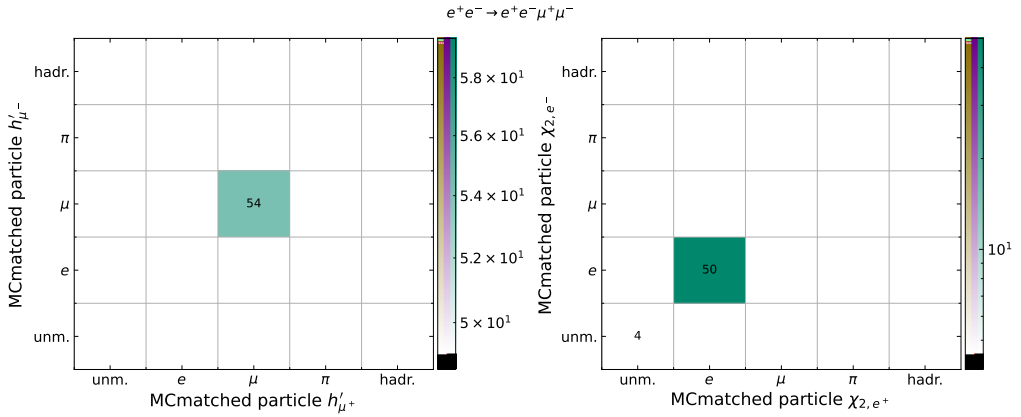

Figure A.8.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to e^+e^-\mu^+\mu^-$ sample.

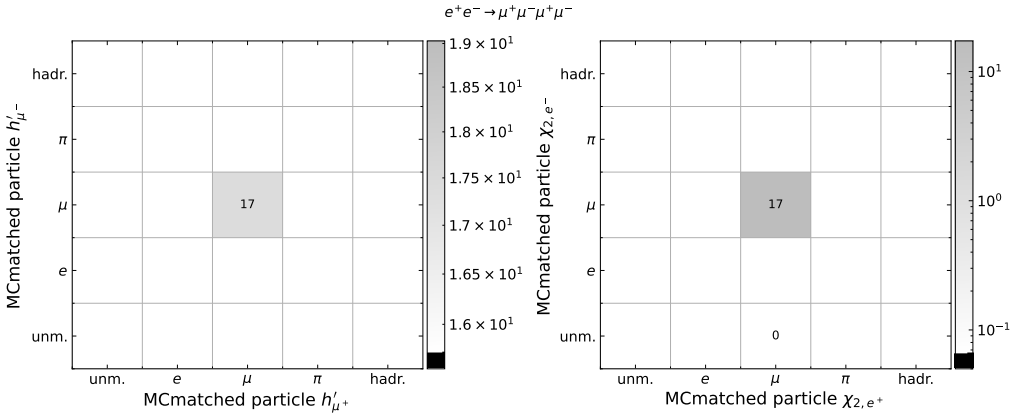

Figure A.9.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to \mu^+\mu^-\mu^+\mu^-$ sample.

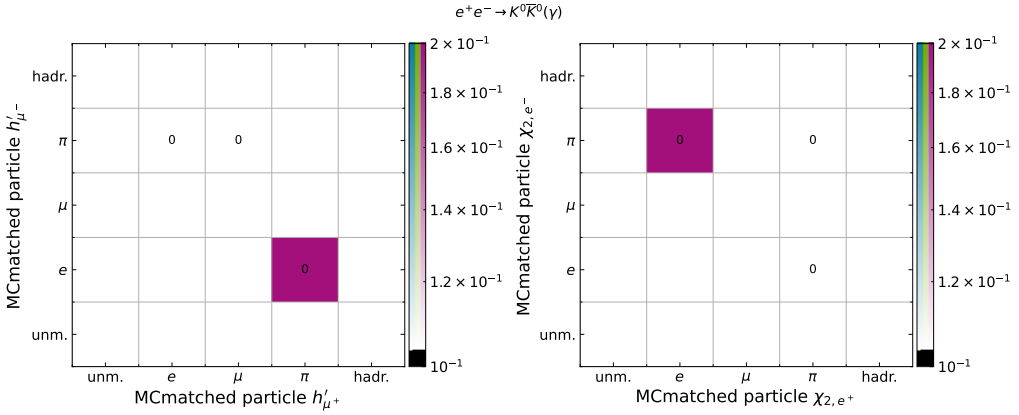

Figure A.10.: True particles used to reconstruct the h' and χ_2 candidates for the $e^+e^- \to K^0\overline{K}^0(\gamma)$ sample.

A.2. Training Details for AEs

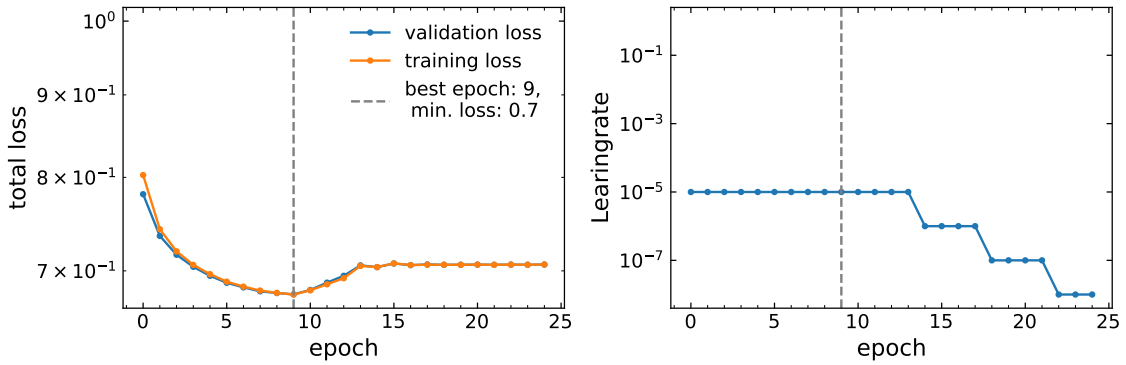

Figure A.11.: Training details for the training of an AE with 1-dimensional latent space. The total loss is equal to the MSE.

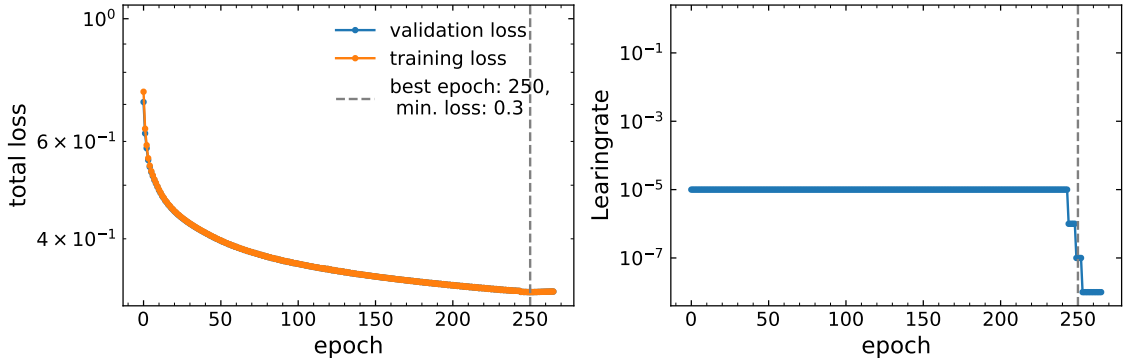

Figure A.12.: Training details for the training of an AE with 2-dimensional latent space. The total loss is equal to the MSE.

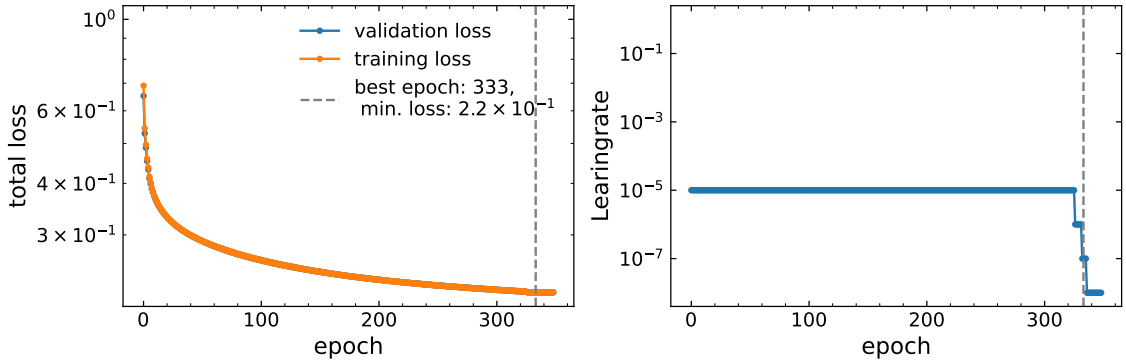

Figure A.13.: Training details for the training of an AE with 3-dimensional latent space. The total loss is equal to the MSE.

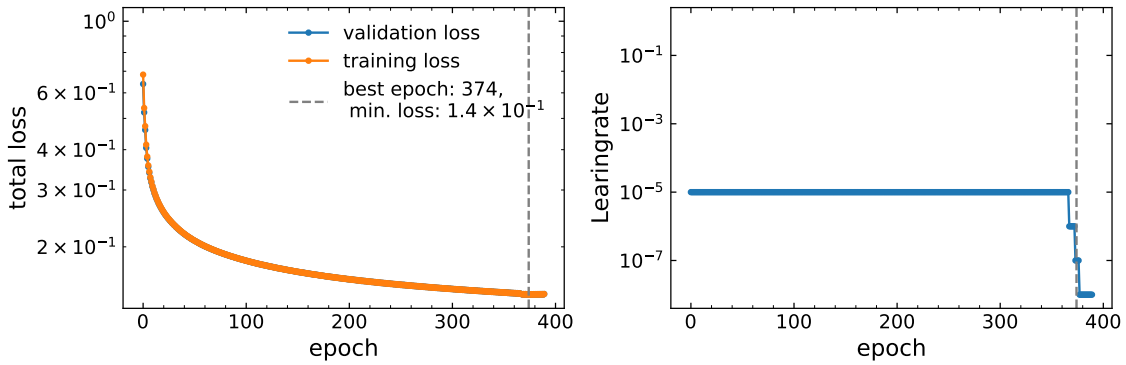

Figure A.14.: Training details for the training of an AE with 4-dimensional latent space. The total loss is equal to the MSE.

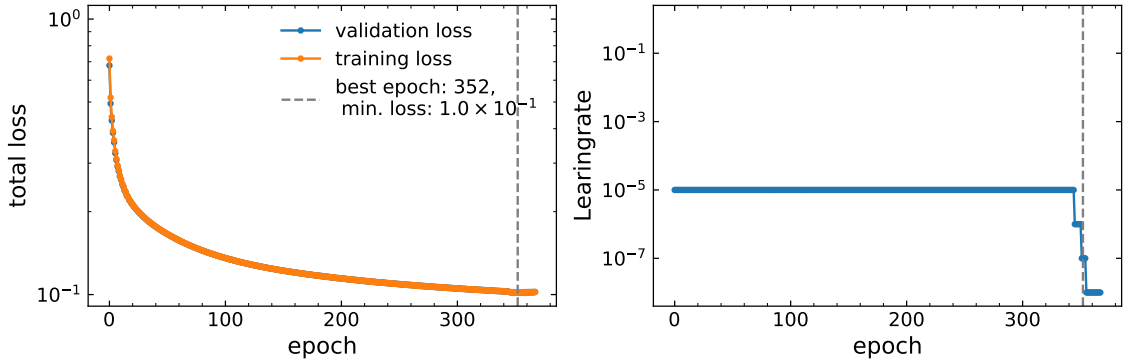

Figure A.15.: Training details for the training of an AE with 5-dimensional latent space. The total loss is equal to the MSE.

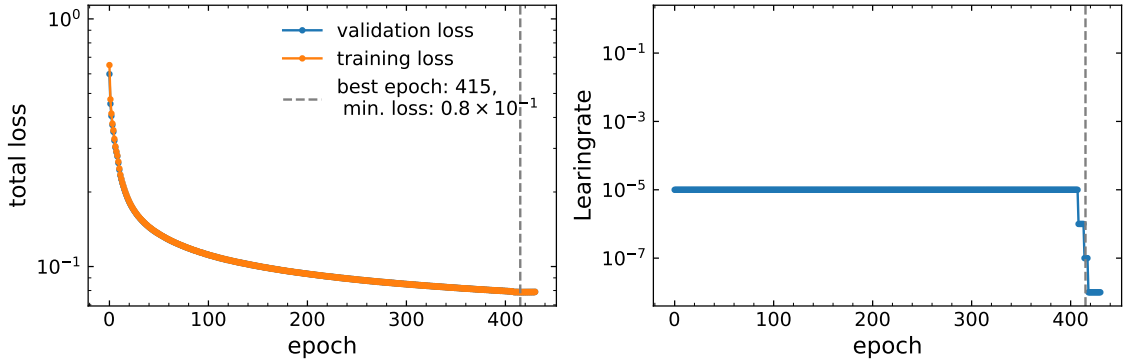

Figure A.16.: Training details for the training of an AE with 6-dimensional latent space. The total loss is equal to the MSE.

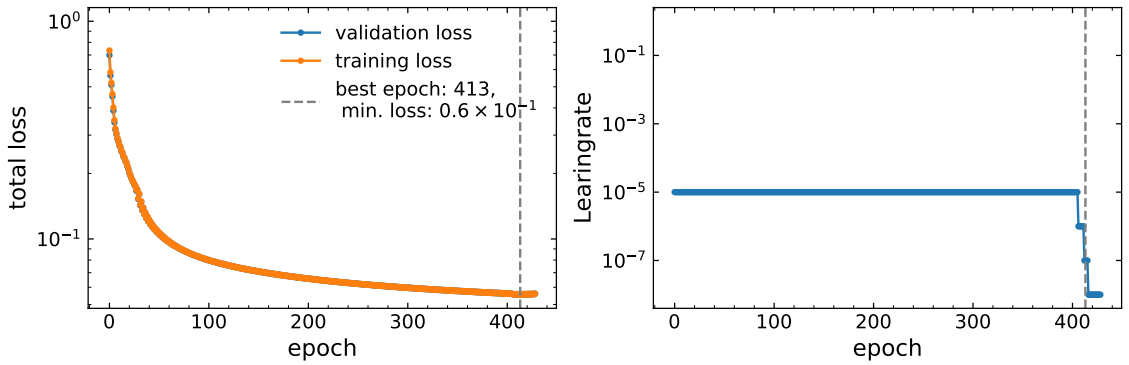

Figure A.17.: Training details for the training of an AE with 7-dimensional latent space. The total loss is equal to the MSE.

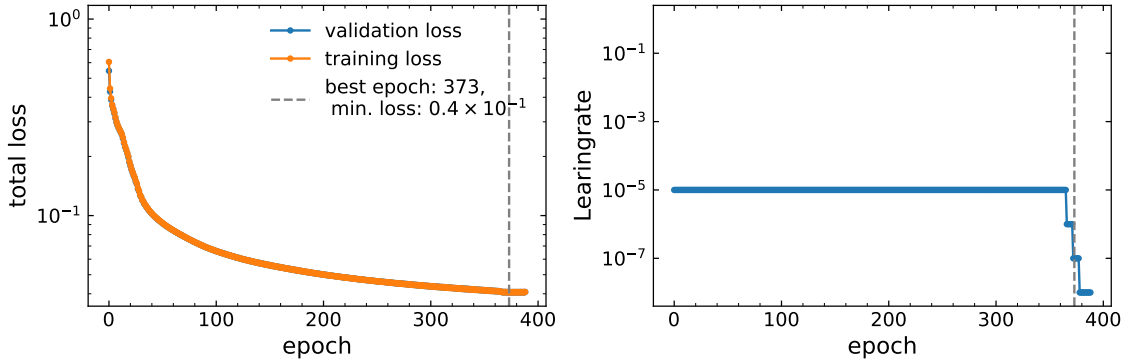

Figure A.18.: Training details for the training of an AE with 8-dimensional latent space. The total loss is equal to the MSE.

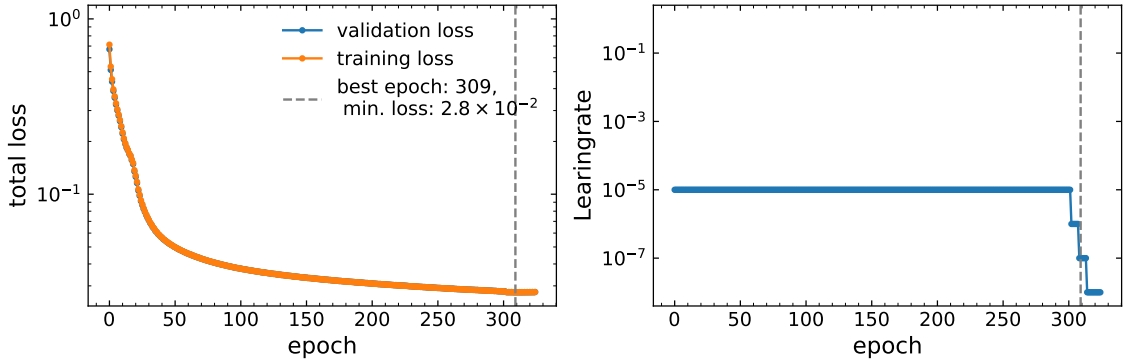

Figure A.19.: Training details for the training of an AE with 9-dimensional latent space. The total loss is equal to the MSE.

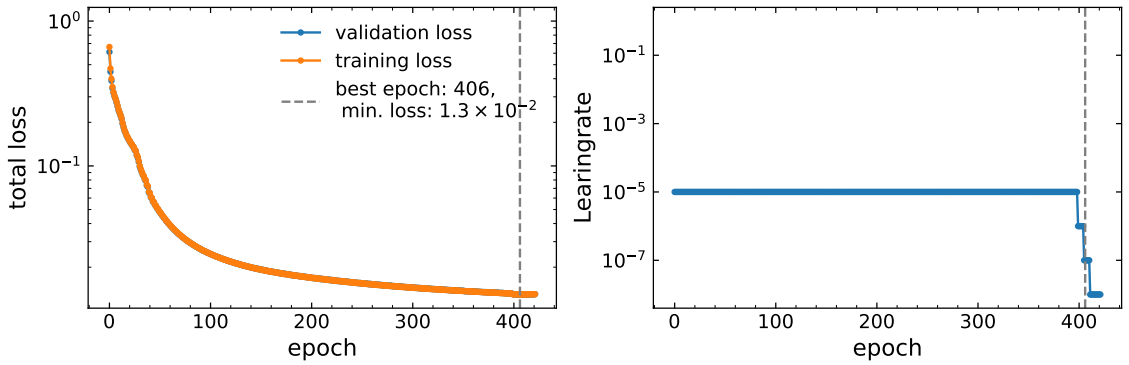

Figure A.20.: Training details for the training of an AE with 10-dimensional latent space. The total loss is equal to the MSE.

A.3. MSE for AEs

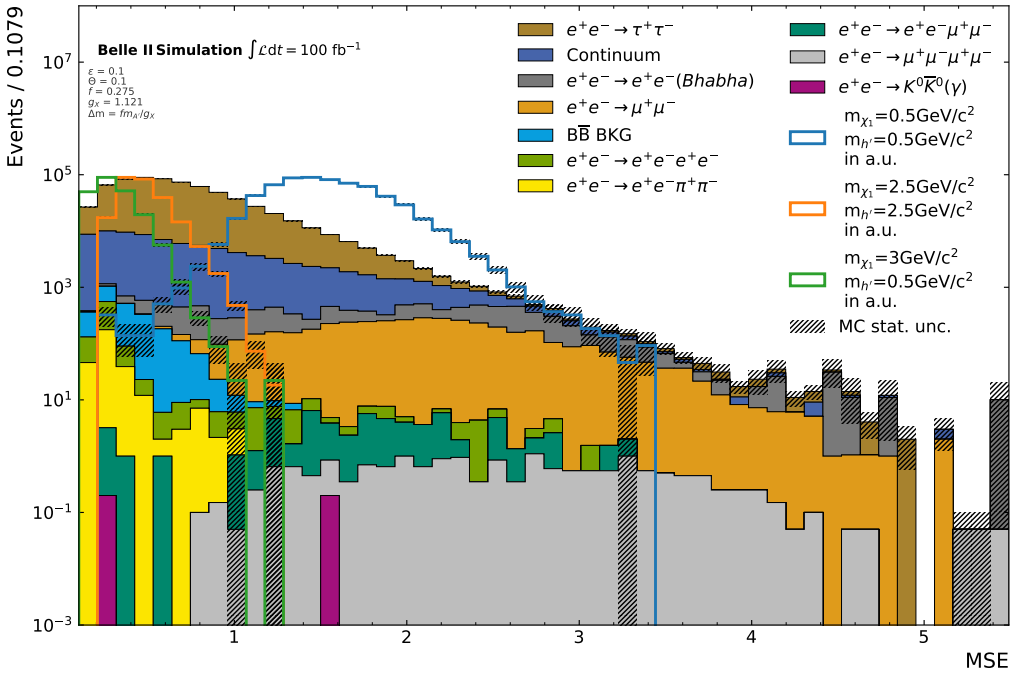

Figure A.21.: Distribution of the MSE for the 1-dimensional AE.

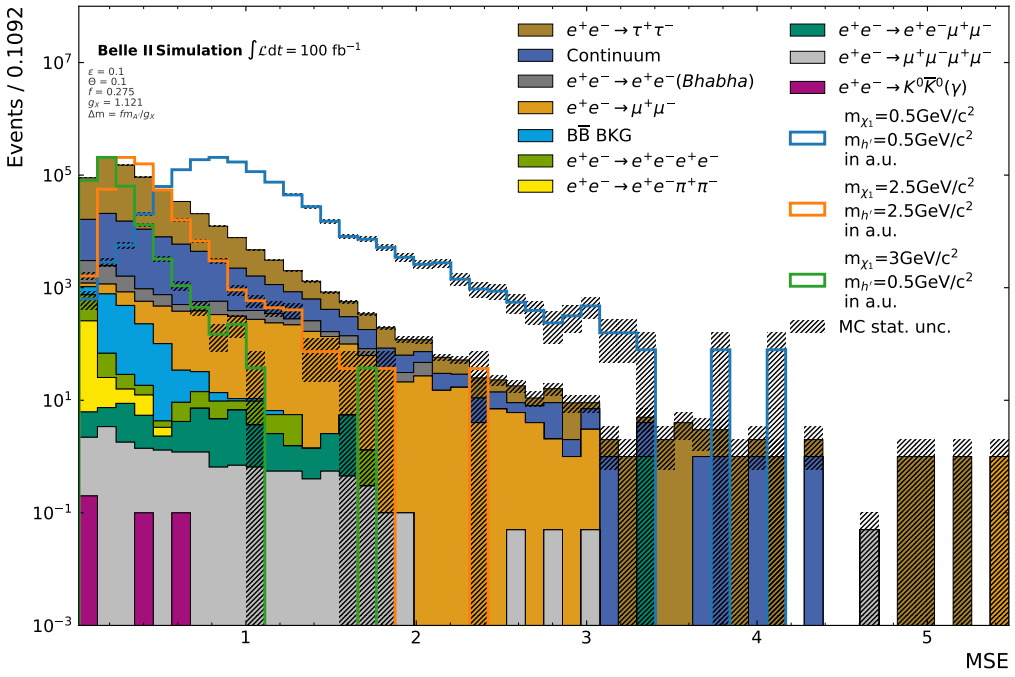

Figure A.22.: Distribution of the MSE for the 2-dimensional AE.

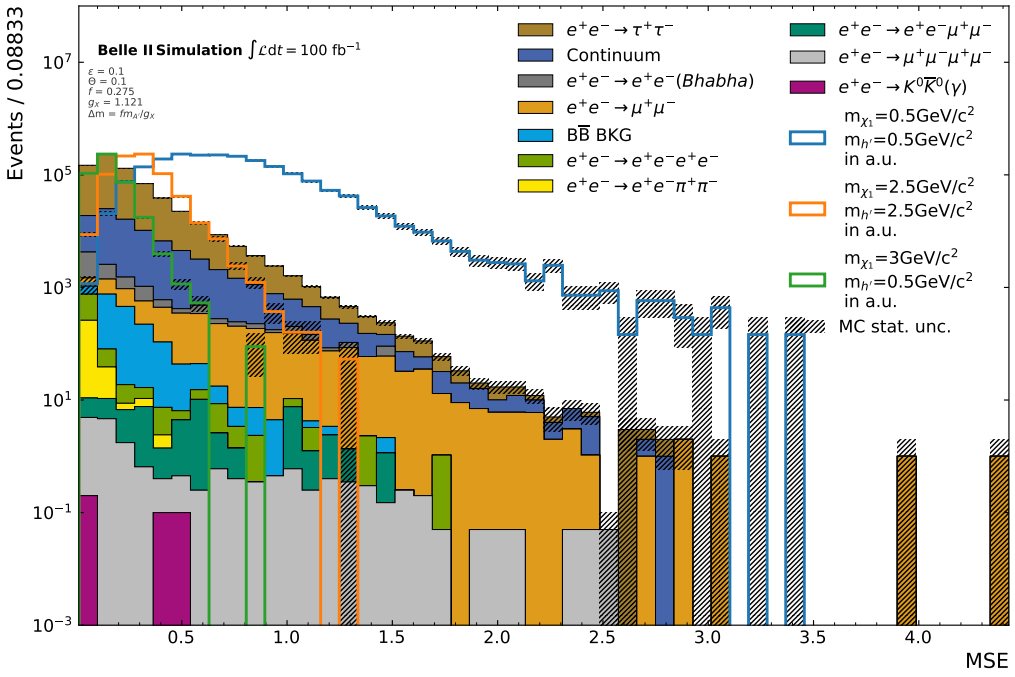

Figure A.23.: Distribution of the MSE for the 3-dimensional AE.

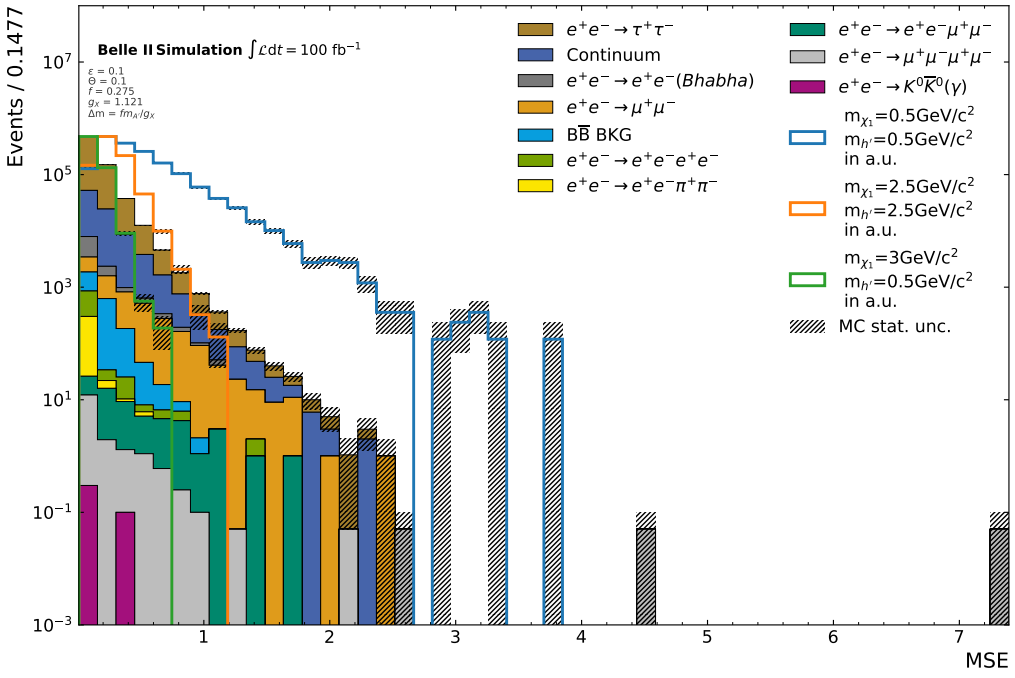

Figure A.24.: Distribution of the MSE for the 4-dimensional AE.

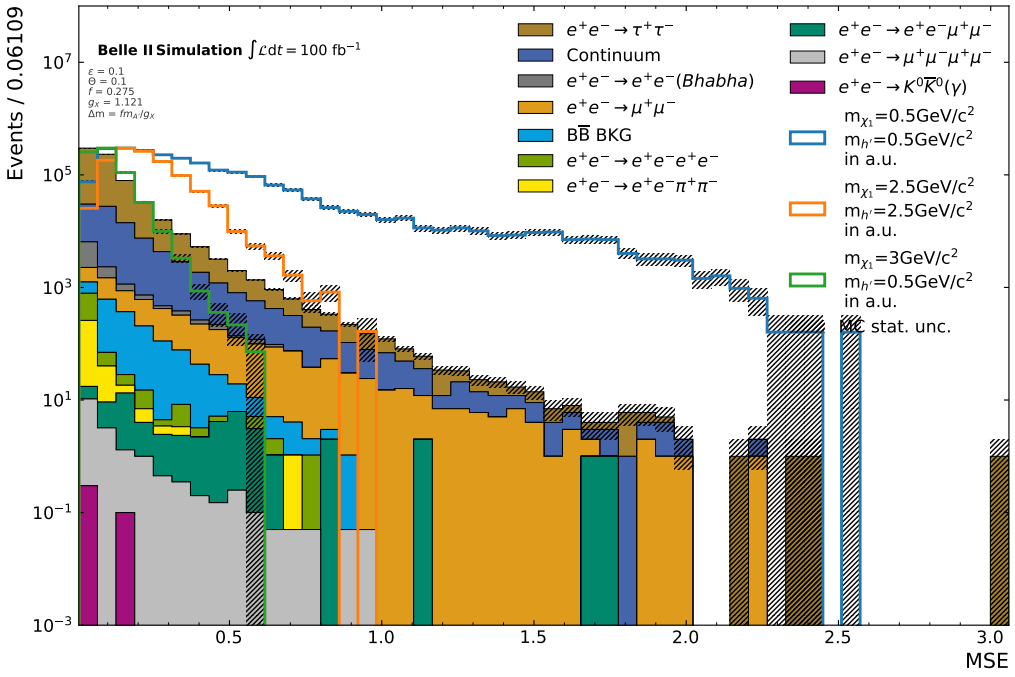

Figure A.25.: Distribution of the MSE for the 5-dimensional AE.

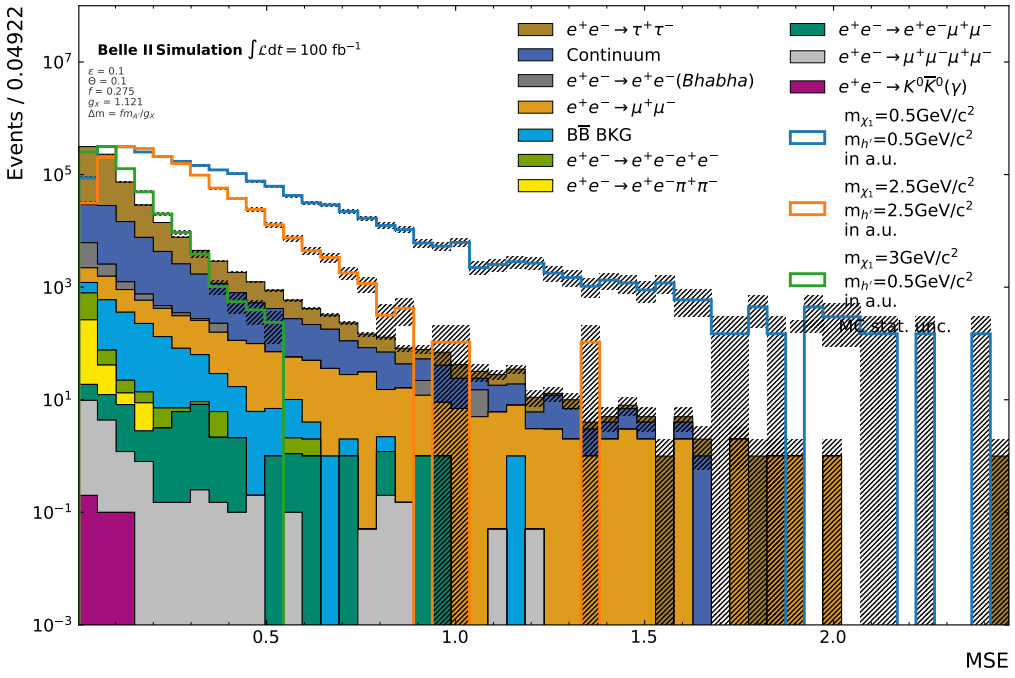

Figure A.26.: Distribution of the MSE for the 6-dimensional AE.

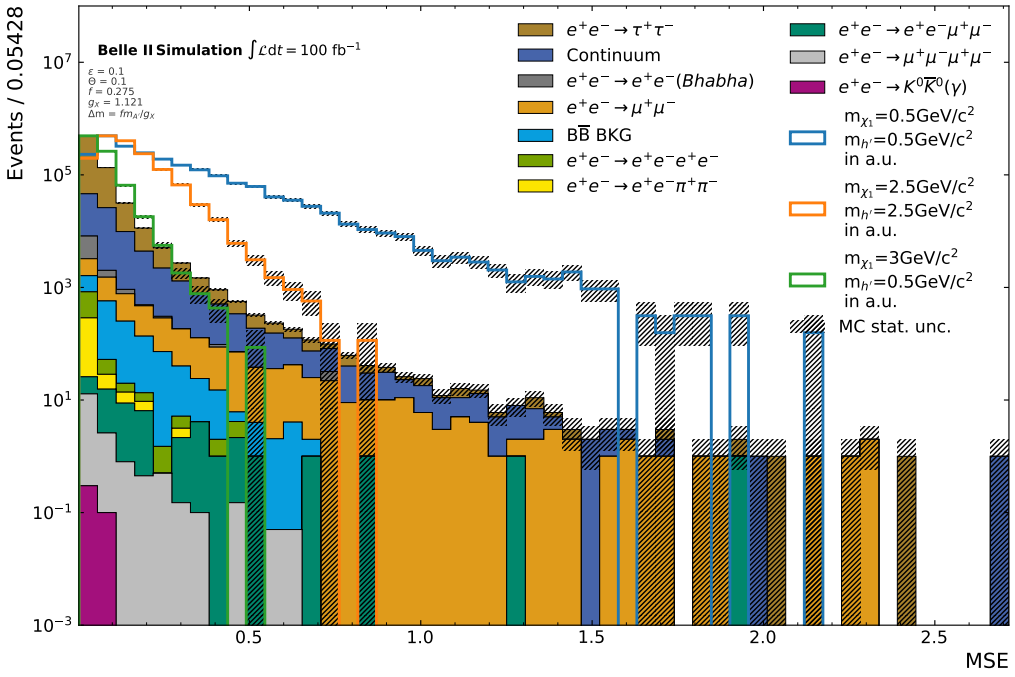

Figure A.27.: Distribution of the MSE for the 7-dimensional AE.

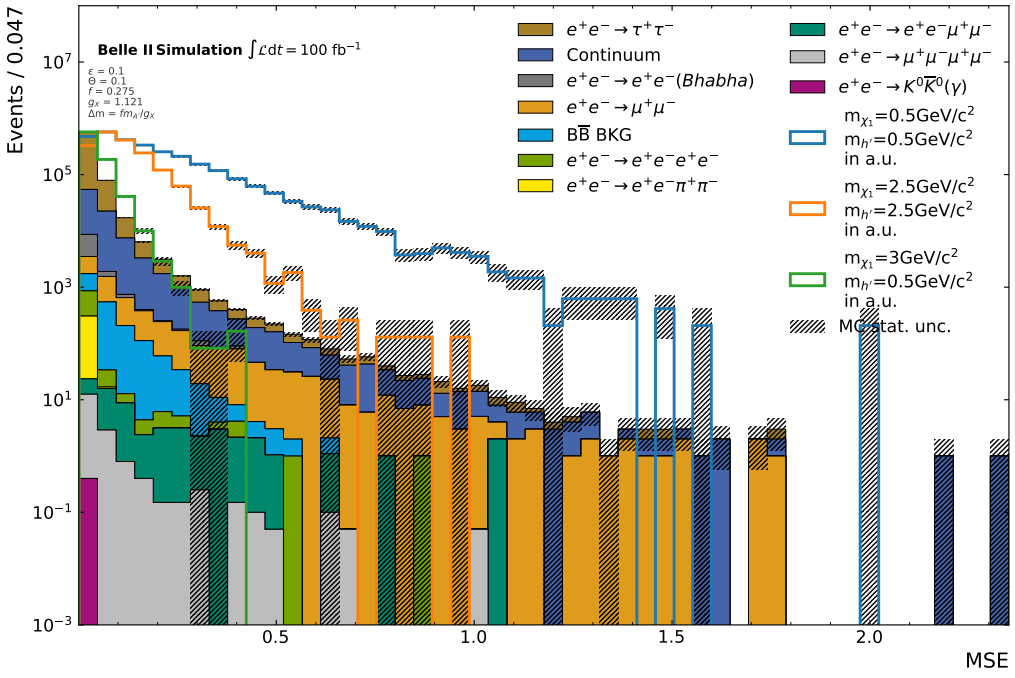

Figure A.28.: Distribution of the MSE for the 8-dimensional AE.

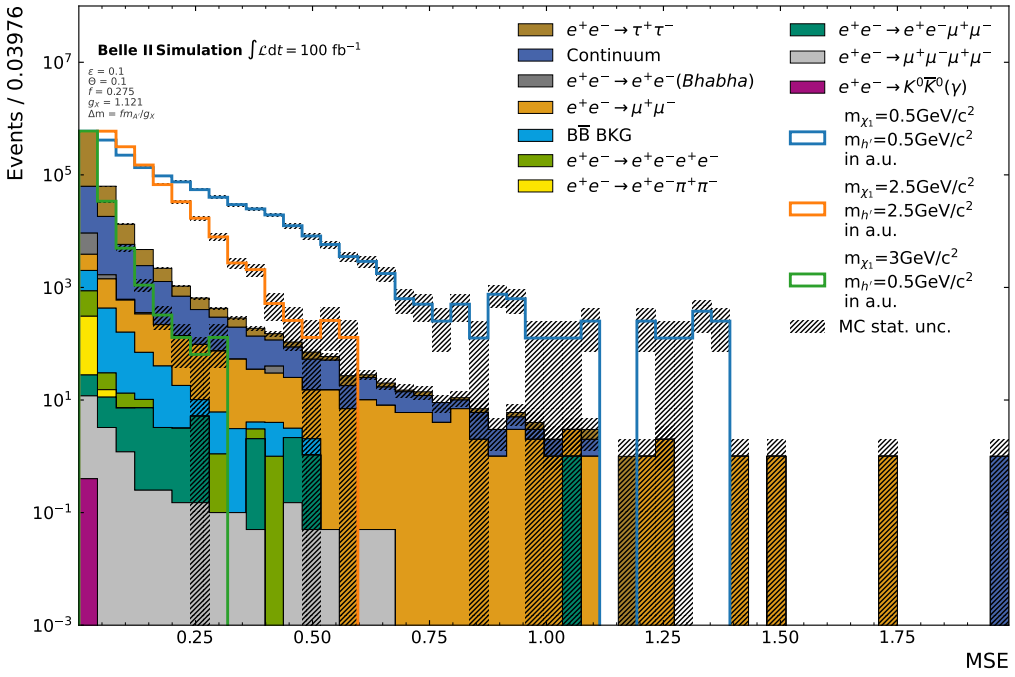

Figure A.29.: Distribution of the MSE for the 9-dimensional AE.

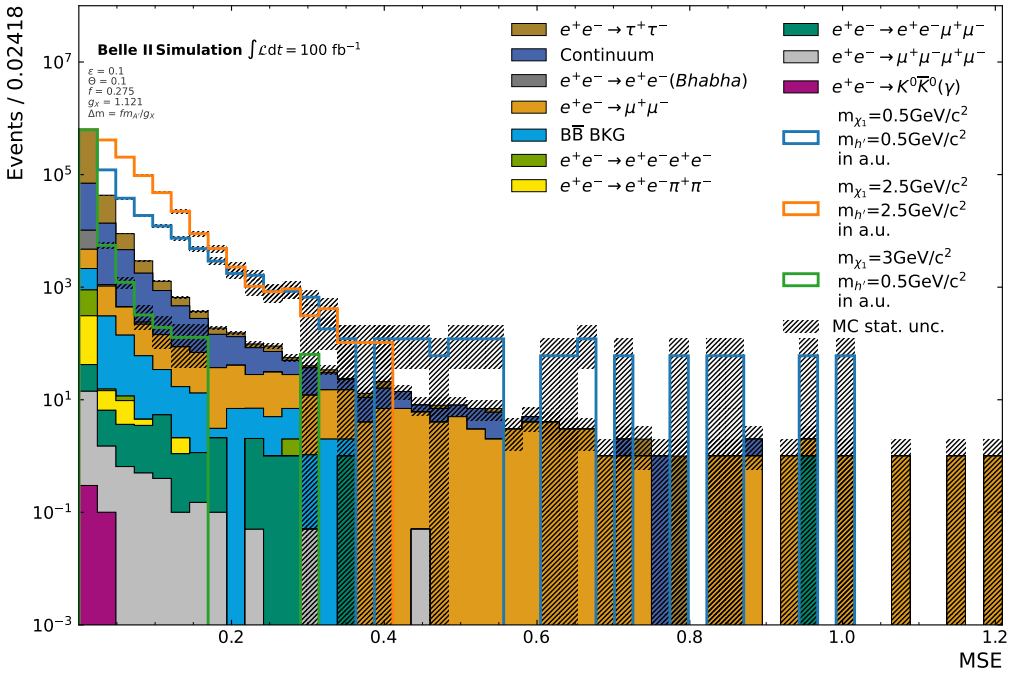

Figure A.30.: Distribution of the MSE for the 10-dimensional AE.

A.4. Latentspace of AEs

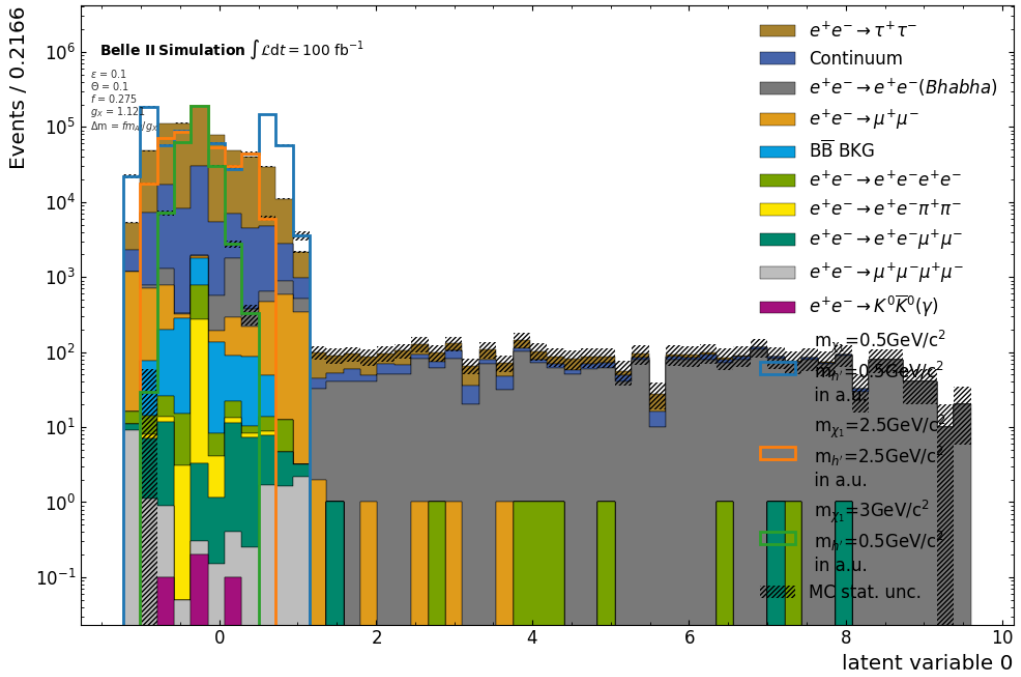

Figure A.31.: Latent variables and their correlations for the 1-dimensional AE for the background samples and the three example signals.

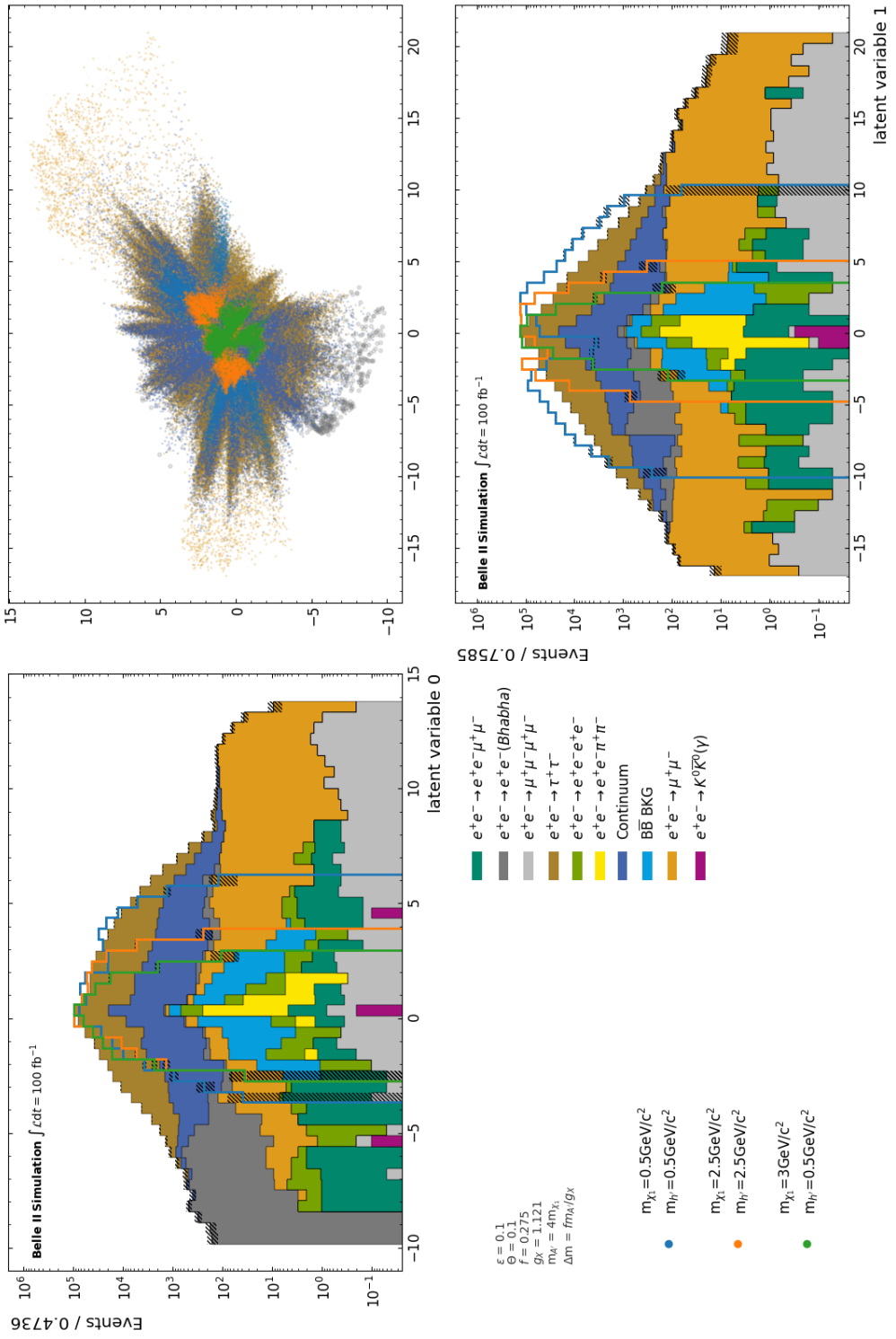

Figure A.32.: Latent variables and their correlations for the 2-dimensional AE for the background samples and the three example signals.

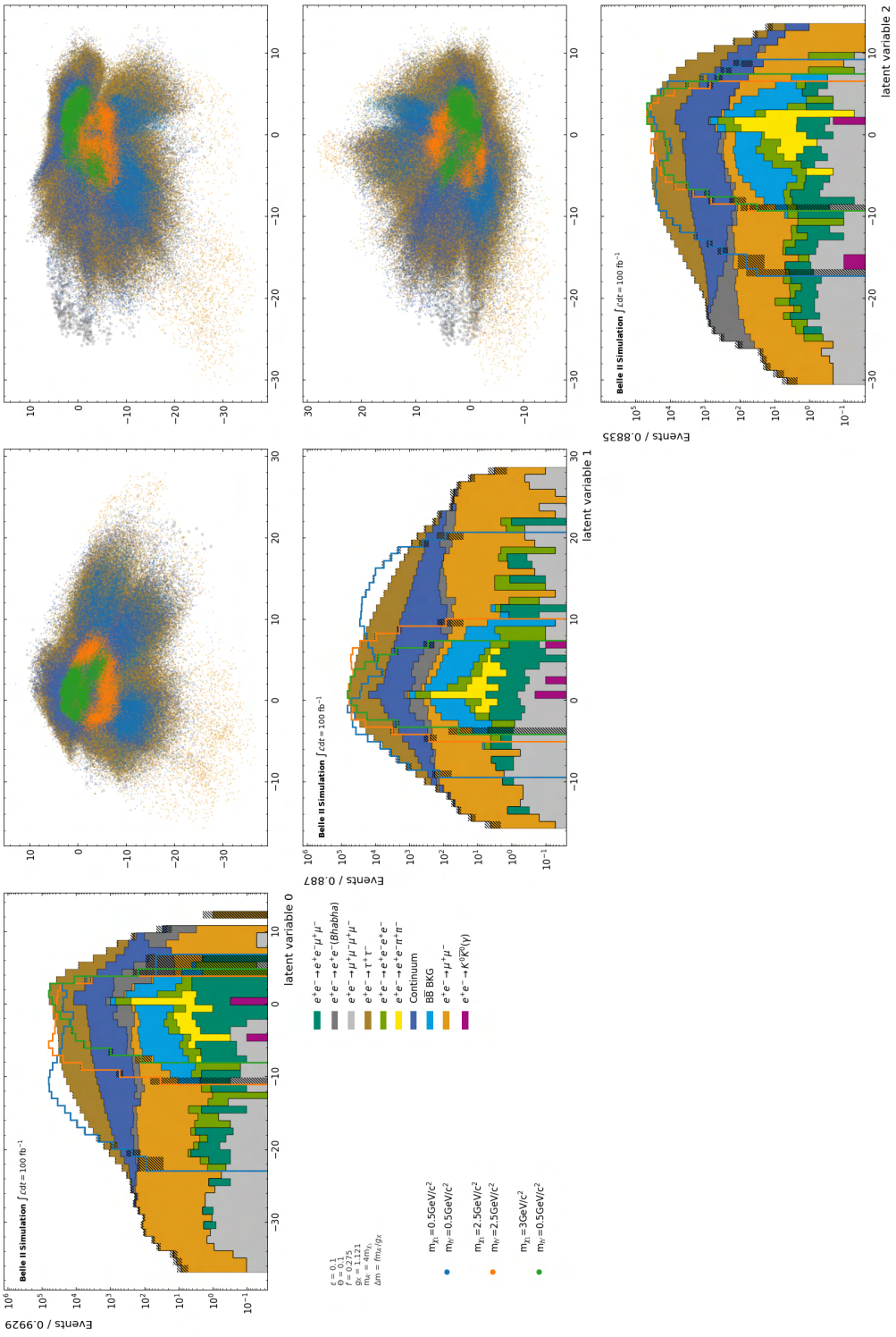

Figure A.33.: Latent variables and their correlations for the 3-dimensional AE for the background samples and the three example signals.

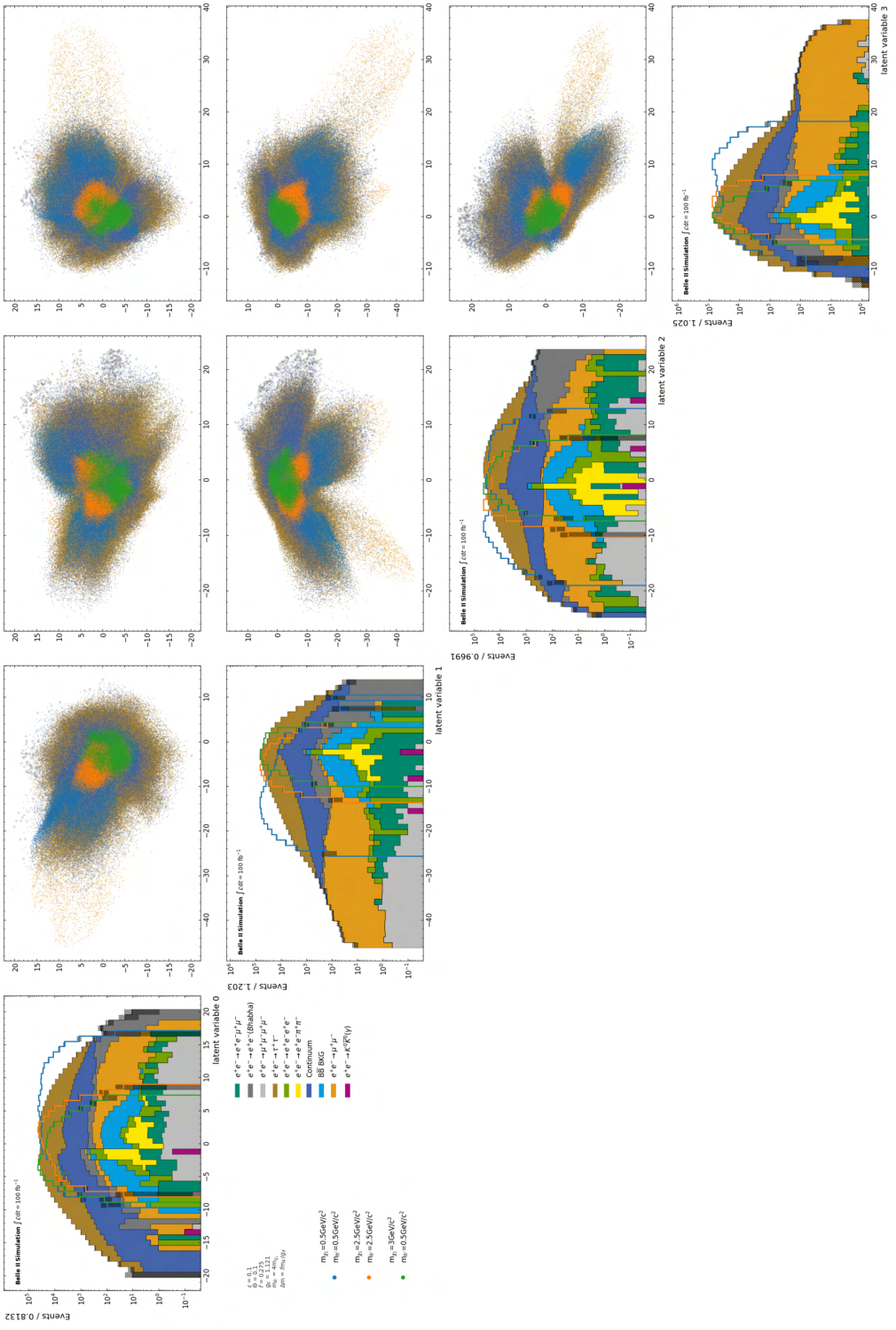

Figure A.34.: Latent variables and their correlations for the 4-dimensional AE for the background samples and the three example signals.

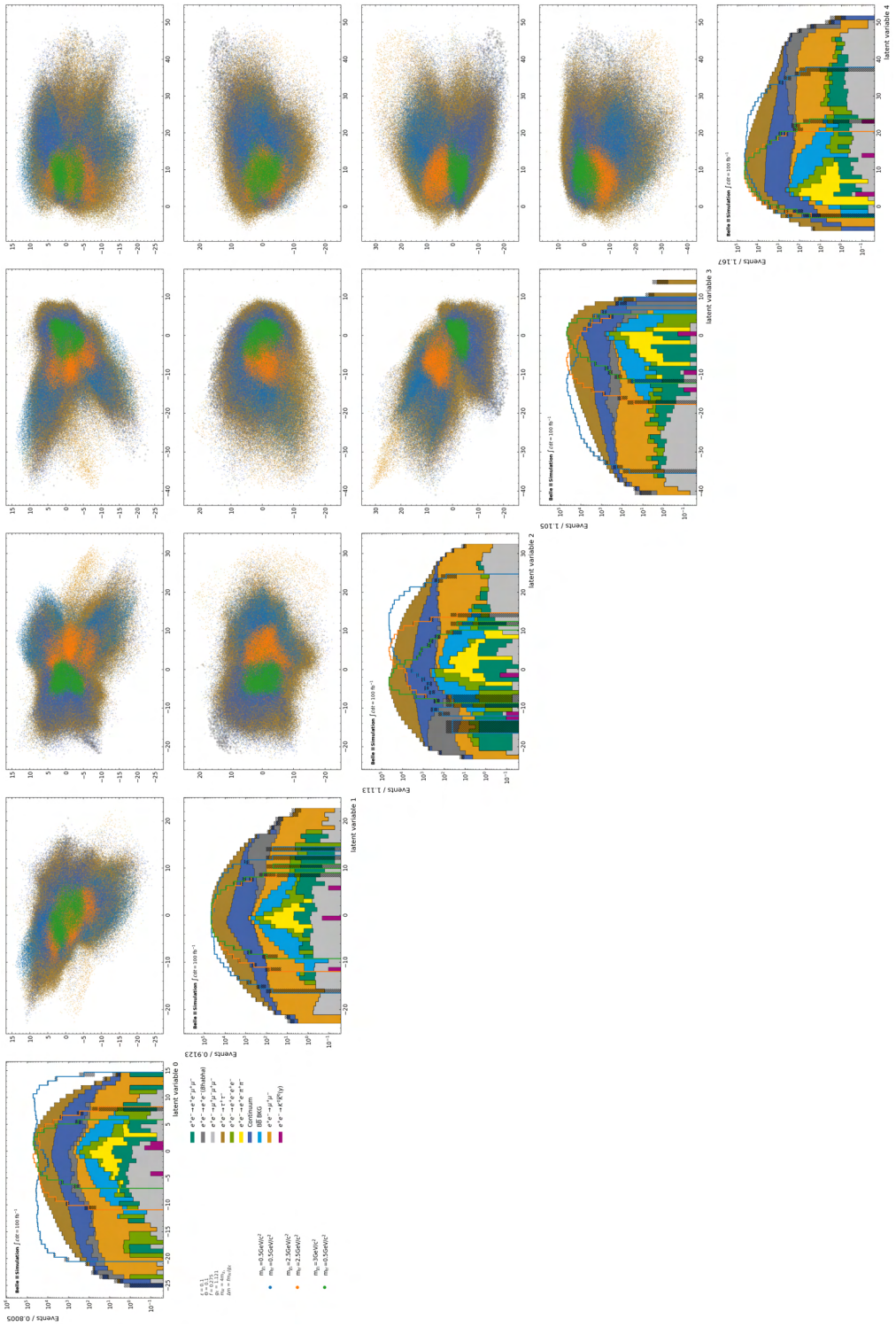

Figure A.35.: Latent variables and their correlations for the 5-dimensional AE for the background samples and the three example signals.

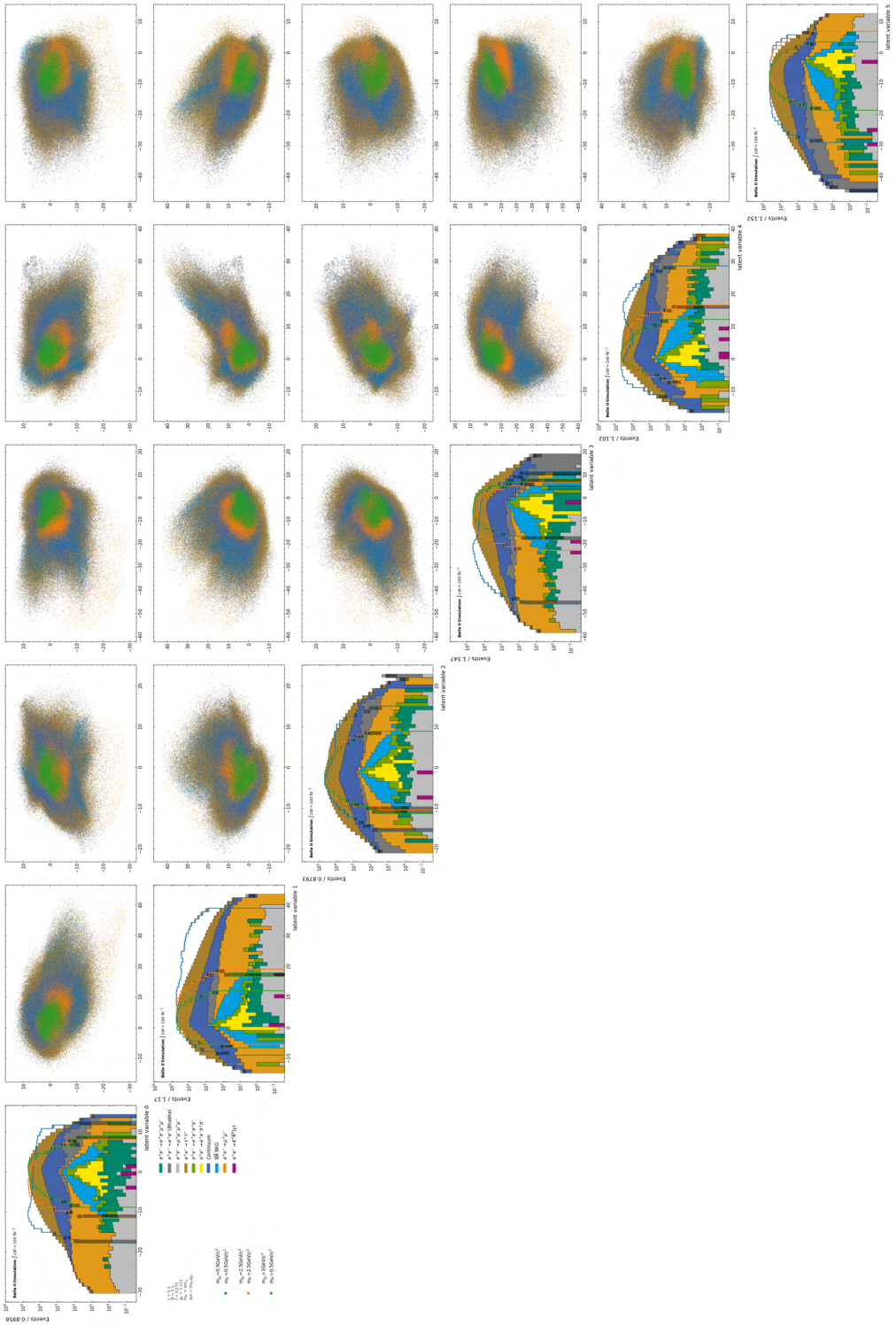

Figure A.36.: Latent variables and their correlations for the 6-dimensional AE for the background samples and the three example signals.

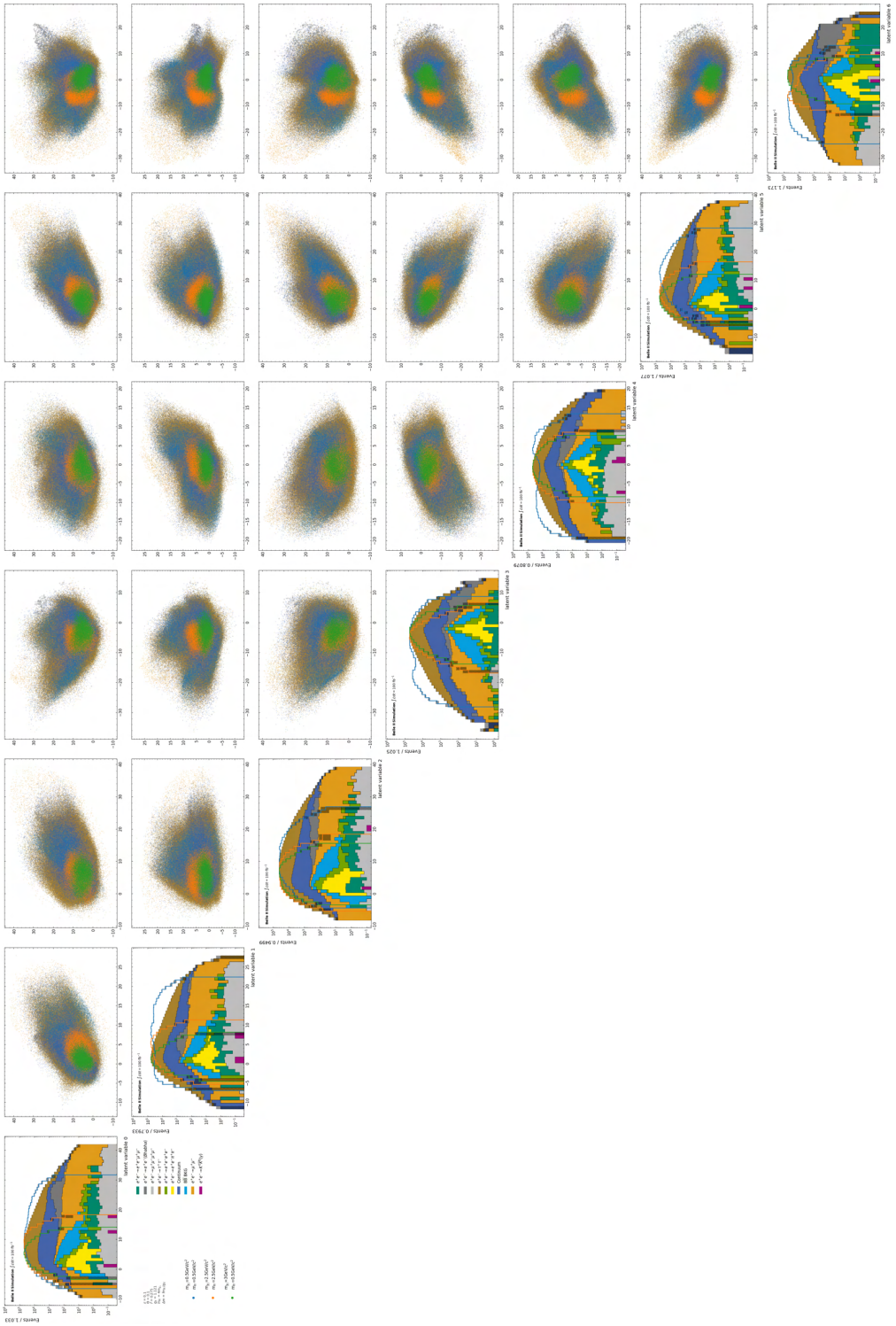

Figure A.37.: Latent variables and their correlations for the 7-dimensional AE for the background samples and the three example signals.

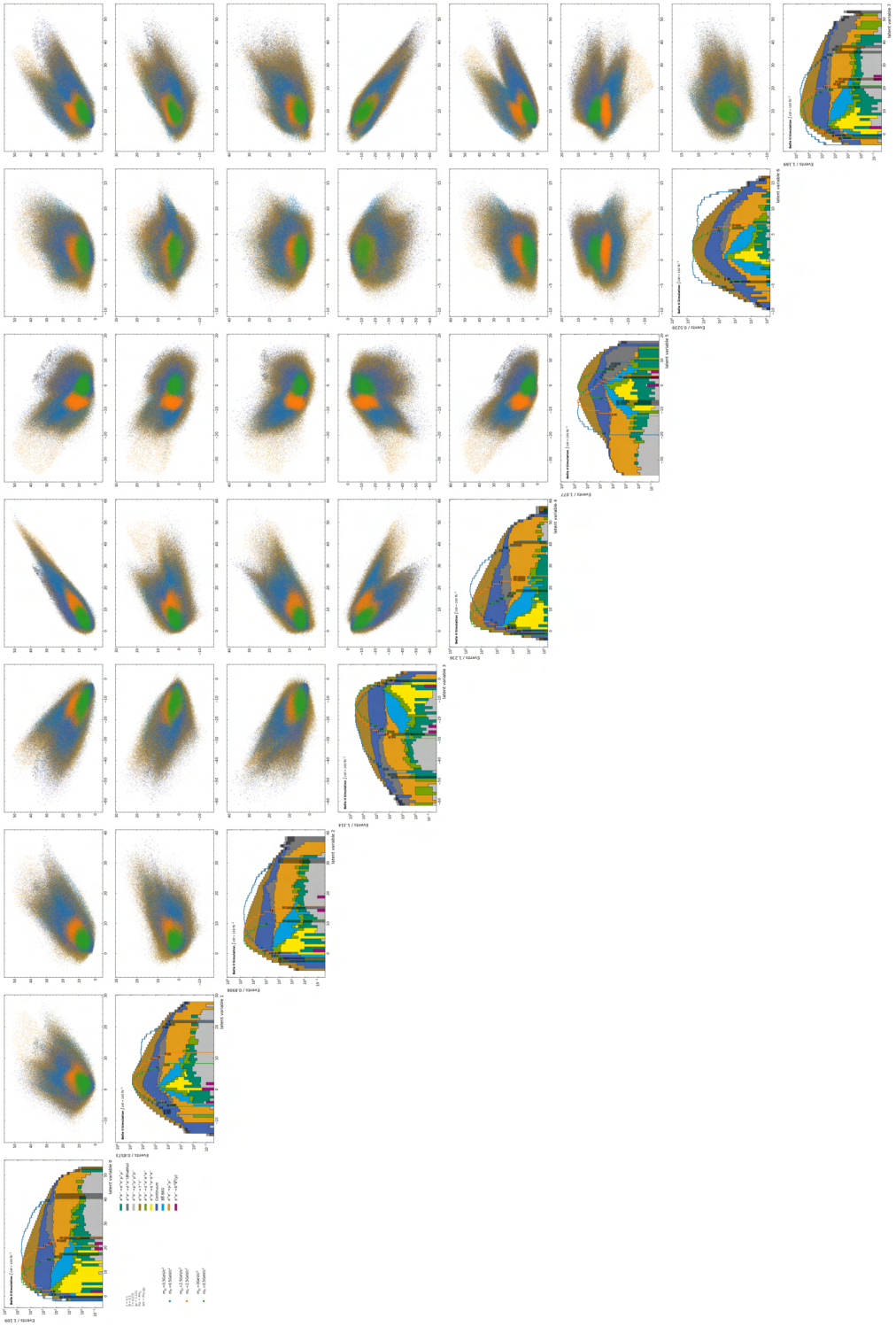

Figure A.38.: Latent variables and their correlations for the 8-dimensional AE for the background samples and the three example signals.

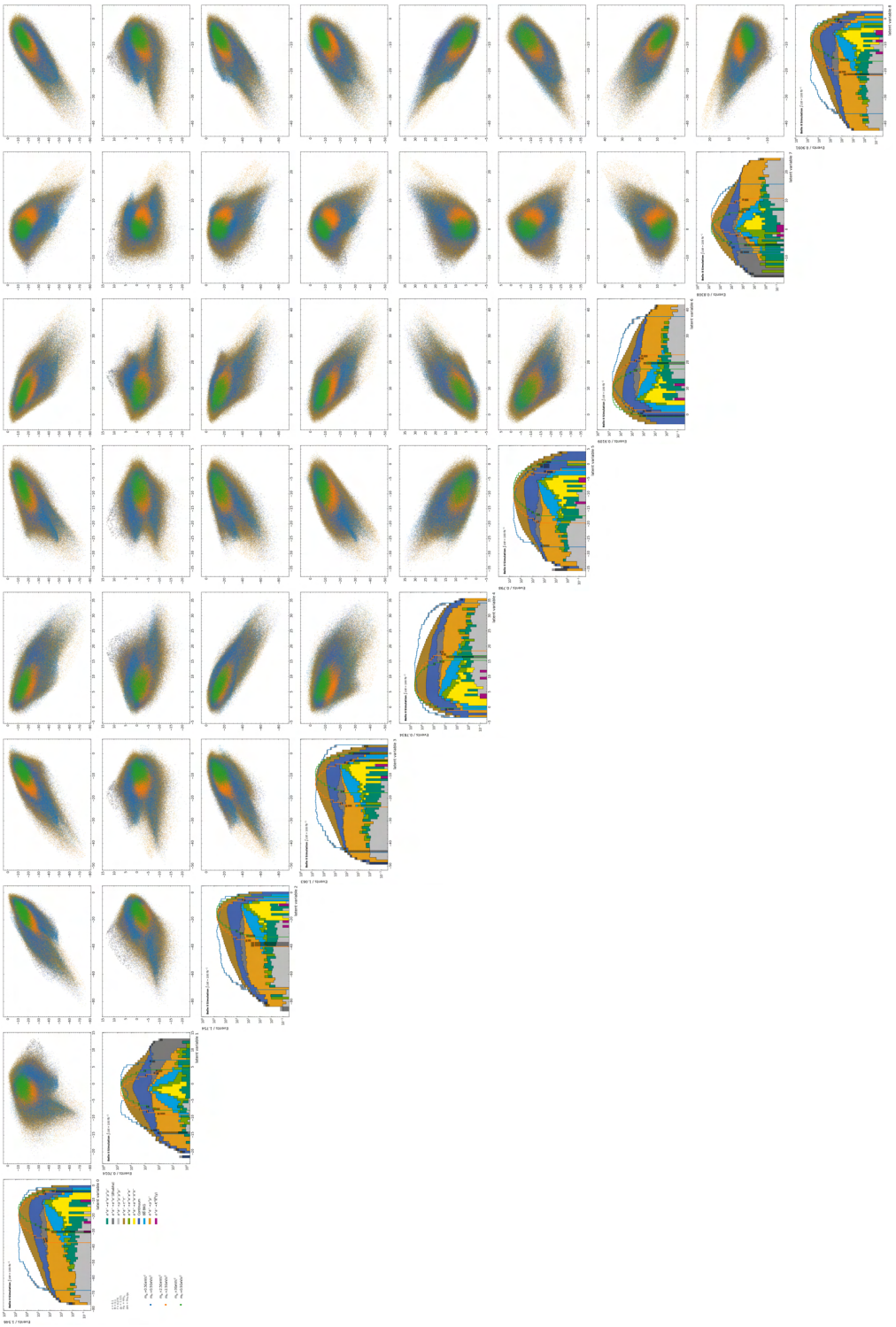

Figure A.39.: Latent variables and their correlations for the 9-dimensional AE for the background samples and the three example signals.

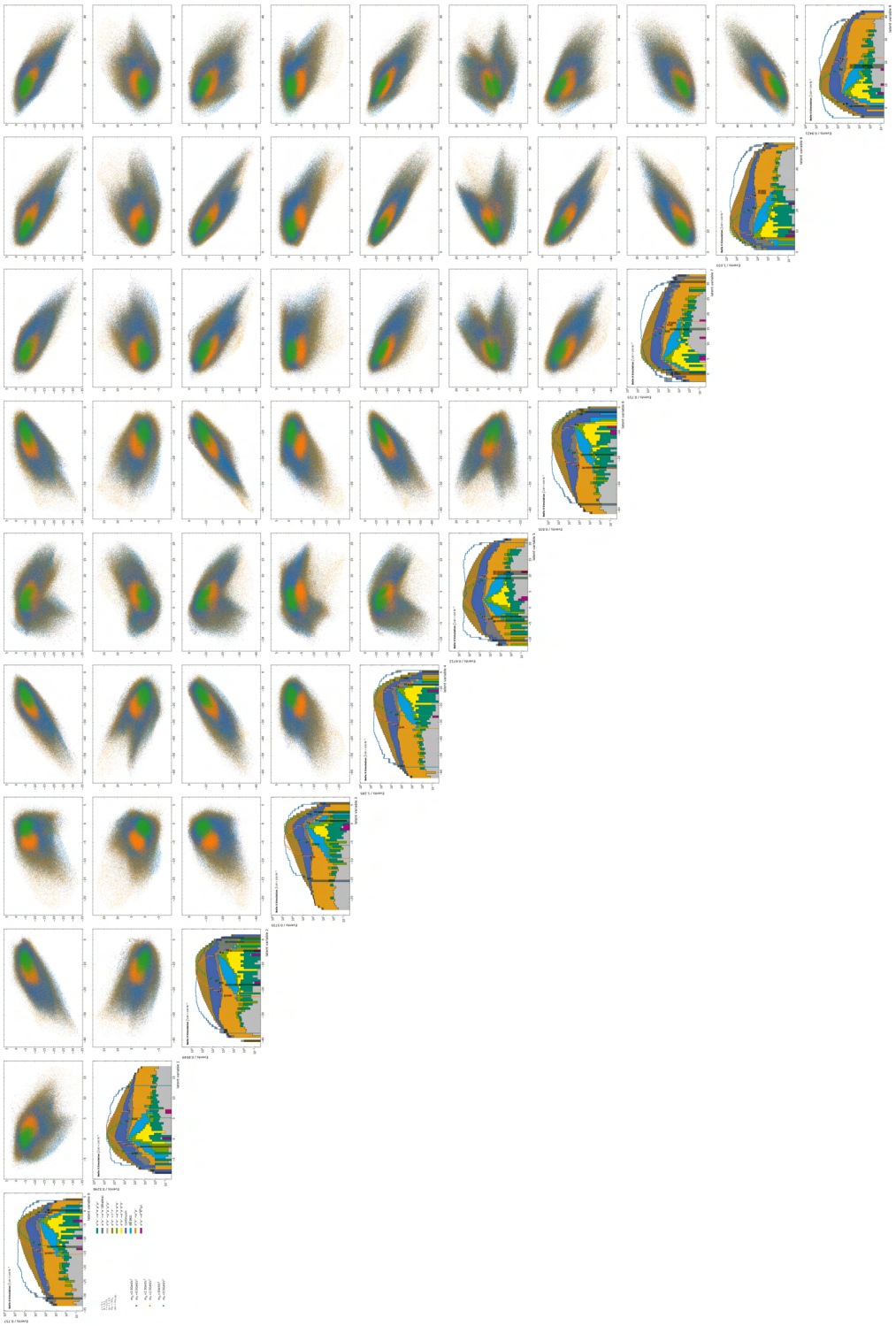

Figure A.40.: Latent variables and their correlations for the 10-dimensional AE for the background samples and the three example signals.

A.5. Training Details for VAEs

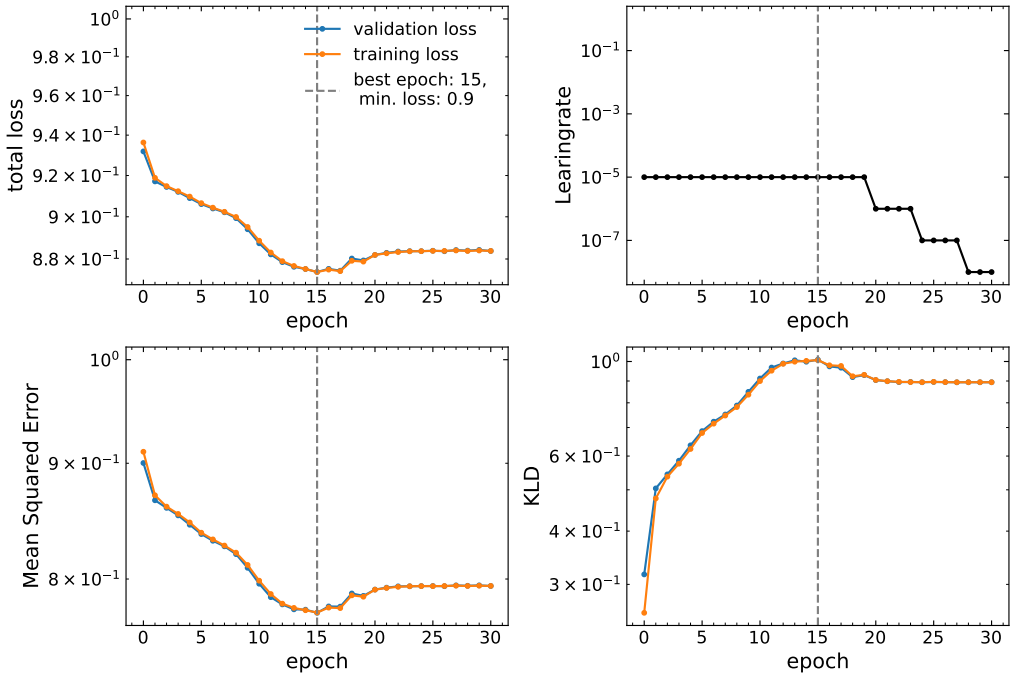

Figure A.41.: Training details for the training of an VAE with 1-dimensional latent space (top). The total loss is equal to the MSE.

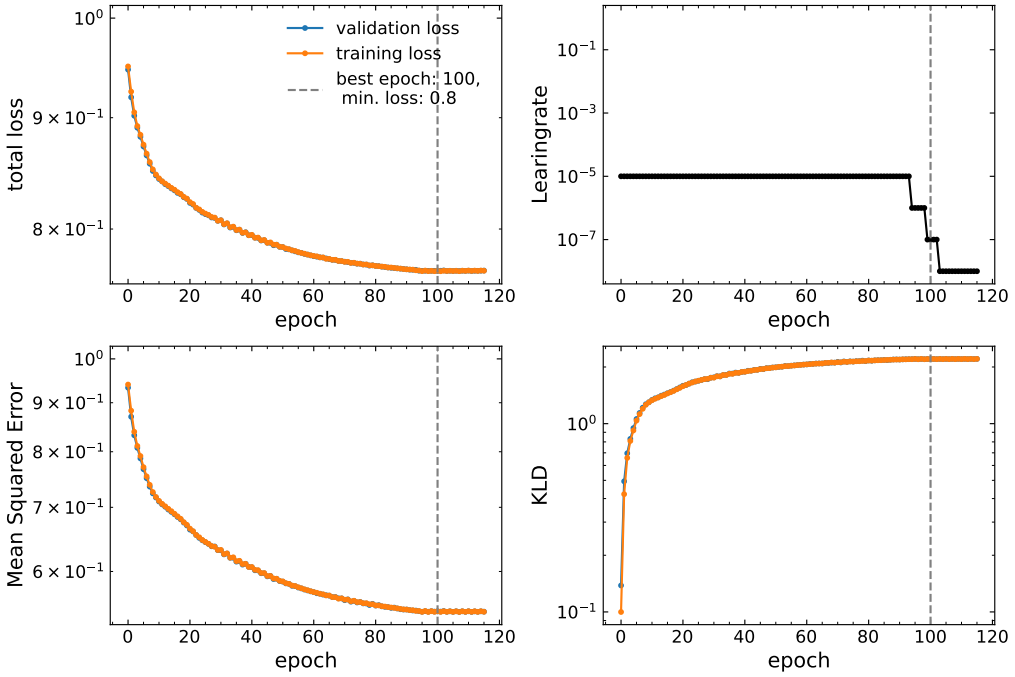

Figure A.42.: Training details for the training of an VAE with 2-dimensional latent space (top). The total loss is equal to the MSE.

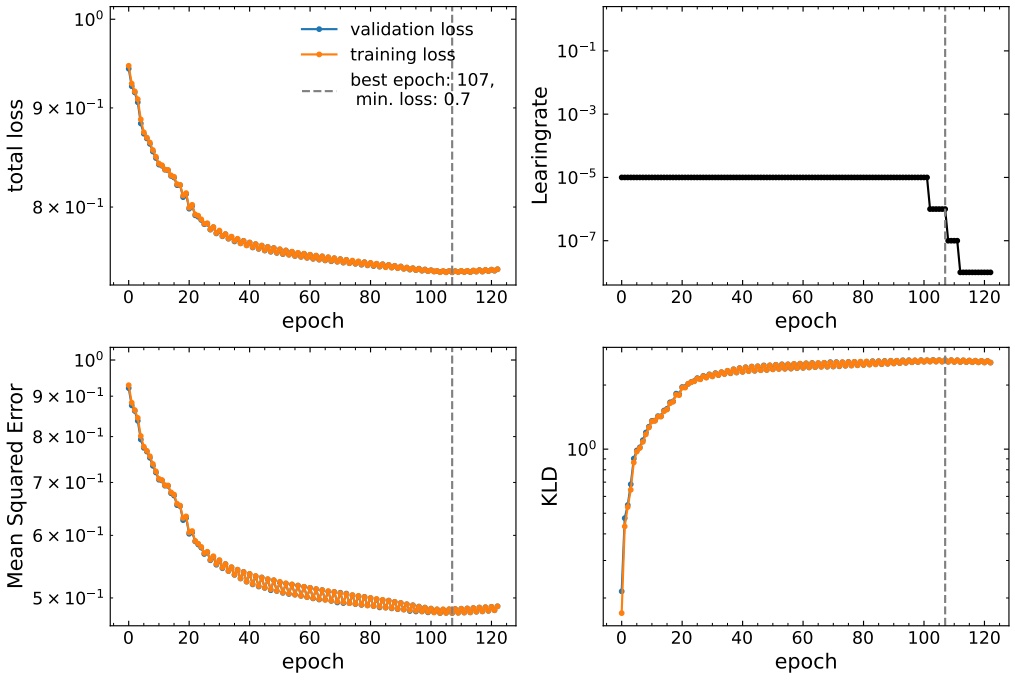

Figure A.43.: Training details for the training of an VAE with 3-dimensional latent space (top). The total loss is equal to the MSE.

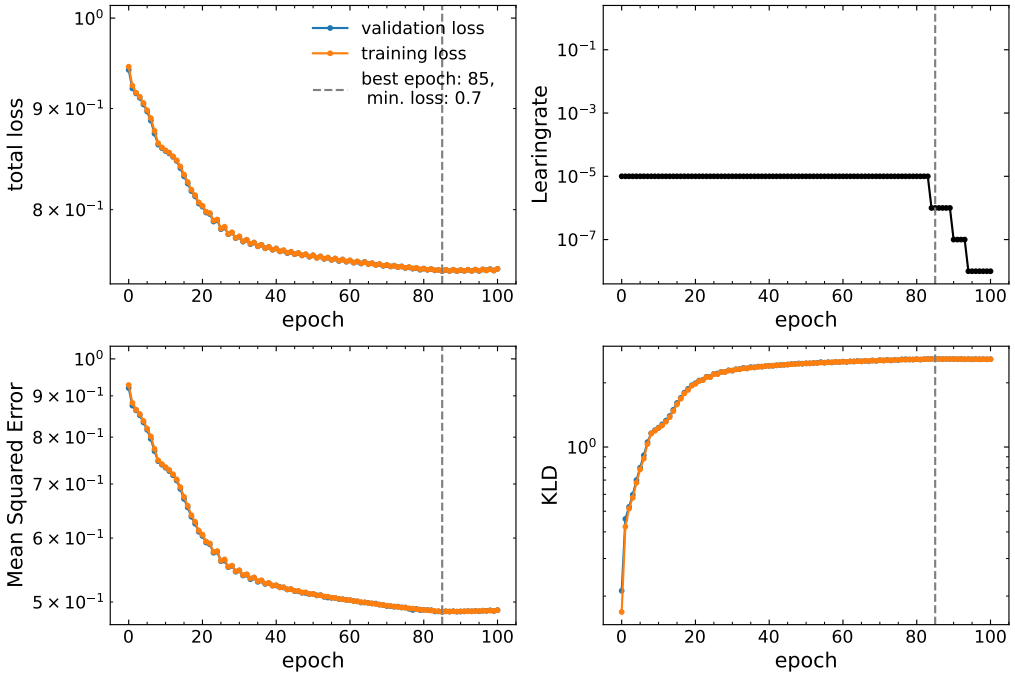

Figure A.44.: Training details for the training of an VAE with 4-dimensional latent space (top). The total loss is equal to the MSE.

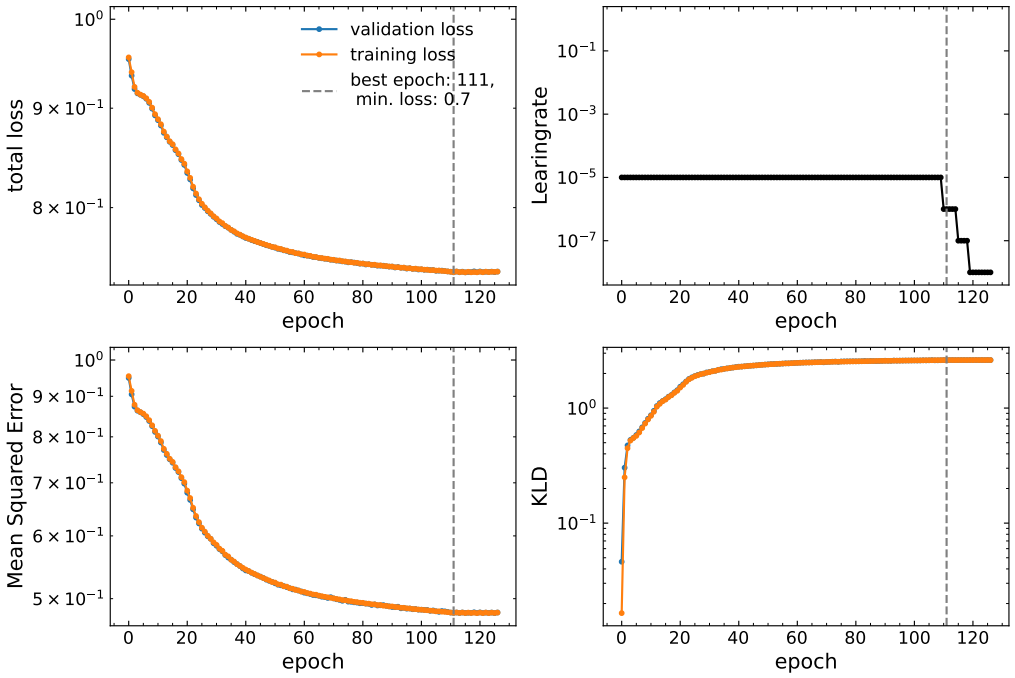

Figure A.45.: Training details for the training of an VAE with 5-dimensional latent space (top). The total loss is equal to the MSE.

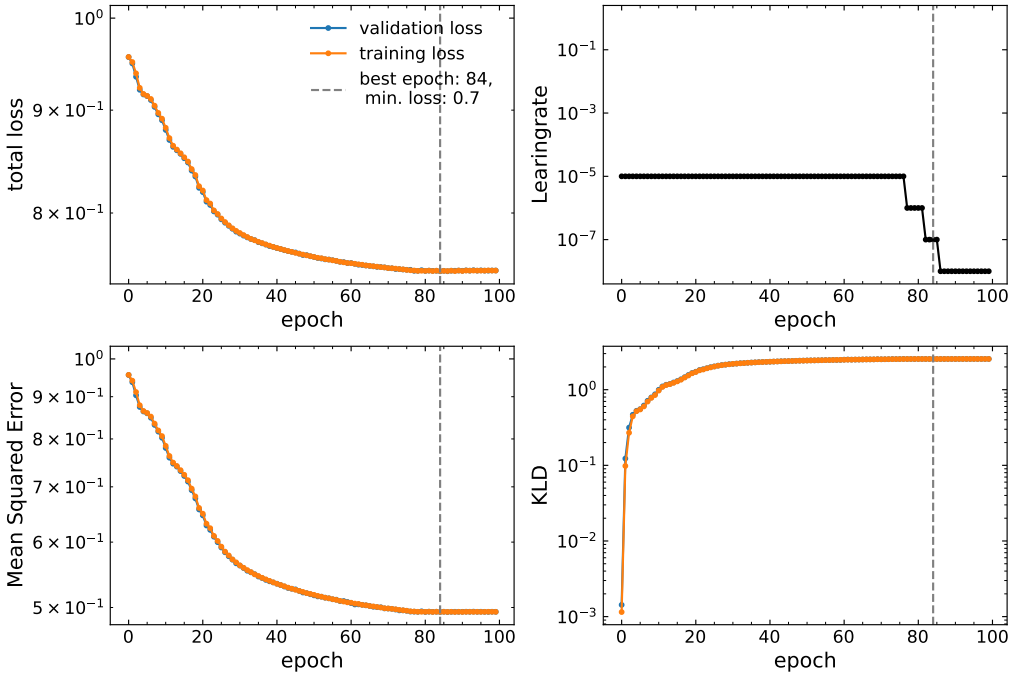

Figure A.46.: Training details for the training of an VAE with 6-dimensional latent space (top). The total loss is equal to the MSE.

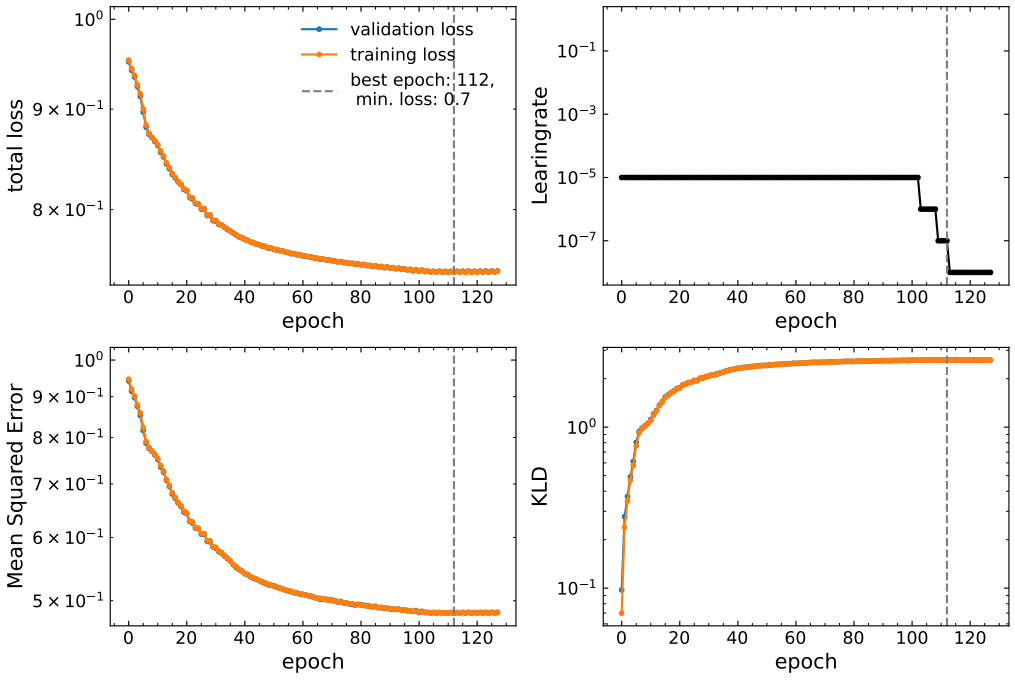

Figure A.47.: Training details for the training of an VAE with 7-dimensional latent space (top). The total loss is equal to the MSE.

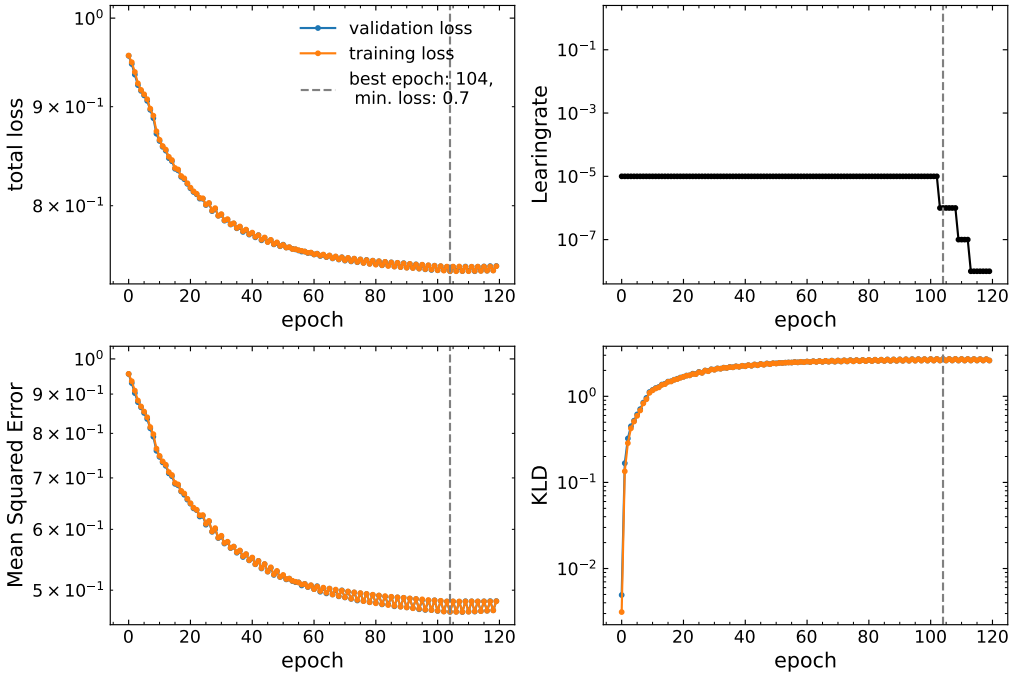

Figure A.48.: Training details for the training of an VAE with 8-dimensional latent space (top). The total loss is equal to the MSE.

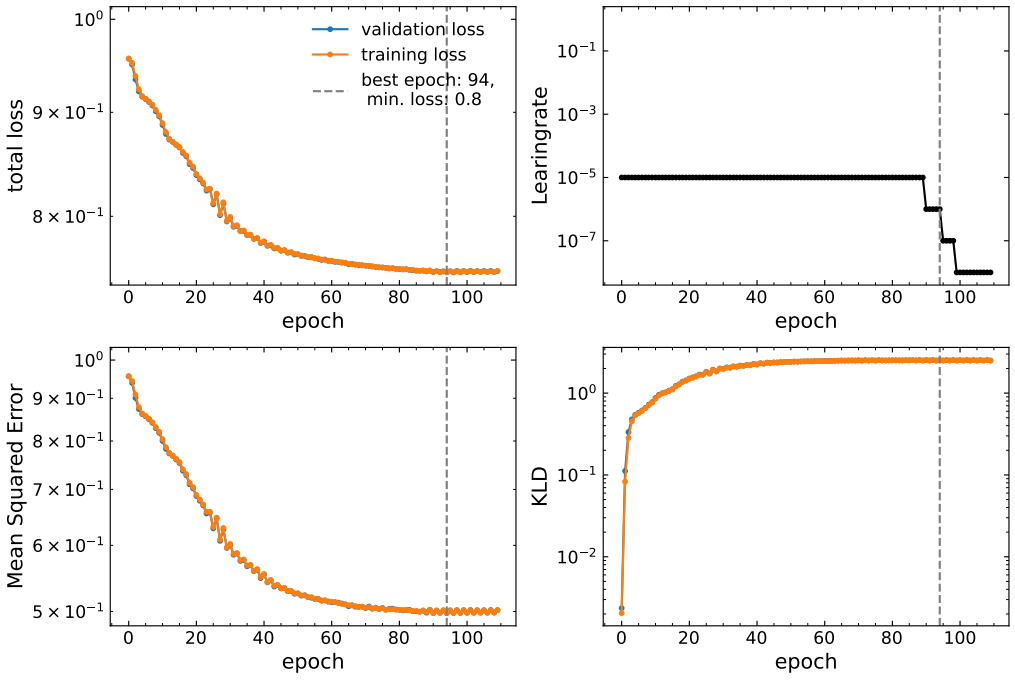

Figure A.49.: Training details for the training of an VAE with 9-dimensional latent space (top). The total loss is equal to the MSE.

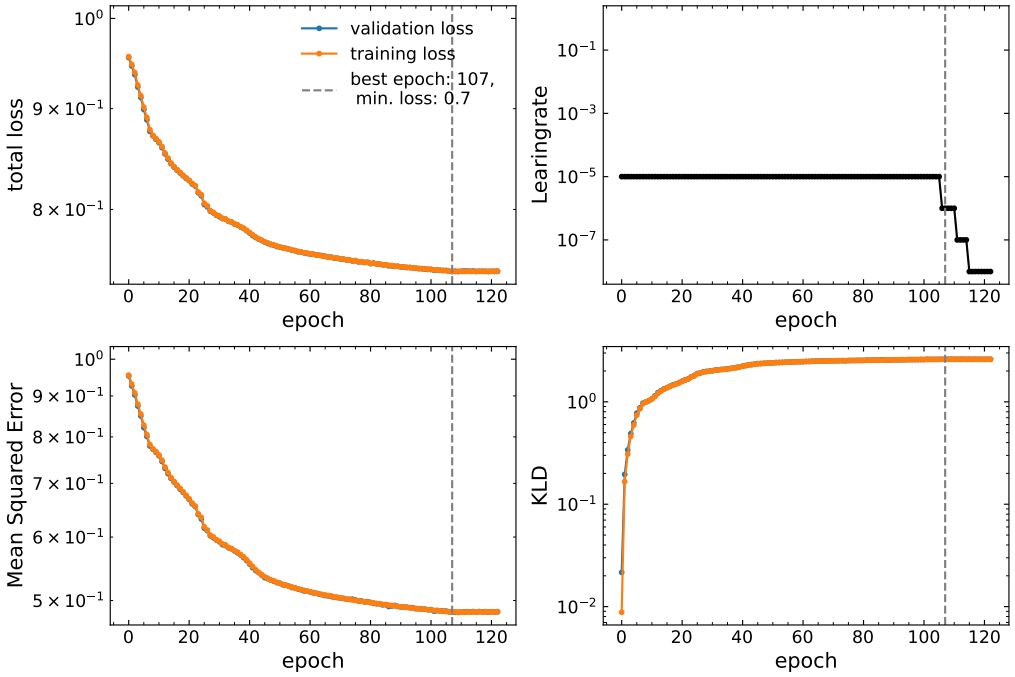

Figure A.50.: Training details for the training of an VAE with 10-dimensional latent space (top). The total loss is equal to the MSE.

A.6. MSE for VAEs

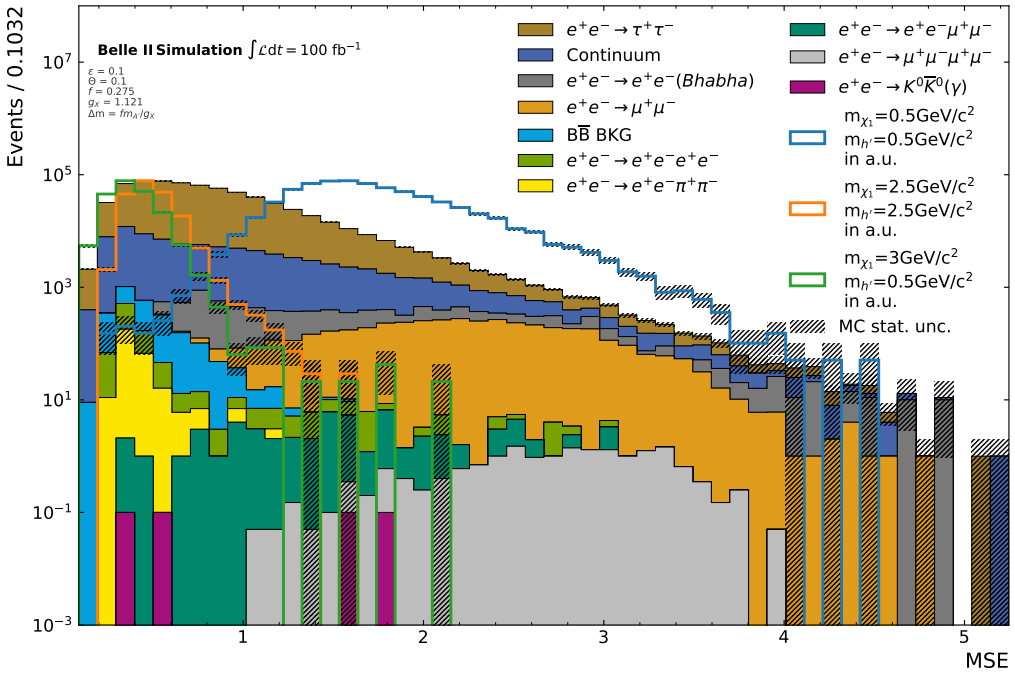

Figure A.51.: Distribution of the MSE for the 1-dimensional VAE.

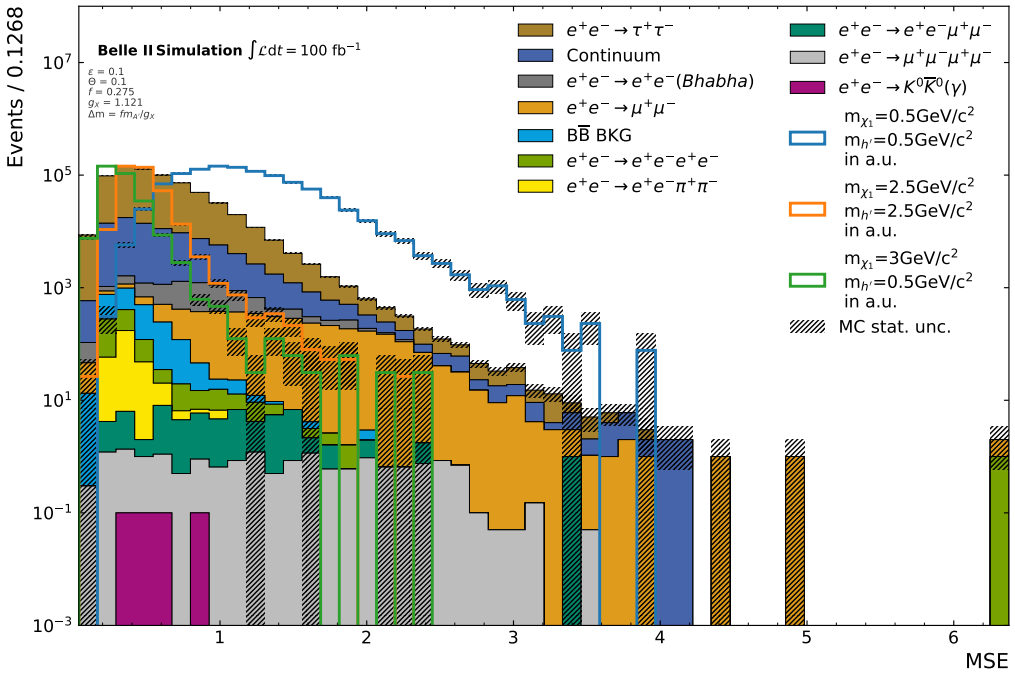

Figure A.52.: Distribution of the MSE for the 2-dimensional VAE.

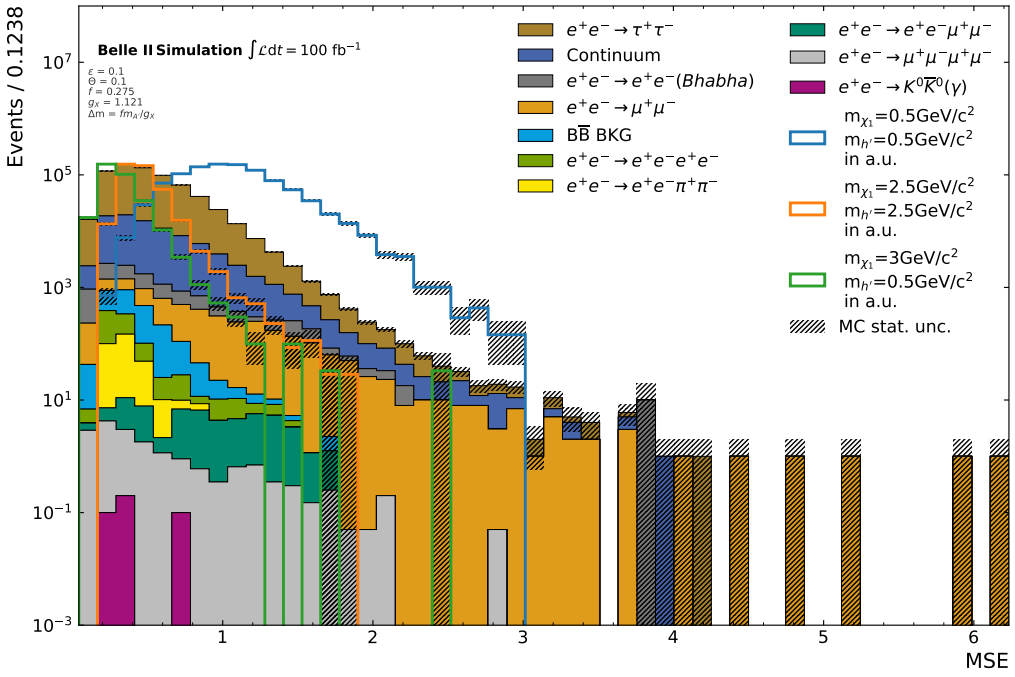

Figure A.53.: Distribution of the MSE for the 3-dimensional VAE.

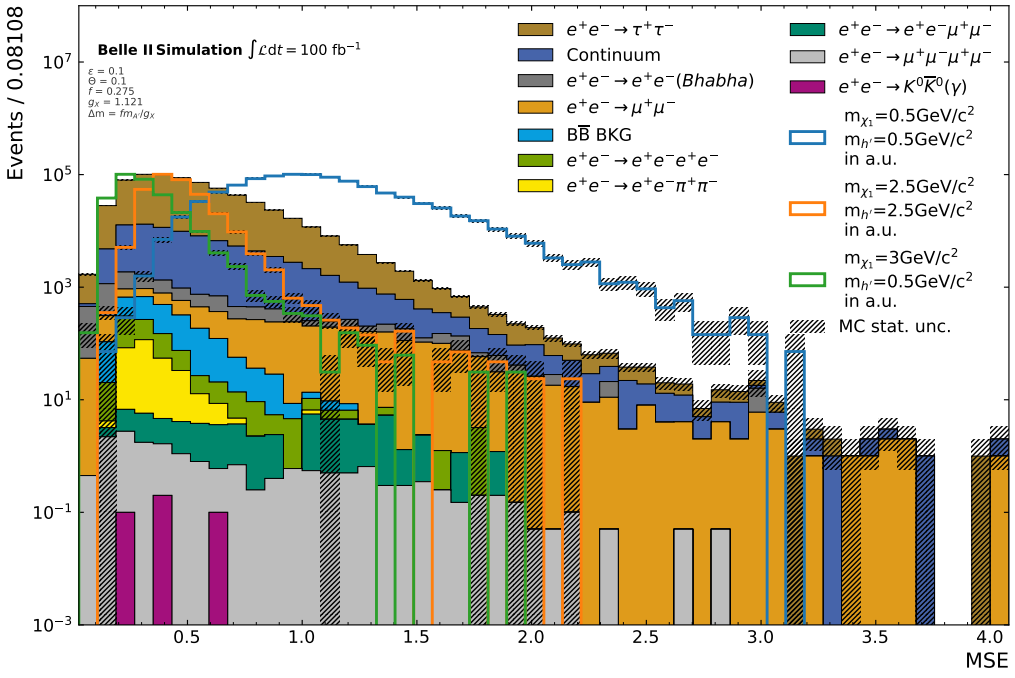

Figure A.54.: Distribution of the MSE for the 4-dimensional VAE.

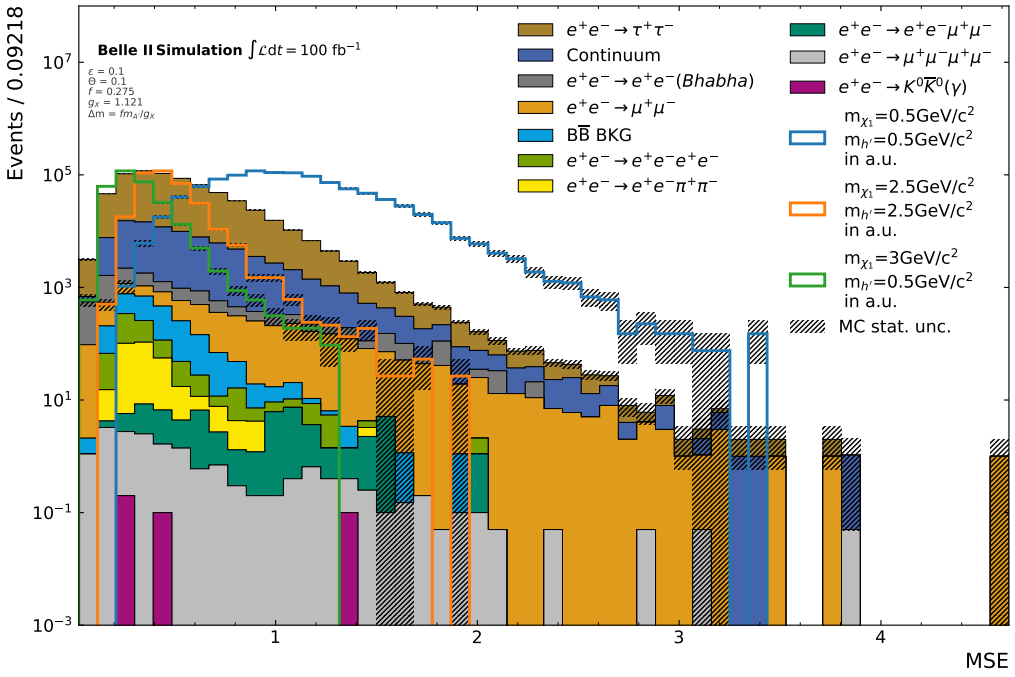

Figure A.55.: Distribution of the MSE for the 5-dimensional VAE.

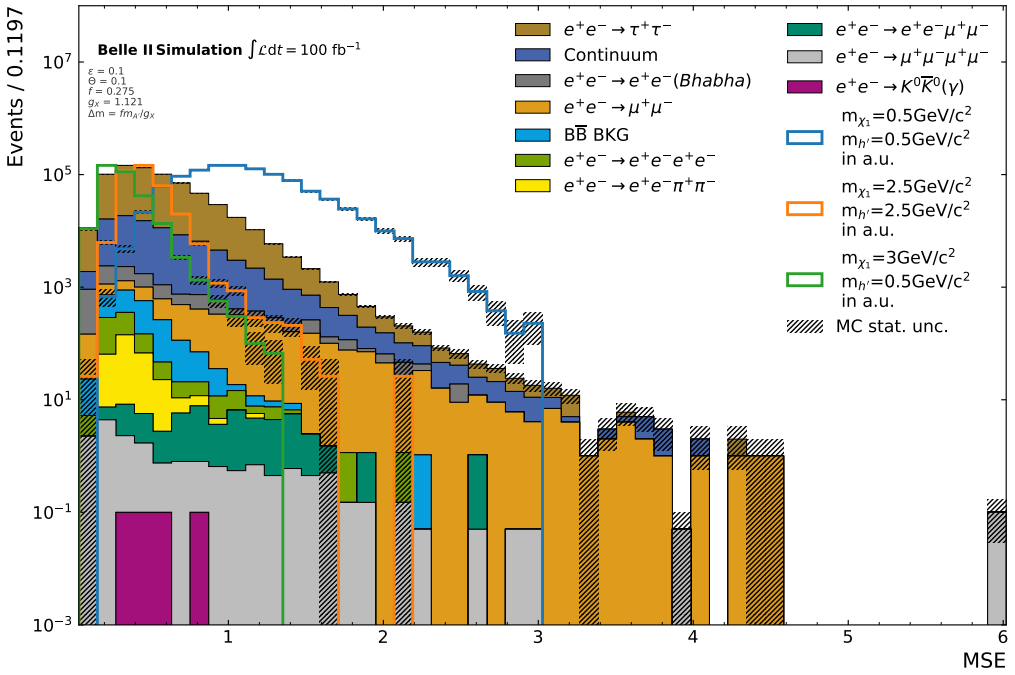

Figure A.56.: Distribution of the MSE for the 6-dimensional VAE.

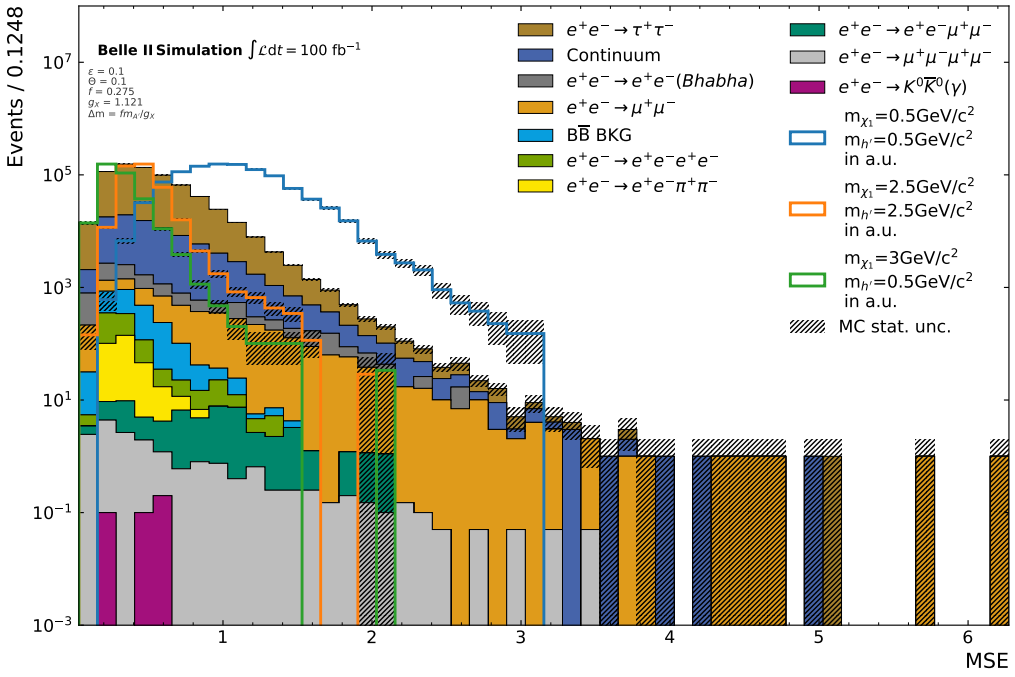

Figure A.57.: Distribution of the MSE for the 7-dimensional VAE.

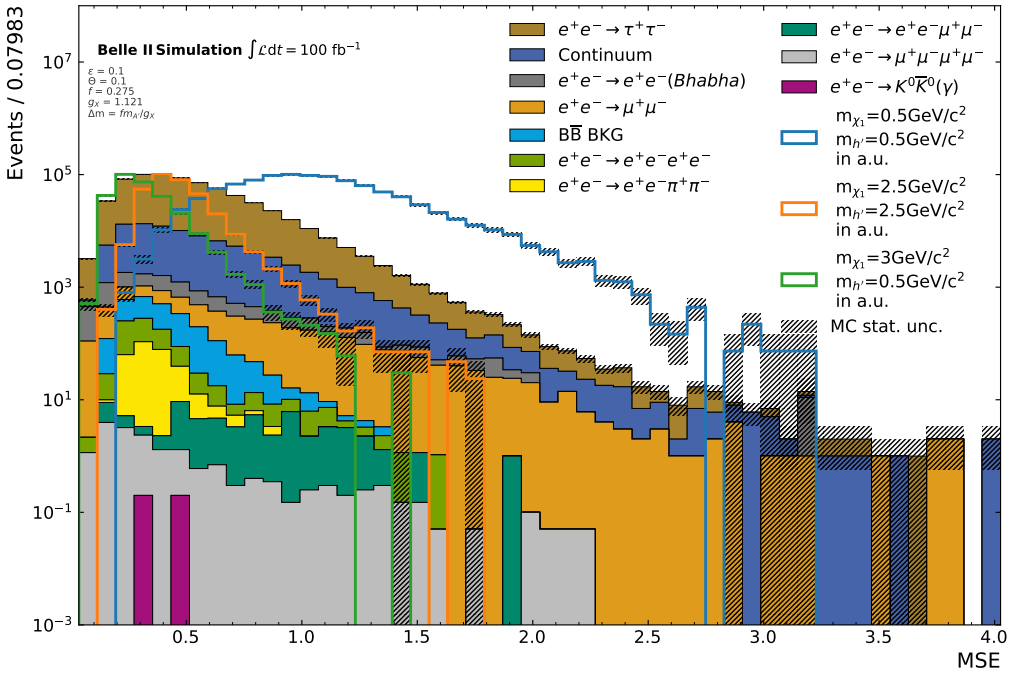

Figure A.58.: Distribution of the MSE for the 8-dimensional VAE.

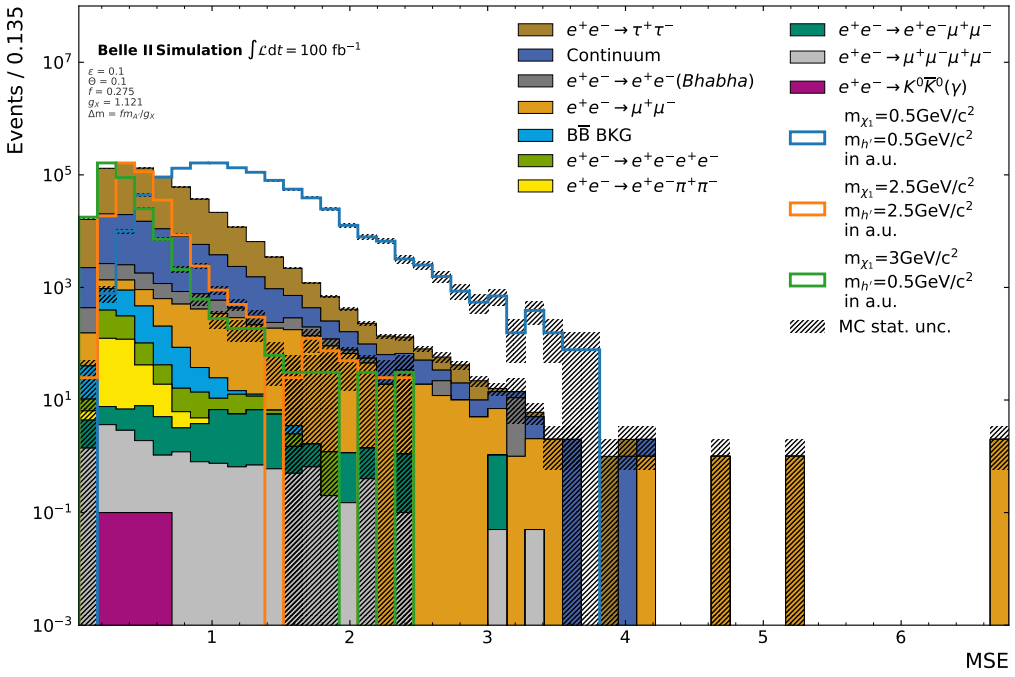

Figure A.59.: Distribution of the MSE for the 9-dimensional VAE.

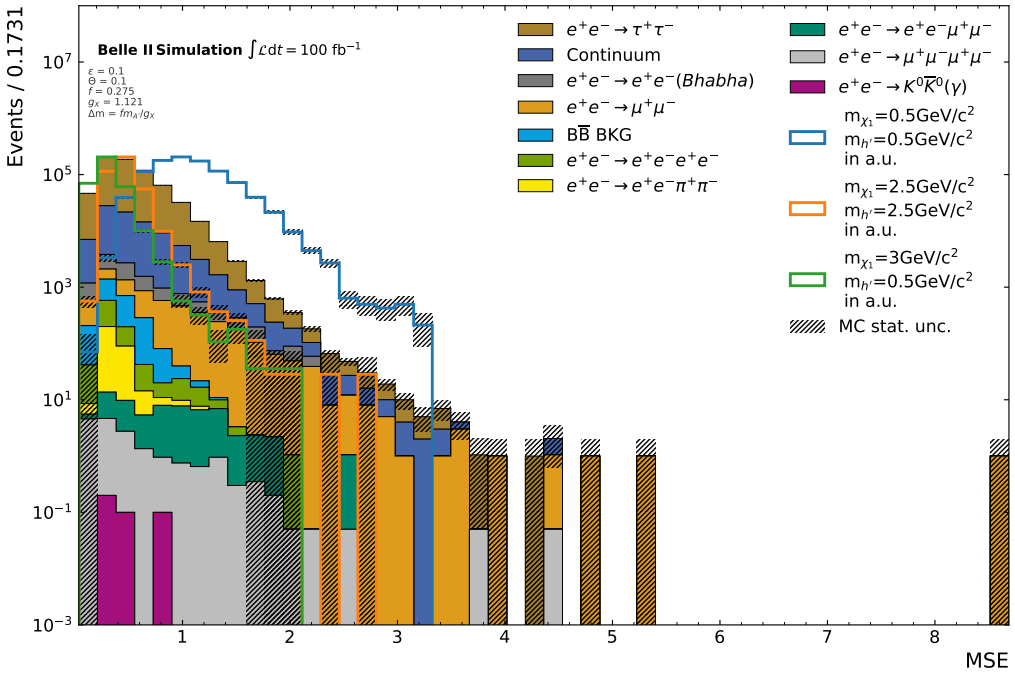

Figure A.60.: Distribution of the MSE for the 10-dimensional VAE.

A.7. Latentspace of VAEs

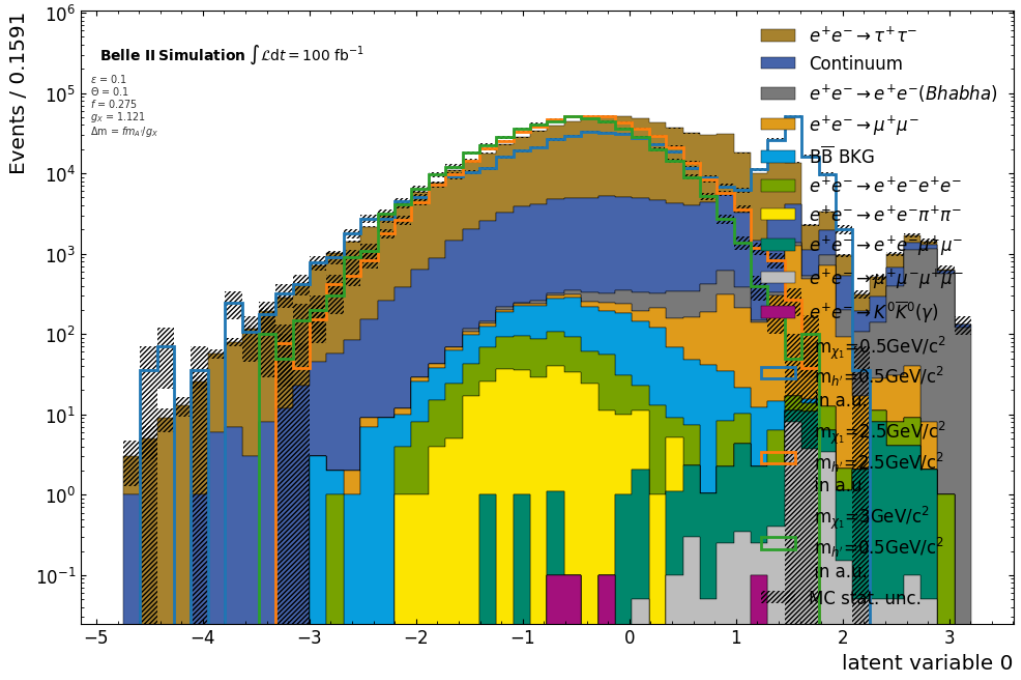

Figure A.61.: Latent variables and their correlations for the 1-dimensional VAE for the background samples and the three example signals.

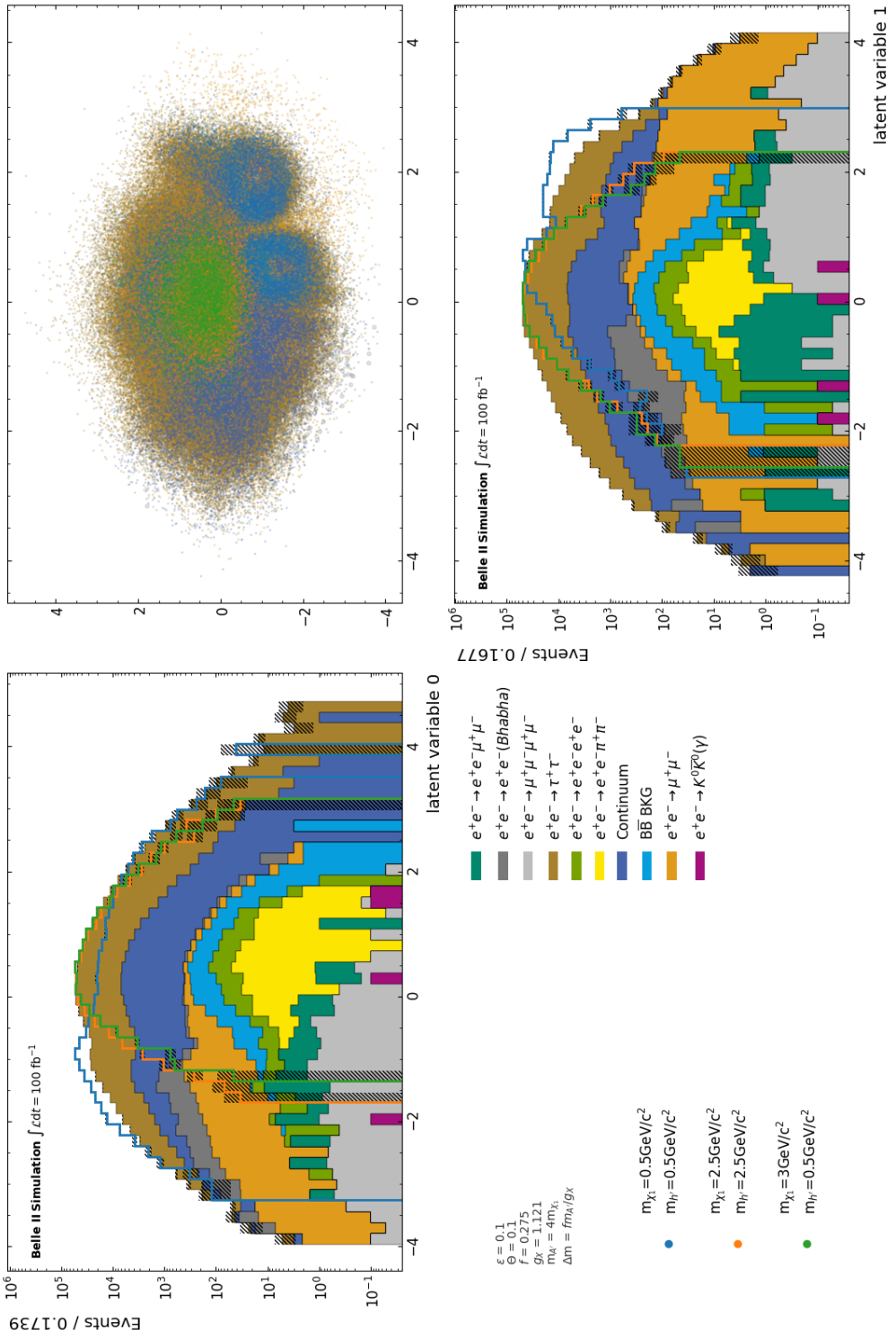

Figure A.62.: Latent variables and their correlations for the 2-dimensional VAE for the background samples and the three example signals.

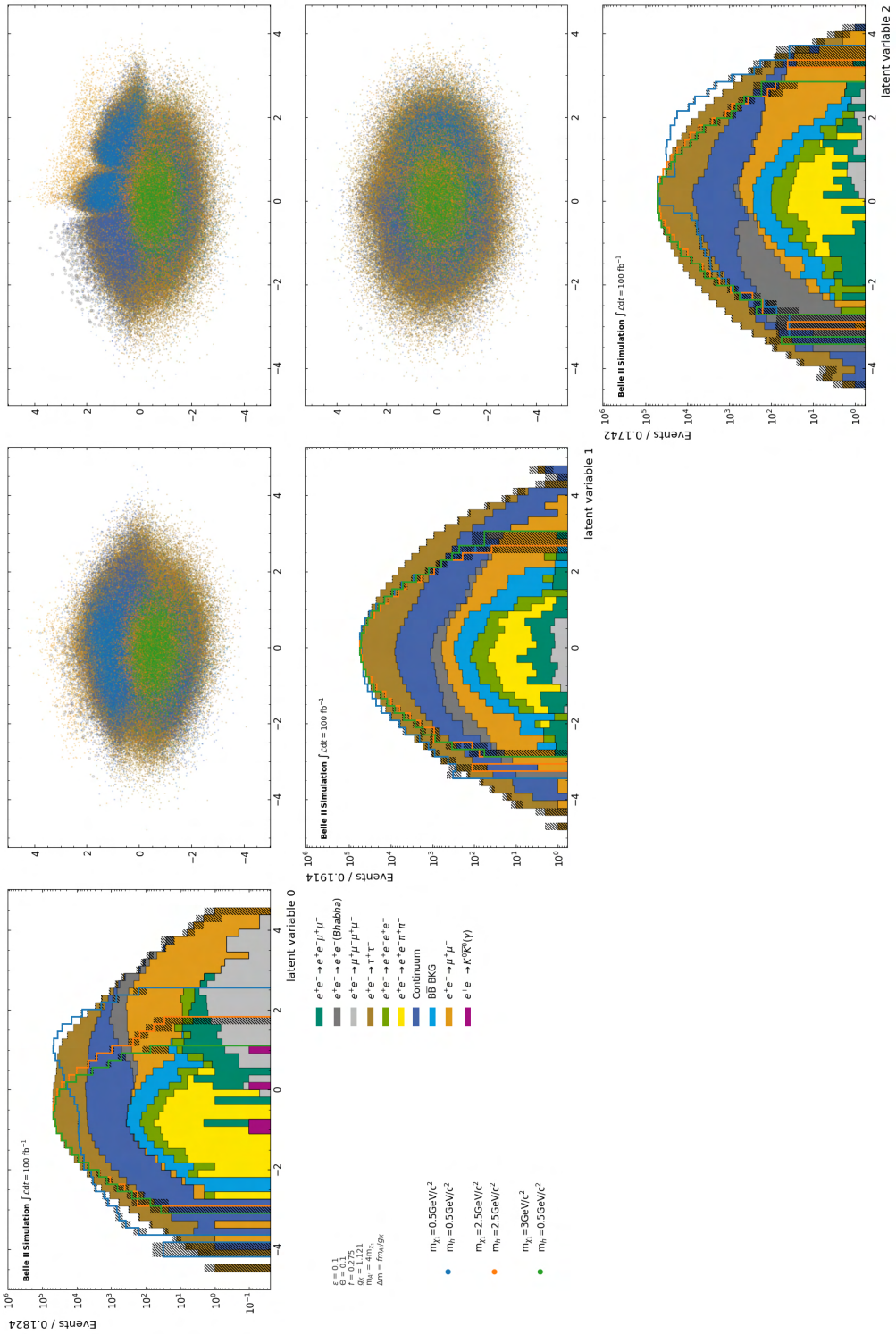

Figure A.63.: Latent variables and their correlations for the 3-dimensional VAE for the background samples and the three example signals.

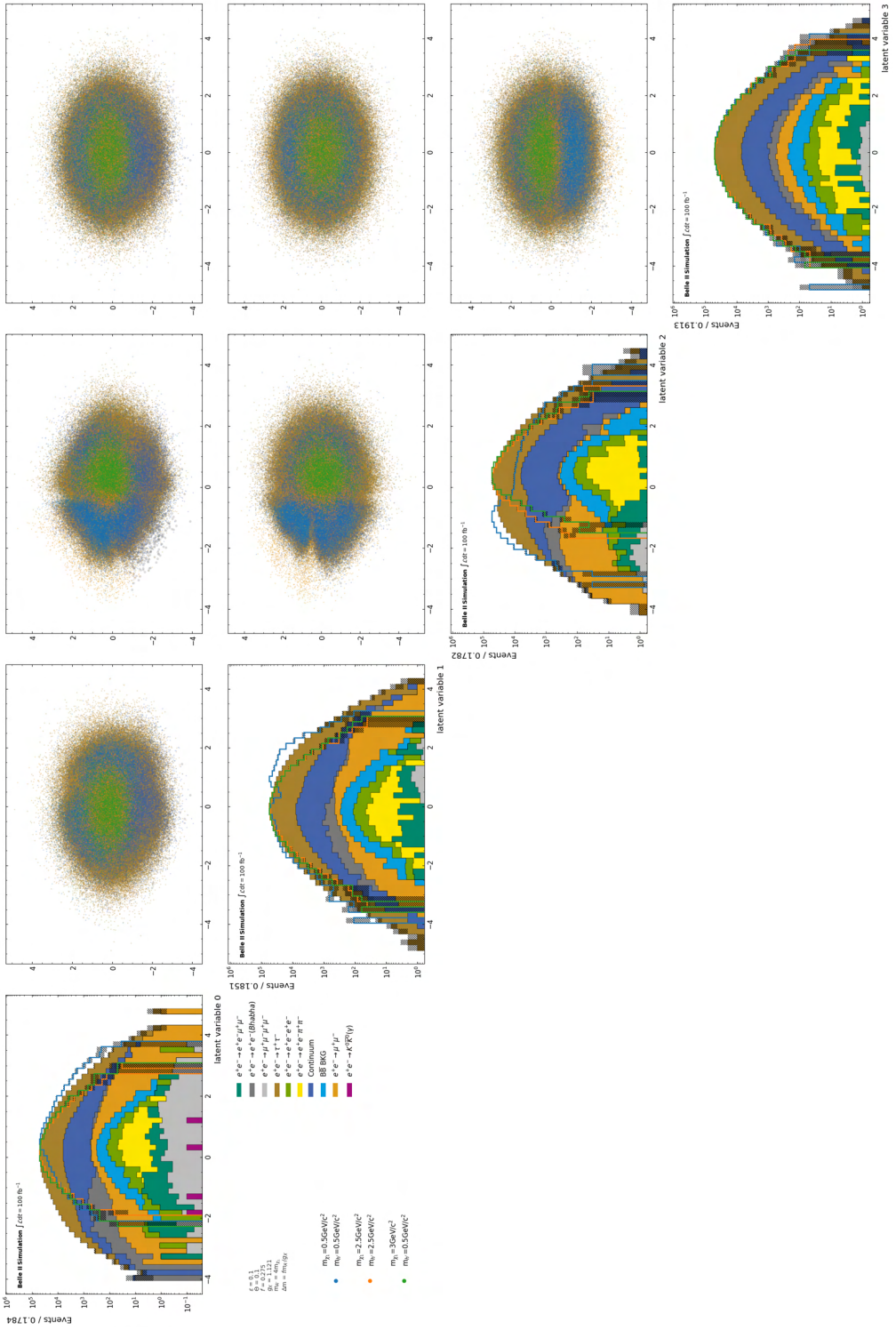

Figure A.64.: Latent variables and their correlations for the 4-dimensional VAE for the background samples and the three example signals.

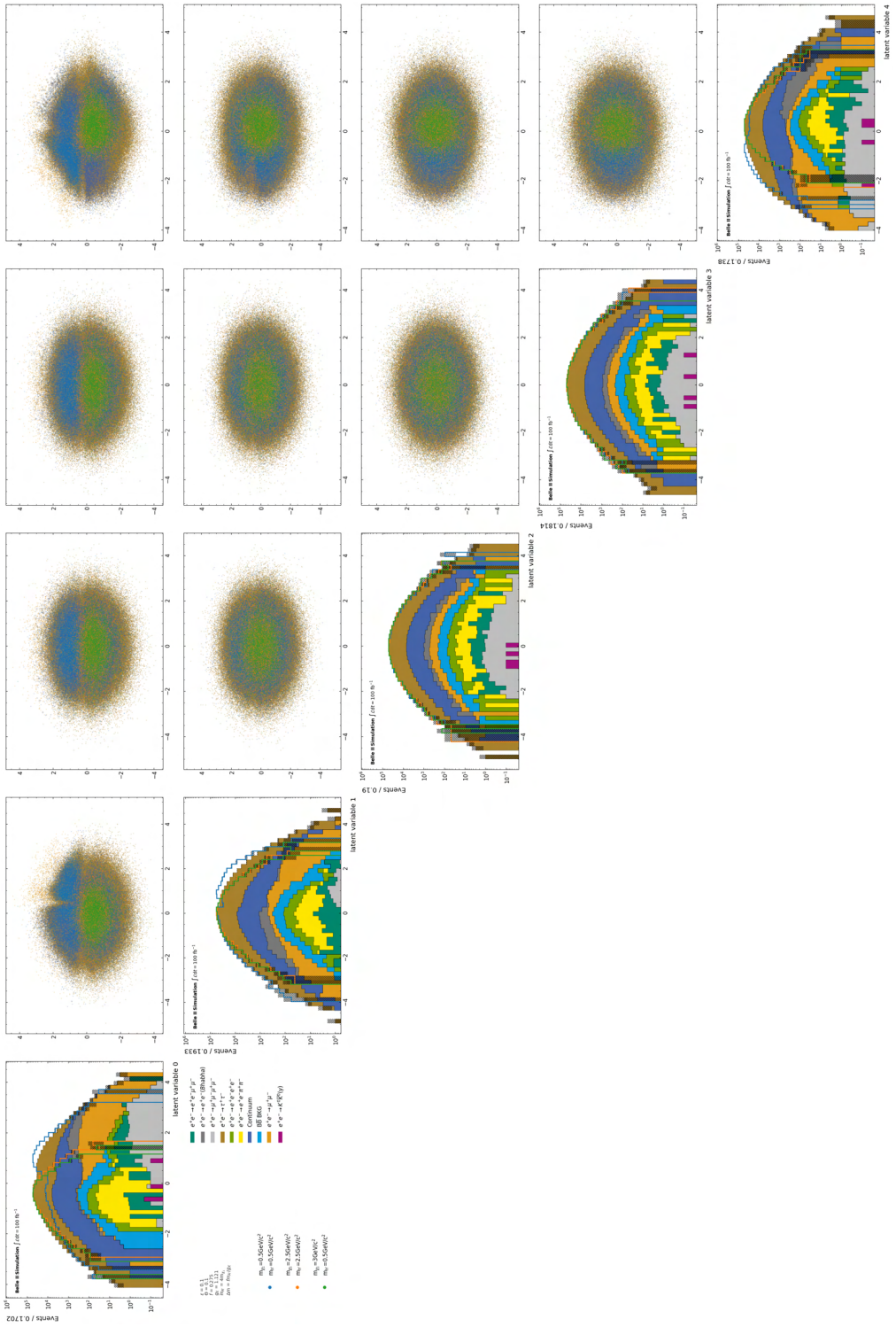

Figure A.65.: Latent variables and their correlations for the 5-dimensional VAE for the background samples and the three example signals.

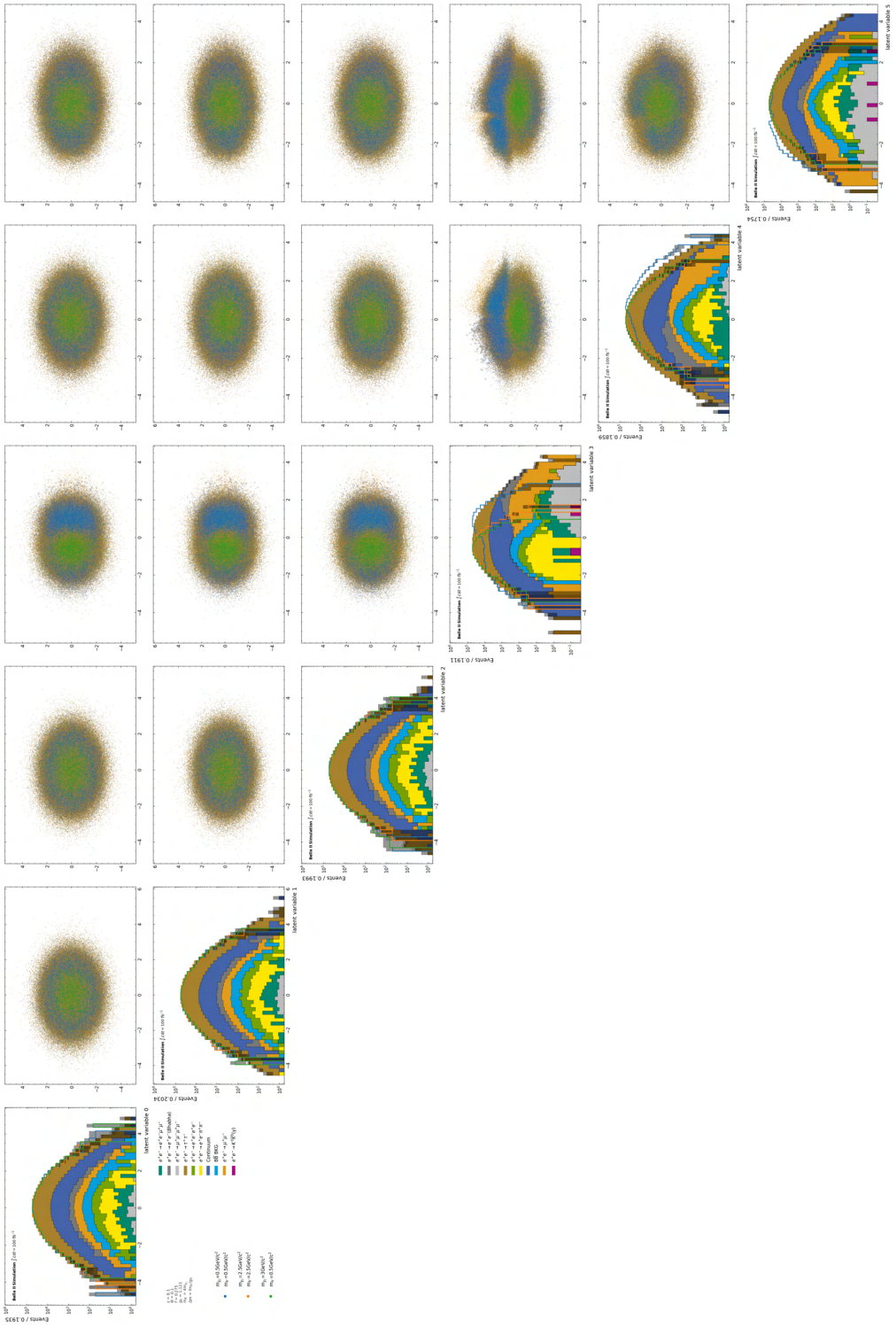

Figure A.66.: Latent variables and their correlations for the 6-dimensional VAE for the background samples and the three example signals.

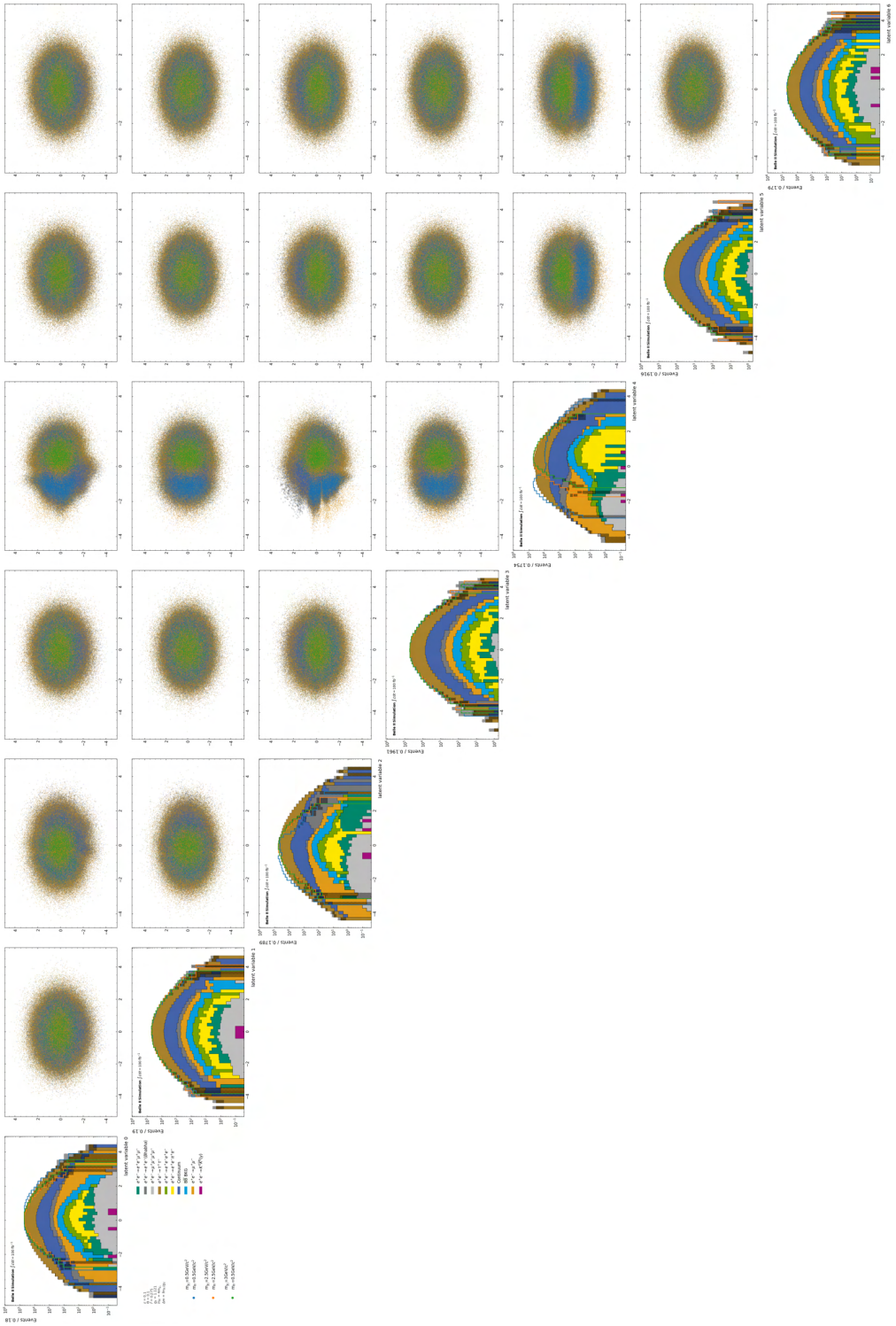

Figure A.67.: Latent variables and their correlations for the 7-dimensional VAE for the background samples and the three example signals.

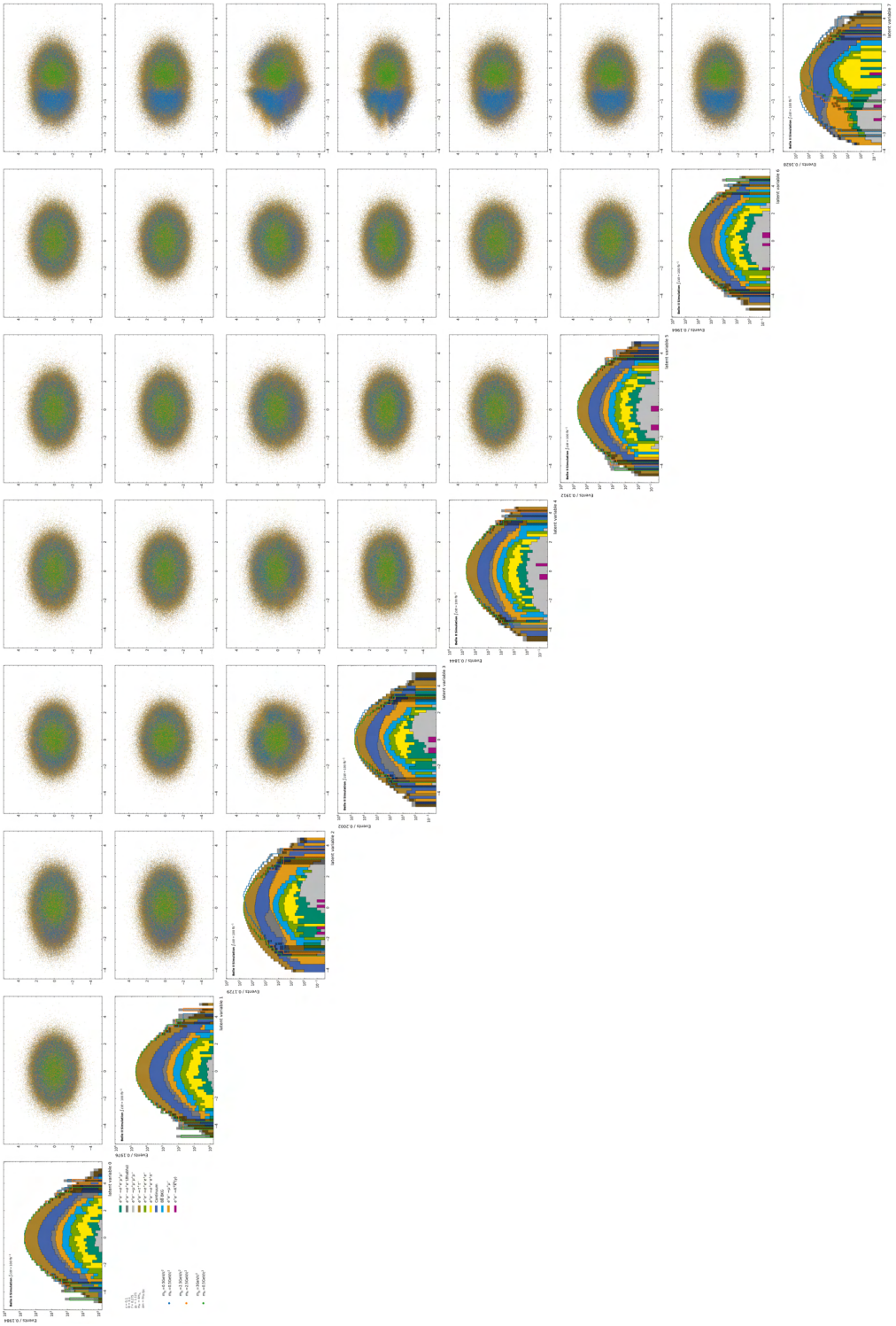

Figure A.68.: Latent variables and their correlations for the 8-dimensional VAE for the background samples and the three example signals.

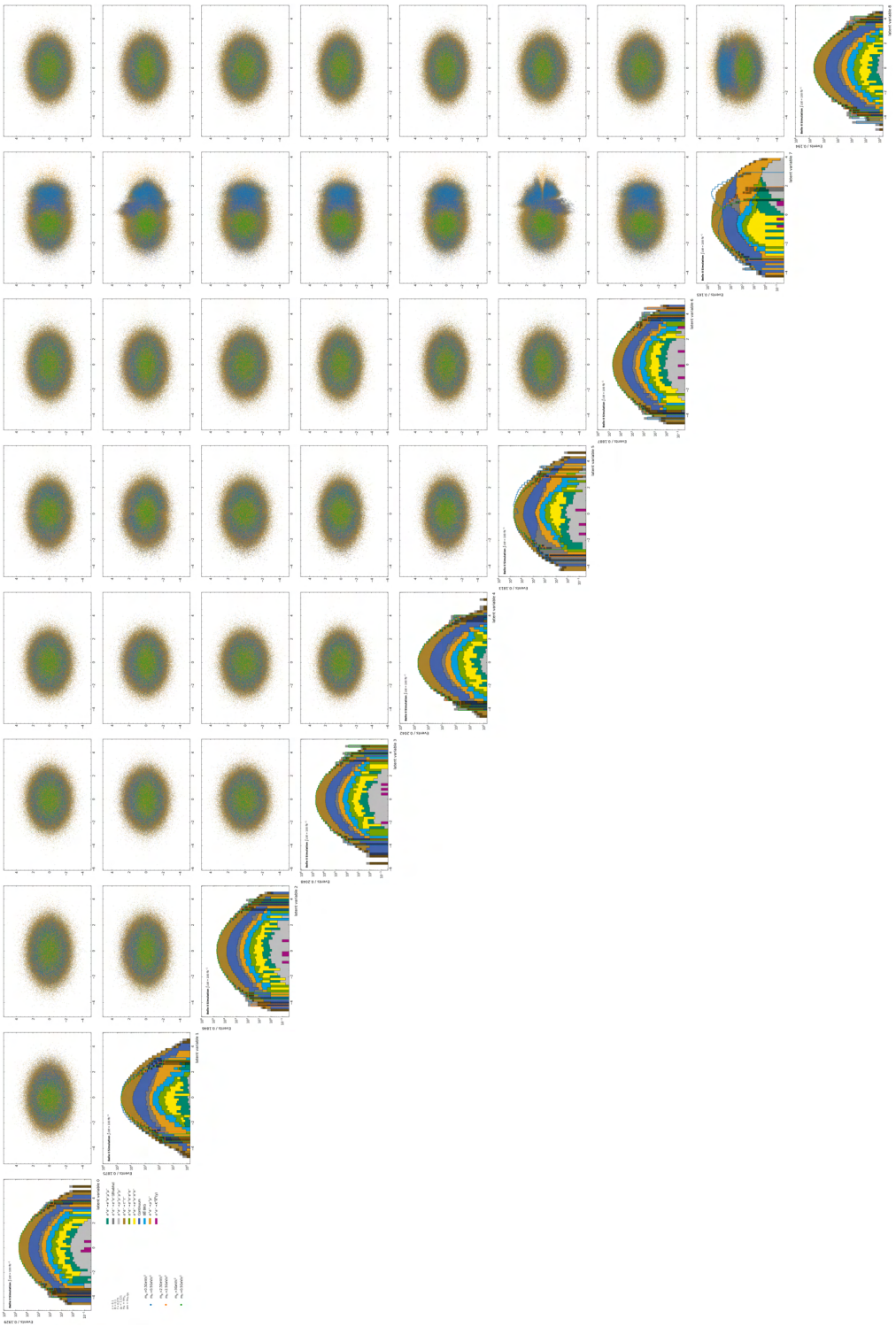

Figure A.69.: Latent variables and their correlations for the 9-dimensional VAE for the background samples and the three example signals.

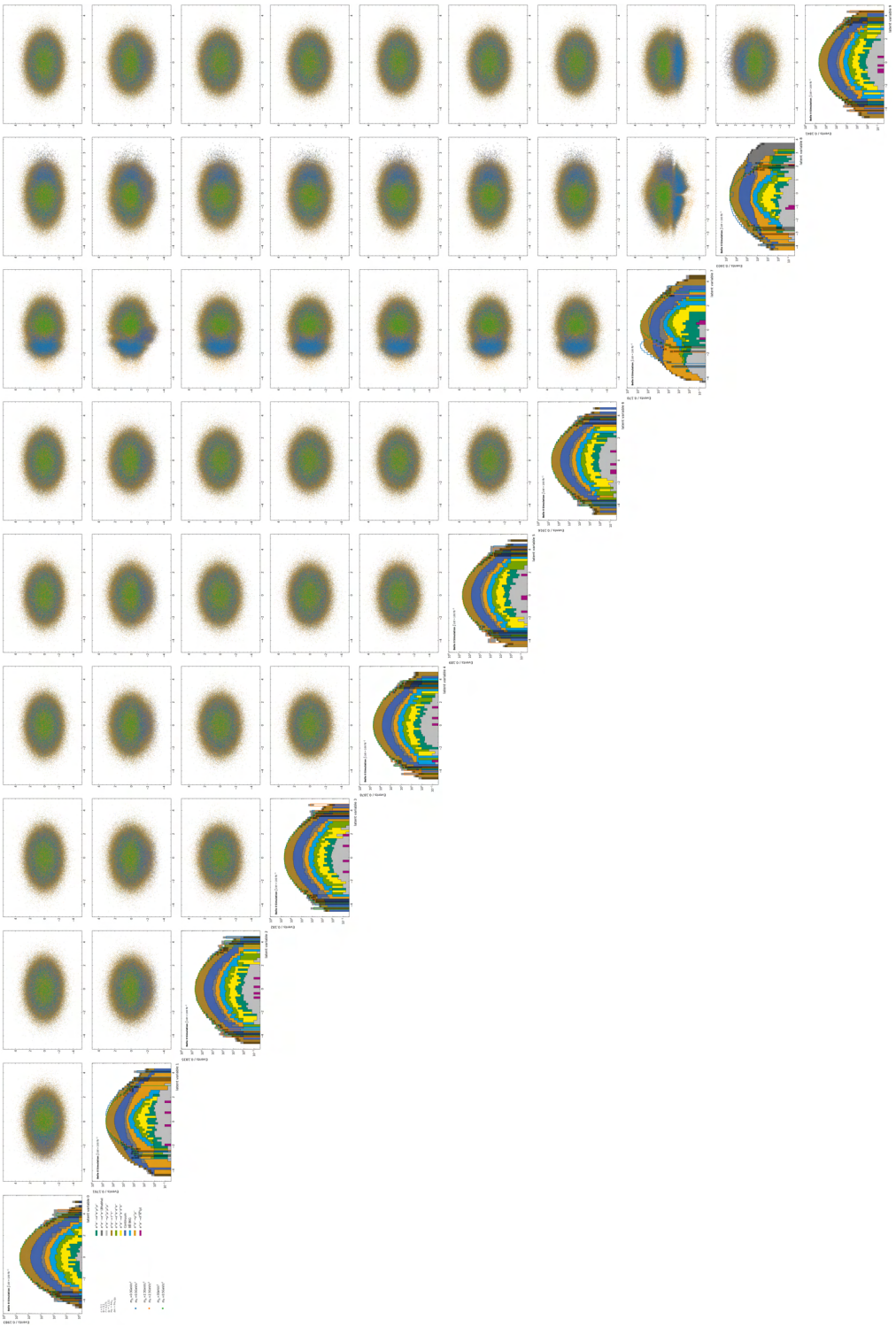

Figure A.70.: Latent variables and their correlations for the 10-dimensional VAE for the background samples and the three example signals.

A.8. Training Details for DVAEs

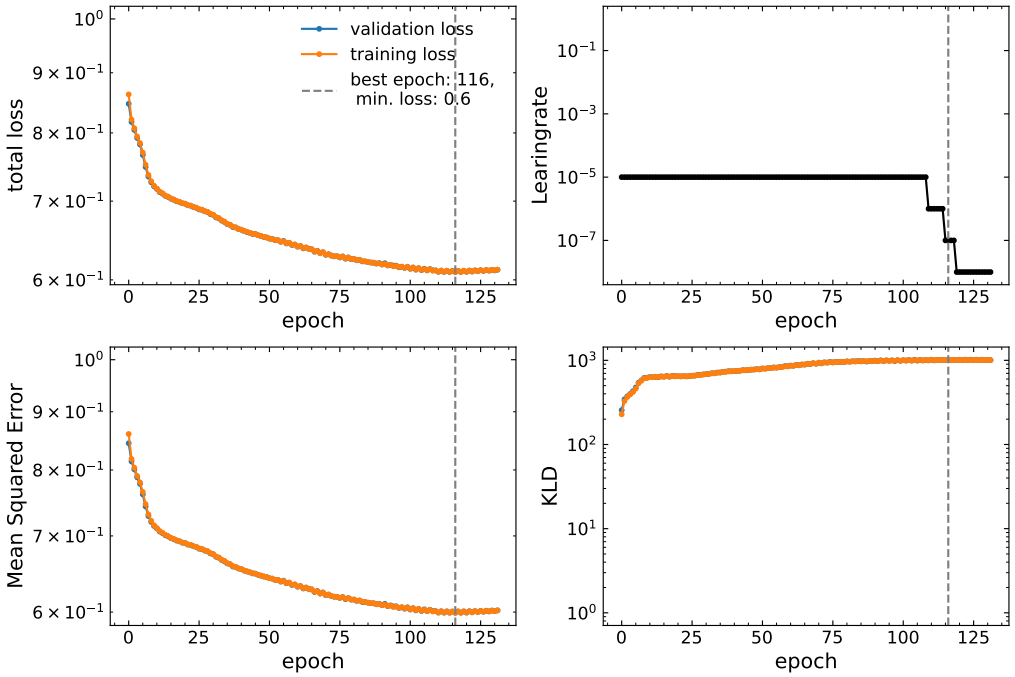

Figure A.71.: Training details for the training of an DVAE with 2-dimensional latent space (top). The total loss is equal to the MSE.

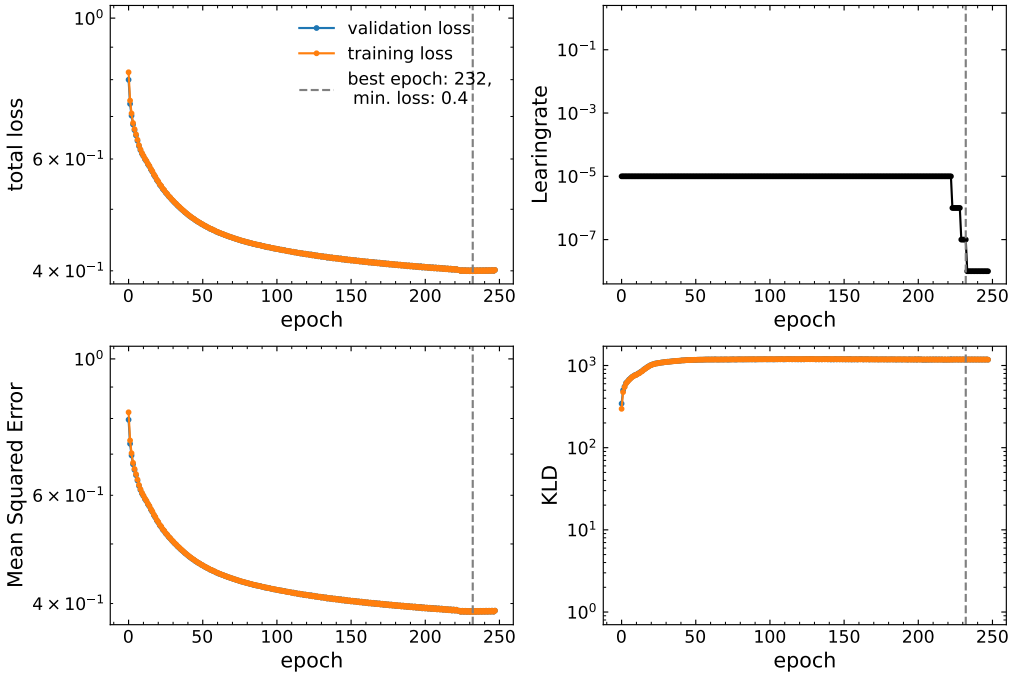

Figure A.72.: Training details for the training of an DVAE with 3-dimensional latent space (top). The total loss is equal to the MSE.

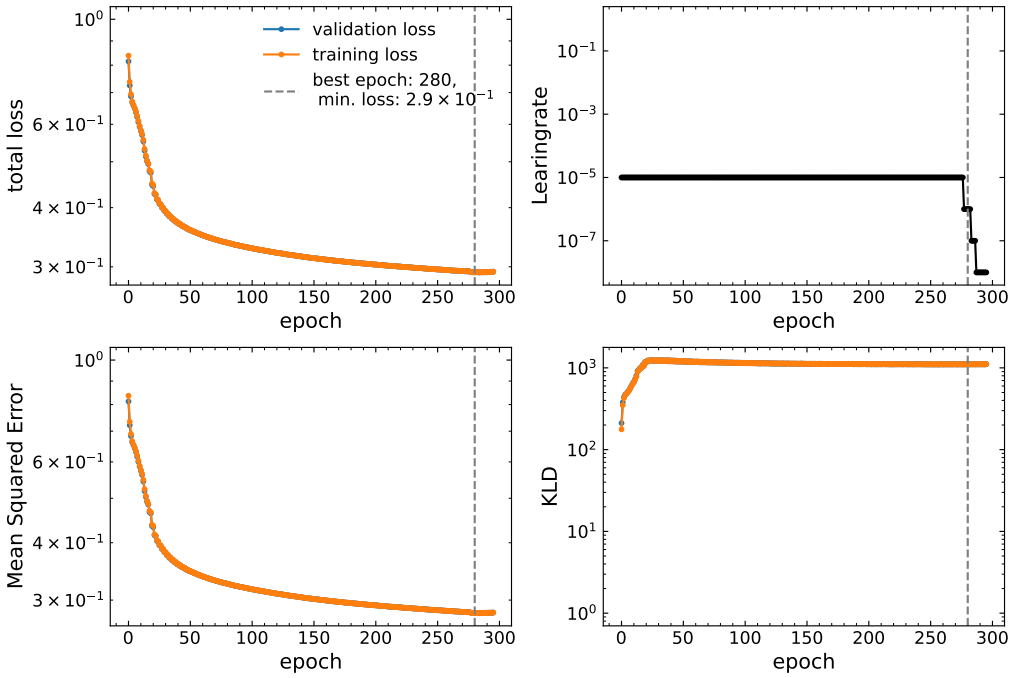

Figure A.73.: Training details for the training of an DVAE with 4-dimensional latent space (top). The total loss is equal to the MSE.

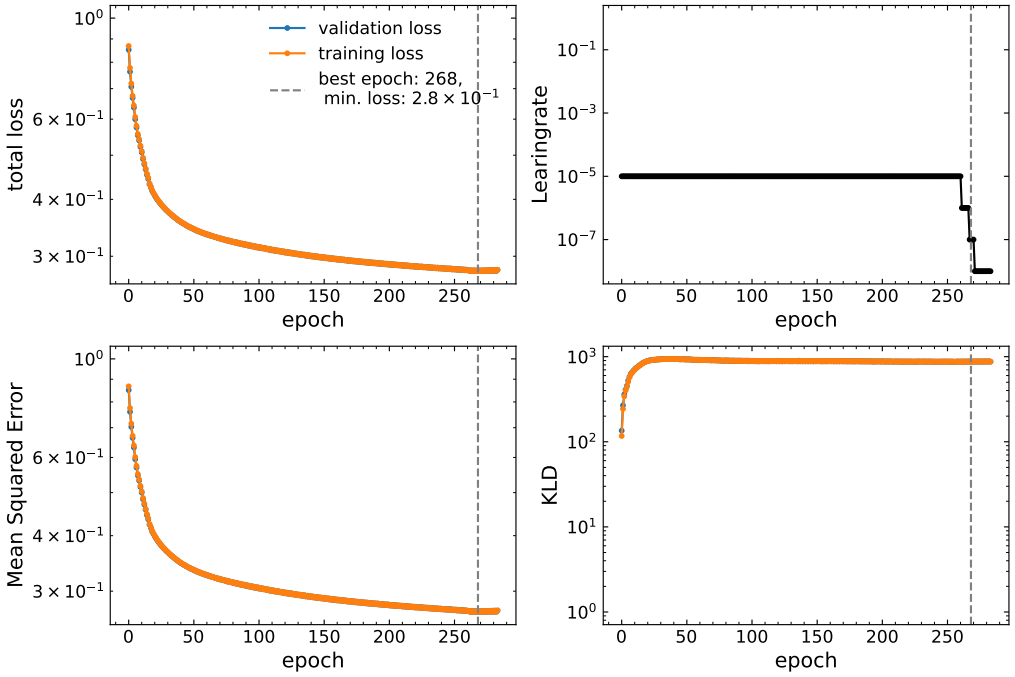

Figure A.74.: Training details for the training of an DVAE with 5-dimensional latent space (top). The total loss is equal to the MSE.

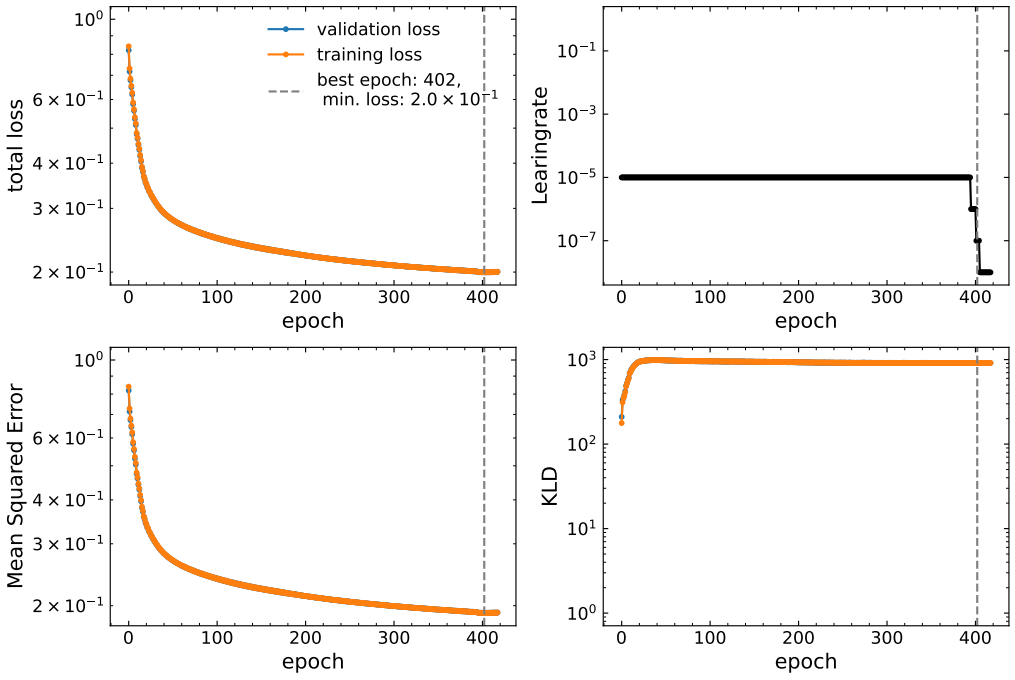

Figure A.75.: Training details for the training of an DVAE with 6-dimensional latent space (top). The total loss is equal to the MSE.

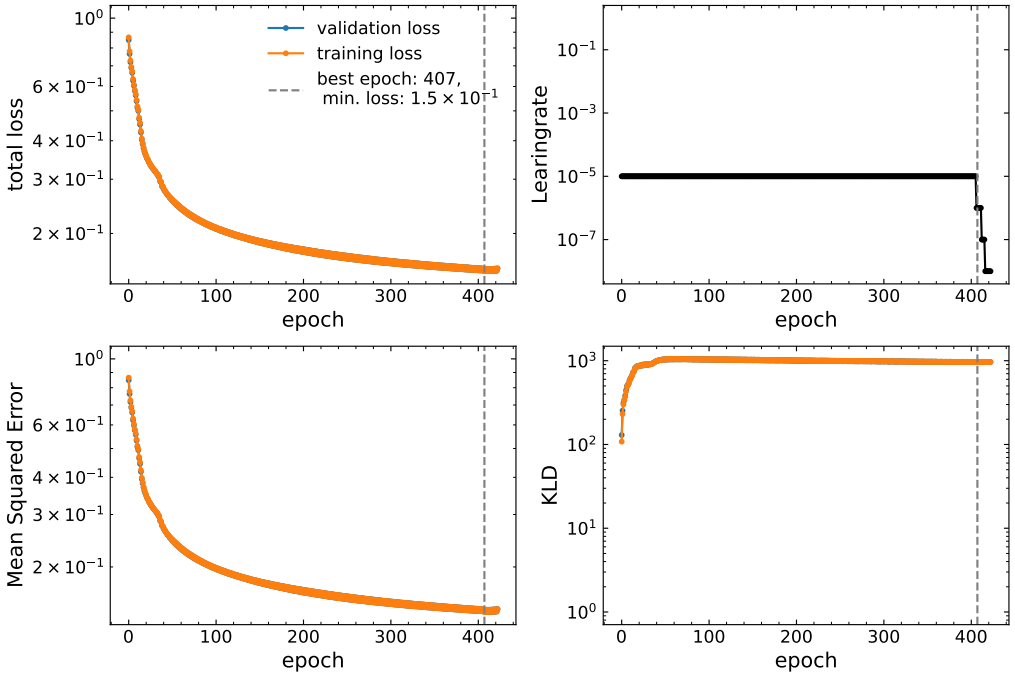

Figure A.76.: Training details for the training of an DVAE with 7-dimensional latent space (top). The total loss is equal to the MSE.

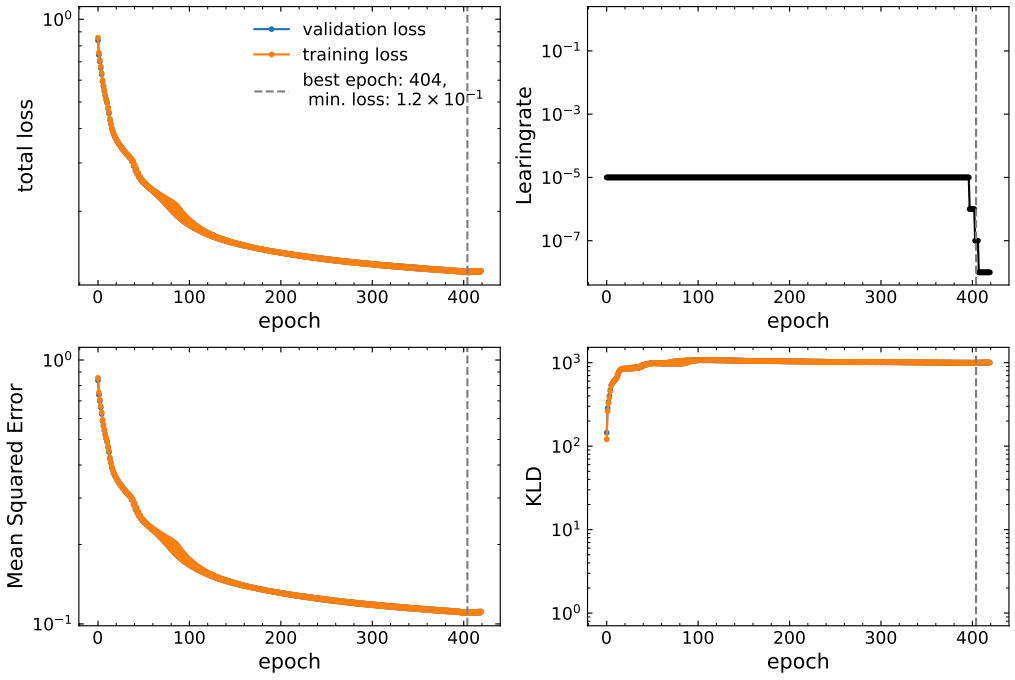

Figure A.77.: Training details for the training of an DVAE with 8-dimensional latent space (top). The total loss is equal to the MSE.

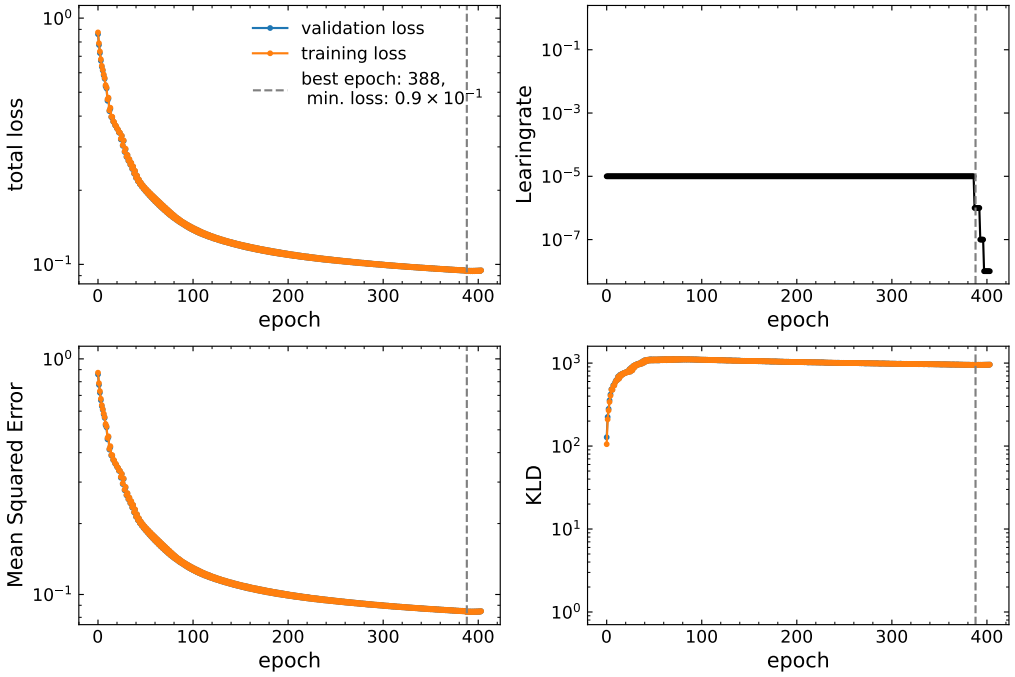

Figure A.78.: Training details for the training of an DVAE with 9-dimensional latent space (top). The total loss is equal to the MSE.

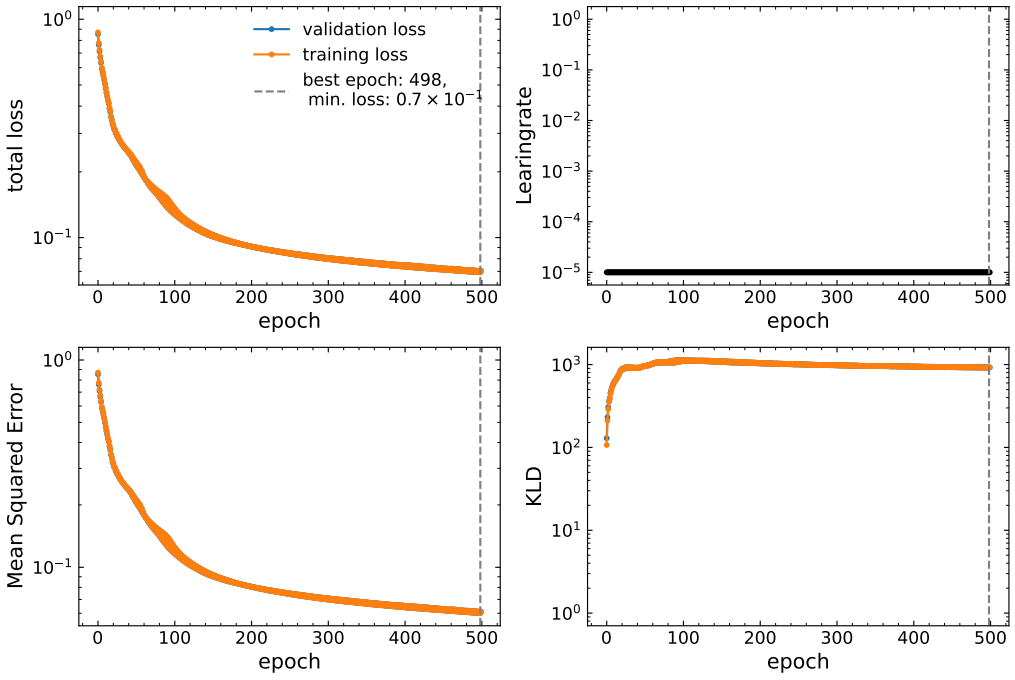

Figure A.79.: Training details for the training of an DVAE with 10-dimensional latent space (top). The total loss is equal to the MSE.

A.9. MSE for DVAEs

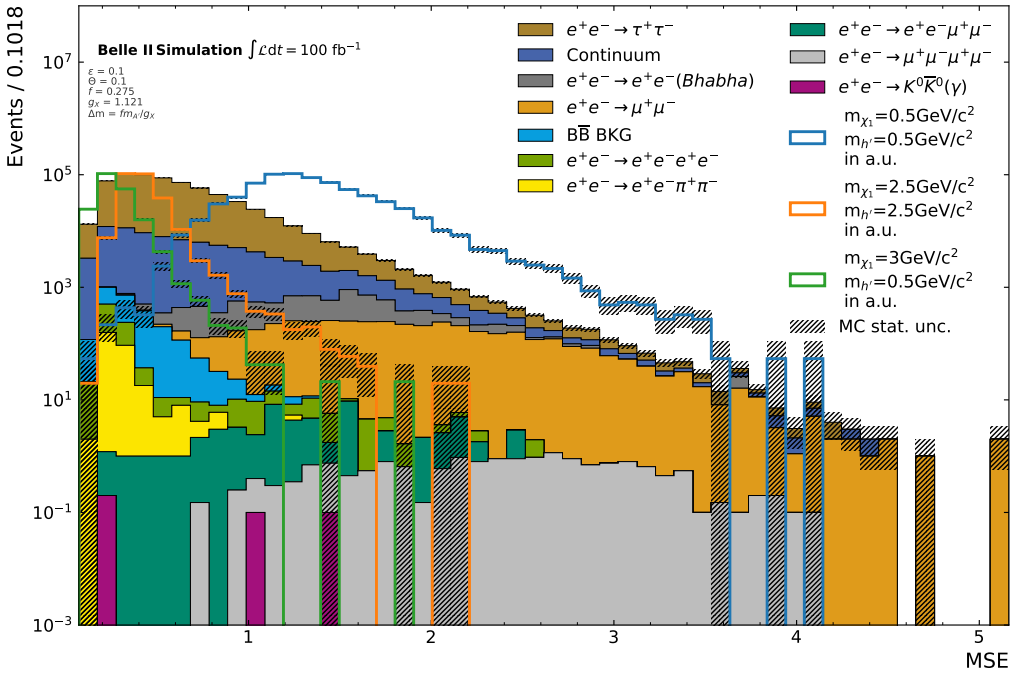

Figure A.80.: Distribution of the MSE for the 2-dimensional DVAE.

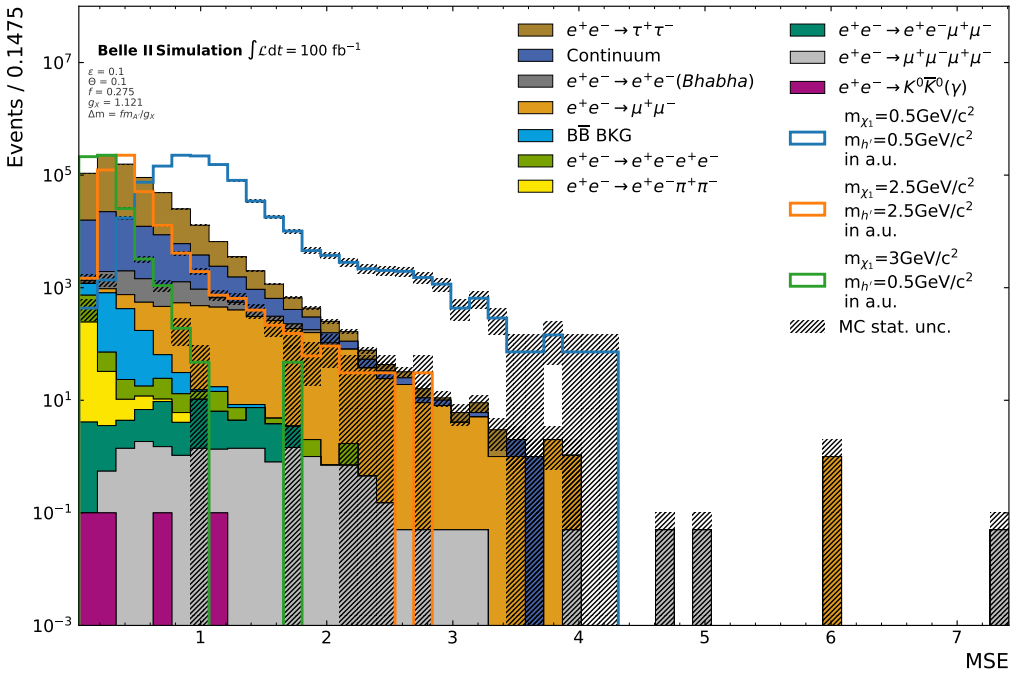

Figure A.81.: Distribution of the MSE for the 3-dimensional DVAE.

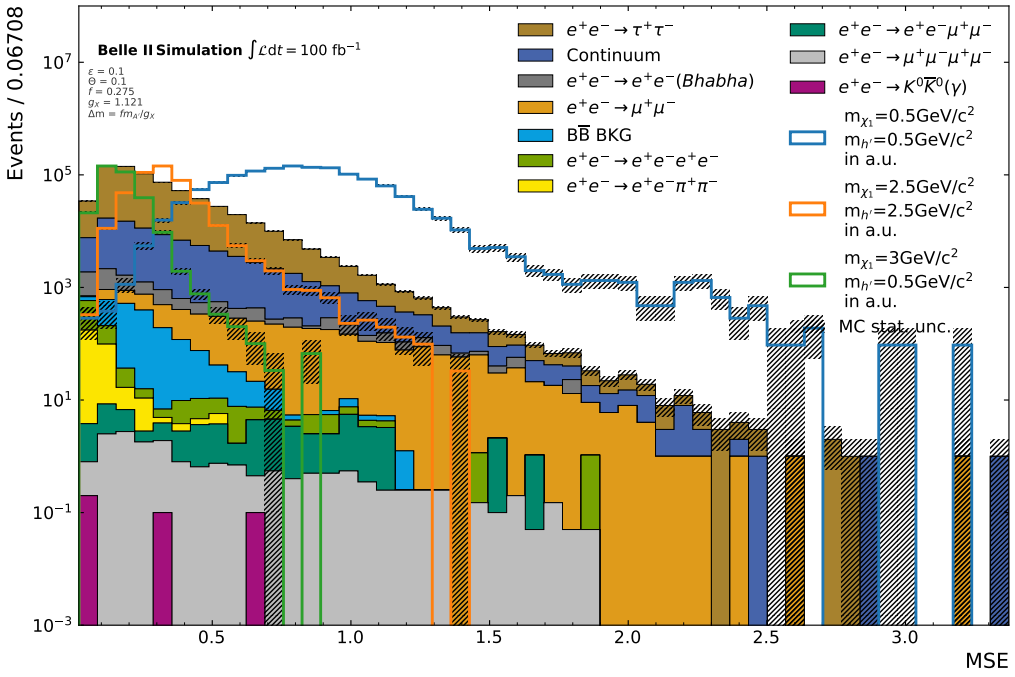

Figure A.82.: Distribution of the MSE for the 4-dimensional DVAE.

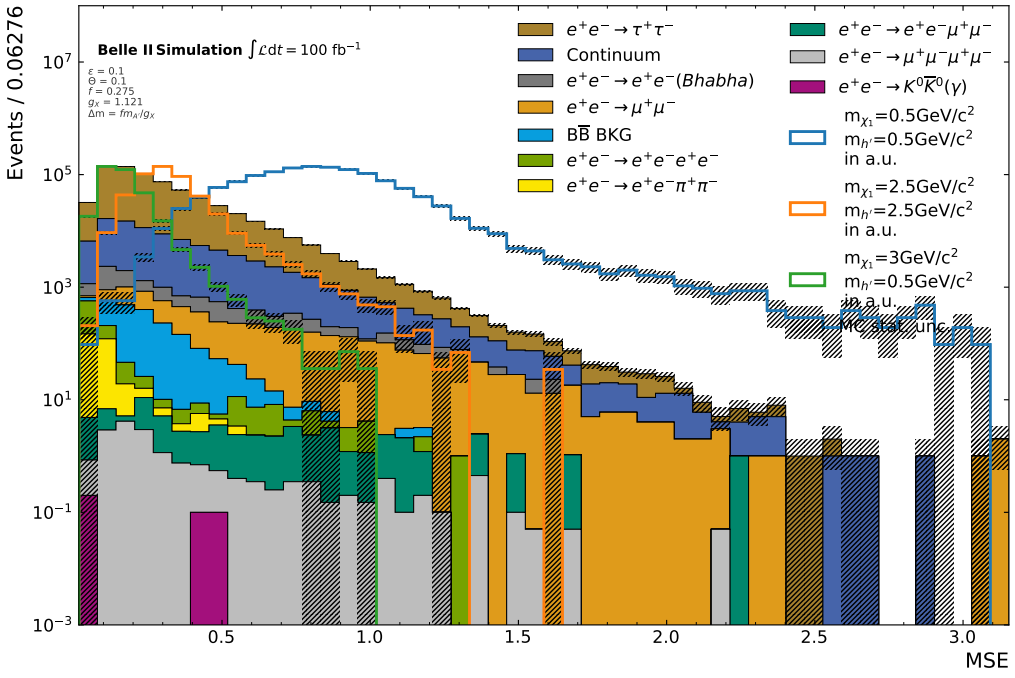

Figure A.83.: Distribution of the MSE for the 5-dimensional DVAE.

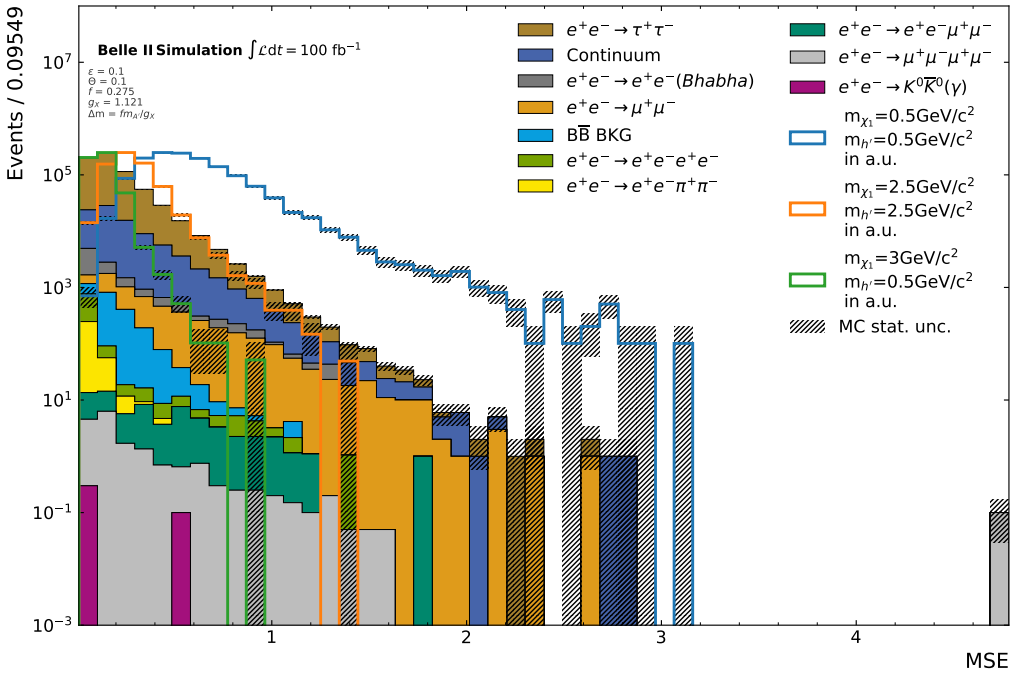

Figure A.84.: Distribution of the MSE for the 6-dimensional DVAE.

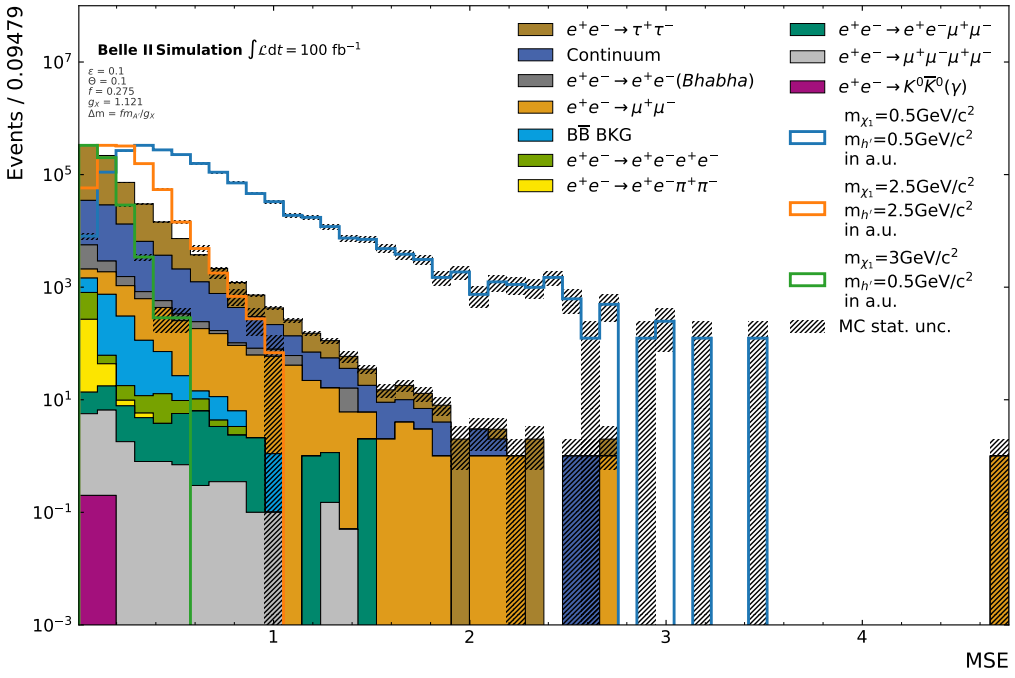

Figure A.85.: Distribution of the MSE for the 7-dimensional DVAE.

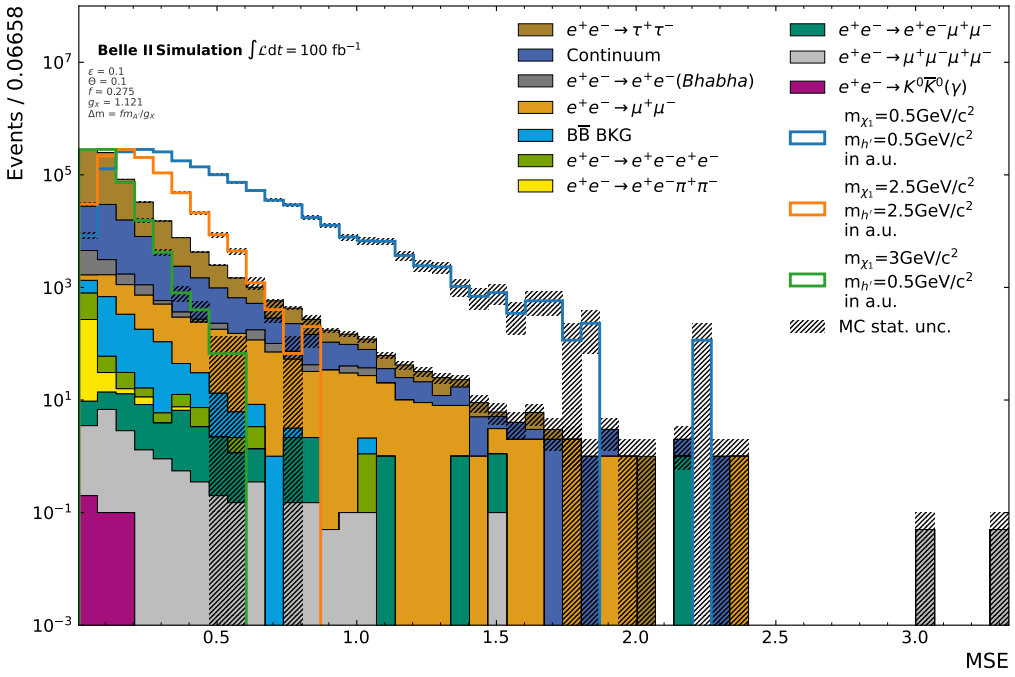

Figure A.86.: Distribution of the MSE for the 8-dimensional DVAE.

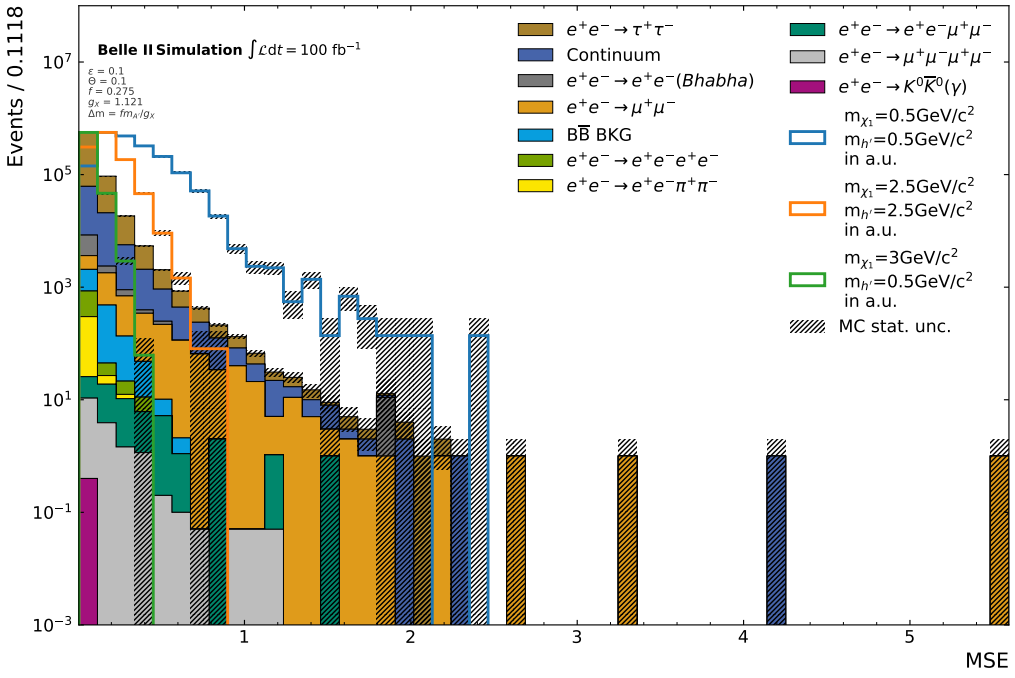

Figure A.87.: Distribution of the MSE for the 9-dimensional DVAE.

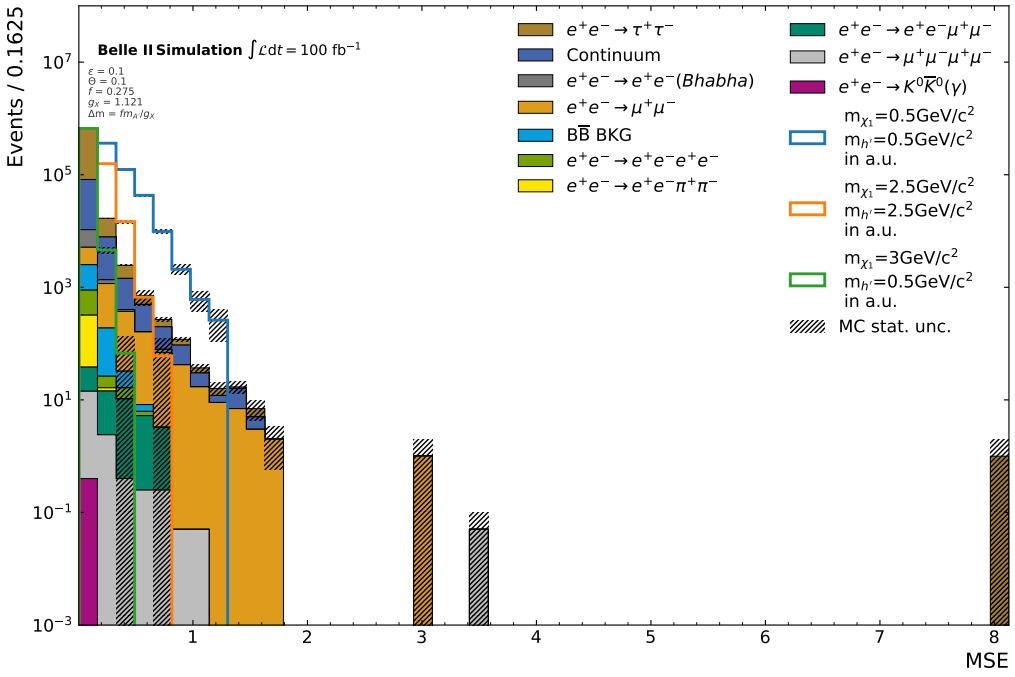

Figure A.88.: Distribution of the MSE for the 10-dimensional DVAE.

A.10. Latentspace of DVAEs

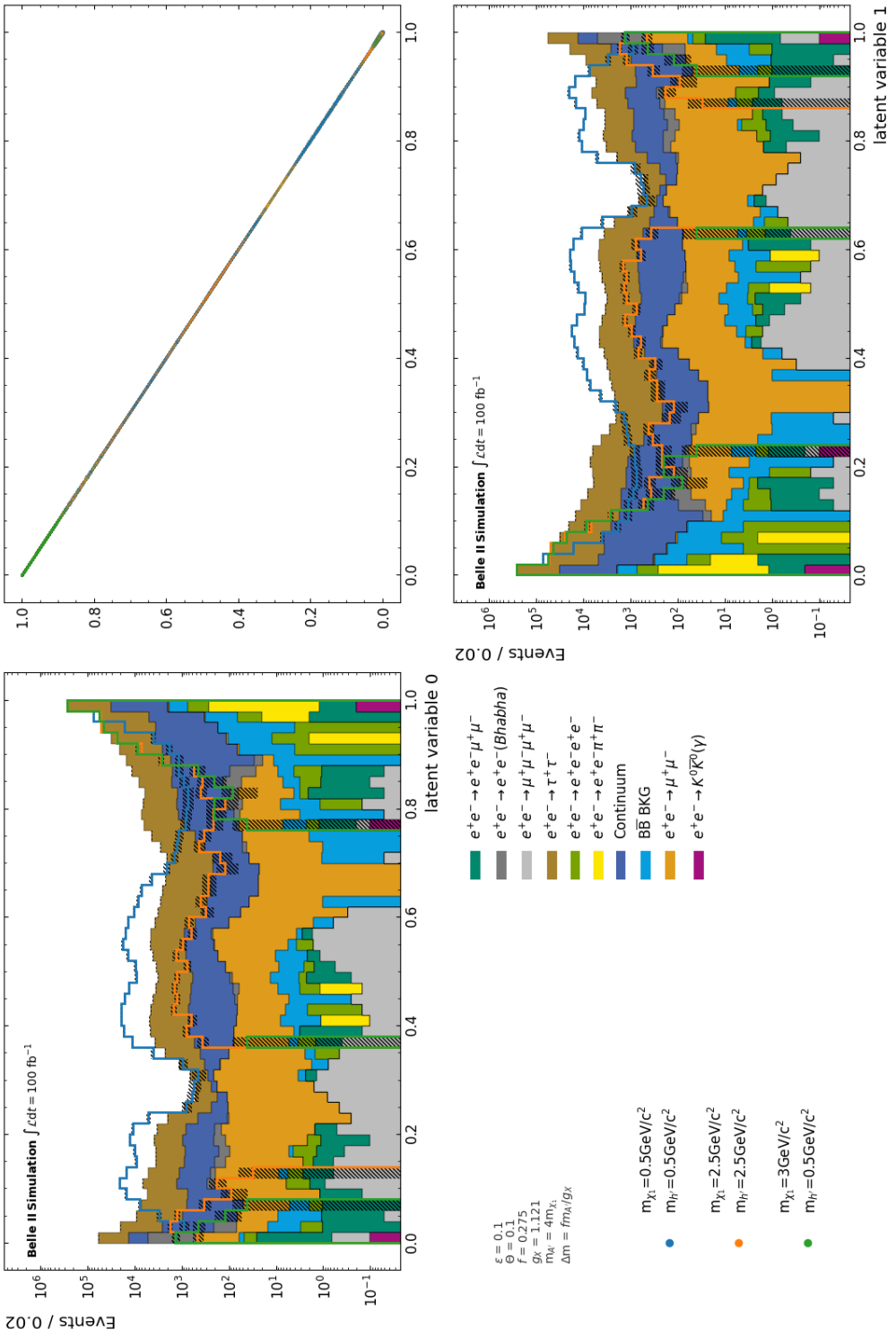

Figure A.89.: Latent variables and their correlations for the 2-dimensional DVAE for the background samples and the three example signals.

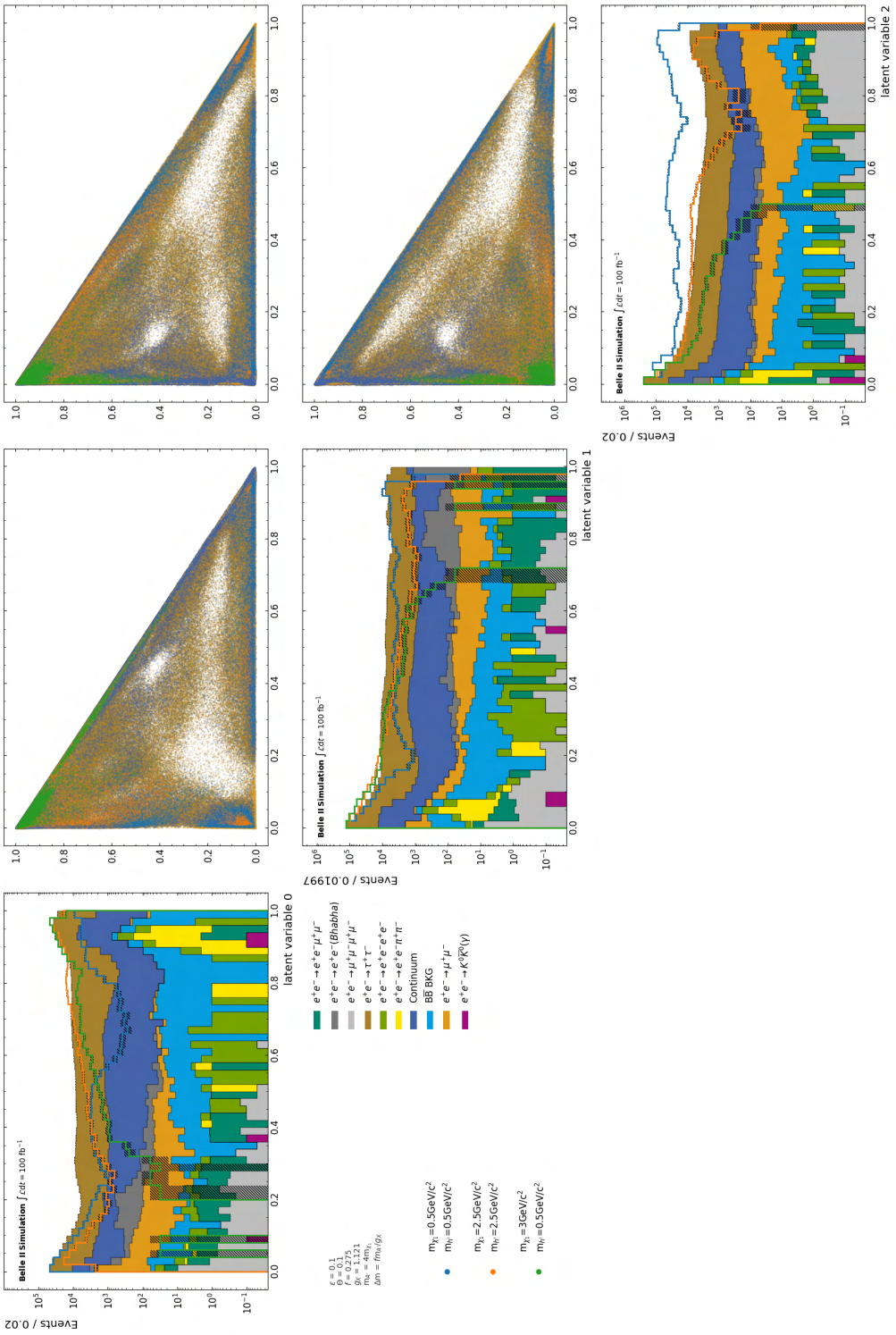

Figure A.90.: Latent variables and their correlations for the 3-dimensional DVAE for the background samples and the three example signals.

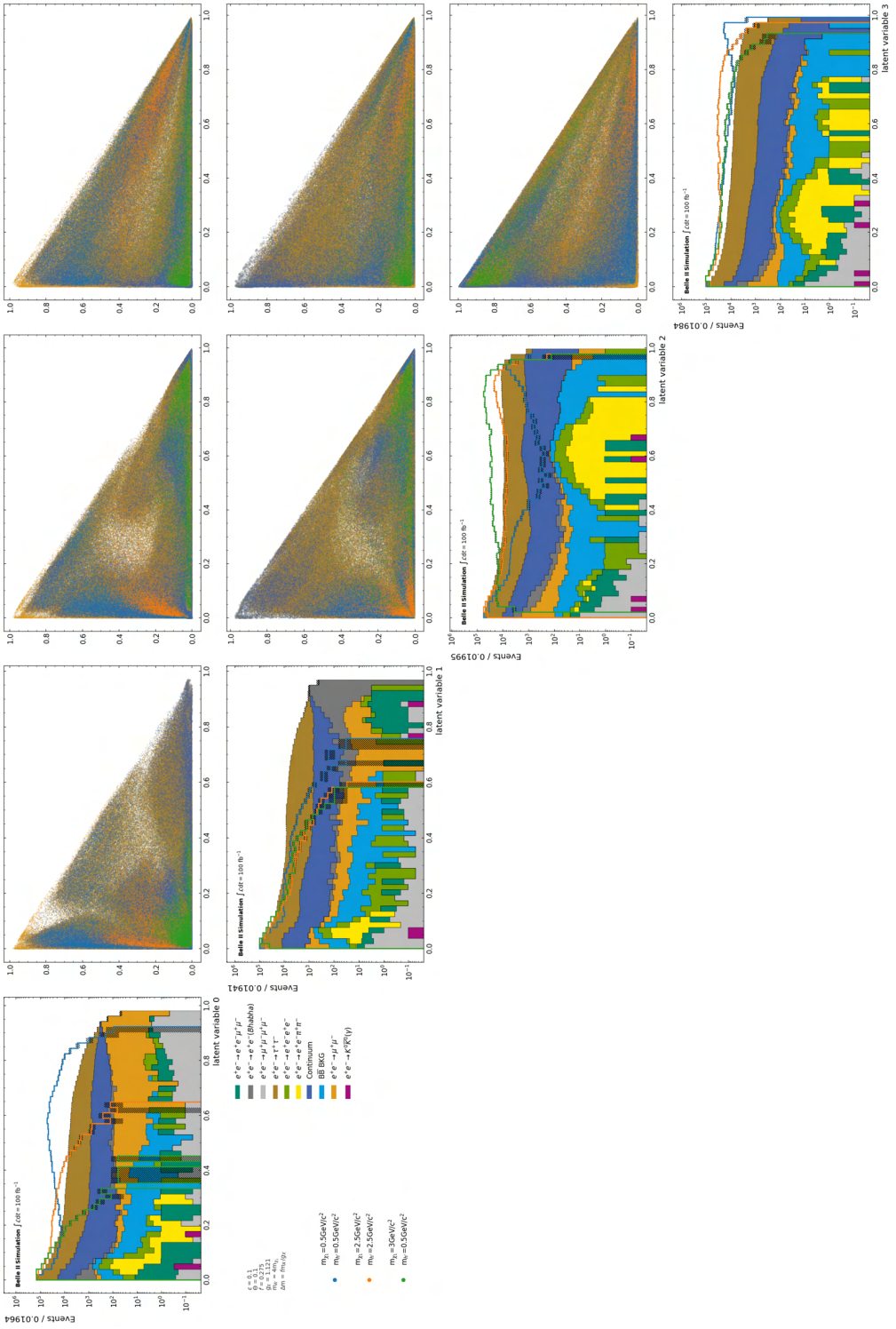

Figure A.91.: Latent variables and their correlations for the 4-dimensional DVAE for the background samples and the three example signals.

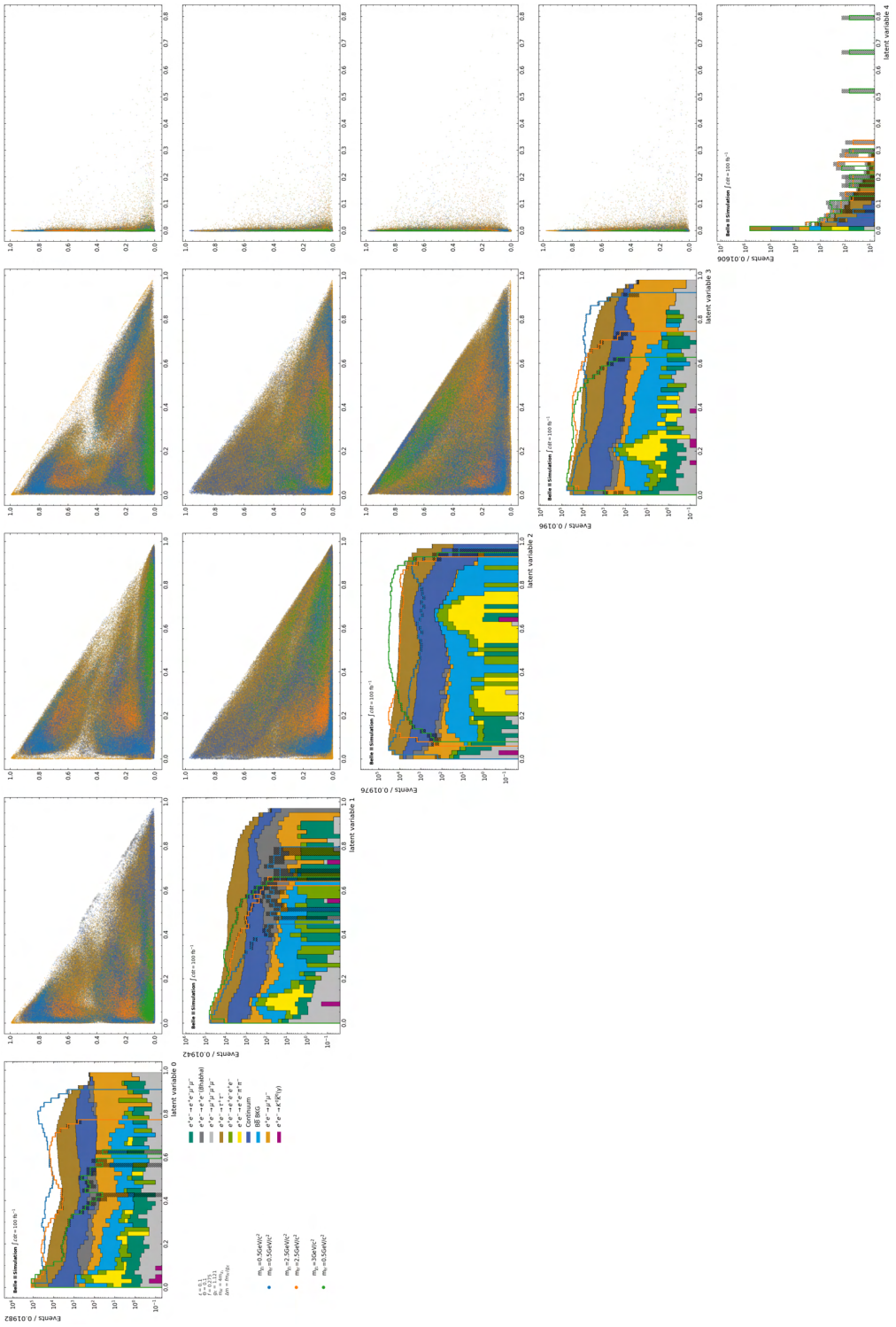

Figure A.92.: Latent variables and their correlations for the 5-dimensional DVAE for the background samples and the three example signals.

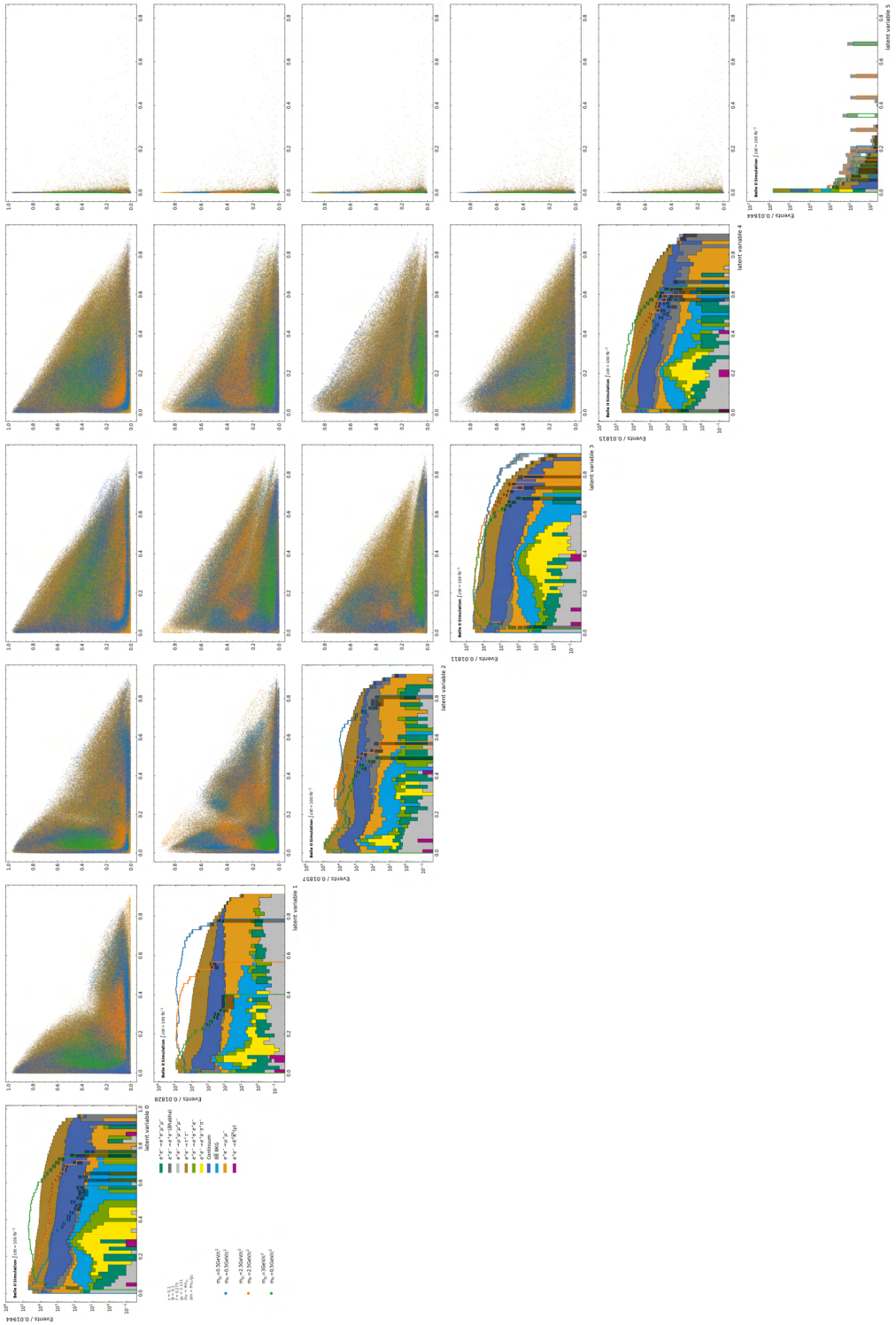

Figure A.93.: Latent variables and their correlations for the 6-dimensional DVAE for the background samples and the three example signals.

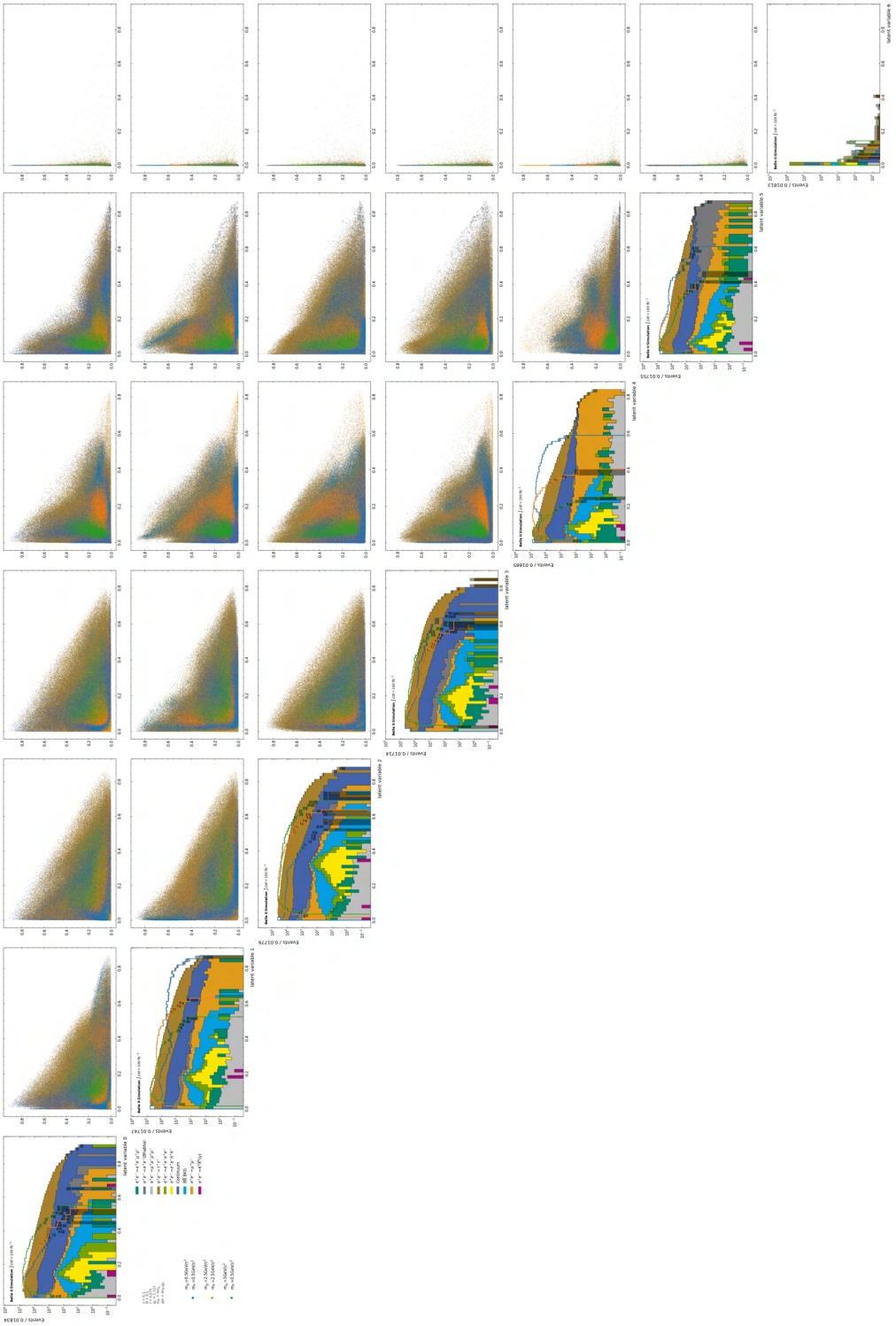

Figure A.94.: Latent variables and their correlations for the 7-dimensional DVAE for the background samples and the three example signals.

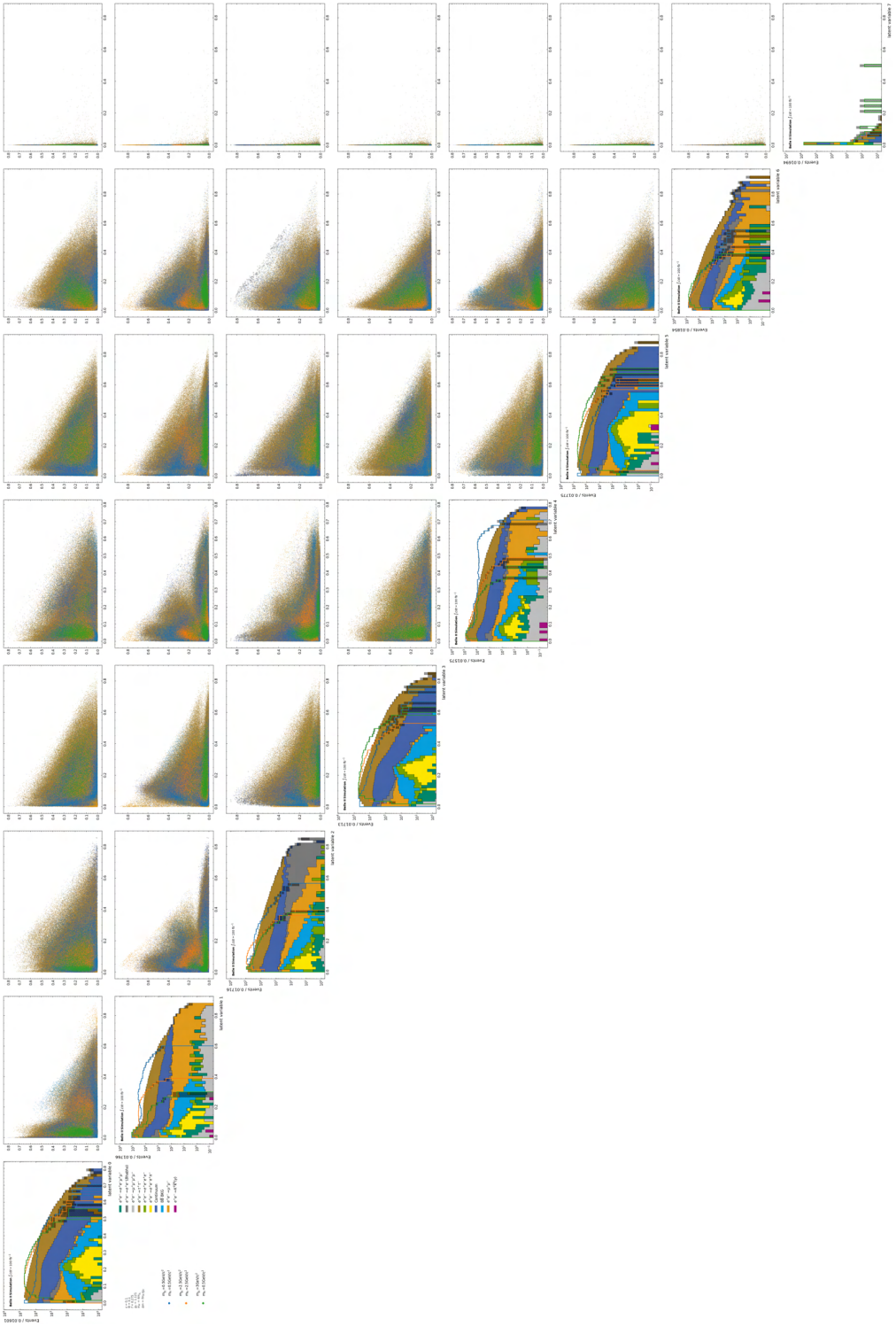

Figure A.95.: Latent variables and their correlations for the 8-dimensional DVAE for the background samples and the three example signals.

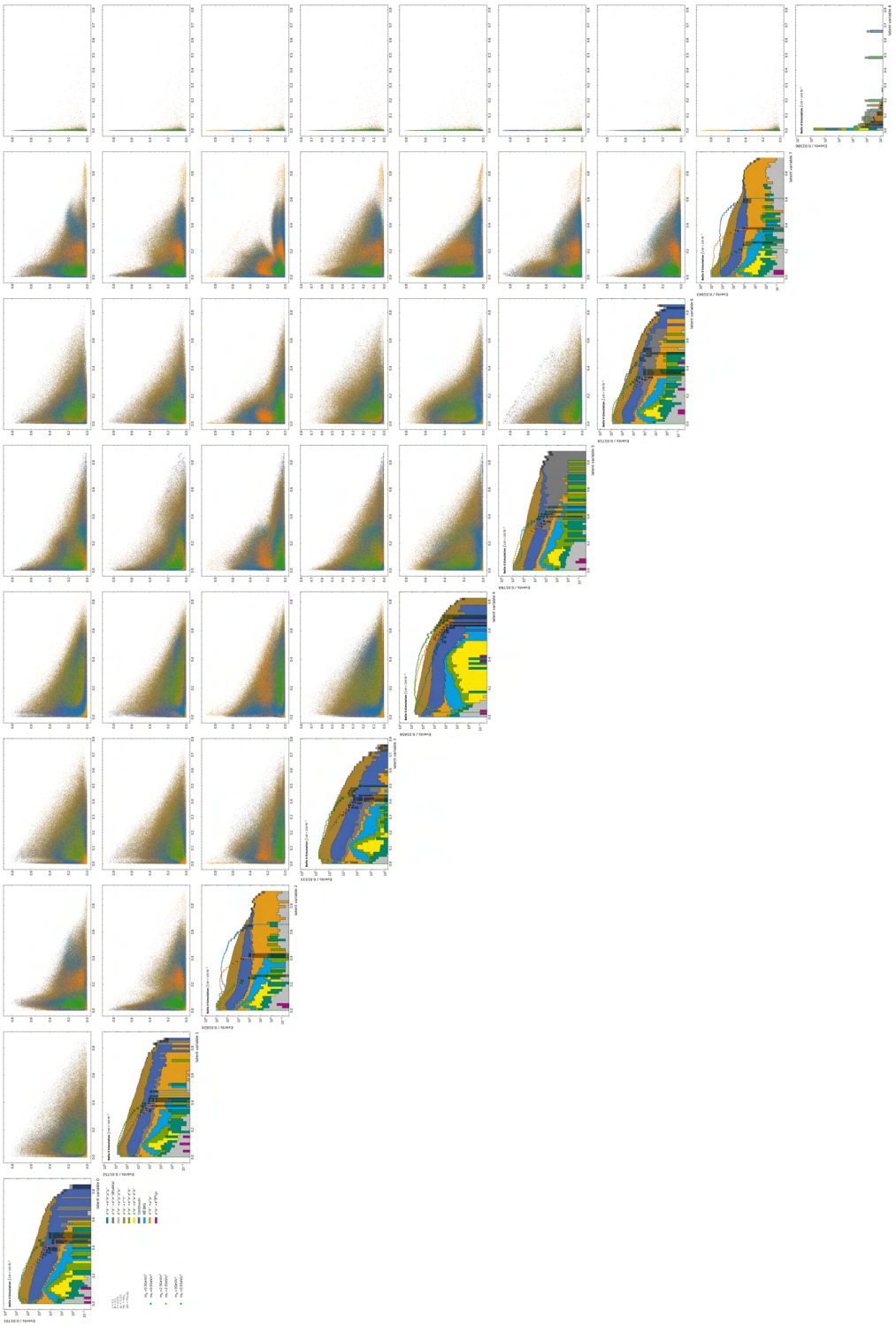

Figure A.96.: Latent variables and their correlations for the 9-dimensional DVAE for the background samples and the three example signals.

Figure A.97.: Latent variables and their correlations for the 10-dimensional DVAE for the background samples and the three example signals.

A.11. PFOM Optimization for AEs

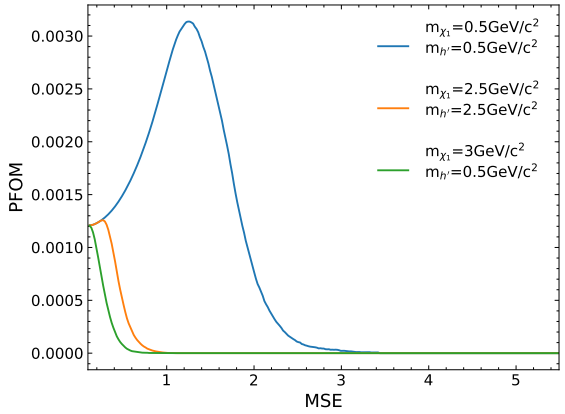

Figure A.98.: Punzi Figure of Merit (PFOM) over MSE for the three example signals for the 1-dimensional AE.

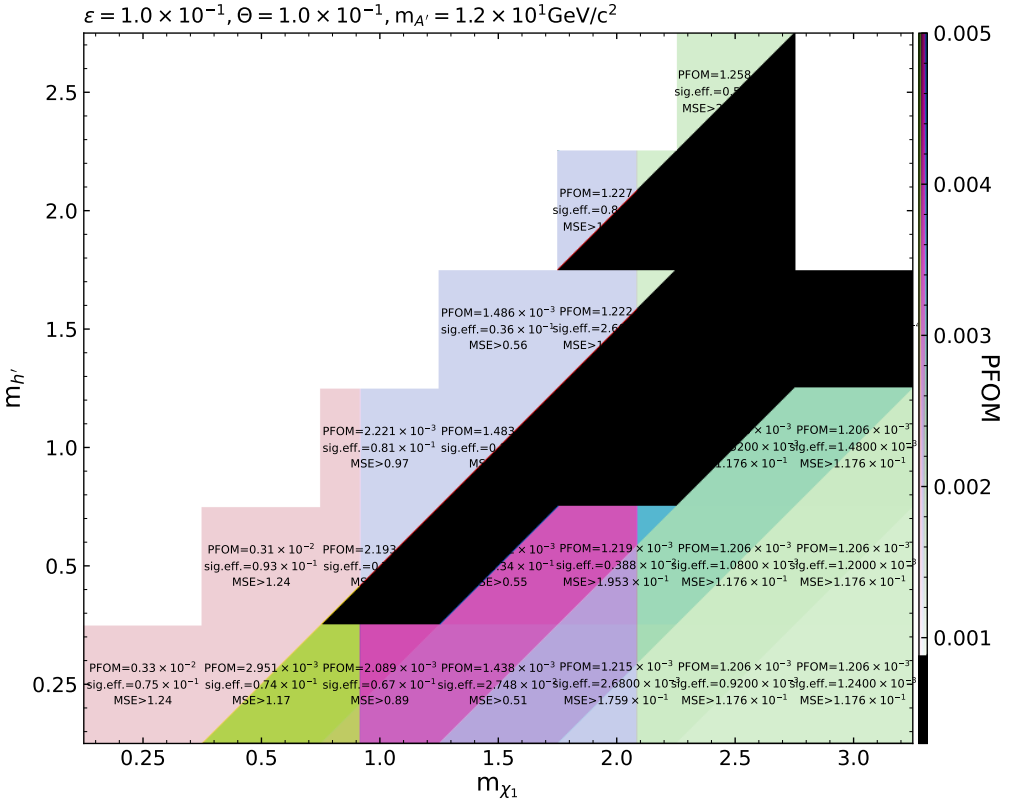

Figure A.99.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 1-dimensional AE for each mass configuration of the signal.

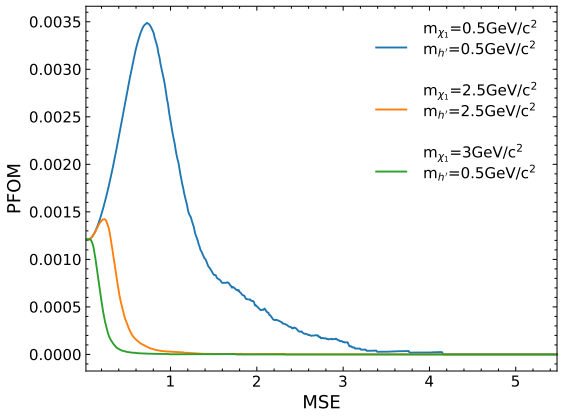

Figure A.100.: PFOM over MSE for the three example signals for the 2-dimensional AE.

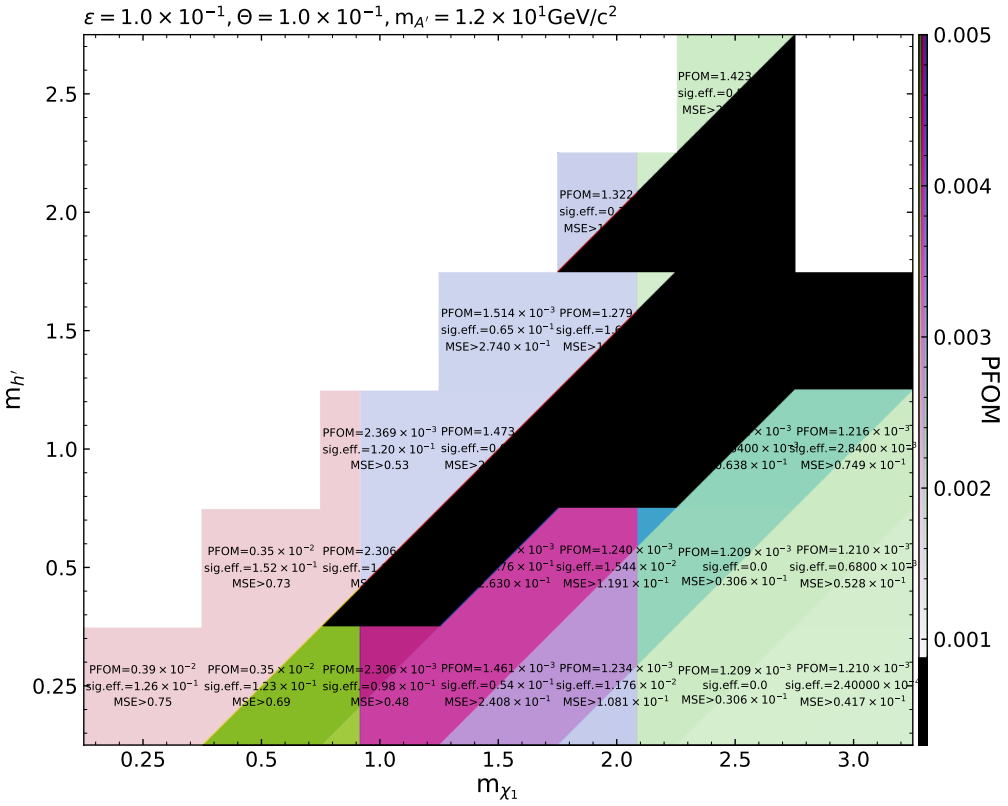

Figure A.101.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 2-dimensional AE for each mass configuration of the signal.

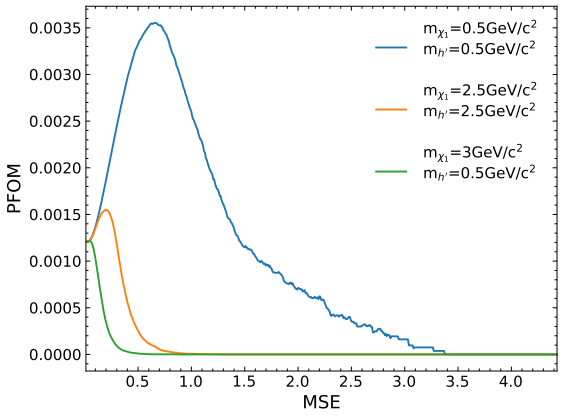

Figure A.102.: PFOM over MSE for the three example signals for the 3-dimensional AE.

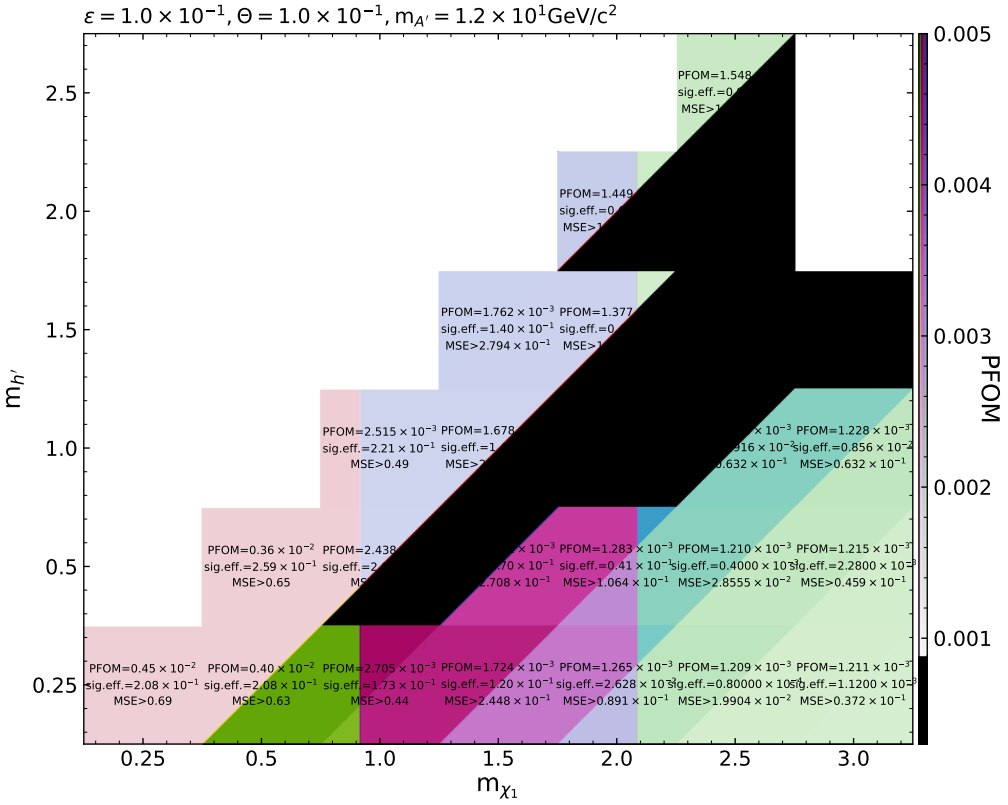

Figure A.103.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 3-dimensional AE for each mass configuration of the signal.

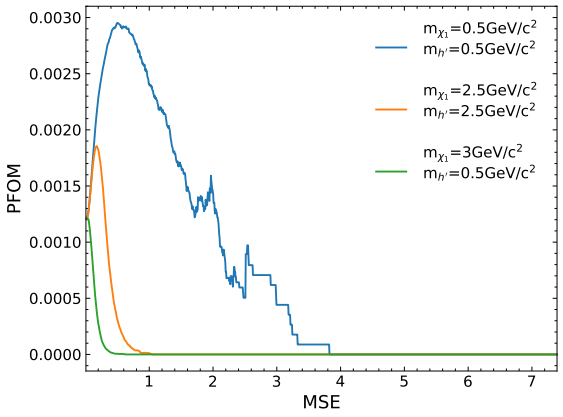

Figure A.104.: PFOM over MSE for the three example signals for the 4-dimensional AE.

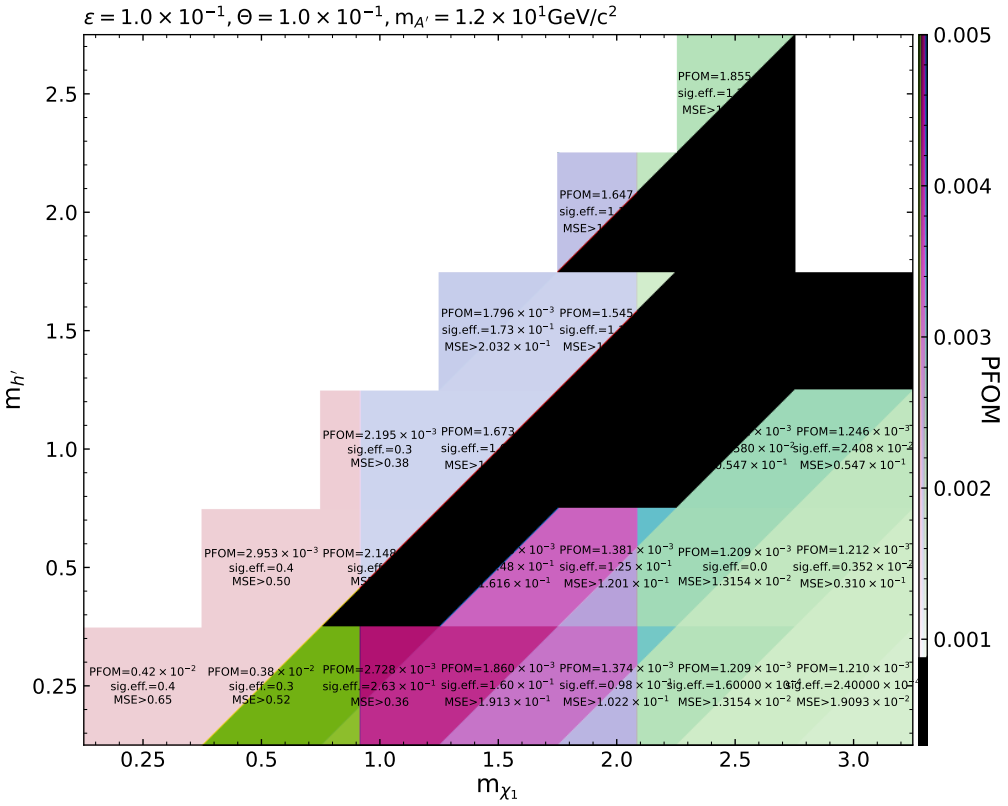

Figure A.105.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 4-dimensional AE for each mass configuration of the signal.

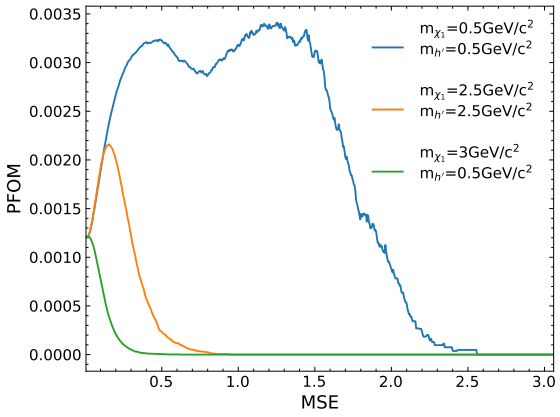

Figure A.106.: PFOM over MSE for the three example signals for the 5-dimensional AE.

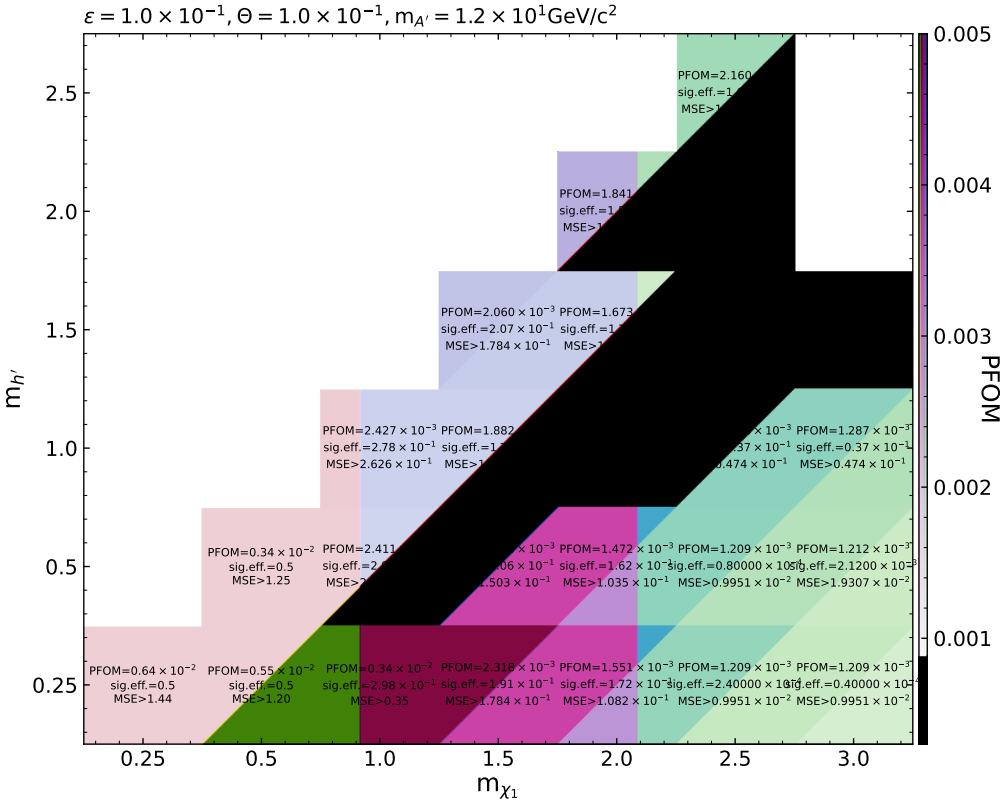

Figure A.107.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 5-dimensional AE for each mass configuration of the signal.

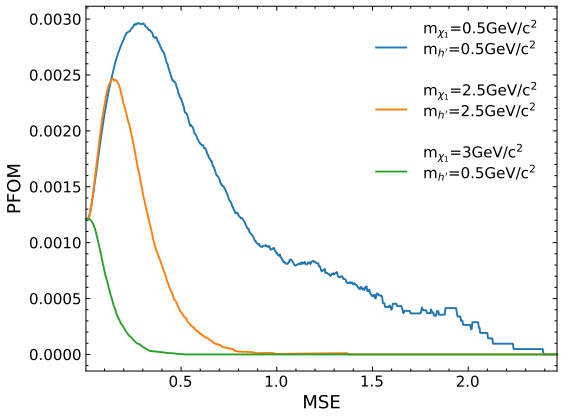

Figure A.108.: PFOM over MSE for the three example signals for the 6-dimensional AE.

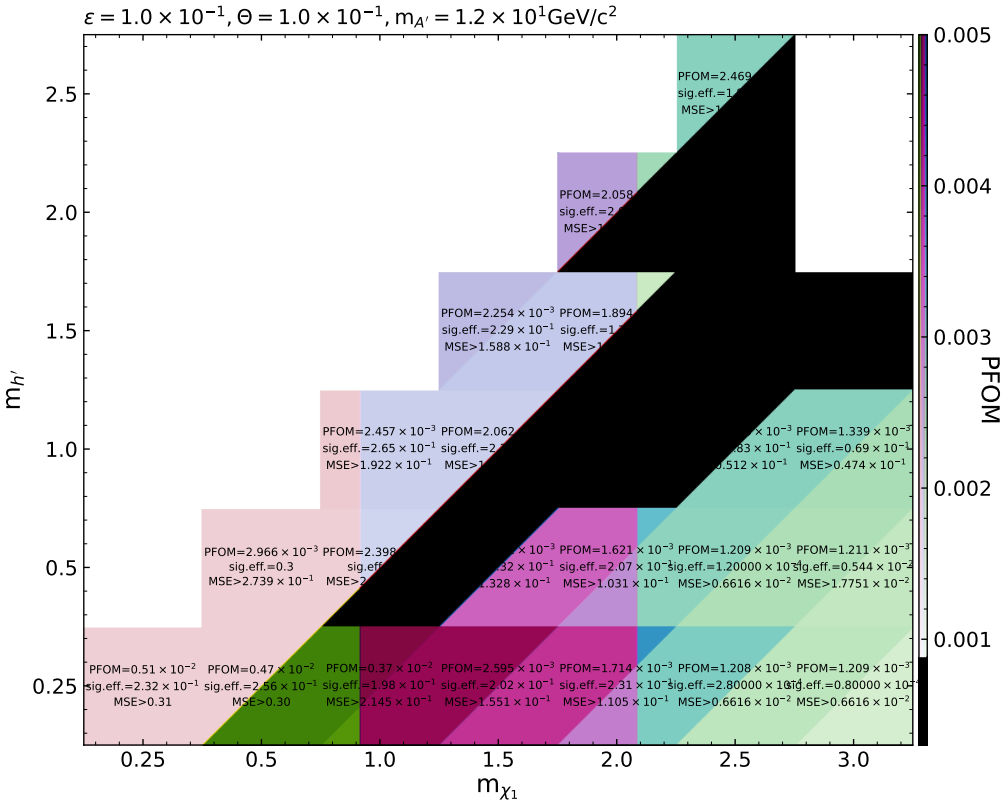

Figure A.109.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 6-dimensional AE for each mass configuration of the signal.

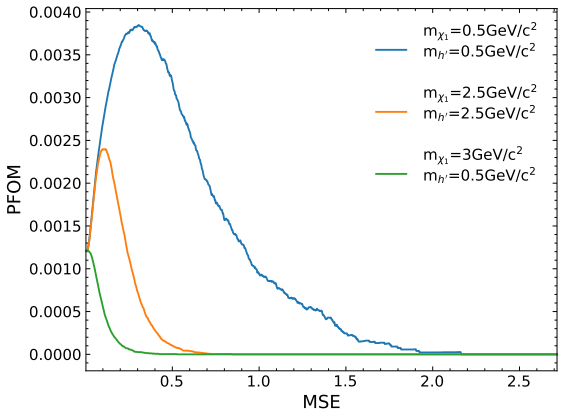

Figure A.110.: PFOM over MSE for the three example signals for the 7-dimensional AE.

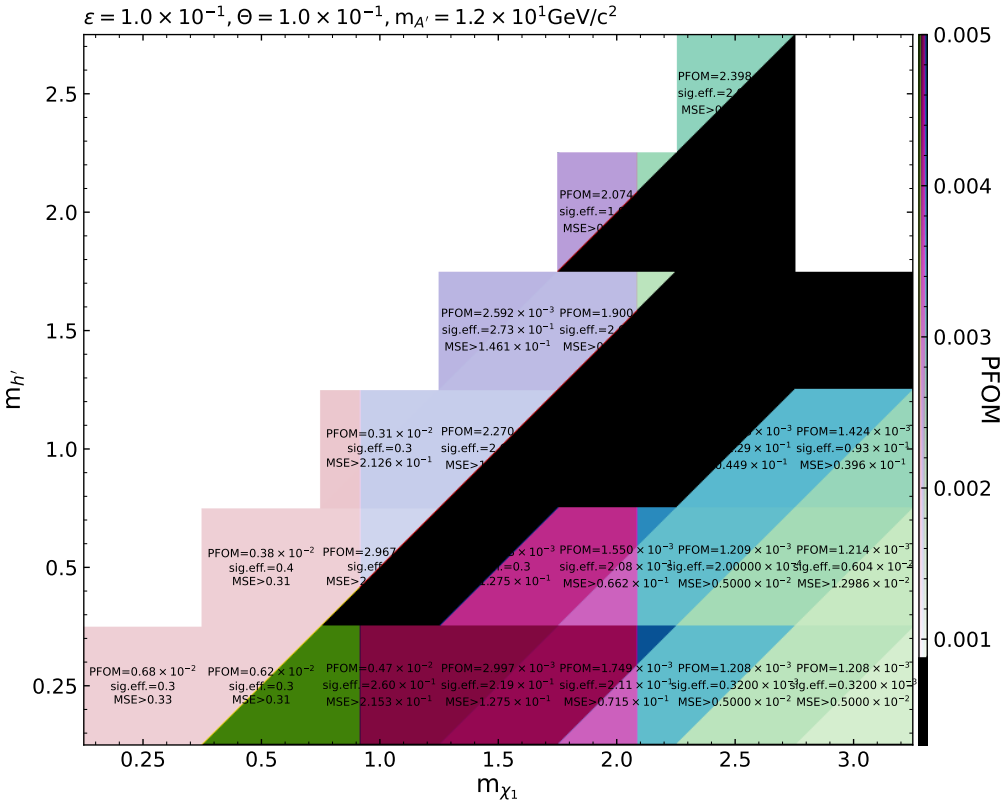

Figure A.111.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 7-dimensional AE for each mass configuration of the signal.

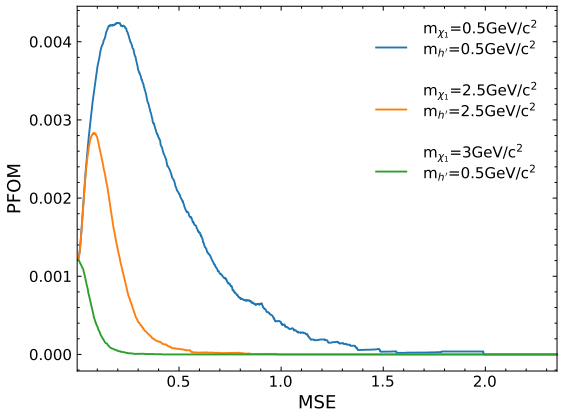

Figure A.112.: PFOM over MSE for the three example signals for the 8-dimensional AE.

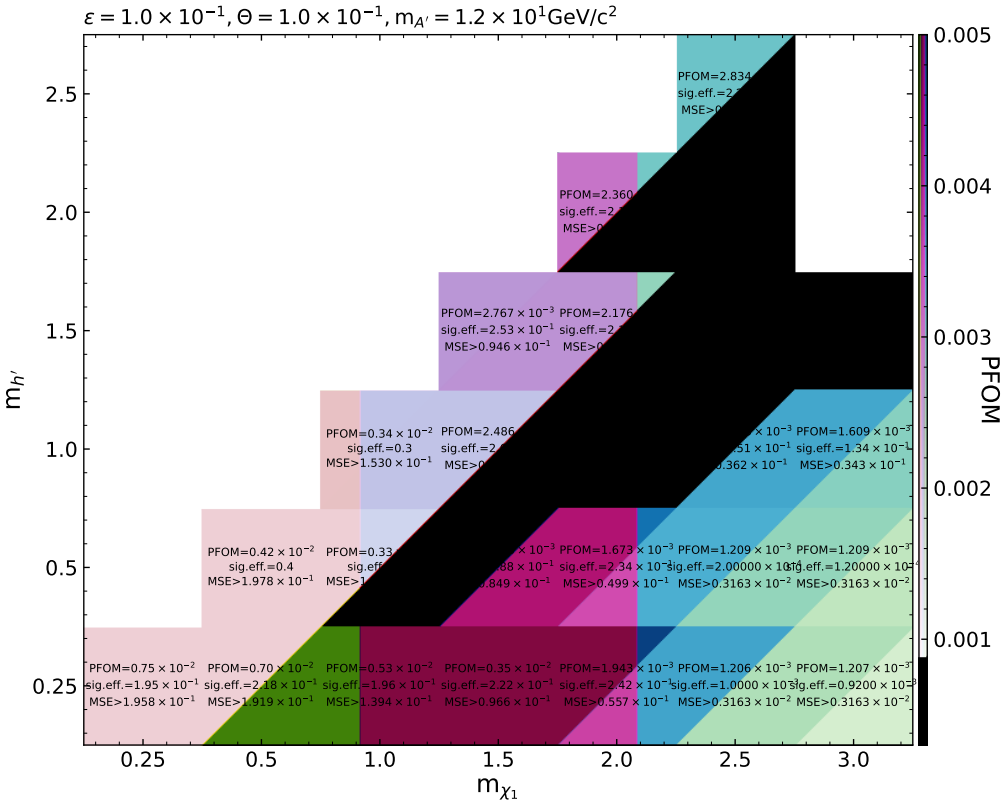

Figure A.113.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 8-dimensional AE for each mass configuration of the signal.

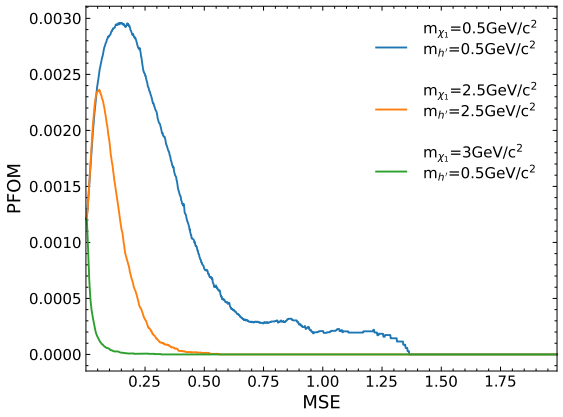

Figure A.114.: PFOM over MSE for the three example signals for the 9-dimensional AE.

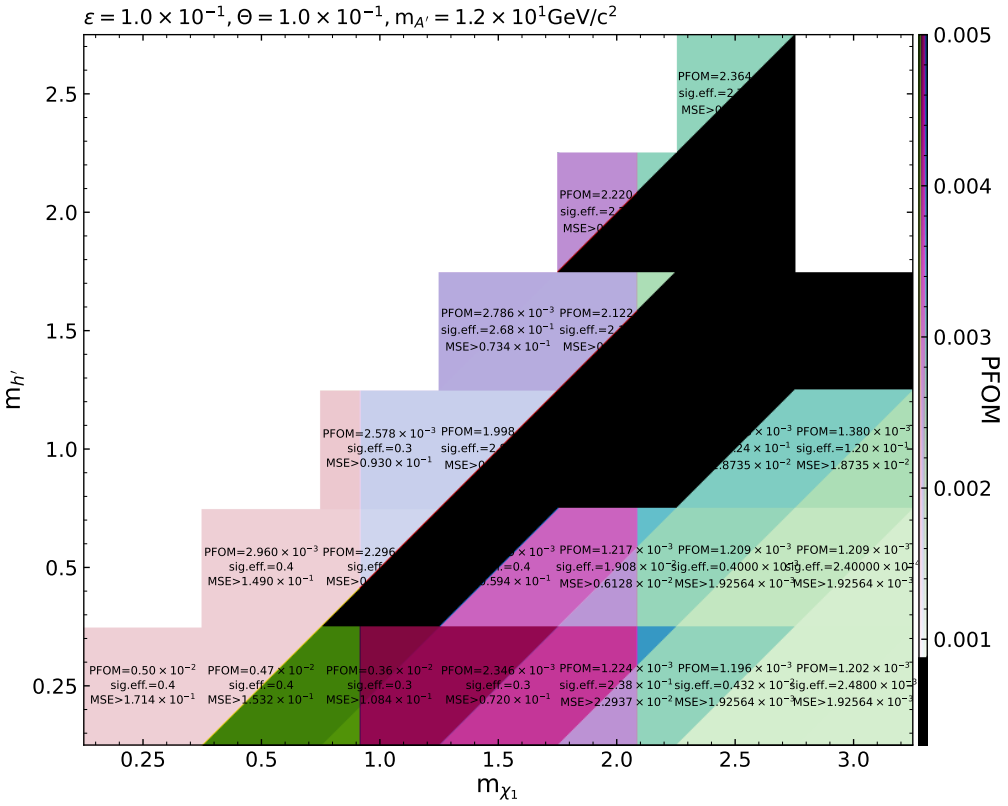

Figure A.115.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 9-dimensional AE for each mass configuration of the signal.

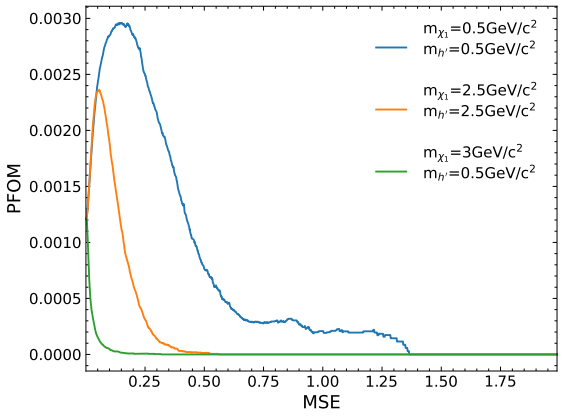

Figure A.116.: PFOM over MSE for the three example signals for the 10-dimensional AE.

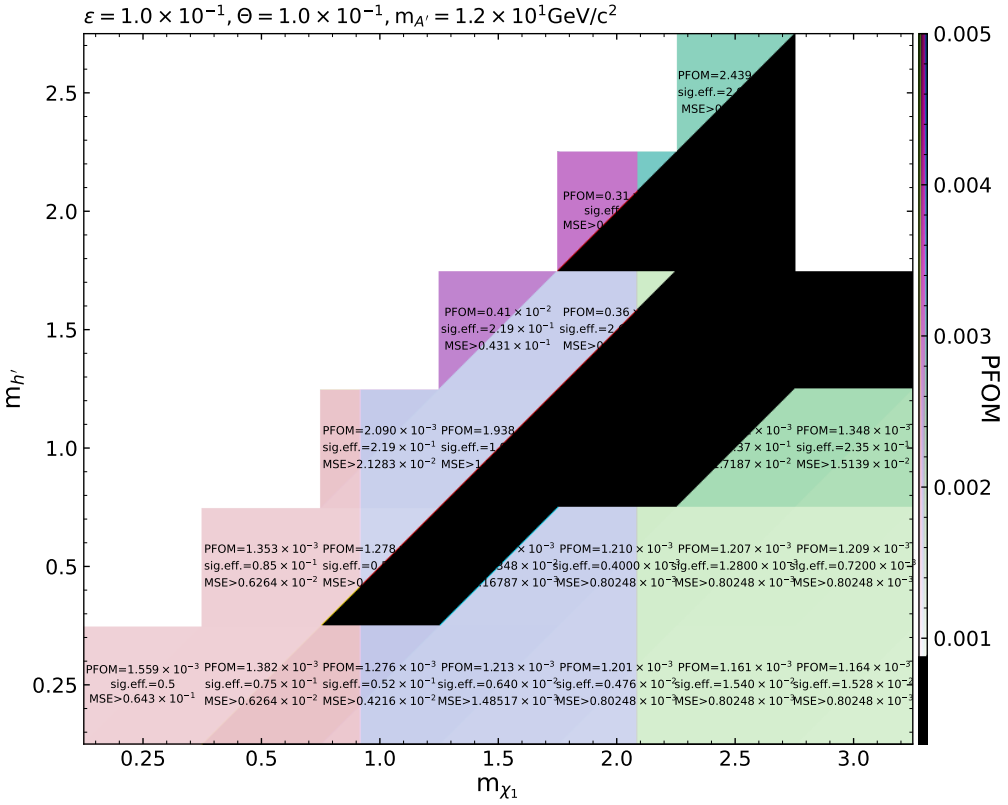

Figure A.117.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 10-dimensional AE for each mass configuration of the signal.

A.12. PFOM Optimization for VAEs

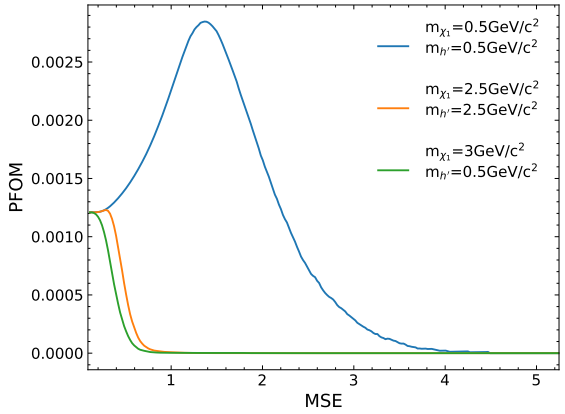

Figure A.118.: PFOM over MSE for the three example signals for the 1-dimensional VAE.

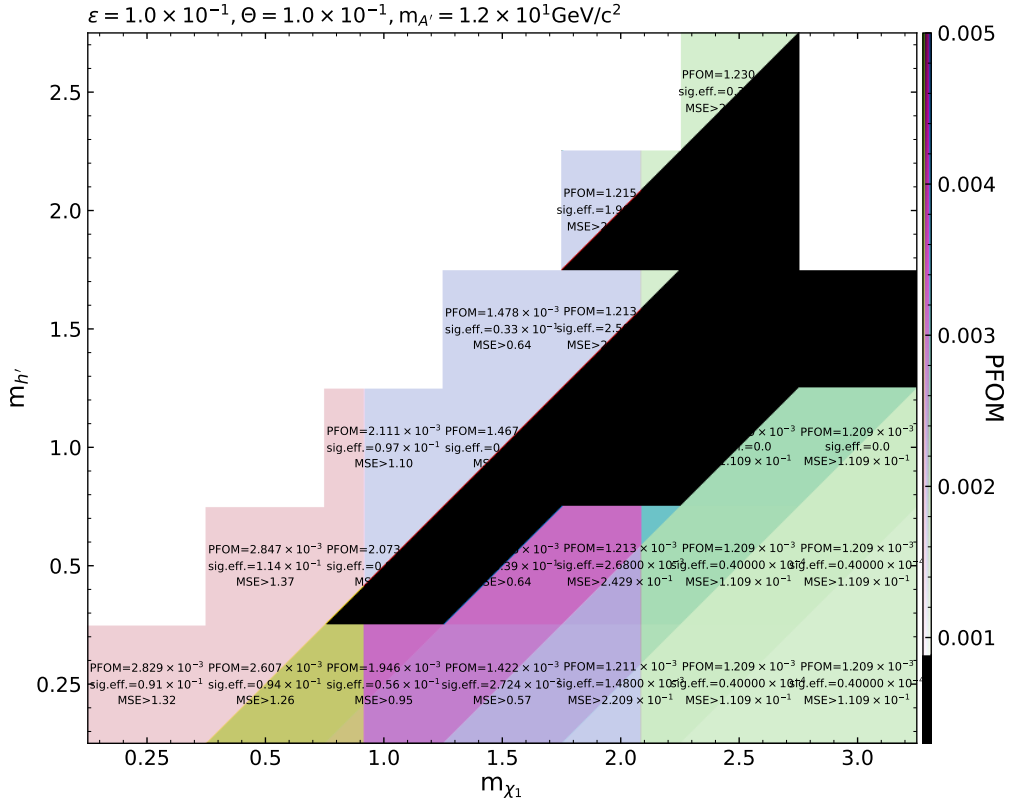

Figure A.119.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 1-dimensional VAE for each mass configuration of the signal.

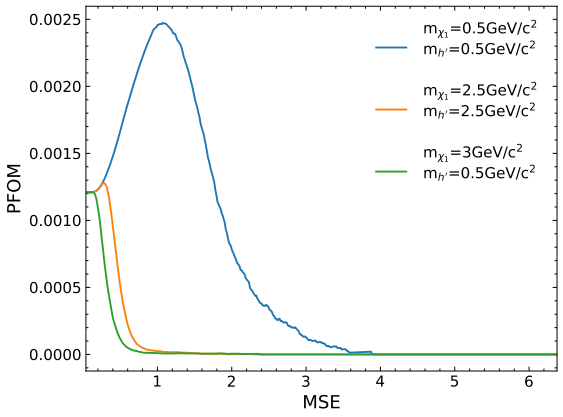

Figure A.120.: PFOM over MSE for the three example signals for the 2-dimensional VAE.

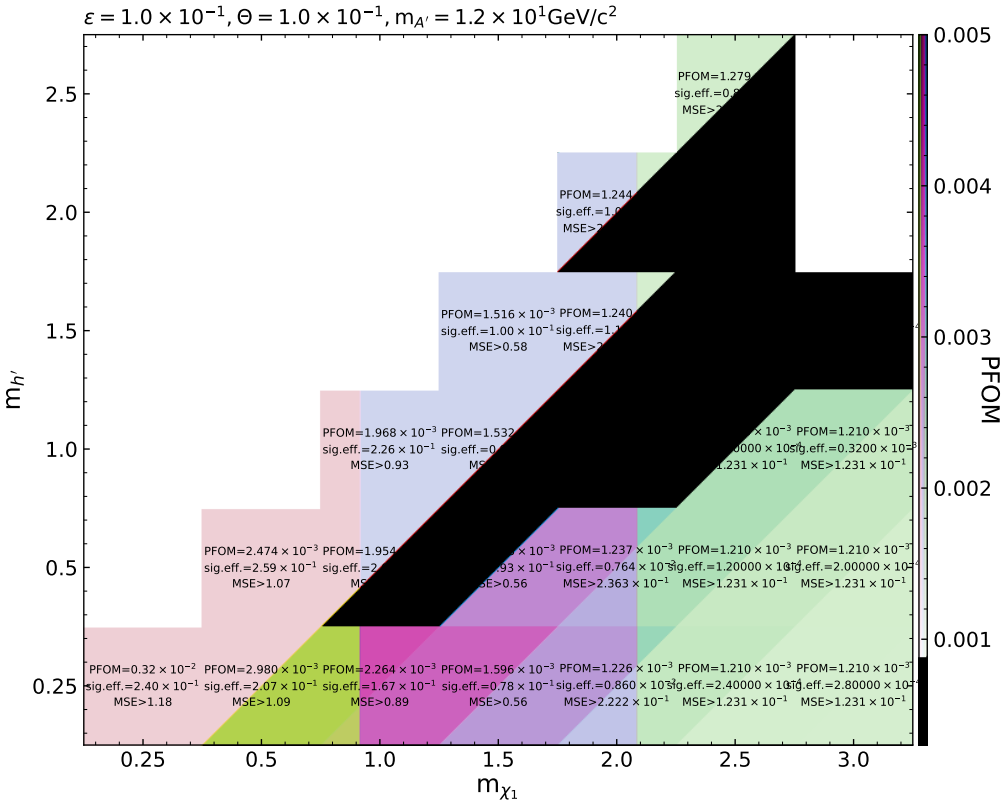

Figure A.121.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 2-dimensional VAE for each mass configuration of the signal.

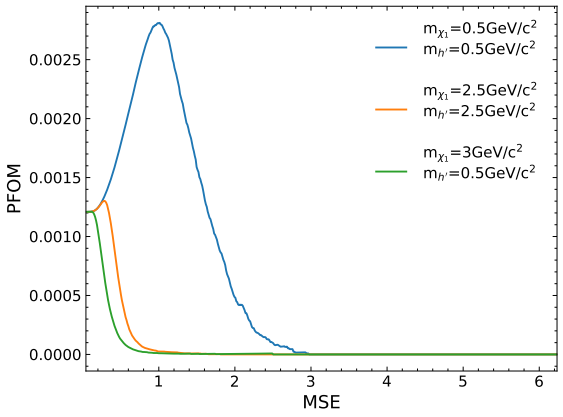

Figure A.122.: PFOM over MSE for the three example signals for the 3-dimensional VAE.

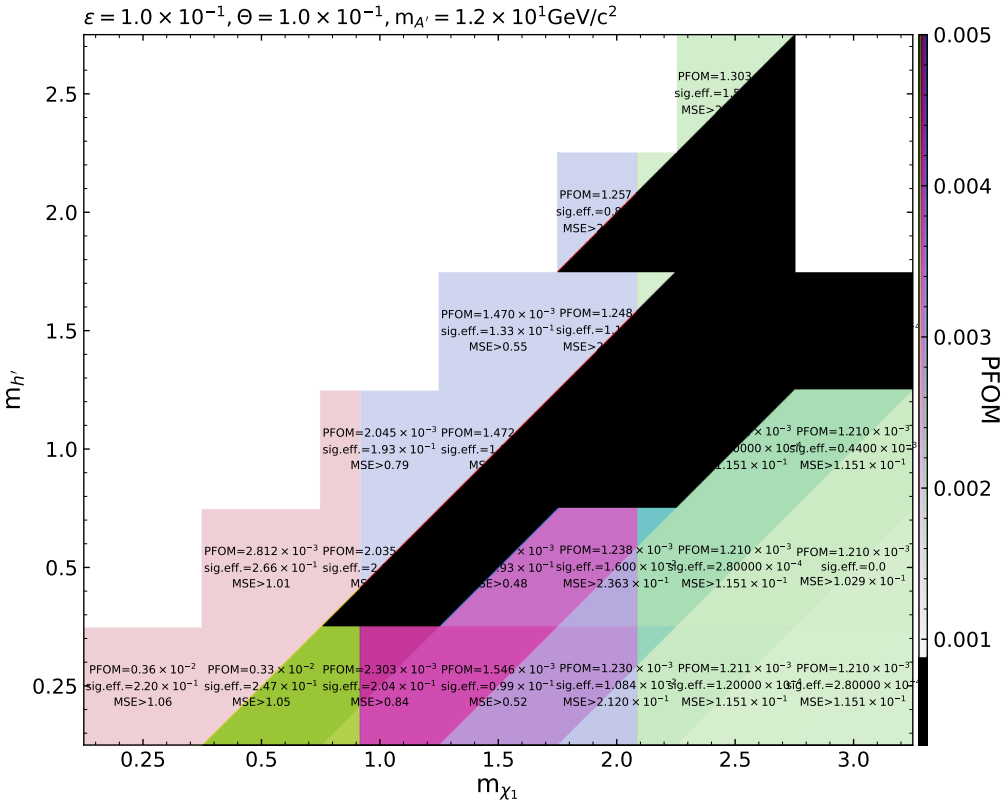

Figure A.123.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 3-dimensional VAE for each mass configuration of the signal.

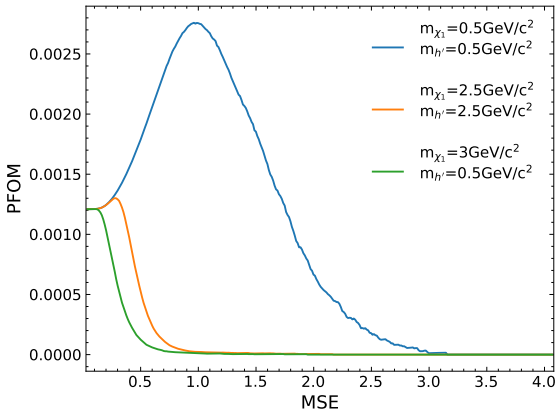

Figure A.124.: PFOM over MSE for the three example signals for the 4-dimensional VAE.

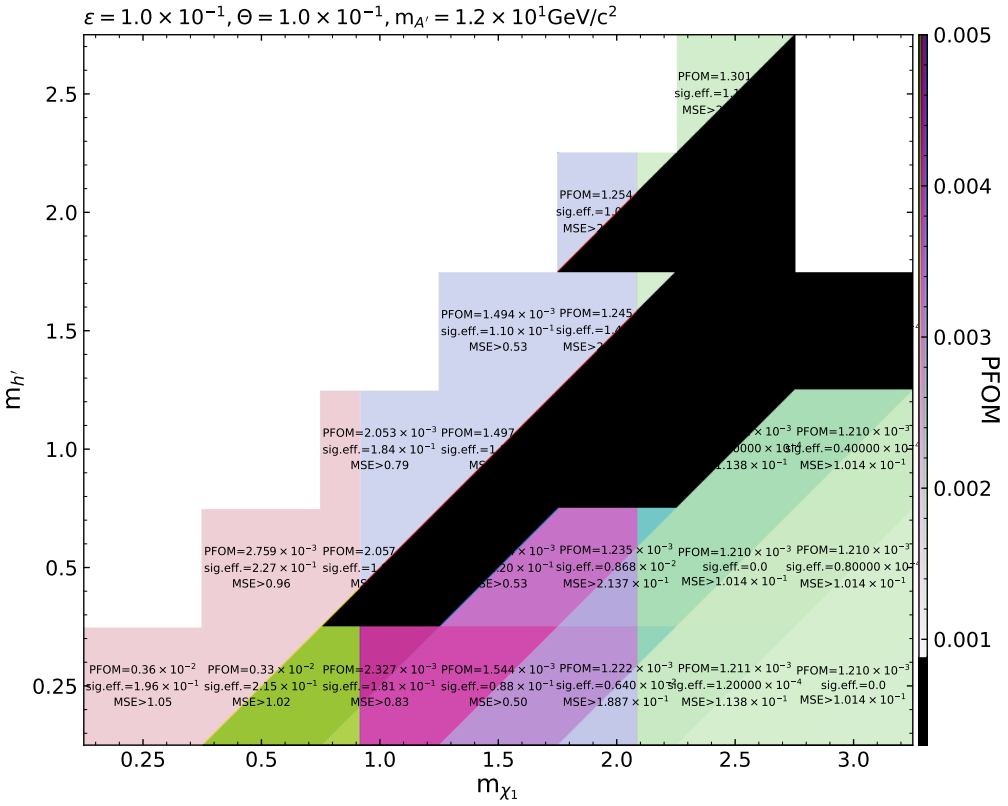

Figure A.125.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 4-dimensional VAE for each mass configuration of the signal.

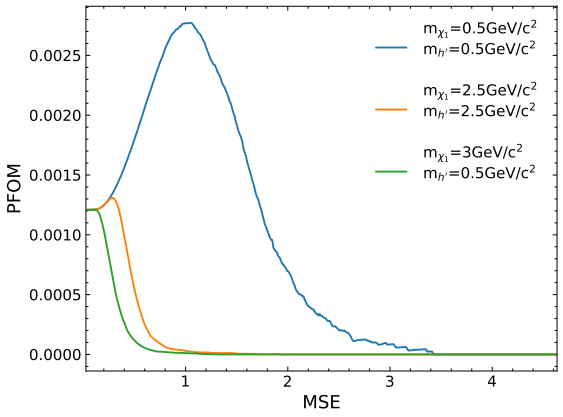

Figure A.126.: PFOM over MSE for the three example signals for the 5-dimensional VAE.

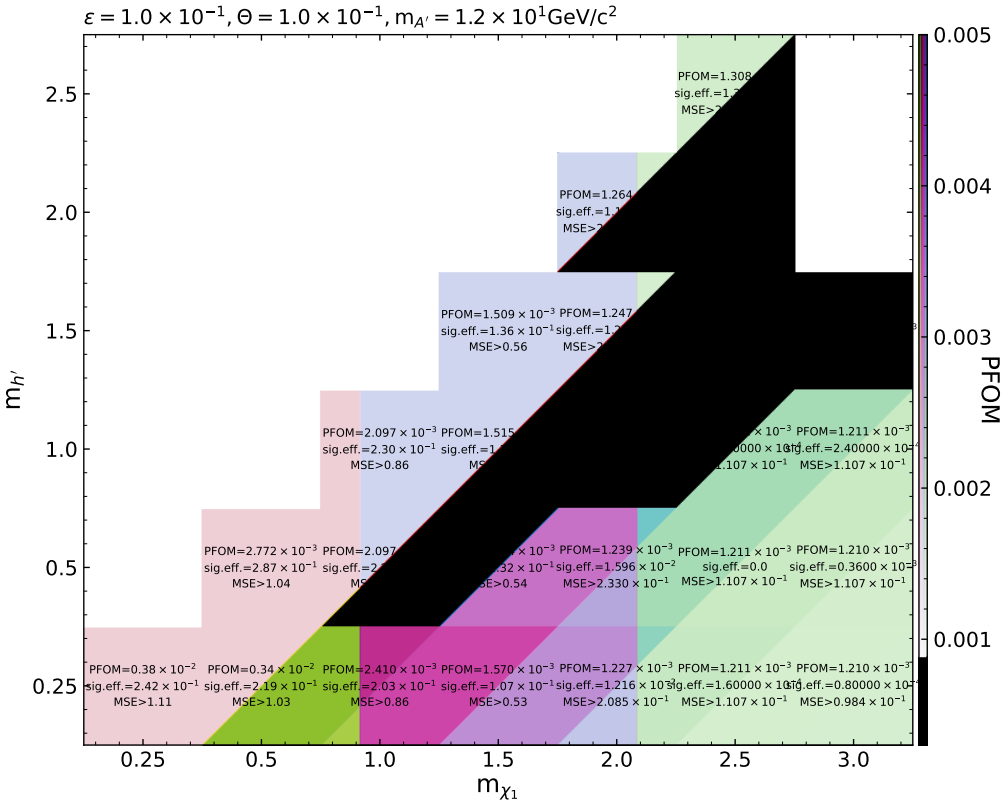

Figure A.127.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 5-dimensional VAE for each mass configuration of the signal.

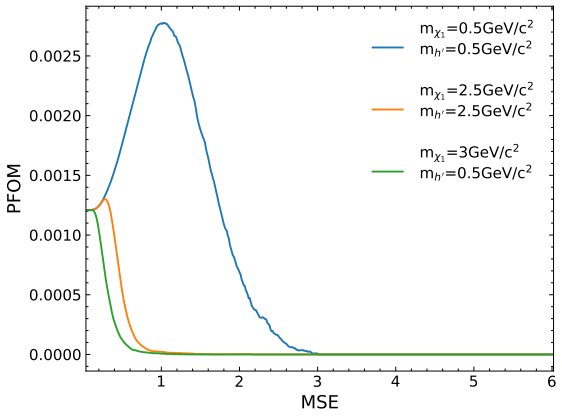

Figure A.128.: PFOM over MSE for the three example signals for the 6-dimensional VAE.

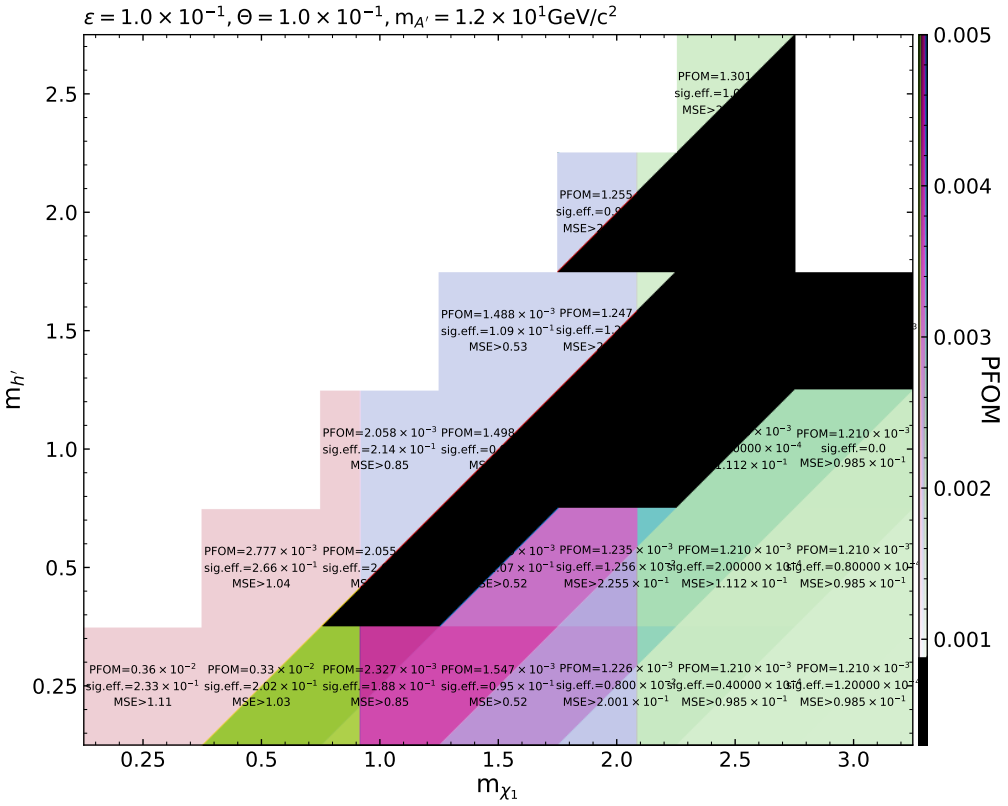

Figure A.129.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 6-dimensional VAE for each mass configuration of the signal.

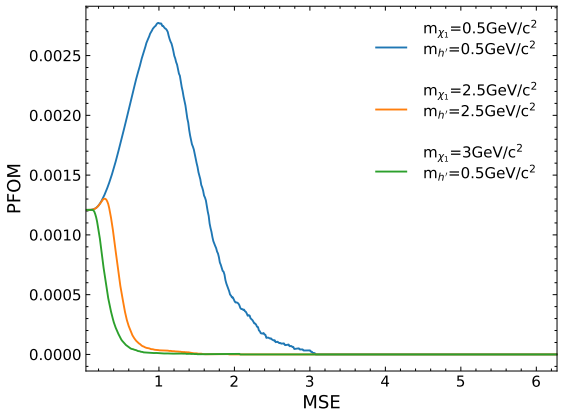

Figure A.130.: PFOM over MSE for the three example signals for the 7-dimensional VAE.

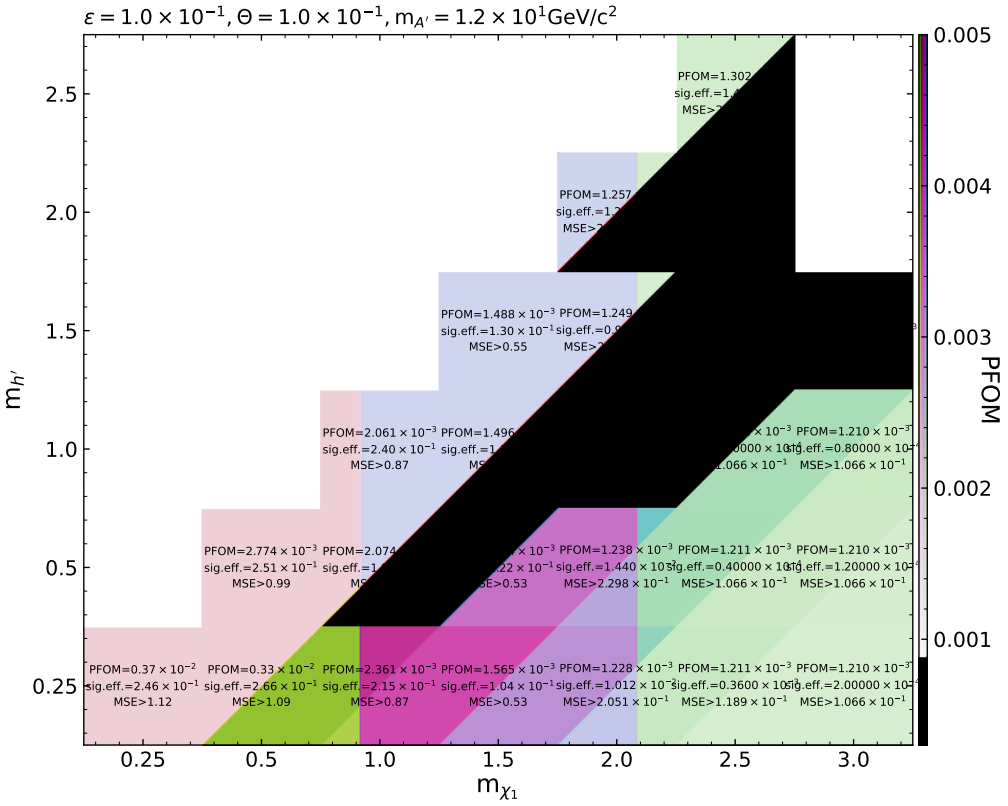

Figure A.131.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 7-dimensional VAE for each mass configuration of the signal.

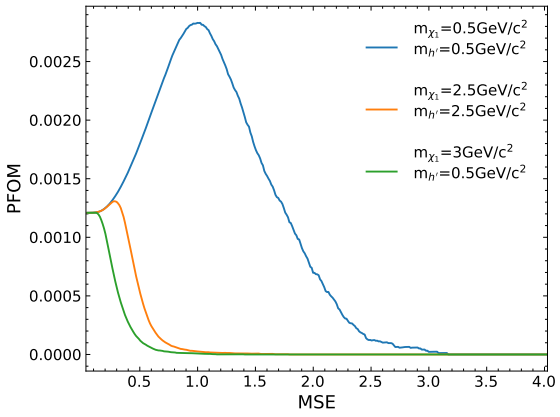

Figure A.132.: PFOM over MSE for the three example signals for the 8-dimensional VAE.

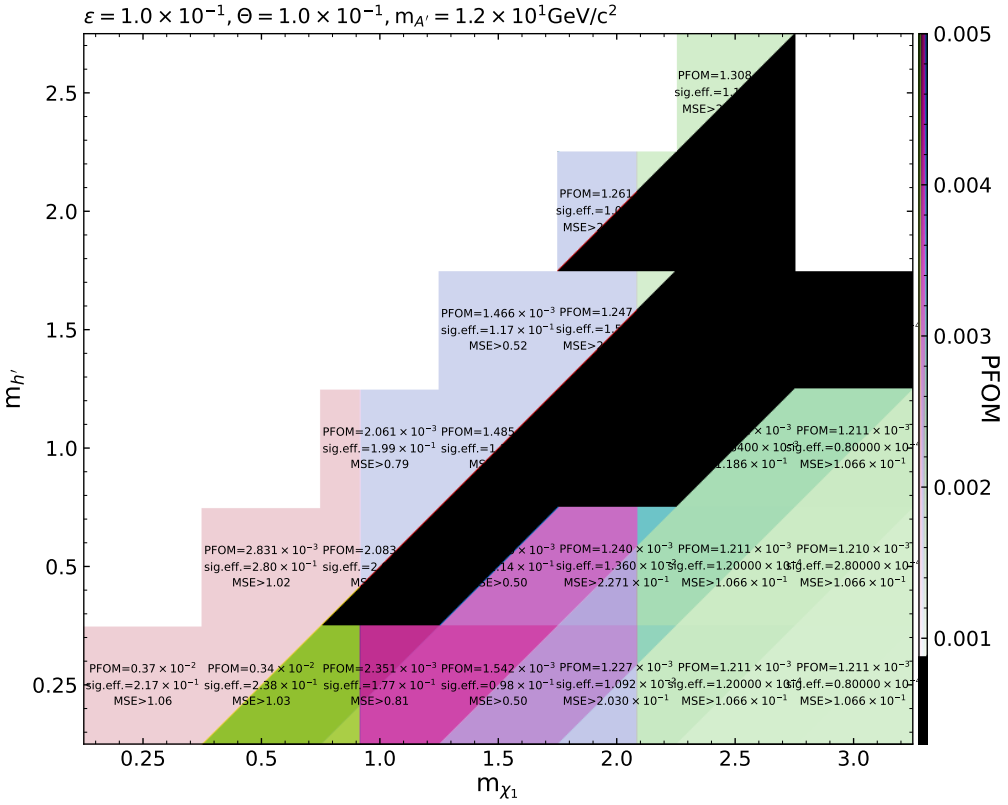

Figure A.133.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 8-dimensional VAE for each mass configuration of the signal.

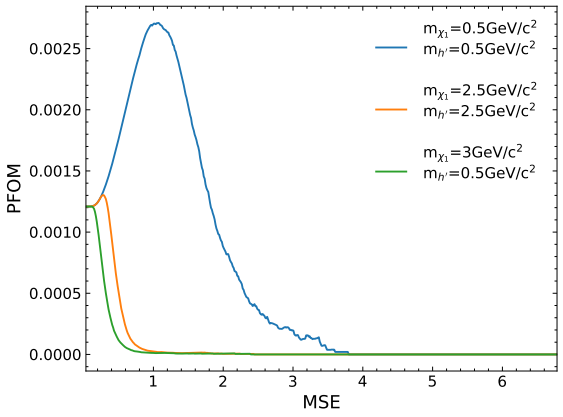

Figure A.134.: PFOM over MSE for the three example signals for the 9-dimensional VAE.

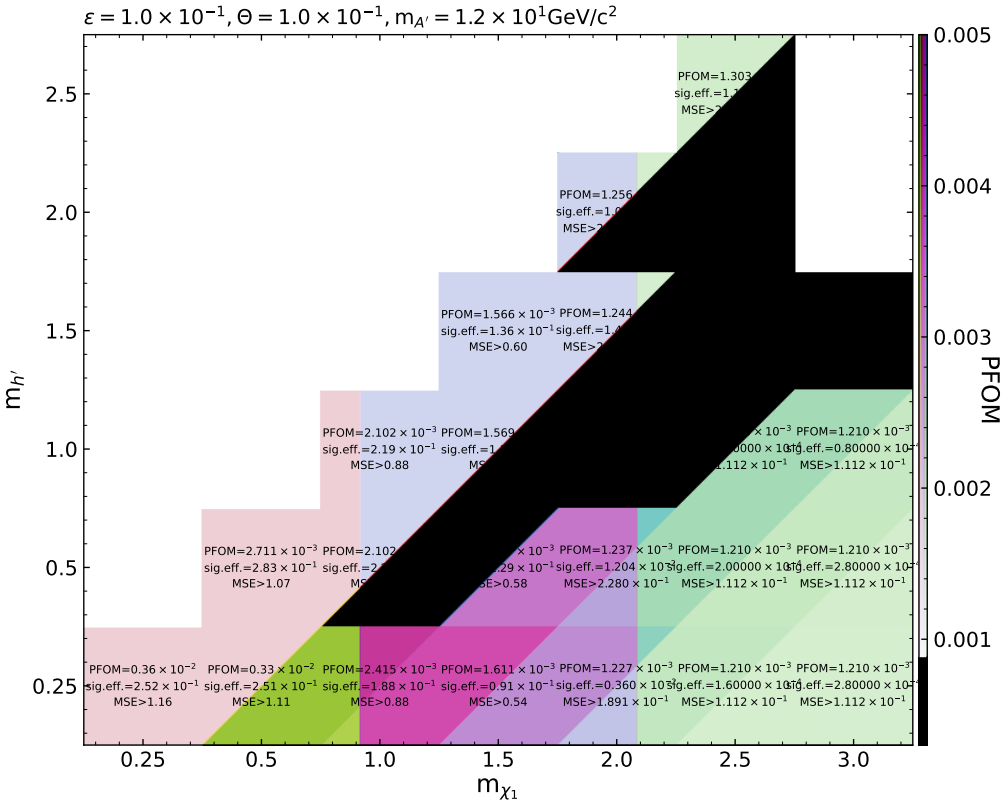

Figure A.135.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 9-dimensional VAE for each mass configuration of the signal.

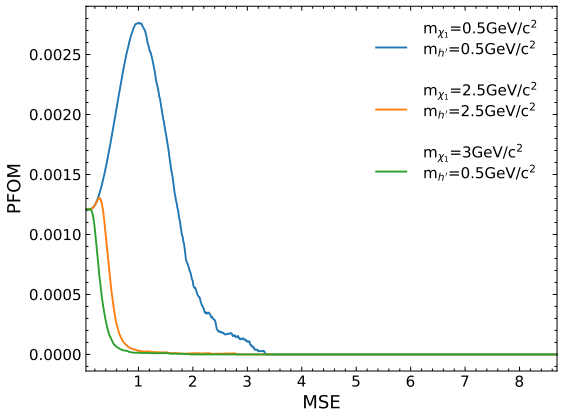

Figure A.136.: PFOM over MSE for the three example signals for the 10-dimensional VAE.

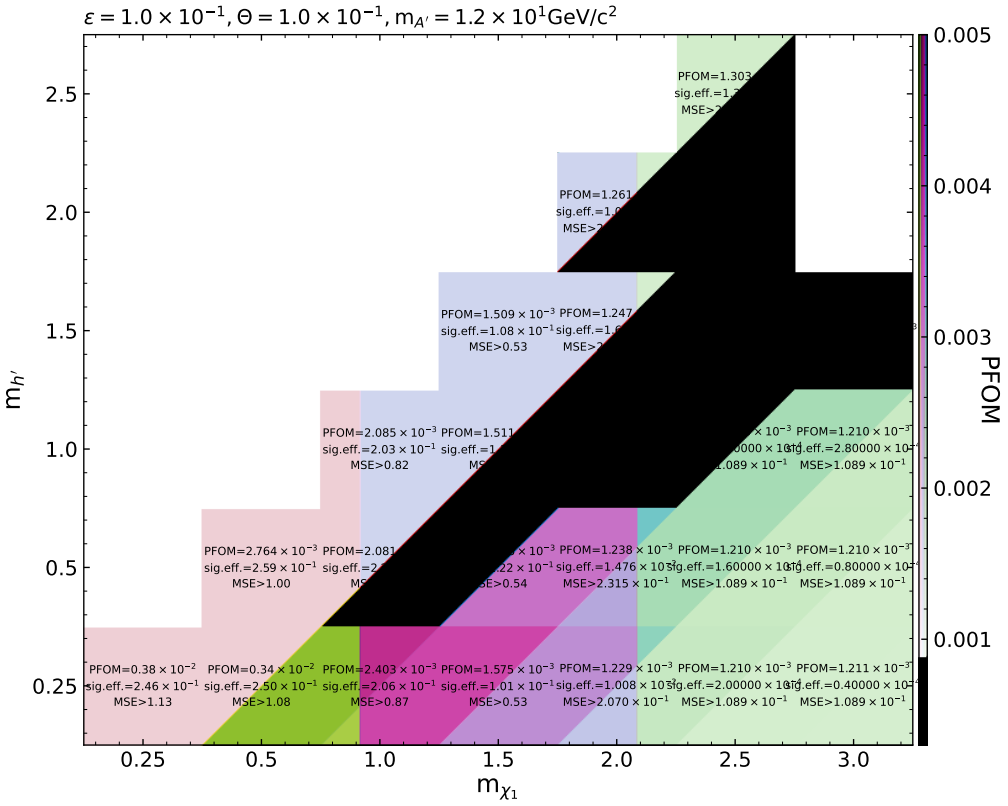

Figure A.137.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 10-dimensional VAE for each mass configuration of the signal.

A.13. PFOM Optimization for DVAEs

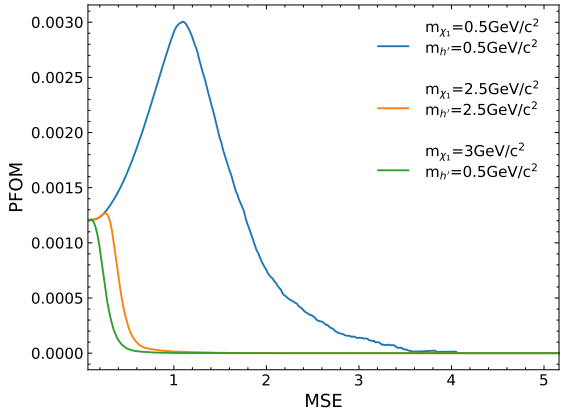

Figure A.138.: PFOM over MSE for the three example signals for the 2-dimensional DVAE.

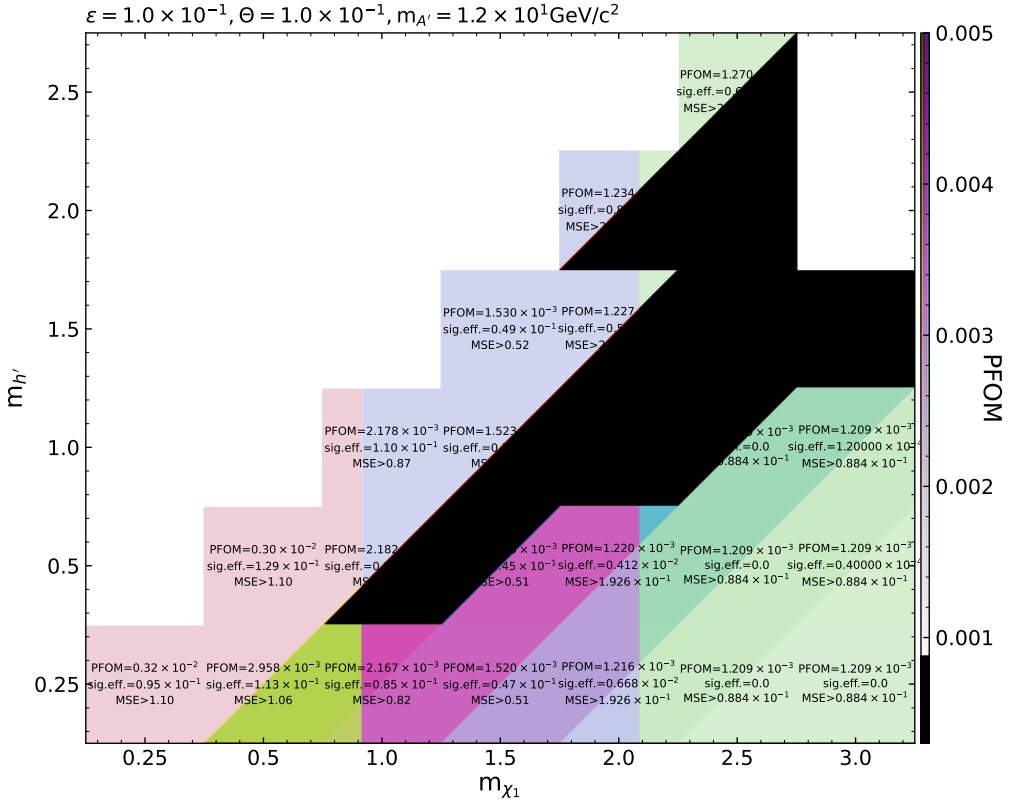

Figure A.139.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 2-dimensional DVAE for each mass configuration of the signal.

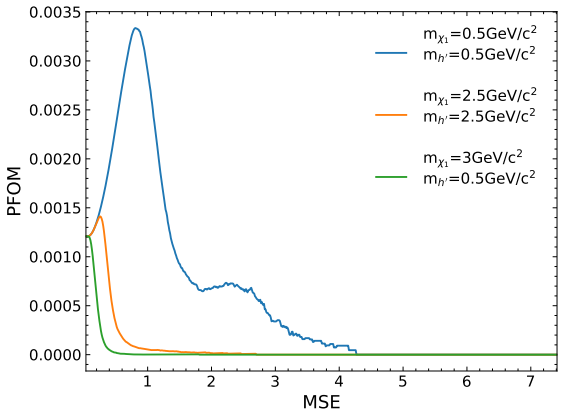

Figure A.140.: PFOM over MSE for the three example signals for the 3-dimensional DVAE.

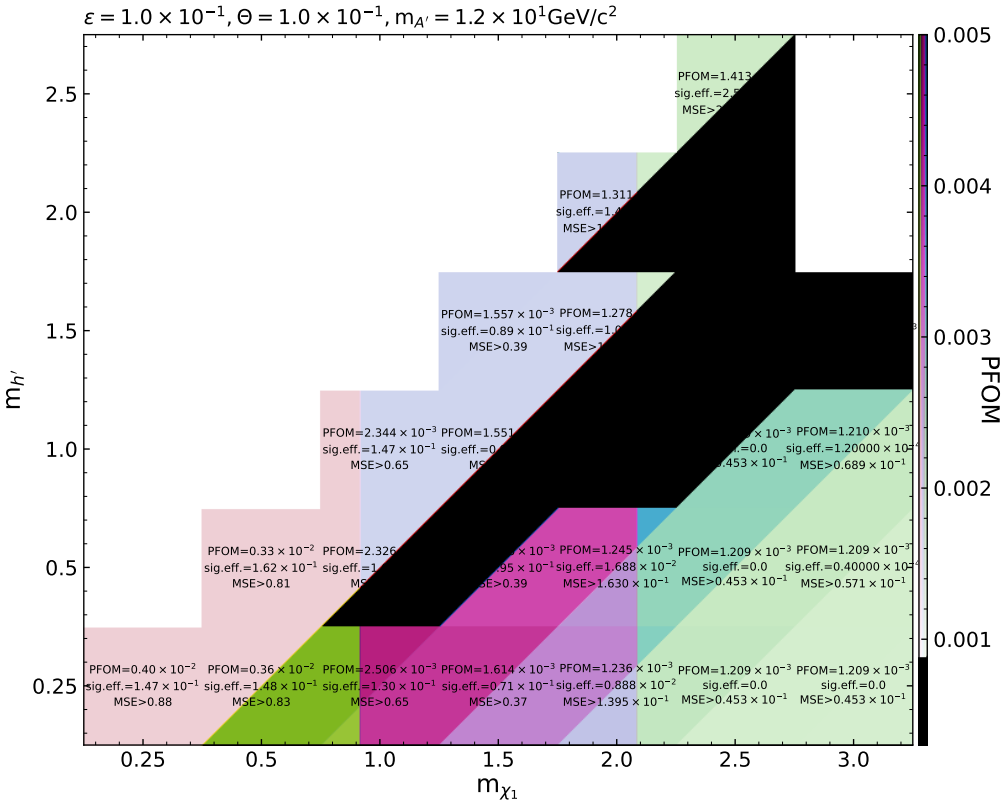

Figure A.141.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 3-dimensional DVAE for each mass configuration of the signal.

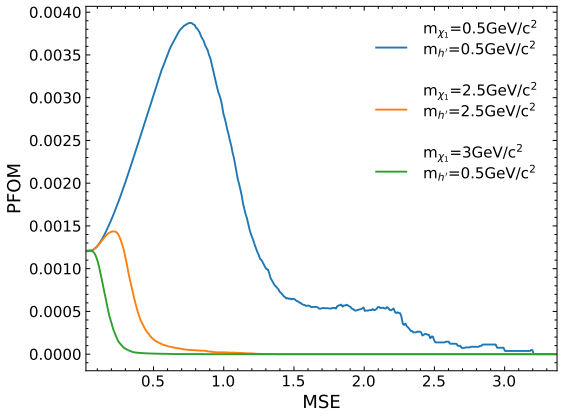

Figure A.142.: PFOM over MSE for the three example signals for the 4-dimensional DVAE.

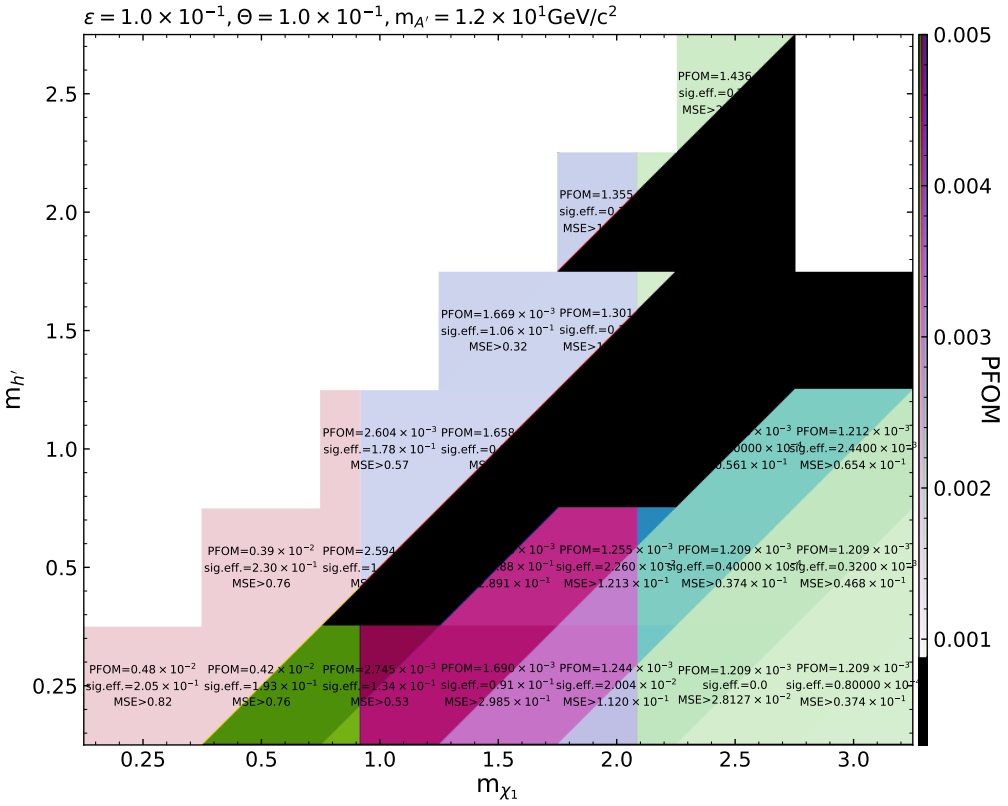

Figure A.143.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 4-dimensional DVAE for each mass configuration of the signal.

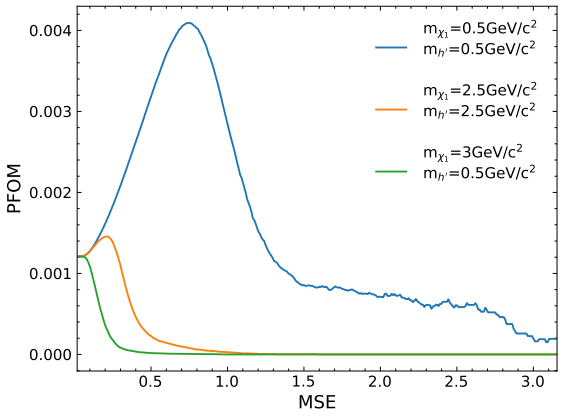

Figure A.144.: PFOM over MSE for the three example signals for the 5-dimensional DVAE.

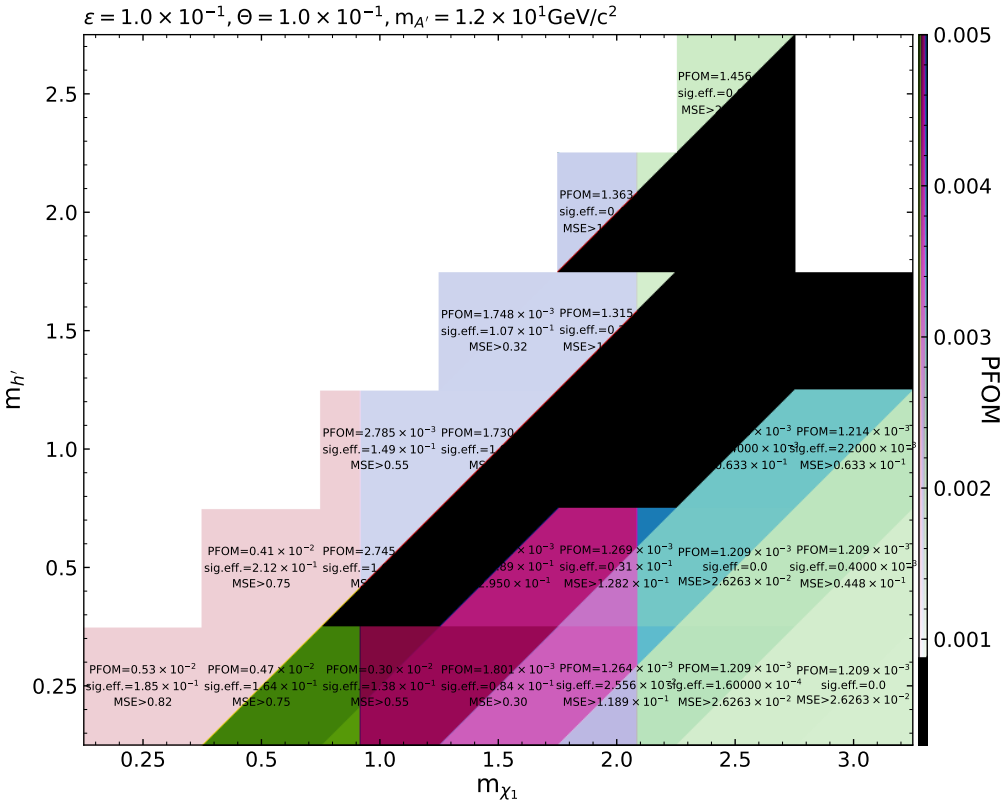

Figure A.145.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 5-dimensional DVAE for each mass configuration of the signal.

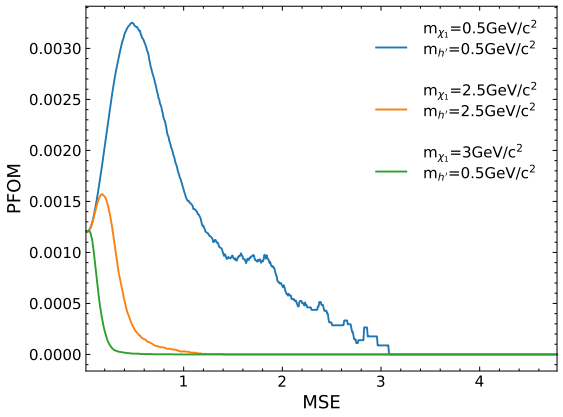

Figure A.146.: PFOM over MSE for the three example signals for the 6-dimensional DVAE.

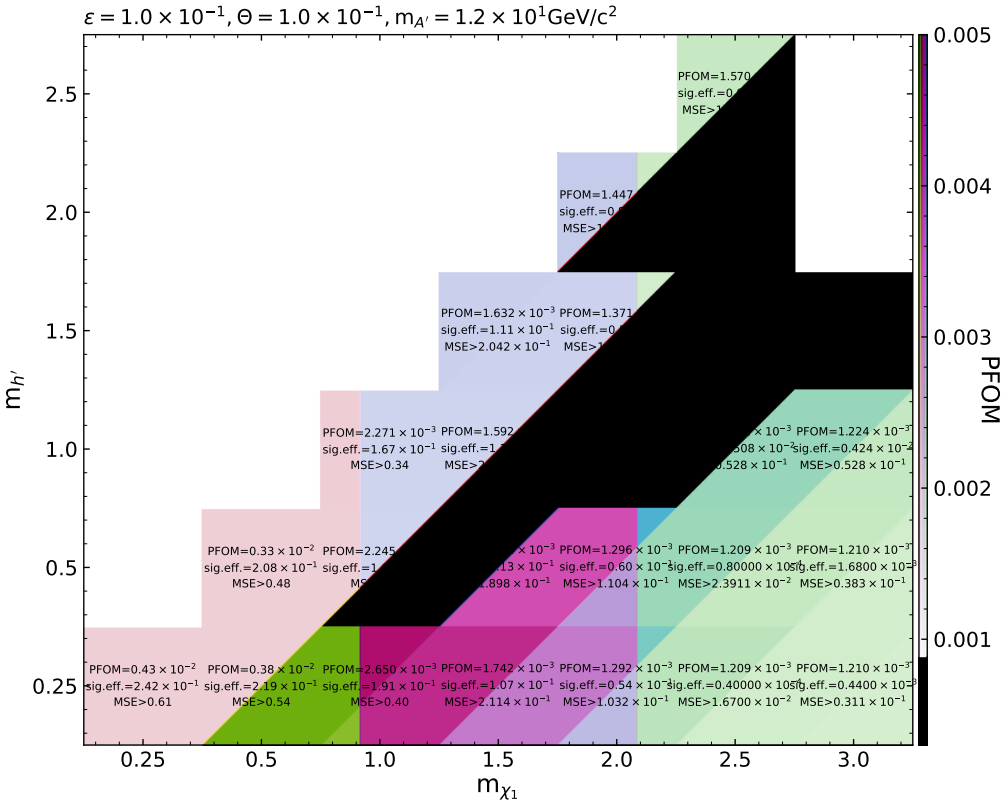

Figure A.147.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 6-dimensional DVAE for each mass configuration of the signal.

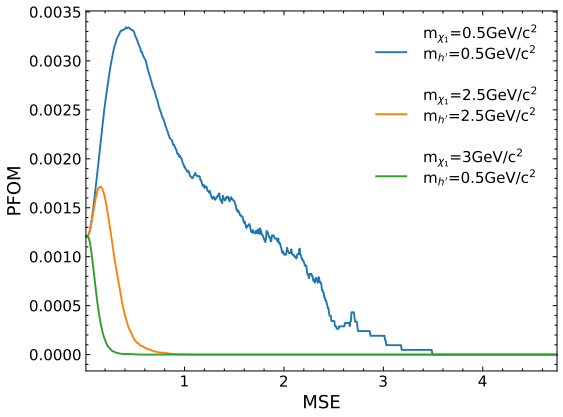

Figure A.148.: PFOM over MSE for the three example signals for the 7-dimensional DVAE.

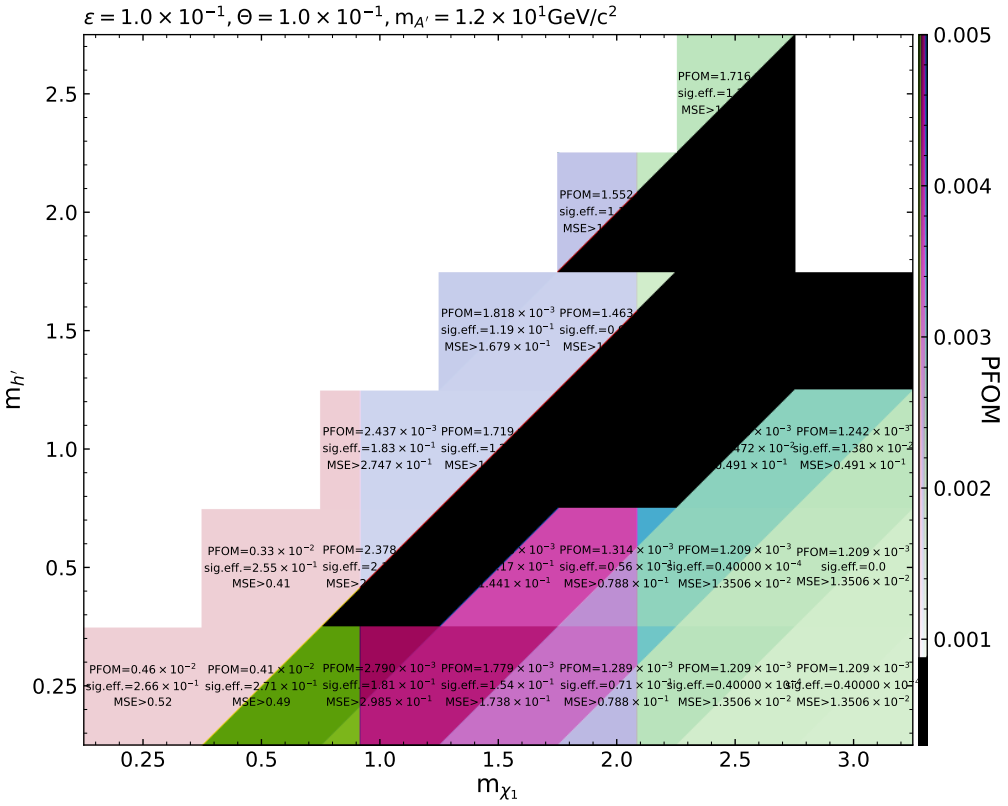

Figure A.149.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 7-dimensional DVAE for each mass configuration of the signal.

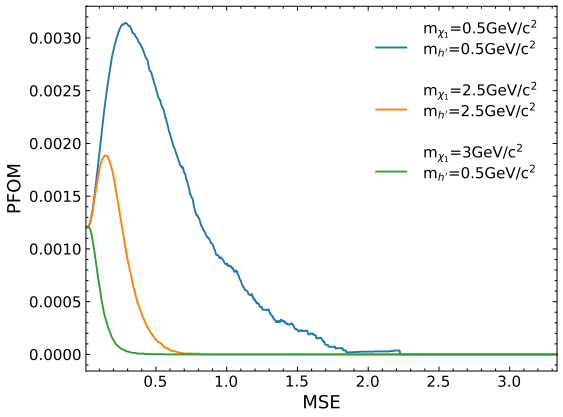

Figure A.150.: PFOM over MSE for the three example signals for the 8-dimensional DVAE.

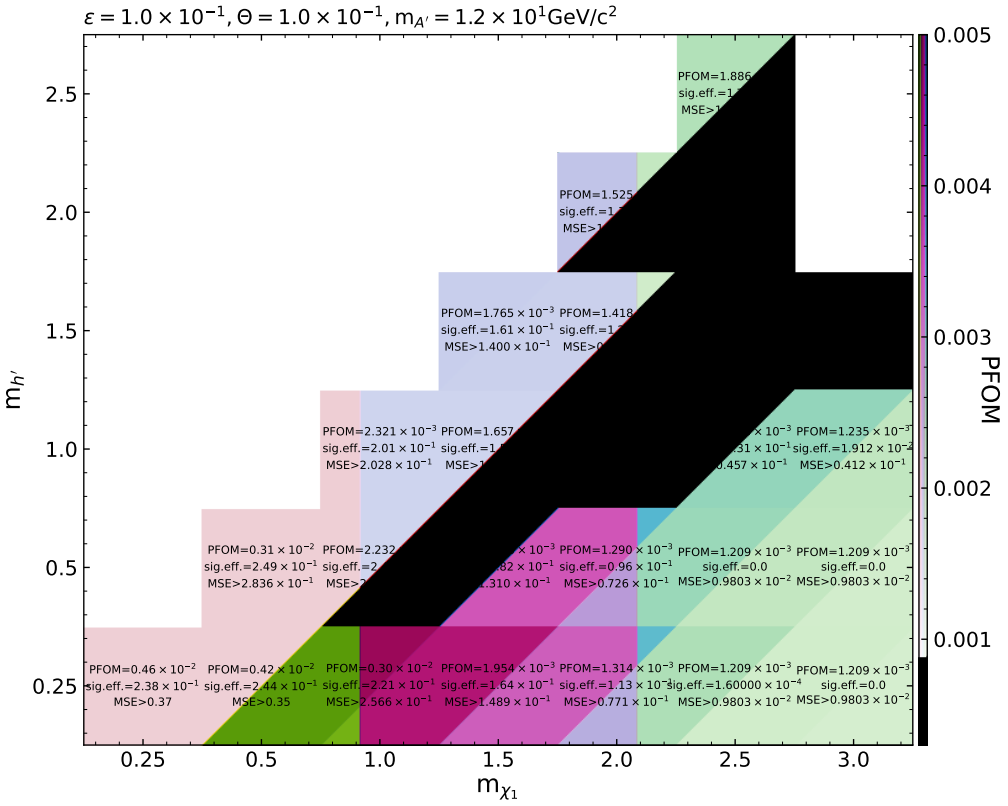

Figure A.151.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 8-dimensional DVAE for each mass configuration of the signal.

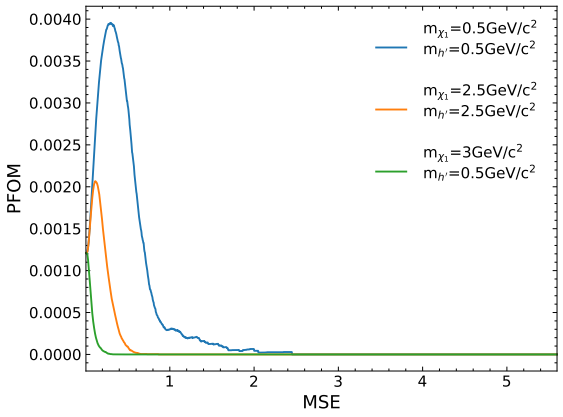

Figure A.152.: PFOM over MSE for the three example signals for the 9-dimensional DVAE.

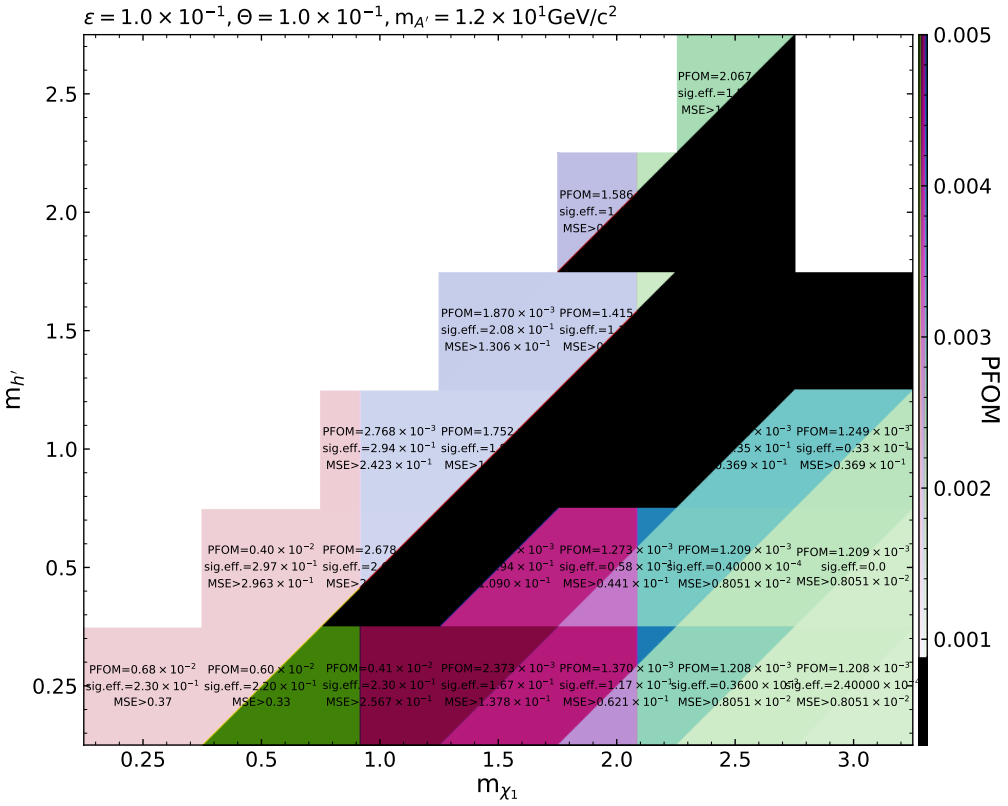

Figure A.153.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 9-dimensional DVAE for each mass configuration of the signal.

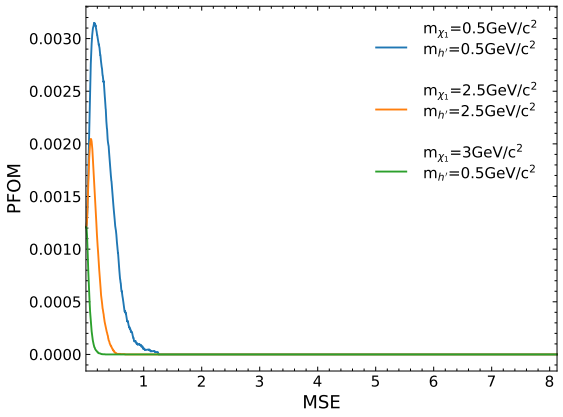

Figure A.154.: PFOM over MSE for the three example signals for the 10-dimensional DVAE.

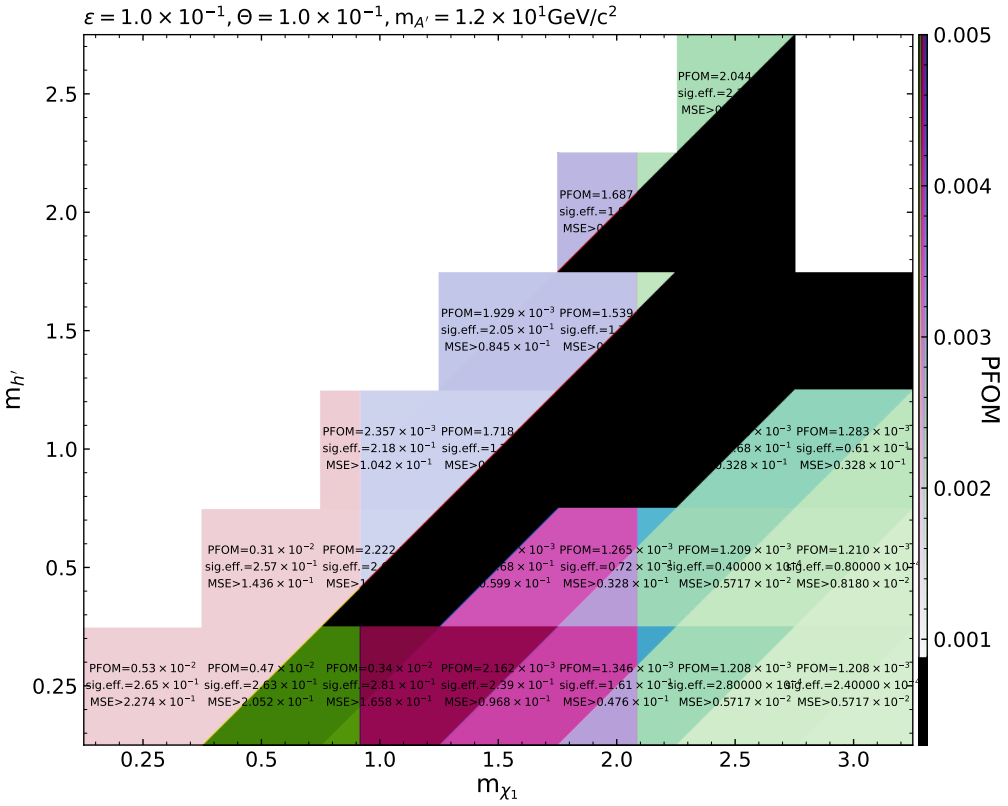

Figure A.155.: PFOM, signal efficiency and optimal MSE value yielded by the PFOM optimization for the 10-dimensional DVAE for each mass configuration of the signal.

A.14. Data-MC Comparison for 8-dimensional AE

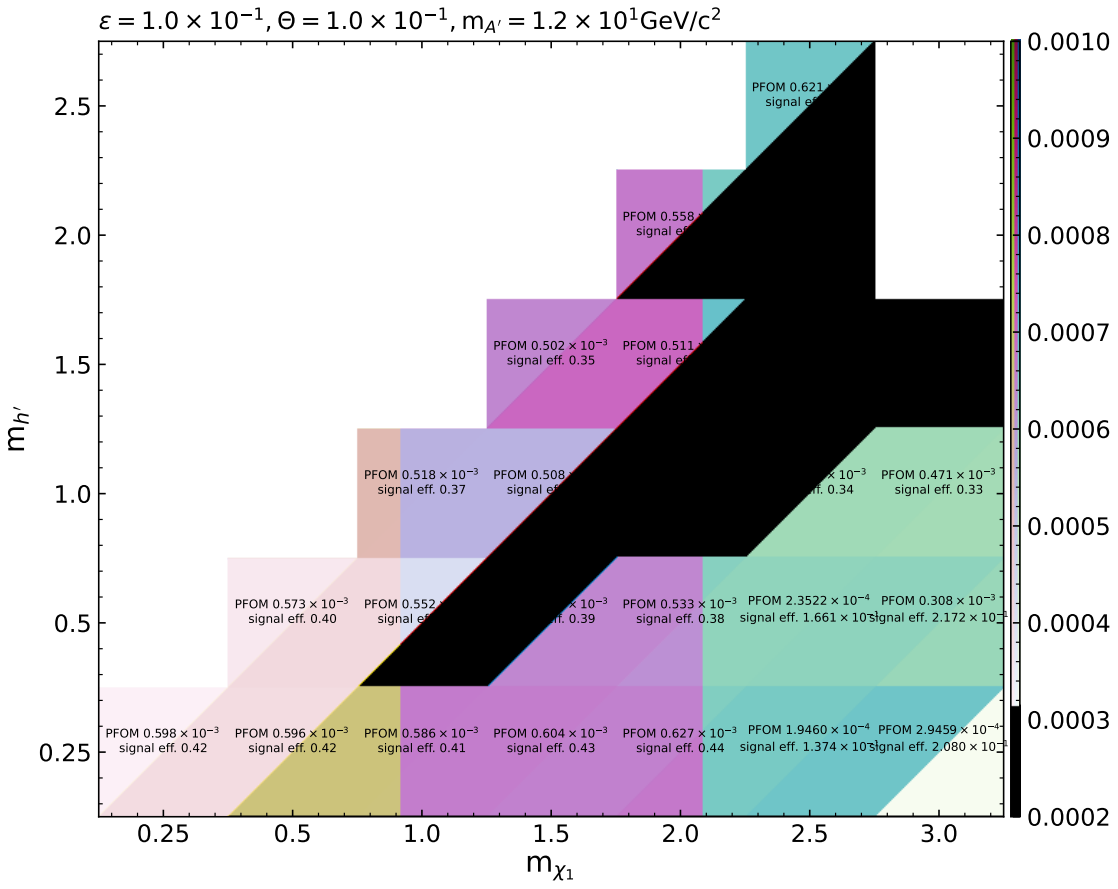

Figure A.156.: PFOM and signal efficiency after selecting only events passing the L1-Trigger for 3 full tracks and the High Level Trigger (HLT). The signal efficiencies are calculated with respect to the total number of simulated events (25000).

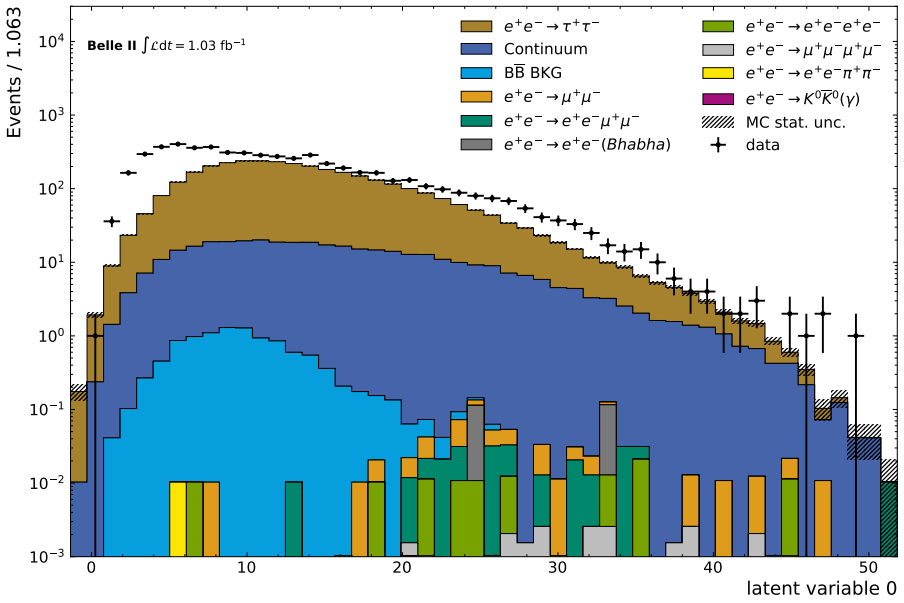

Figure A.157.: Distribution of latent variable 0 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

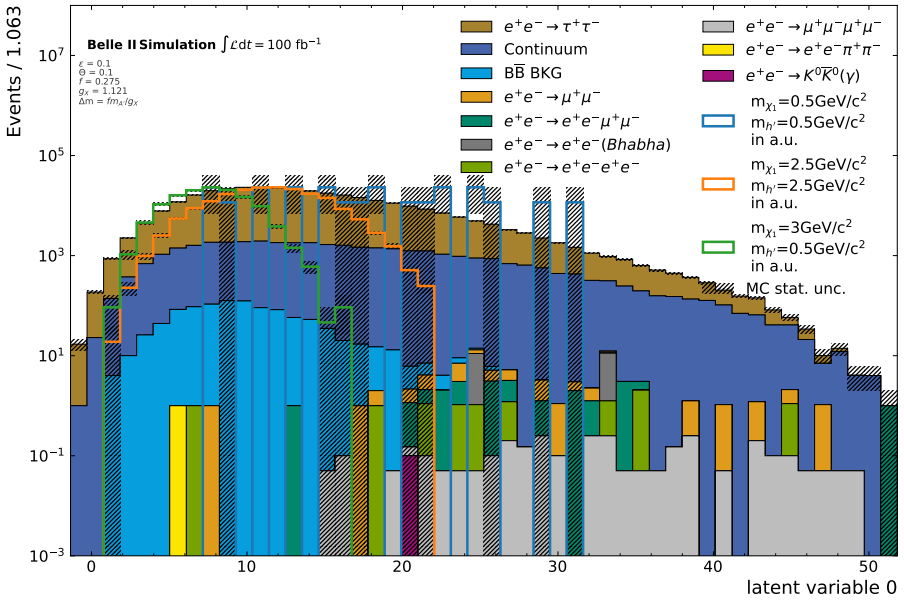

Figure A.158.: Distribution of latent variable 0 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

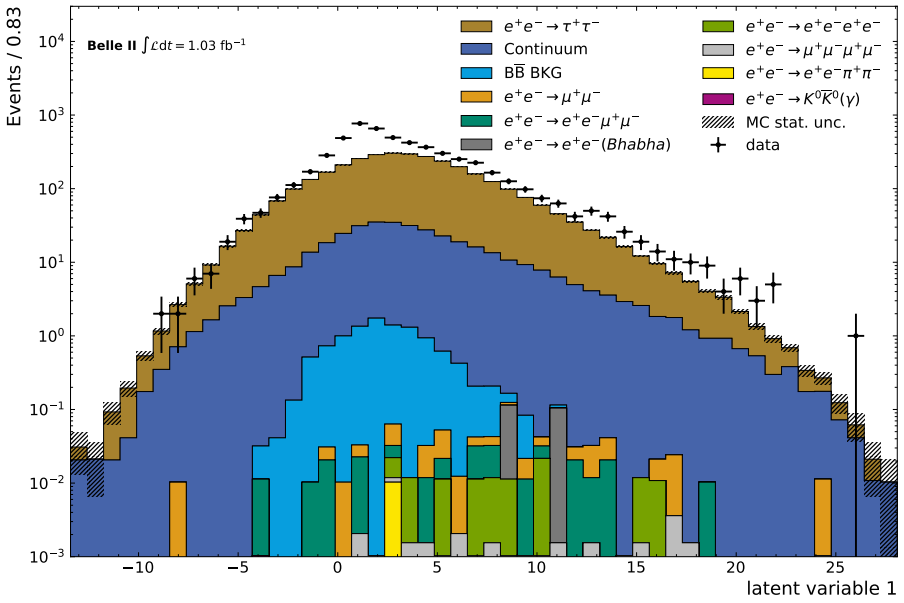

Figure A.159.: Distribution of latent variable 1 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

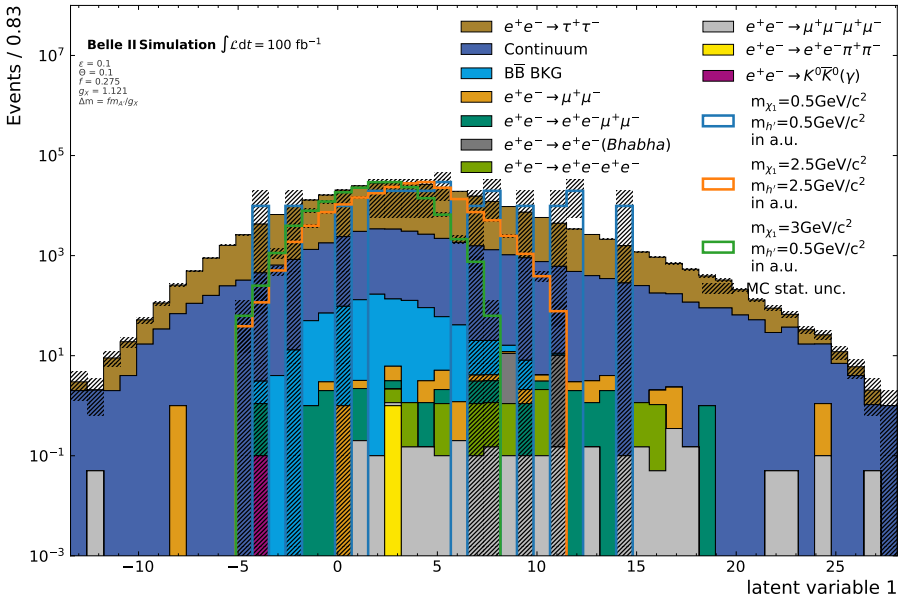

Figure A.160.: Distribution of latent variable 1 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

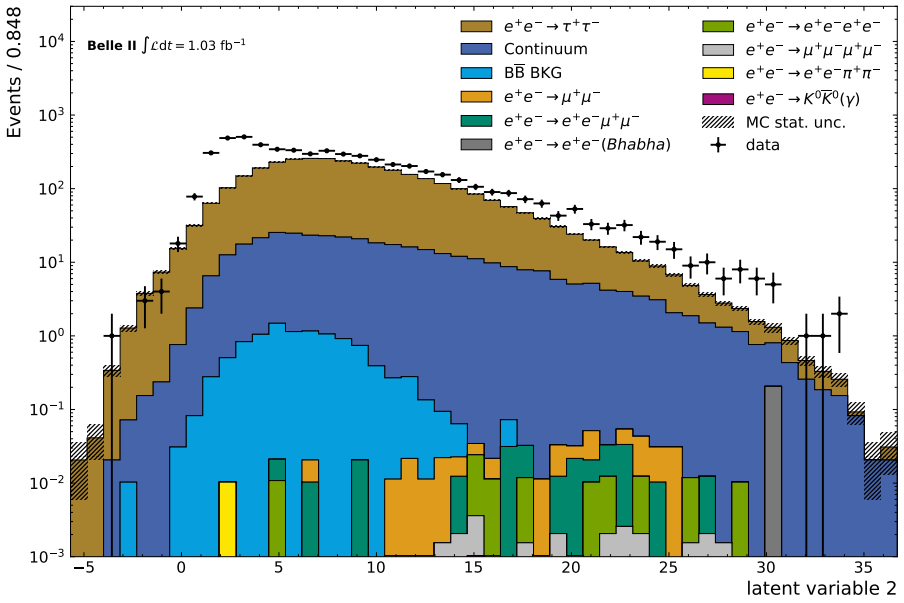

Figure A.161.: Distribution of latent variable 2 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

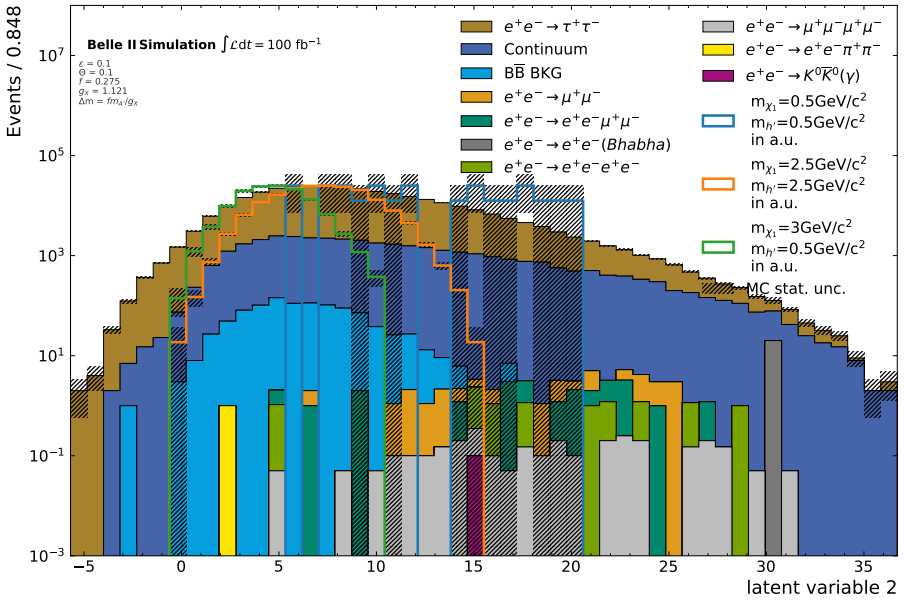

Figure A.162.: Distribution of latent variable 2 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

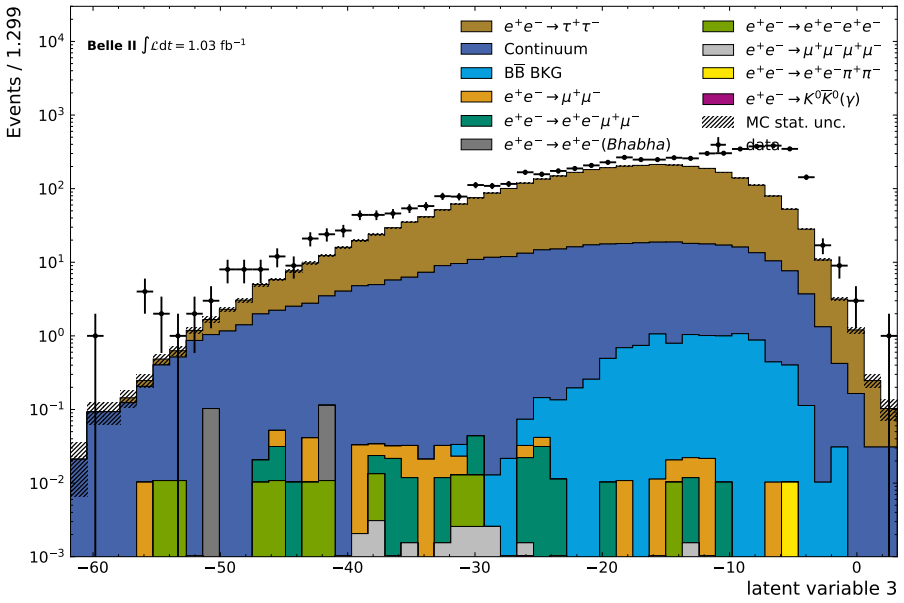

Figure A.163.: Distribution of latent variable 3 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

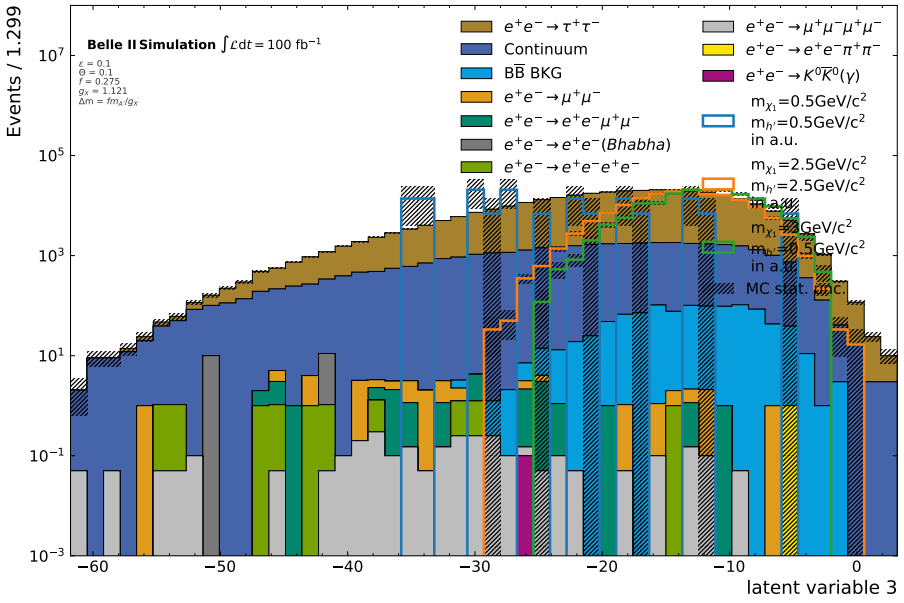

Figure A.164.: Distribution of latent variable 3 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

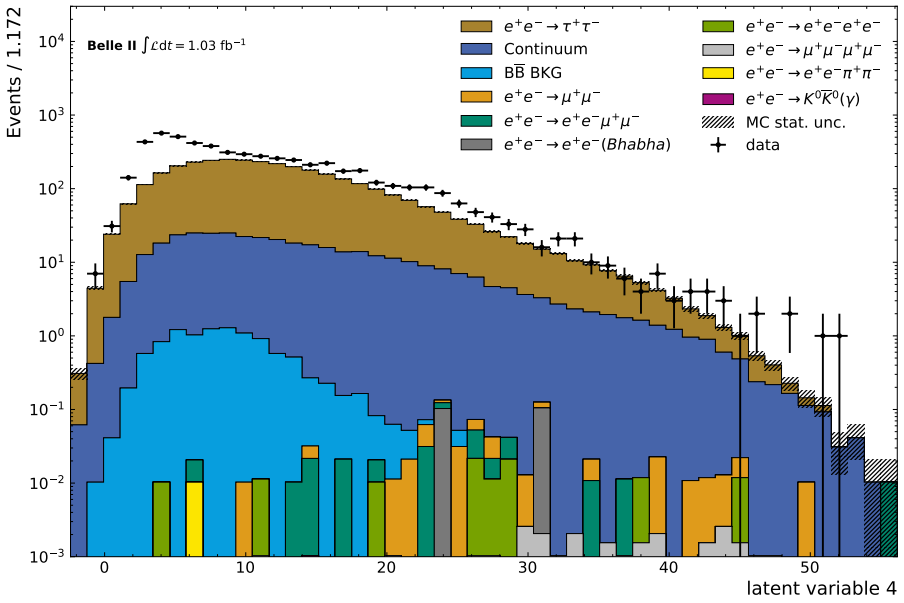

Figure A.165.: Distribution of latent variable 4 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

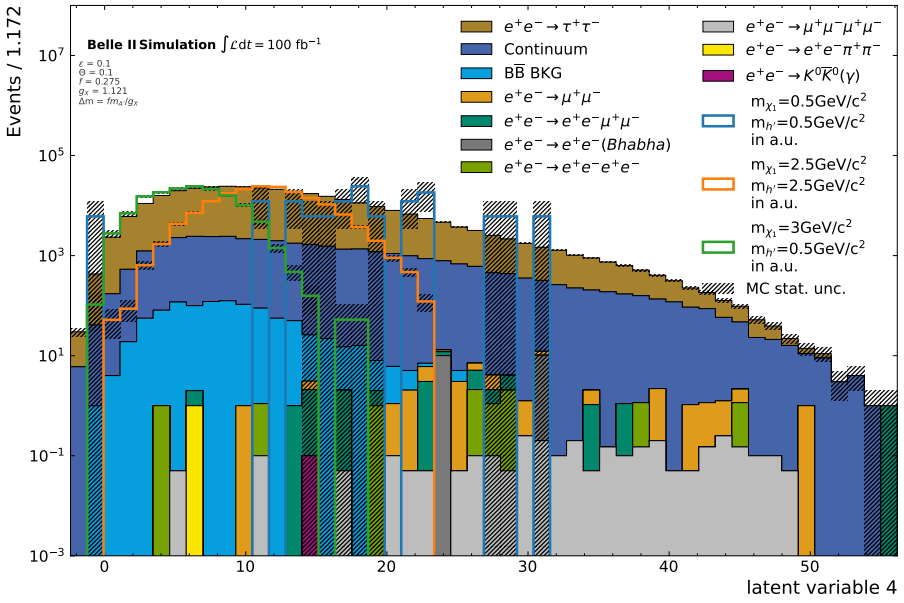

Figure A.166.: Distribution of latent variable 4 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

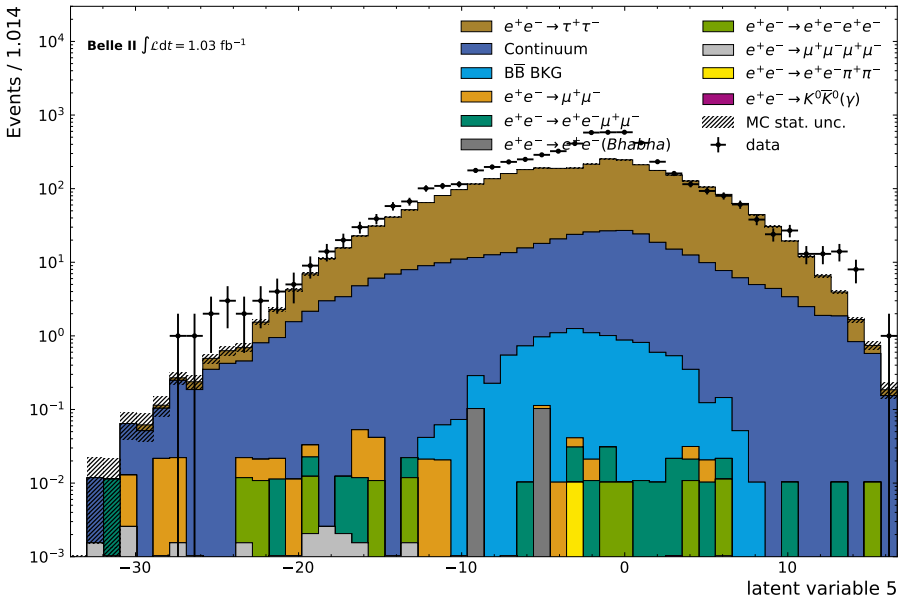

Figure A.167.: Distribution of latent variable 5 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

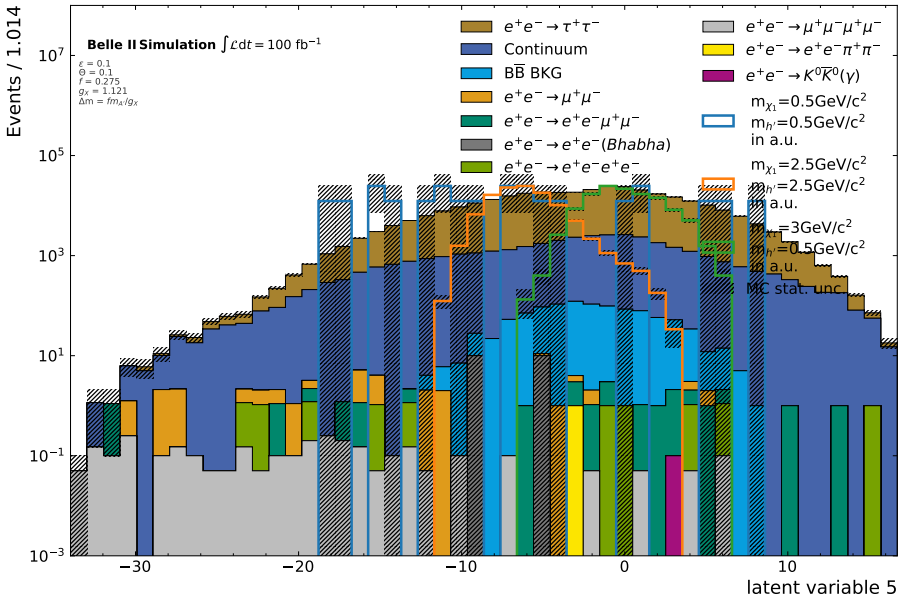

Figure A.168.: Distribution of latent variable 5 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

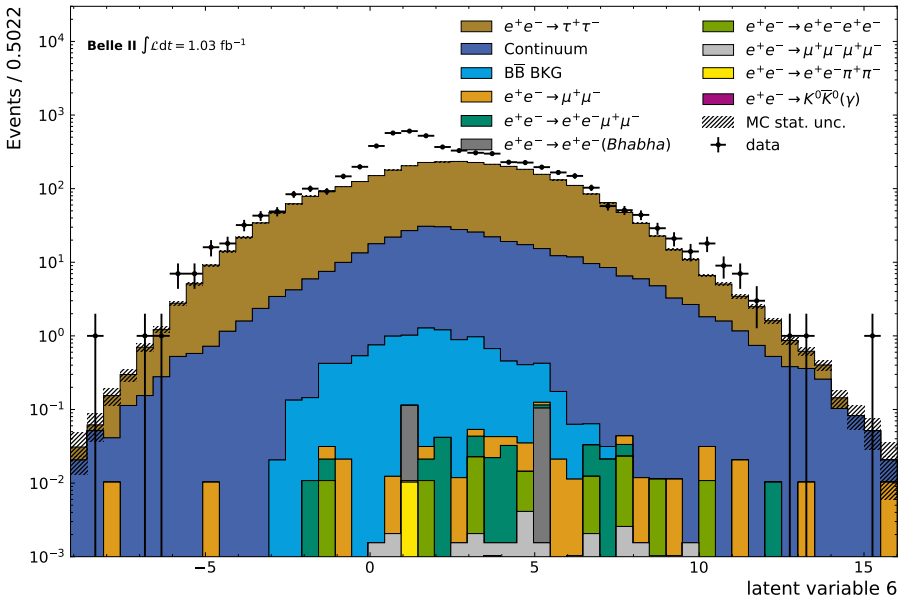

Figure A.169.: Distribution of latent variable 6 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

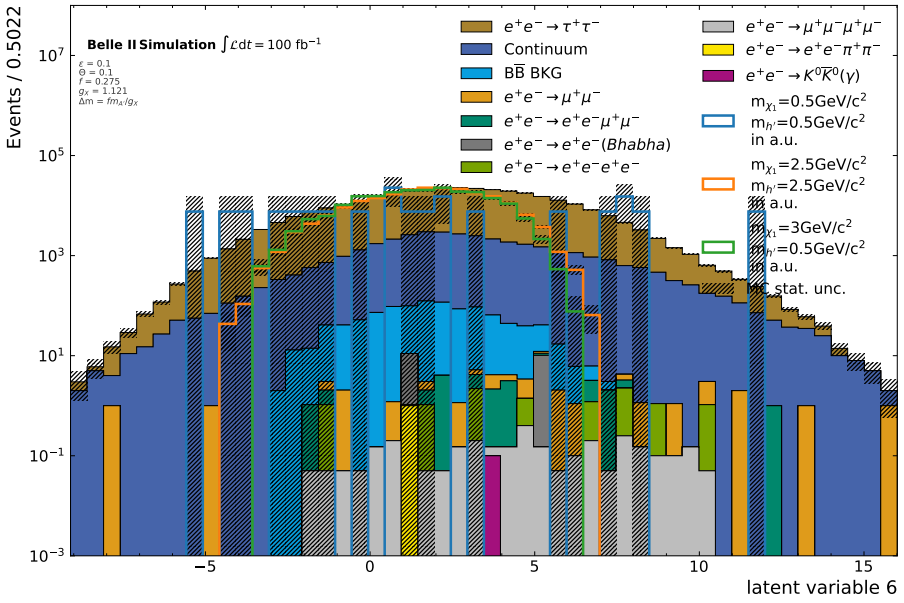

Figure A.170.: Distribution of latent variable 6 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

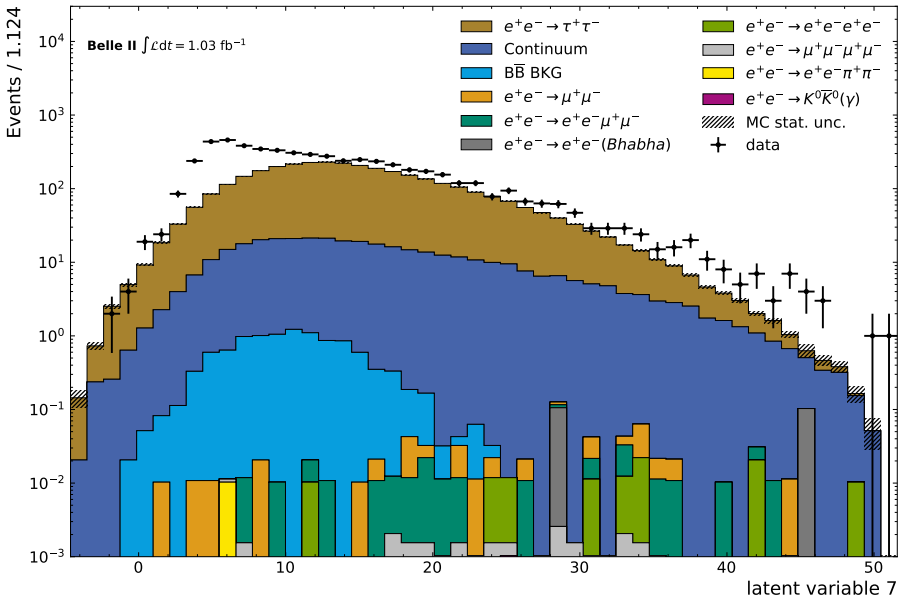

Figure A.171.: Distribution of latent variable 7 of the 8-dimensional AE The background samples are scaled to data luminosity of $\int \mathcal{L} = 1.03\,\text{fb}^{-1}$. For both, data and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

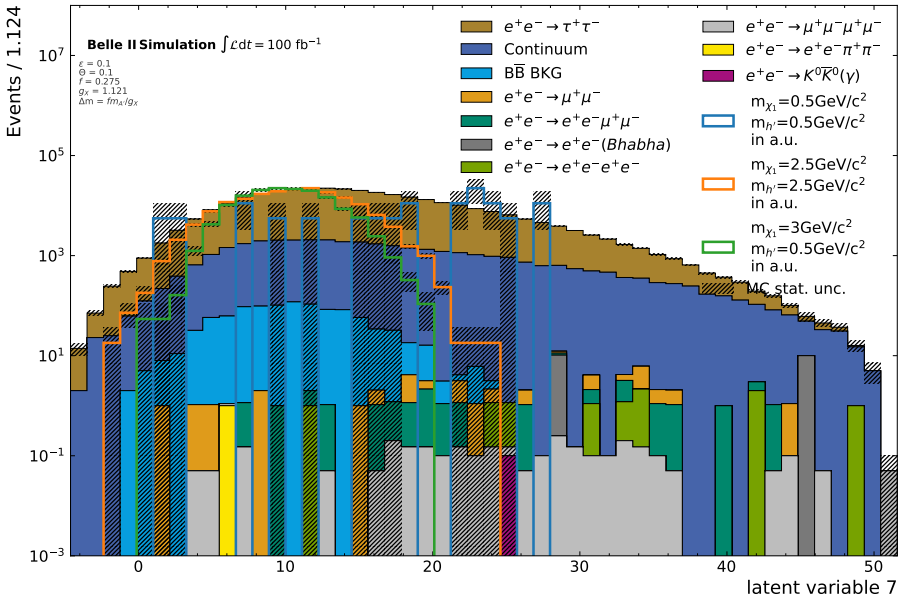

Figure A.172.: Distribution of latent variable 7 of the 8-dimensional AE for the background samples and the three example signals. For both, the signal and background samples, only events that passed the L1 trigger for three full tracks and the HLT are shown.

B. List of Figures

C. List of Tables