Learning to discover: expressive Gaussian mixture models for multi-dimensional simulation and parameter inference in the physical sciences

IOP Publishing Mach. Learn.: Sci. Technol. 3 (2022) 015021 https://doi.org/10.1088/2632-2153/ac4a3b

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
31 August 2021

REVISED
5 January 2022

ACCEPTED FOR PUBLICATION
11 January 2022

PUBLISHED
28 January 2022

PAPER

Learning to discover: expressive Gaussian mixture models for multi-dimensional simulation and parameter inference in the physical sciences

Stephen B Menary and Darren D Price*

Department of Physics & Astronomy, University of Manchester, Manchester, United Kingdom
* Author to whom any correspondence should be addressed.

E-mail: darren.price@manchester.ac.uk

Keywords: statistical inference, scientific discovery, machine learning, simulation

Abstract

We show that density models describing multiple observables with (1) hard boundaries and (2) dependence on external parameters may be created using an auto-regressive Gaussian mixture model. The model is designed to capture how observable spectra are deformed by hypothesis variations, and is made more expressive by projecting data onto a configurable latent space. It may be used as a statistical model for scientific discovery in interpreting experimental observations, for example when constraining the parameters of a physical model or tuning simulation parameters according to calibration data. The model may also be sampled for use within a Monte Carlo simulation chain, or used to estimate likelihood ratios for event classification. The method is demonstrated on simulated high-energy particle physics data considering the anomalous electroweak production of a Z boson in association with a dijet system at the Large Hadron Collider, and the accuracy of inference is tested using a realistic toy example. The developed methods are domain agnostic; they may be used within any field to perform simulation or inference where a dataset consisting of many real-valued observables has conditional dependence on external parameters.

1. Introduction

In the physical sciences, we often use statistical methods to make quantifiable statements about how compatible experimental observations are with different hypotheses about nature. These frameworks, typically frequentist or Bayesian, usually require us to model the expected probability density function (PDF) for all possible observations, conditioned on the hypotheses of interest. Finding such a parameterization for the PDF can be very challenging when data are multi-dimensional.

Within experimental particle physics, often the problem is simplified by observing only one or two dimensions of the data at a time following some initial data selections. For these low-dimensional measurements, we are able to approximate the PDF either parametrically or using histograms, allowing for statistical interpretation of the data. To ensure these simplified measurements contain maximum sensitivity to the processes of interest, hereafter referred to as the 'signal' in contrast with the 'background' of all other processes contained in the dataset, we only select data in regions of phase space for which the frequency of signal is high relative to the background. We note several disadvantages of this approach:

(a) By analyzing data only in select regions of phase space, we lose any potentially useful information contained within other regions.
(b) Different hypotheses may predict different distributions of the data in the high-dimensional space. However, we lose this information when collapsing data into one or two dimensions.
(c) When analyzing histograms, the binning of data discards finely-grained information about the shape of the distribution.

(d) The experimentalist must manually design the selection criteria, observables and binning, making it difficult to ensure that an analysis provides fully optimized sensitivity to all accessible regions of the theory parameter space.

If the expected signal and background PDFs can be modeled parametrically in a space spanning all data dimensions, the PDF ratio contains the expected signal-to-background-ratio at every point in phase space. This means that we do not require restrictive data selections to optimize statistical sensitivity to the signal component. We also do not require binning. The information described above is therefore retained and may be used to provide greater exclusion and discovery potential for all possible new physics models[1].

It has recently been demonstrated that machine-learned density models may be constructed which describe PDFs (or PDF ratios) in a high-dimensional observable space [1–9]. Provided that model bias can be mitigated and systematic uncertainties properly described, these can be used to perform parameter inference or construct likelihood ratios for event classification[2].

Many PDF models may also be sampled from, which is not the case when exclusively modeling the PDF ratio. This has several benefits:

(a) We can verify that the distribution obtained by sampling the model is well-behaved when compared with the training data. Such cross checks are desirable in the physical sciences, where rigorous data interpretation is emphasized.
(b) It may be used to generate new datasets at arbitrary points in parameter space, which the model accomplishes by interpolating between the external parameter values at which training data were provided.
(c) We can numerically estimate the expected distribution of a test-statistic under different parameter hypotheses, instead of assuming an asymptotic form. This aids in the estimation of rigorous frequentist confidence limits.
(d) Once trained, sampling from the density model may be more computationally efficient than running the full simulation package used to generate training data. In this context, density models provide a compelling alternative to other stochastic generative models such as generative adversarial networks [13] and variational auto-encoders [14, 15] for performing steps in a simulation chain [16–19].

In this work, we will show that density models describing multiple observables with (1) a complex multi-dimensional distribution, (2) hard boundaries and (3) dependence on external parameters may be created using an *auto-regressive Gaussian mixture model* (GMM) [5, 6, 20, 21][3]. The model is made more expressive by projecting data onto a configurable latent space. The method is designed to capture how observable spectra are continuously deformed as the external parameters are varied, behaviour which is common in the physical sciences. We hope that this work will provide users with a simple but expressive way to model such datasets in their own domains.

To study the performance of our method on a high-dimensional dataset of physically realistic observables, we use simulations of particle physics data sensitive to anomalies in the electroweak production of a Z boson in association with a dijet system. We demonstrate the degree to which our trained density models can describe this data, capturing how it is deformed as two physical parameters are varied. We then use a toy example, in which we can access the ground-truth PDF, to demonstrate that accurate parameter estimates and exclusion limits may be obtained using our method. This is not possible using the physical example because we do not have access to the ground-truth PDF with which to compare.

This paper is structured as follows. In section 2 we describe the generation of training data used throughout the paper, and explain the physical basis behind it. In section 3 we describe how data are transformed onto the latent space and how the density model is built. We then discuss several features of the model. In section 4 we construct a 12-dimensional model to study the ability to describe a highly multi-dimensional dataset. In section 5 we construct a 4-dimensional model with dependence on two external parameters to study the ability to learn the parameter dependence. In section 6 we study the accuracy of inference using our toy example. In section 7 we conclude.

[1] Here we consider only the optimization of statistical sensitivity and assume that the PDFs can be modeled with sufficient accuracy and well-described systematic uncertainties. This may be challenging in a real-world analysis which includes data-driven constraints and regions with large systematic effects.

[2] See e.g. [10–12] for alternative approaches for enhancing sensitivity to new physics models using machine-learned classifiers and anomaly detection.

[3] Whilst we were unable to find examples which combine all these properties, Bishop [20] and Variani *et al* [21] provide examples of GMMs for density estimation parameterized using neural networks and Papamakarios *et al* [5] and Uria *et al* [6] of auto-regressive density estimation used to model multi-dimensional data.

Whilst these experiments demonstrate that the method is performant on datasets of realistic observables within the domain of high-energy physics, we emphasize that it may be used to model any dataset of continuous observables for which a high-dimensional PDF is deformed by parameter variations, regardless of scientific domain, provided that appropriate training data may be provided.

2. Experimental setup

To test our method in a real-world environment, we consider the electroweak production of a Z boson in association with a dijet system occurring in high-energy proton–proton collisions at the Large Hadron Collider. This process is labeled EW Zjj in the remainder of this text. It is often referred to as the Vector Boson Fusion production of a Z boson.

We choose to model the EW Zjj process for several reasons. Firstly, it provides a number of physically interesting observables which are correlated, challenging our method to capture a feature-rich high-dimensional distribution. Secondly, there exist new physics models which are expected to continuously deform this distribution in distinct ways as different parameters-of-interest are varied. Finally, it is a process of interest for current and future LHC experiments. Nonetheless, we emphasize that the EW Zjj process is intended to be a representative example using which we test the ability of our method to overcome general modeling challenges, and we hope that the method may be used to model smoothly-varying parameter-dependent high-dimensional datasets in any domain.

Each 'event' is the observation of many particles created by a single proton–proton collision. High-energy physics datasets typically consists of $\mathcal{O}(100-100\,\mathrm{M})$ events, depending on the pre-selection criteria applied. By identifying the particles produced, and measuring their kinematic properties and other high-level 'observables', we study the processes which contributed to their production.

The EW Zjj process is characterized by a final state of two jets of hadrons along with two oppositely charged electrons or muons which are produced by a Z-boson decay. Since the EW Zjj process is defined by a t-channel exchange of a colour-neutral weak boson between the two incoming partons, these jets are typically separated by a wider rapidity than in the dominant background process which contains a t-channel exchange of a gluon. As a result, experimental analyses often select events with a large dijet rapidity separation (or large invariant mass) to enhance the proportion of signal within their sample. We may measure the event rate as a function of many observables. We expect that the presence of certain new particles/forces will induce distortions in the shape or magnitude of these spectra relative to the precise predictions of the Standard Model of Particle Physics (SM). These measurements enable a rich discovery potential for new natural phenomena and the derivation of constraints on the theoretical models describing them.

The binned one-dimensional kinematic spectra of particles produced via EW Zjj in high-energy proton–proton collisions were recently measured [22, 23] by the ATLAS experiment [24]. Exclusion limits were derived for several parameters of the SM effective field theory (SMEFT) in the Warsaw basis [25], which characterize the presence of any novel physics phenomena in such interactions. In this work, we consider how EW Zjj events are affected by variations of the SMEFT parameters c_{HWB} and $\tilde{c}_W$. These parameters extend the SM Lagrangian $\mathcal{L}_{\mathrm{SM}}$ by the addition of two non-renormalizable terms with mass dimension six. These additional terms modify how electroweak bosons interact with one another, impacting the rate and expected kinematic distribution of EW Zjj events. These modifications reflect the indirect effects of new physics interactions above some energy scale Λ which is not directly probed by the experiment. We will assume $\Lambda = 1$ TeV throughout, noting that other choices simply correspond to a re-scaling of c_{HWB} and $\tilde{c}_W$ within this parameterization. The effective Lagrangian is [25–27]:

$$\mathcal{L} = \mathcal{L}_{\mathrm{SM}} + \frac{c_{\mathrm{HWB}}}{\Lambda^2} H^\dagger \tau^I H W^I_{\mu\nu} B^{\mu\nu} + \frac{\tilde{c}_W}{\Lambda^2} \epsilon^{IJK} \tilde{W}^{I\nu}_{\mu} W^{J\rho}_{\nu} W^{K\mu}_{\rho}, \tag{1}$$

where H is the Higgs doublet, τ are the Pauli matrices, $W^{\mu\nu}$ and $B^{\mu\nu}$ are the electroweak field strength tensors, ε are anti-symmetric tensors with $\epsilon_{012} = \epsilon_{0123} = 1$, $\tilde{W}^{\mu\nu} = \frac{1}{2}\epsilon^{\mu\nu}_{\rho\sigma} W^{\rho\sigma}$ and we neglect Hermitian conjugates. In this work, we use simulated events to construct high-dimensional statistical models which describe many of the kinematic observables considered in the ATLAS analysis. Of the six parameters constrained within the ATLAS analysis, we choose to study c_{HWB} and $\tilde{c}_W$ because they are shown to vary the expected PDF in distinctly different ways. Simultaneously modeling both parameters therefore provides a more ambitious test for the efficacy of our methods.

Ground truth events are generated using the `Madgraph5` (`MG5`) [28] program with perturbative calculations at leading order in the strong coupling constant. This models the primary high-energy interaction of interest, simulating the resultant array of particles and their properties. Subsequent hadronization of these particles and modeling of the underlying event [29, 30] are simulated using `Pythia8` [31, 32]. Definition and selection of stable and detectable particles produced in the collision is performed

using `Rivet` [33]. Neural networks are implemented using `TensorFlow` v2.4.3 interfaced with `Keras` v2.4.0 [34, 35]. SMEFT interactions are implemented in `MG5` using the `SMEFTSim` [36] package. 1M datapoints are generated at the Standard Model value of $(c_{\mathrm{HWB}},\ \tilde{c}_W) = (0, 0)$. 400k datapoints are generated in increments of 0.1 on the interval $\tilde{c}_W \in [-0.4,\ 0.4]$ with $c_{\mathrm{HWB}} = 0$, excluding the SM configuration. 200k datapoints are generated in a 2D grid with increments of 0.2 on the interval $\tilde{c}_W \in [-0.4,\ 0.4]$ and increments of 2 on the interval $c_{\mathrm{HWB}} \in [-4,\ 4]$, excluding pairs with $c_{\mathrm{HWB}} = 0$.

All objects are defined at particle level, i.e. after parton showering and hadronization (as they would appear in a particle detector). Testing our method on such a dataset demonstrates that it fulfills the key objective of this work: to effectively model a high-dimensional PDF of physically realistic observables with external parameter dependence. Since the method is not restricted to any particular experiment or domain, we do not simulate the effects of detector efficiency and resolution when generating our training data. However, we note that end-users who wish to perform (for example) parameter estimation using detector-level experimental data can accomplish this by simulating the impact of their detector when generating their own training data. We expect this to smear the PDF, but not impact the key modeling challenges identified above. We emphasize that there are no practical barriers preventing the modeling of detector-level datasets for use within a given experimental context.

2.1. EW *Zjj* event selection and observable definitions

Selection requirements and observables of interest are chosen based on the recent ATLAS measurement [22], and the ATLAS co-ordinate system [24] is used throughout with all observables defined in the laboratory reference frame.

All final state objects are required to satisfy a pseudorapidity of $|\eta| \leqslant 5$. Electrons and muons are 'dressed' [37] with photons within a cone of $\Delta R \leqslant 0.1$. Electrons are required to satisfy $p_{\mathrm{T}} \geqslant 25$ GeV and have $|\eta| < 2.47$ excluding $1.37 < |\eta| < 1.52$ where p_{T} is the momentum component transverse to the beamline. Muons are required to satisfy $p_{\mathrm{T}} \geqslant 25$ GeV and $|\eta| < 2.4$. Jets arise from collimated streams of stable particles and are clustered [38] from all final state particles excluding muons and neutrinos using the anti-k_{T} algorithm [39] within a cone of $\Delta R \leqslant 0.4$. Reconstructed jets are required to satisfy $p_{\mathrm{T}} \geqslant 30$ GeV and have a rapidity of $|y| < 4.4$. Jets are rejected if they fall within $\Delta R \leqslant 0.2$ of a selected electron, to reflect the limitations of a real detector in accurately distinguishing jets and electrons produced at small angular separations.

Events are required to have at least two selected electrons or muons, where the two leptons with the highest p_{T} are used to define the dilepton system and are required to have opposite charge. Events are also required to contain two selected jets, and the two jets with the highest p_{T} are used to define the dijet system. The following observables are calculated from the selected objects:

- m_{ll}, $p_{\mathrm{T}}^{\mathrm{ll}}$ and $|y^{\mathrm{ll}}|$ are respectively the mass, transverse momentum and absolute rapidity of the dilepton system.
- m_{jj}, $p_{\mathrm{T}}^{\mathrm{jj}}$ and $|y^{\mathrm{jj}}|$ are respectively the mass, transverse momentum and absolute rapidity of the dijet system.
- $p_{\mathrm{T}}^{\mathrm{j1}}$ and $p_{\mathrm{T}}^{\mathrm{j2}}$ are the transverse momenta of the highest and second-highest p_{T} jets.
- $\Delta\phi(j,j)$ is the angular spread of the dijet system in a plane transverse to the beamline, measured clockwise with respect to the highest rapidity jet and defined on a domain of $[-\pi,\ \pi]$.
- $|\Delta y(j,j)|$ is the absolute rapidity spread of the dijet system.
- N_{jet} is the number of selected jets, and N_{gapjet} is the number of selected jets which have a rapidity in the interval bounded by the rapidities of the two highest p_{T} jets.

Table 1 shows the intervals over which these observables are defined. Events are rejected if any observable falls outside of its interval. The total selection efficiency is estimated to be 64% using the events simulated under the SM hypothesis.

3. Method overview

Consider that we measure datapoints $x \in \mathbb{X}$ on an n-dimensional observable space $\mathbb{X} \equiv \mathbb{R}^n$. The PDF is $p(x|\theta)$, where $\theta \in \Theta$ represents the set of parameters of interest and nuisance parameters. This conditional dependence allows us to constrain a set of possible physical models according to their consistency with experimental observations.

3.1. Gaussian mixture models

We can model a conditional *one-dimensional* density $p(x|\theta)$ by simulating data for a variety of θ and fitting this with a conditional GMM. This parameterizes the density as a linear sum of Gaussian distributions according to:

Table 1. Closed intervals over which observables are selected for experiments performed on simulated EW *Zjj* data. Events are rejected if they fail any selection requirement.

Observable	Closed interval
m_{ll}	[75, 105] GeV
$p_{\mathrm{T}}^{\mathrm{ll}}$	[0, 900] GeV
y^{ll}	[0, 2.2]
m_{jj}	[150, 5000] GeV
$p_{\mathrm{T}}^{\mathrm{jj}}$	[0, 900] GeV
y^{jj}	[0, 4.4]
$p_{\mathrm{T}}^{\mathrm{j1}}$	[60, 1200] GeV
$p_{\mathrm{T}}^{\mathrm{j2}}$	[40, 1200] GeV
$\Delta\phi\,(j,j)$	$[-\pi,\pi]$
$\lvert\Delta y\,(j,j)\rvert$	[0, 8.8]
N_{jet}	[0, 5]
N_{gapjet}	[0, 2]

$$p_\phi\,(x|\theta) = \sum_{g=1}^{N_G} f_{\phi,g}\,(\theta)\cdot\mathcal{N}\left(x;\,\mu_{\phi,g}\,(\theta)\,;\,\sigma_{\phi,g}\,(\theta)\right), \tag{2}$$

where N_G labels the number of Gaussian modes; $\mathcal{N}$ is a Gaussian PDF; $f_{\phi,g}$, $\mu_{\phi,g}$ and $\sigma_{\phi,g}$ are respectively the amplitude, mean and width of the g^{th} Gaussian subject to $\sum_{g=1}^{N_G} f_{\phi,g} = 1$ and $f_{\phi,g} \geqslant 0\ \forall\ g$; ϕ label the parameters of a neural network used to capture the functional forms of $f_{\phi,g}$, $\mu_{\phi,g}$ and $\sigma_{\phi,g}$ (see e.g. [20, 21]).

We use mixture models in this work because they allow us to model arbitrarily complex positive-definite distributions which can be analytically normalized to unity and easily sampled from. This is achieved by writing the density as the linear sum of simple parametric probability distributions. They are often used to model multi-modal data [40], and are well-suited for our probability spectra which we can imagine as being composed from a series of overlapping local probability masses. Each local mass may be modeled as having a different dependence on the external parameters θ, allowing us to express how every region of the spectrum is deformed when θ is varied. In this work we use Gaussian distributions to model each local mass of density. This is because they are simple distributions (each defined by only two parameters) which are peaked in the center and smoothly vary to 0 without excessively sharp or sparse tails, ensuring continuity in the model and retaining the local nature of the probability mass. They are also easily normalized and sampled from.

However, there are several ways in which the shape of $p(x|\theta)$ may not be well-suited to a GMM:

(a) GMMs naturally model a smooth turn-off at the boundaries of a distribution, whereas the data distribution may have hard boundaries due to strict physical constraints or event pre-selection.
(b) The structural features of the PDF, and any deformations induced by variations of θ, must be smooth and wide enough to be modulated by the Gaussian modes.
(c) In order to deform the PDF *downwards*, the model must contain a Gaussian mode with finite amplitude local to the deformation, the amplitude of which can be modulated downwards without impacting the rest of the distribution[4].

Points (b) and (c) mean that a GMM which is dominated by few wide Gaussian modes will have limited ability to describe local deformations of the PDF as θ is varied. Instead, we wish to have a distribution which is described by *a spectrum of many narrow overlapping Gaussian modes* and which contains *no deformations narrower than the Gaussians themselves*. We will now show that these conditions may be achieved by transforming the data and using a suitable network architecture to model $f_{\phi,g}$, $\mu_{\phi,g}$ and $\sigma_{\phi,g}$. We find that this method resolves the failure conditions listed above in the experiments presented.

[4] A density model must be positive definite everywhere. For a GMM, we enforce this by only allowing positive amplitudes for the Gaussian modes. To deform the PDF upwards in some local region, we can add a new Gaussian mode with positive amplitude. However, a downwards deformation cannot be similarly accounted for by adding a new Gaussian mode with negative amplitude, as this is not allowed. This can only be described if the nominal model already contained a narrow Gaussian mode with positive amplitude local to the deformation. In this case we capture the downwards deformation by modulating the amplitude downwards, from positive to less-positive.

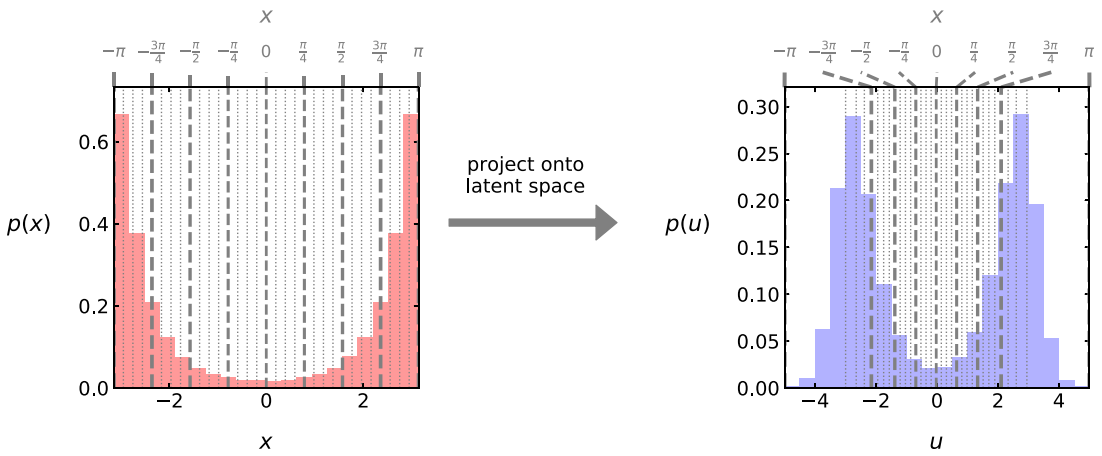


Figure 1. Left: probability density $p(x)$ with $x = \Delta\phi(j,j)$, evaluated using `MG5` events assuming $c_{\mathrm{HWB}} = \tilde{c}_W = 0$. Right: probability density after projecting onto the the latent space using the method described in figure 2.

3.2. Modeling a single observable

Datapoints are projected by a function $h : x \mapsto u \in \mathbb{U}$ onto a latent space $\mathbb{U} \equiv \mathbb{R}^n$. The properties of the projection may be tuned to optimize the performance of a GMM describing the density $p_\phi(u|\theta)$. We will now explore this idea using our EW *Zjj* example.

Consider the case where $x = \Delta\phi(j,j)$ is the only observable. Figure 1 (left) shows the probability density $p(x)$ for the SM case of $c_{\mathrm{HWB}} = \tilde{c}_W = 0$. This plot is obtained by histogramming the datapoints simulated using `MG5`. We note that this distribution has hard physical boundaries at $[-\pi, \pi]$ which a GMM would be unable to model. Figure 1 (right) shows the probability density of the same datapoints after projecting x onto the latent space. This distribution is designed to be well described by a series of overlapping narrow Gaussian modes. We will now describe how this projection function $h(x)$ was derived, then train a GMM to model this spectrum for a variety of $\tilde{c}_W$.

To derive $h(x)$, we first construct a response curve $Q_x(x)$ between the physical boundaries of x. This is written as:

$$Q_x(x) = (1-f)\cdot D_x(x) + f\cdot L_x(x), \tag{3}$$

where $D_x(x)$ is the cumulative distribution function of the data simulated at the SM and $L_x(x)$ is a linear function. The hyperparameter f is tuned to ensure that wide regions in $\mathbb{X}$ are not collapsed onto narrow regions in $\mathbb{U}$, whilst also providing a smooth turn-off at the boundaries of the distribution. This function is shown as the solid black line in figure 2 (left). We then construct a response curve $Q_u(u)$ over the latent space, shown as the solid blue line in figure 2 (middle), defined as the cumulative distribution function of a target function $\tilde{q}_u(u)$ given by:

$$\tilde{q}_u(u) = \frac{1}{1+\exp[\alpha(u-\beta)-\gamma]}\cdot\frac{1}{1+\exp[-\alpha(u+\beta)-\gamma]}. \tag{4}$$

This function, shown in figure 2 (right) using values of $(\alpha,\beta,\gamma) = (4,3,1)$, is heuristically designed to be flat in the centre and smooth at the edges. This encourages the optimal GMM description to contain many narrow overlapping Gaussian modes. We note that it may seem natural to choose a Gaussian distribution for $\tilde{q}_u(u)$ (see e.g. [9]), however this will often result in a GMM which is dominated by a single wide Gaussian mode, violating our target behaviour. The mapping function between $\mathbb{X}$ and $\mathbb{U}$ is defined as $h(x) = Q_u^{-1}(Q_x(x))$, and its derivation is shown visually as the green dotted line connecting the points x^* and u^* in figure 2 (left and middle).

We compute $Q_u(u)$ as a piecewise-linear function over the interval $u \in [-5, 5]$. Whilst the domain of u could be extended arbitrarily far so that all sampled points $u^* \in \mathbb{U}$ are mapped onto the physically allowed domain of $\mathbb{X}$, we found that limiting the domain improved numerical stability in our experiments by avoiding dilute tails in the latent distribution.

We now apply the projection function h to all our datasets with nonzero values of $\tilde{c}_W$[5]. It is crucial that h are derived using data at a single point in parameter space (here $\tilde{c}_W = 0$) and applied to the data at all values

[5] For simplicity, in this section we only consider variations of $\tilde{c}_W$ and fix $c_{\mathrm{HWB}} = 0$ throughout.

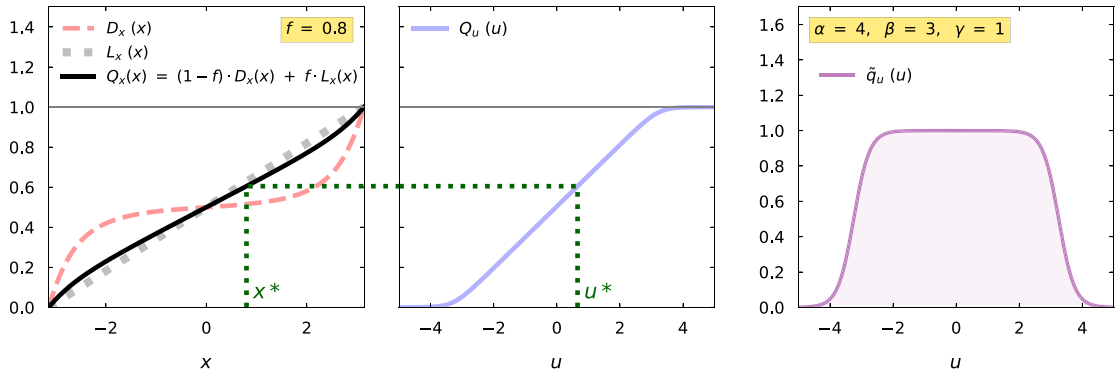


Figure 2. Left: response curve over the data space, $Q_x\,(x)$, derived as the linear sum of $D_x\,(x)$ and $L_x\,(x)$. Middle: response curve over the latent space, $Q_u\,(u)$, derived as the cumulative distribution function of $\tilde{q}_u\,(u)$. Right: heuristic function $\tilde{q}_u\,(u)$. The green dotted line connecting $Q_x\,(x)$ with $Q_u\,(u)$ visually represents how a datapoint at x^* is transformed onto u^* in the latent space.

of $\tilde{c}_W$. As $\tilde{c}_W$ is varied, the probability density $p\,(u|\tilde{c}_W)$ is deformed. This is modeled as $p_\phi\,(u|\tilde{c}_W)$ where the neural network parameters ϕ are trained using maximum likelihood estimation evaluated over the simulated training data for all $\tilde{c}_W$, i.e.:

$$\mathbb{V}\,(\phi) = \frac{1}{\sum w}\cdot\sum_{\tilde{c}_W,x,w} w\cdot\log p_\phi\,(h\,(x)\,|\tilde{c}_W)\,, \tag{5}$$

$$\phi \to \underset{\phi}{\text{argmax}}\ \ \mathbb{V}\,(\phi)\,, \tag{6}$$

where w label Monte Carlo event weights, used to account for how integration of probabilities is handled within a particular simulation package [29, 30], if applicable.

We train a GMM with $N_G = 30$ individual modes to describe the probability density. Figure 3 (top row) compares the training data and post-fit model $p_\phi\,(u|\tilde{c}_W)$ at values of $\tilde{c}_W = \{-0.4,\ 0,\ 0.4\}$. Thin coloured lines show the decomposition into individual Gaussian modes. As $\tilde{c}_W$ is varied, we see that deformations in the spectrum are captured by modulating the amplitudes, positions and widths of the narrow Gaussian modes.

Figure 3 (middle row) shows the ratio between the training data and the model PDF, offset to 0 so we study the residual difference between the two. This demonstrates that systematic mis-modelling is below 5% except in the sparsely populated tails of the distribution for all three values of $\tilde{c}_W$. The dark shaded band around the data shows the Poisson estimate of the statistical uncertainty. The thickness of this band is comparable with the residual difference between the data and the model, suggesting that this residual is mostly dominated by random fluctuations in the data.

Figure 3 (bottom row) shows the ratio between $p_\phi\,(u|\tilde{c}_W)$ and $p_\phi\,(u|0)$, the model PDF evaluated at $\tilde{c}_W = 0$, once again offset to 0 so we study the residual difference between the two. This quantifies how the shape of the distribution is deformed when translating across $\tilde{c}_W$. Training data are also shown, demonstrating that the model has captured how the spectrum is deformed as $\tilde{c}_W$ is varied.

3.3. Extending to multiple observables

When modeling d observables on the latent space, we write an auto-regressive probability density:

$$p_\phi\,(u|\theta) = \prod_{i=1}^{d} p_{\phi,i}\,(u_i|u_{<i},\theta)\,, \tag{7}$$

where i label observables and $u_{<i}$ is the list of all prior latent observables. The conditional probability density for each u_i is modeled using a GMM parameterized by a neural network with parameters ϕ according to:

$$p_{\phi,i}\,(u_i|u_{<i},\theta) = \sum_{g=1}^{N_G} f_{\phi,g,i}\,(u_{<i},\theta)\cdot\mathcal{N}\left(u_i;\ \mu_{\phi,g,i}\,(u_{<i},\theta);\ \sigma_{\phi,g,i}\,(u_{<i},\theta)\right), \tag{8}$$

where $f_{\phi,g,i}$, $\mu_{\phi,g,i}$ and $\sigma_{\phi,g,i}$ are respectively the amplitude, mean and width of the g^{th} Gaussian for observable index i. By including $u_{<i}$ as input to the network, it now captures the dependence on *both* external parameters *and* preceding observables. This means that high-dimensional observable correlations may be described by the model.

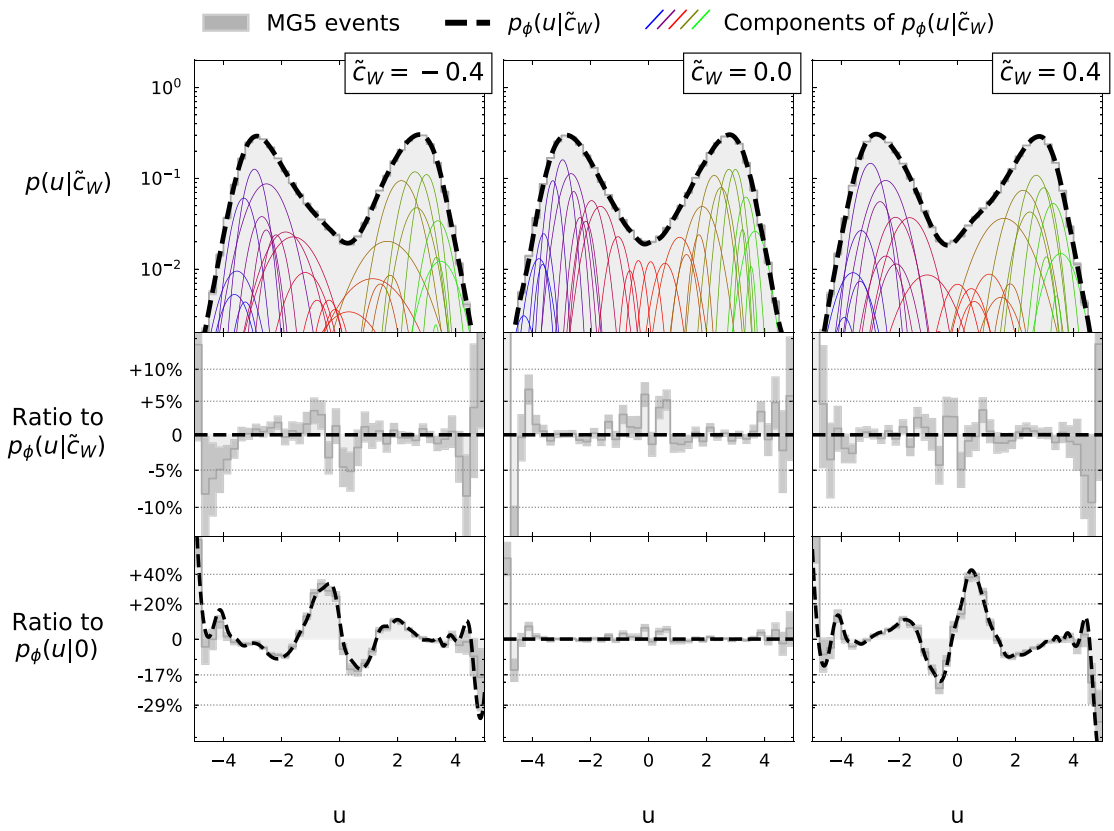


Figure 3. Gaussian mixture model over the latent space (black dashed line) for the one-dimensional example of $x = \Delta\phi\,(j,j)$. We show the comparison with `MG5` events (grey line) when $\tilde{c}_W = -0.4$ (left), $\tilde{c}_W = 0$ (middle) and $\tilde{c}_W = 0.4$ (right), with $c_{\mathrm{HWB}} = 0$ throughout. Dark shaded regions show the estimated statistical uncertainty due to the finite number of `MG5` events. Lower panels show the ratio with respect to $p_\phi\,(u|\tilde{c}_W)$ (middle row) and $p_\phi\,(u|0)$ (bottom row).

3.4. Neural network architecture

Figure 4 shows a schematic diagram of the neural network architecture used to model the GMM for latent observable $u_i \in \left[u_i^{\min}, u_i^{\max}\right]$. Fully connected layers at depth l are shown in grey and labelled *Dense*, with a number of neurons equal to N_l as specified and an activation function shown in parentheses. These are either *linear*, equivalent to applying no activation function, or *LeakyReLU* [41] with a negative gradient of 0.2 defined for input x according to:

$$\mathrm{LeakyReLU}\,(x) = \begin{cases} x & \text{if } x \geqslant 0 \\ 0.2 \cdot x & \text{if } x < 0. \end{cases} \tag{9}$$

Inputs θ and $u_{<i}$ of lengths N_θ and N_u respectively are compressed onto the interval $[-2,2]$ and fed into initial layers of size N_1 and N_2. The configurable constants $\{A_1, A_2, B_1, B_2\}$ determine the width of these layers. The outputs are concatenated and fed into a sequence of C layers of width $N_1 + N_2$. The constant C determines the ultimate depth of the network. The outputs are then fed into three separate channels, which will separately assign the Gaussian amplitudes $\vec{f}_i$, means $\vec{\mu}_i$ and widths $\vec{\sigma}_i$. In each channel, activations x pass through two further dense layers of size $D \cdot N_G$ and N_G, creating three vectors of length N_G. These are scaled by factors of s_f, s_μ and s_σ. These scale factors determine the size of the initial fluctuations around the nominal initial values of $\vec{f}_i$, $\vec{\mu}_i$ and $\vec{\sigma}_i$ which are assigned as follows.

In the $\vec{f}_i$ channel, activations are passed through a Softmax function to ensure the Gaussian amplitudes are positive definite and sum to unity. If $|s_f| \ll 1$ then all components of $\vec{f}_i$ are initially approximately equal. In the $\vec{\mu}_i$ channel, a constant is added to the g^{th} vector component such that the Gaussian modes are initially linearly spaced between $u_i^{\min}$ and $u_i^{\max}$ subject to fluctuations. In the $\vec{\sigma}_i$ channel, Gaussian widths are initialized to fluctuate around a value of f_σ units of $\frac{u_i^{\max} - u_i^{\min}}{N_G}$. The configurable constant f_σ therefore determines how many standard deviations of overlap exist between the initial Gaussian modes. Finally, a constant of $\epsilon = 10^{-4}$ is added to prevent the evaluation of Gaussian modes with zero width. We note that these transformations impact the gradients of the loss function with respect to the three different channels,

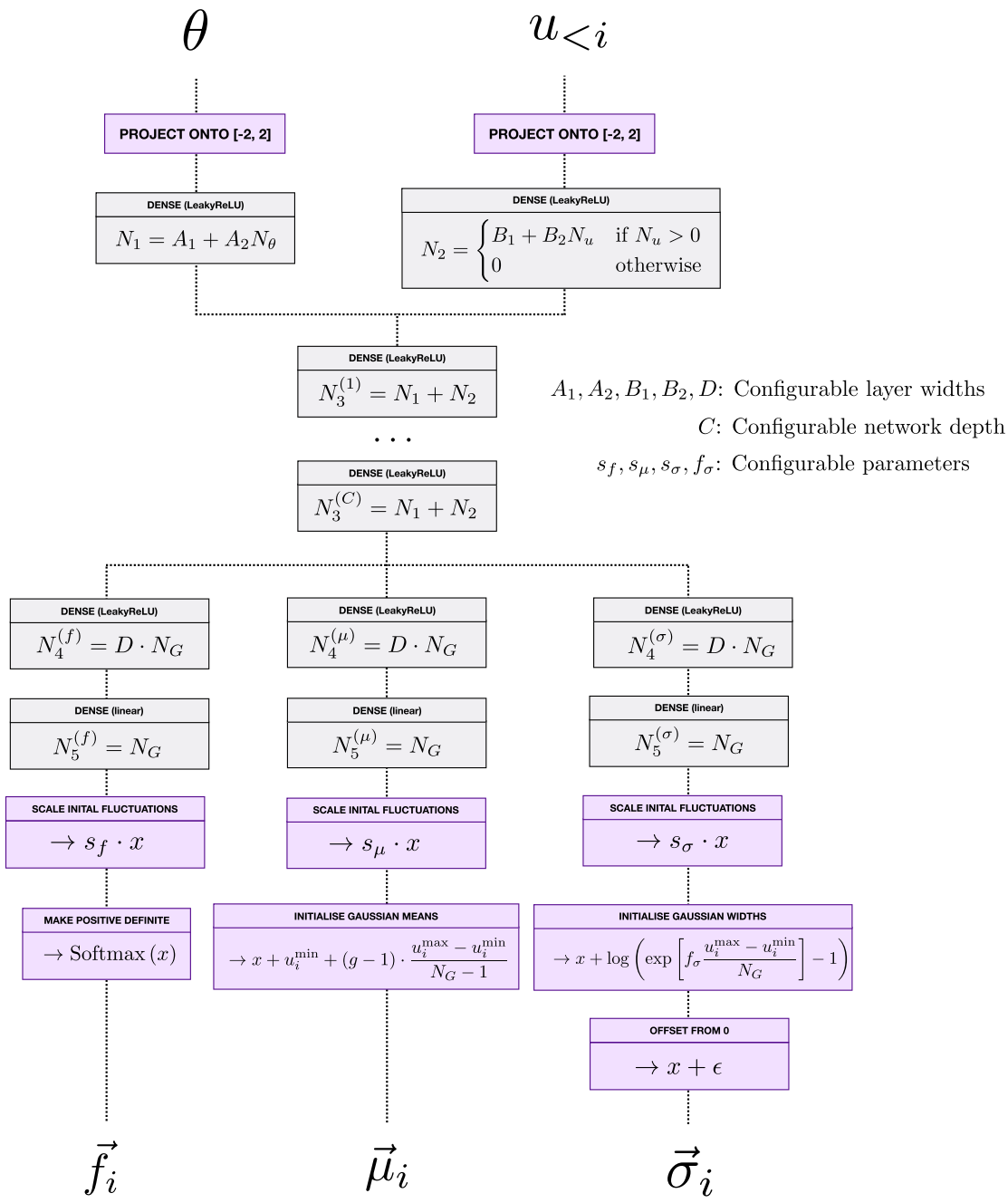


Figure 4. Structure of the neural network implemented for observable $u_i \in \left[u_i^{\min}, u_i^{\max}\right]$. Configurable parameters $\{A_1, A_2, B_1, B_2, D\}$ determine the width of the fully connected Dense layers, which have nodes equal to the N provided, and C determines the number of intermediate Dense layers. Configurable constants $\{s_f, s_\mu, s_\sigma\}$ determine the scale of initial perturbations, while f_σ configures the initial Gaussian widths.

leading to different learning rates for the amplitudes, means and widths respectively. This likely impacts the post-fit model, and future optimization may be achieved by controlling the balance of these gradients to preferentially enhance model updates in one channel.

The resulting network contains $\mathcal{O}\left((N_1+N_2)^{2C} + (N_1+N_2+N_G)\,DN_G\right)$ trainable parameters. Model optimization is performed using the `Adam` [42] algorithm with a learning rate of λ_{lr}. An adaptive learning rate is used, such that λ_{lr} is multiplied by a factor of $\lambda_{\mathrm{lr}}^{\text{update factor}} < 1$ if the training loss does not improve for $\lambda_{\mathrm{lr}}^{\text{patience}}$ epochs. This mitigates underfitting when the initial λ_{lr} is large. Network biases are initialized to zero and weights are drawn randomly from a uniform distribution over the interval $\pm 10/(3\sqrt{N_{\mathrm{in}}})$ where N_{in} is the number of input neurons. This mitigates vanishing/exploding activations and gradients in the initial state.

3.5. Impact of transforming the likelihood

The function h performs a monotonic one-dimensional change of variables between x and u. The probability density $p_u(u)$ over the latent space may therefore be transformed into a probability density over the original data space $p_x(x)$ according to:

$$p_x(x) = p_u(h(x)) \cdot \left| \frac{dh(x)}{dx} \right|, \tag{10}$$

where $h(x)$ is evaluated using a piecewise linear function calculated from the training data, and so $\left|\frac{dh(x)}{dx}\right|$ is a step function over x. Whilst it leads to a tractable density over x, equation (10) contains no dependence on θ. This means that statistical inference is equivalent when performed on $\mathbb{U}$ and $\mathbb{X}$. Applying such a transformation is therefore not necessary, and we will always perform inference using observations in the latent representation unless stated otherwise.

We also note that the transformation $h(x)$ must preserve the total probability contained within a span, i.e.

$$\int_{x_1}^{x_2} p_x(x)\,\mathrm{d}x = \int_{h(x_1)}^{h(x_2)} p_u(u)\,\mathrm{d}u, \tag{11}$$

and so we can integrate the probability contained within $[x_1, x_2]$ simply by transforming x_1 and x_2 and performing the integration over the latent space. However, this integration may only be performed analytically when data are one-dimensional.

We do not perform a rotation when transforming between x and u. This secures three desirable features: it ensures a diagonal Jacobian matrix, it retains an easily understood relationship between each component of x and u, and it mitigates potential concerns about loss of generalization [43].

3.6. Complexity of likelihood evaluation

Consider that we wish to model d observables, using d neural networks each containing L hidden layers and W neurons per layer. Assuming that $d \ll W$ and $N_G \ll LW$, the calculation of $p(u|\theta)$ has a complexity of $\mathcal{O}\left(dLW^2\right)$. However, each of the d conditional probability densities may be computed in parallel, resulting in $\mathcal{O}\left(LW^2\right)$ complexity. This may be further accelerated up to a limit of $\mathcal{O}(L)$ by using a GPU for efficient matrix multiplication. Since $u_{<i}$ are used as input to the networks for all $i > 0$, network outputs must be computed separately for every datapoint except in the case of the first observable u_0, for which a single pass through the network can be used to provide the Gaussian parameters needed to evaluate every datapoint.

3.7. Complexity of generative sampling

We have noted that the density model may be sampled, allowing it to be used as a generative model for event simulation. We achieve this by randomly drawing $u_0^* \sim p_{\phi,0}(u_0|\theta)$, $u_1^* \sim p_{\phi,1}(u_1|u_0^*, \theta)$ and so on until a datapoint u^* in d dimensions is constructed. This may be transformed back onto data space using $x^* = h^{-1}(u^*)$.

Since this process is sequential in the latent observables, they may not be simulated in parallel. As with likelihood evaluation, the complexity of sampling is $\mathcal{O}\left(dLW^2\right)$. This may be accelerated up to a limit of $\mathcal{O}(dL)$ using a GPU. Since $p_{\phi,0}(u_0|\theta)$ contains no dependence on other observables, many u_0^* may be sampled using a single evaluation of the network. However, sampling u_i^* for $i > 0$ requires the network to be evaluated for every datapoint.

3.8. Modelling of systematic uncertainties

In this work, we focus on the expressive power of the model and do not consider the impact of systematic uncertainties. However, it is crucial that such uncertainties are accounted for when performing a statistical interpretation on a measured dataset. Here we briefly discuss how this may be done, whilst noting the limitations. We note that cross-section uncertainties may be trivially accounted for, since they do not impact the distribution of events throughout phase space.

We may separate modelling uncertainties into three categories. The first category are uncertainties associated with the simulation of training data which are parameterizable in terms of a nuisance parameter θ_{NP}. These may be accounted for either by including θ_{NP} within the vector θ input to the network, or by training a separate model $r(u, \theta_{\mathrm{NP}}) = p(u|\theta_{\mathrm{NP}}) / p\left(u|\theta_{\mathrm{NP}}^{\mathrm{ref}}\right)$ for some fixed reference $\theta_{\mathrm{NP}}^{\mathrm{ref}}$ and writing:

$$p(u|\theta_{\mathrm{NP}}) = p\left(u|\theta_{\mathrm{NP}}^{\mathrm{ref}}\right) \cdot r(u, \theta_{\mathrm{NP}}). \tag{12}$$

The second category are non-parameterizable uncertainties associated with the simulation of training data. In high energy physics, these may account for poorly understood differences between the simulated data and control measurements. In a binned one-dimensional analysis, they may be mitigated by performing auxiliary observations which are uncorrelated with the observable being modelled and 'transferring' the data-driven constraint on a bin-by-bin basis. Residual uncertainties may then be parameterized according to systematic variations of this transfer procedure. It is challenging to extend such techniques to our model because we must cover possible mismodelling of the high-dimensional observable correlations.

The third category are uncertainties associated with the density model. These biases are caused by the inductive bias of the model as well as under- or over-fitting. Over-fitting may be mitigated using techniques such as regularization, dropout and early stopping, and by limiting model complexity. Under-fitting may be studied by sampling the density model for all simulated θ and showing that the marginal projections are compatible with the simulated data. Quantifying and parameterizing the remaining mismodelling is once again challenging, and we leave this for future work.

We consider overcoming these challenges to be one of the main hurdles facing the use of high-dimensional density models in high energy physics.

3.9. Model optimization

A strength of the proposed method is that there are many ways in which modelling may be improved if under-fitting is observed. These strategies include:

(a) Increase the model capacity by using more complicated networks or larger N_G.
(b) Tune the parameters s_f, s_μ and s_σ, which modulate the size of the initial state perturbations of the Gaussian amplitudes, positions and widths as described in figure 4, to balance the stability of the initial model with the size of perturbations which provide gradients for the learning process.
(c) Tune f_σ to configure the initial width of the Gaussian modes. Whilst narrow modes tend to describe local features of the data, fulfilling the objectives of our model design, training data do not provide significant learning potential for Gaussian modes several standard deviations away. We find that successful training occurs when the value of f_σ balances these effects.
(d) Tune the hyperparameter f or the functional form of $\tilde{q}_u$ to create a latent distribution which is well described by a mixture of narrow Gaussians.
(e) Alter the ordering of the observables, since $p(B|A)$ may be more easily described than $p(A|B)$ for two latent observables A and B.
(f) Alter the training procedure to improve convergence towards likelihood maxima.
(g) Rotate observables onto the eigenvectors of their covariance, reducing strong correlations in the data.

These opportunities for tuning improve the chance of finding a model which captures the salient features of the dataset provided.

4. EW Zjj with 12 observables and no external parameter dependence

In this section we create a density model to describe 12 observables with no external parameter dependence. This demonstrates that the method can learn a joint probability density over a high-dimensional dataset of physically realistic observables. Table 2 shows the observable ordering as well as the f-values used to configure the projection onto the latent space.

We include the two discrete observables N_{gapjet} and N_{jet} in the model. This demonstrates that there are no barriers to modelling continuous and discrete observables at the same time. A discrete observable taking integer values on the inclusive interval $[u_i^{\min}, u_i^{\max}]$ is modelled using a neural network which outputs a categorical probability distribution of length $N_p = 1 + u_i^{\max} - u_i^{\min}$. Inputs θ and $u_{<i}$ are projected onto the interval $[-2, 2]$ and passed through dense layers of size N_1 and N_2 respectively. These are followed by two fully connected layers of size 300 and 200, and an output layer of size N_p. All intermediate layers use a LeakyReLU activation function with a negative gradient of 0.2. The output layer uses a SoftMax activation function to ensure that outputs represent a normalized multinomial probability distribution. The network is trained using a cross entropy loss function and the same training scheme as used to model continuous observables.

Table 3 shows the constants used to configure the remaining neural networks and their training. The networks contain between 27k and 304k trainable parameters. This reflects a degree of over-parameterization of the model, since the number of parameters is the same order of magnitude as the number of training samples. We note that any resultant over-training is mitigated by the use of a GMM which naturally smooths each conditional PDF in the auto-regressive chain. Each network is initially trained for up to 400 epochs,

Table 2. Indices in which observables are ordered when constructing a density model describing EW *Zjj* data with 12 observables and no external parameter dependence. The f values used to project continuous real-valued observables onto the latent space are shown. Indices start from 0.

Observable order: name [projection constant f]								
0:	m_{jj}	$[f=0.2]$	1:	$p_{\mathrm{T}}^{\mathrm{jj}}$	$[f=0.2]$	2:	$\lvert y^{\mathrm{jj}}\rvert$	$[f=0.2]$
3:	$\Delta\phi(j,j)$	$[f=0.8]$	4:	$\Delta y(j,j)$	$[f=0.8]$	5:	$p_{\mathrm{T}}^{\mathrm{j1}}$	$[f=0.2]$
6:	$p_{\mathrm{T}}^{\mathrm{j2}}$	$[f=0.2]$	7:	N_{gapjet}		8:	N_{jet}	
9:	m_{ll}	$[f=0.8]$	10:	$p_{\mathrm{T}}^{\mathrm{ll}}$	$[f=0.2]$	11:	$\lvert y^{\mathrm{ll}}\rvert$	$[f=0.8]$

Table 3. Constants used to construct and train a density model describing EW *Zjj* data with 12 observables and no external parameter dependence.

$N_G=20$	$A_1=200$	$A_2=0$	$B_1=200$	$B_2=50$
$C=3$	$D=3$	$s_f=0.01$	$s_\mu=0.01$	$s_\sigma=0.01$
$f_\sigma=0.5$	batch size $=1$k	$\lambda_{\mathrm{lr}}=0.001$	$\lambda_{\mathrm{lr}}^{\text{update factor}}=0.5$	$\lambda_{\mathrm{lr}}^{\text{patience}}=3$

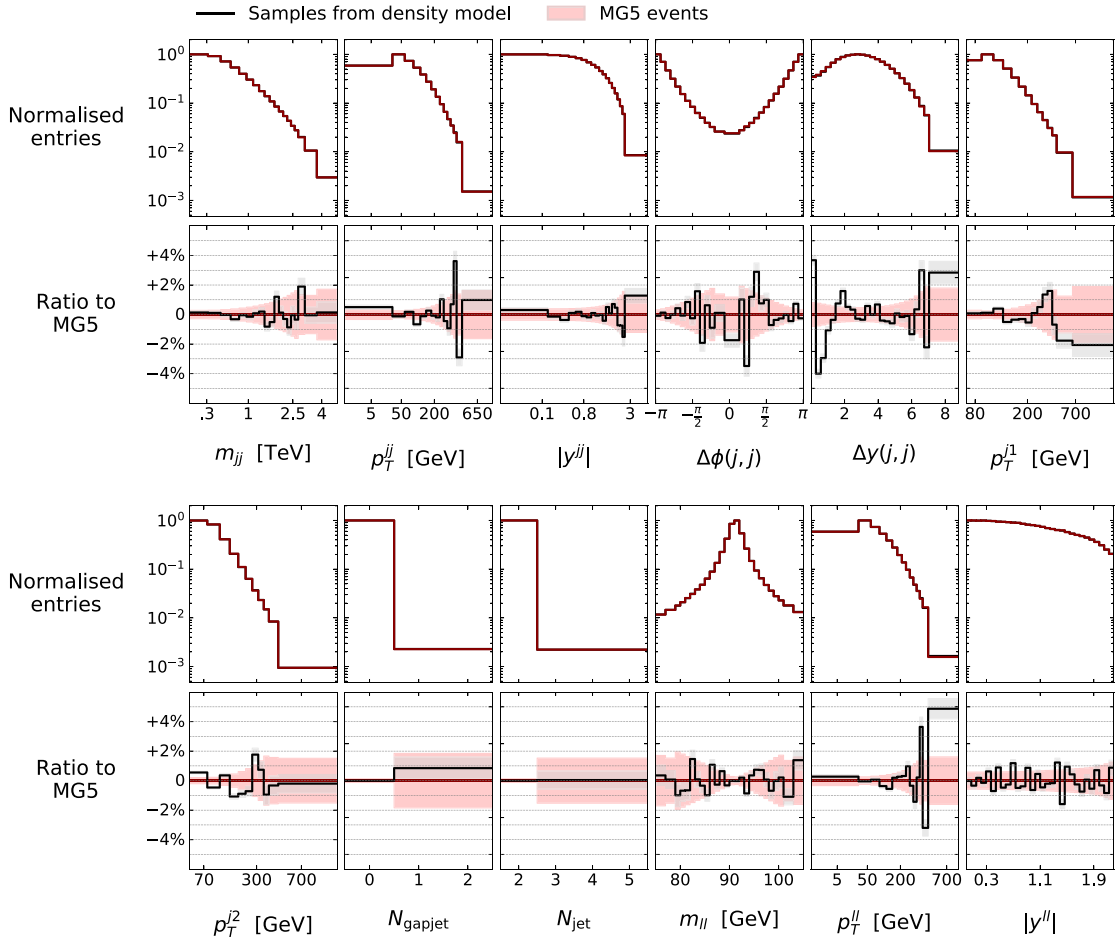


Figure 5. 1D marginal distributions comparing events simulated with MG5 (red) with those sampled from a GMM trained on a latent space (black) with no external parameter dependence. Note that the two spectra are not statistically independent, since the density model was trained using the `MG5` events.

stopping early if the loss function does not improve over a period of 12 epochs. We observe that $\mathcal{O}(10^{-4})$ relative updates to the log-likelihood are important, since they may lead to %-level improvements in the description of the tails. Training should therefore not be halted until a true plateau in the loss function is obtained.

The model is trained using the 640k selected `MG5` events generated assuming the SM hypothesis. To evaluate its performance, we randomly sample 4M datapoints from the model and compare the 1D and 2D marginal distributions with those of the training data. This large number is chosen to reduce fluctuations due to sampling variance.

Figure 5 presents the 1D marginal distributions. For each observable, an upper panel presents the absolute spectrum in units normalized such that the highest bin takes a value of 1, and a lower panel shows a ratio taken with respect to the MG5 events. MG5 events are shown in red and compared with events sampled from the density model, shown in black. Shaded areas present Poisson estimates of the statistical uncertainty arising from finite sample size. We observe that all spectra are well described within a systematic precision of $\pm 5\%$, with many spectra achieving precision similar to the statistical variance of the training data. We note that fewer bins than the expected $\mathcal{O}(32\%)$ lie outside of the uncertainty bands, indicating that the model may be over-trained. Since this work is intended as a proof-of-principle for the method, we make no further attempt to mitigate over-training, whilst noting that this will be important for future applications.

Figure 6 presents the 2D marginal distributions for all pairs of observables as measured using the MG5 events. This demonstrates that complex correlations exist between all observables. Figure 7 presents the 2D marginal distributions using the samples from the density model. Comparing figures 6 and 7 shows that the model has captured the high-dimensional correlations between all pairs of observables. Bins are coloured white if no entries exist, and black if a small number of entries are observed. We note that several fully-white regions of figure 6 are black in figure 7, suggesting that the density model may predict a small non-zero probability in regions of phase space which are unpopulated when simulating from-first-principles, as is the case with MG5.

If the modelled density in such regions is sufficiently small, we expect that this artifact should have minimal impact on inference tasks. This is because any overflow of density into physically-disallowed regions of phase space will mainly cause a small under-estimate of the normalization in physically-allowed regions, where all observed events must necessarily exist. Furthermore, this normalization shift may cancel when considering likelihood ratios. A greater problem may occur when using the density model for event sampling, since events may be generated in the physically-disallowed regions. Whilst not solving this problem at this time, we foresee potential for mitigation using two methods:

(a) Use transformed observables which enforce easily-parameterized boundaries. For example, modelling the pair of observables $\{p_T^{j1}, p_T^{j2}\}$ risks predicting a non-zero density in the unphysical region $p_T^{j2} > p_T^{j1}$. Instead we can model $\{p_T^{j1\prime}, p_T^{j2}\}$ where $p_T^{j1\prime} = p_T^{j1} - p_T^{j2}$ is required to satisfy $p_T^{j1\prime} \geqslant 0$, preventing such unphysical behaviour. A drawback is that we cannot enforce the original boundary limits of p_T^{j1}, because these must now be defined relative to the value of p_T^{j2}. Furthermore, most physical boundary conditions may not be easily enforced by such a transformation, either because they are too complicated or because the user is not aware of them.

(b) In high energy physics, one can model the components of object four-vectors and reconstruct observables accordingly. This naturally imposes many physical constraints, although not all, and once again we cannot enforce simple boundary conditions for high-level observables.

With these caveats, figures 6 and 7 demonstrate that the 2D projections of events sampled using density model are qualitatively very similar to the ground truth events throughout most of the space. The comparison is quantified in figure 8. This shows the pull on the ratio of these histograms, defined as:

$$\text{Pull on } \frac{p_{\text{model}}}{p_{\text{MG5}}} = \frac{\frac{p_{\text{model}} - p_{\text{MG5}}}{p_{\text{MG5}}}}{\Delta\left(\frac{p_{\text{model}}}{p_{\text{MG5}}}\right)}, \tag{13}$$

where p_{model} and p_{MG5} are the densities estimated using events sampled from the density model and MG5 respectively, and $\Delta\left(\frac{p_{\text{model}}}{p_{\text{MG5}}}\right)$ represents the estimated statistical uncertainty on the ratio between them. This dominated by the estimated statistical uncertainty on p_{MG5}. The pull can be interpreted as 'the number of standard deviations by which the ratio differs from unity.' It therefore shows the sign and statistical significance of the difference between the two distributions.

If the density model represents an unbiased fit to the MG5 events then we expect $O(68\%)$ of bins to fall inside the interval $[-1, +1]$. Due to random fluctuations in the event sampling, we still expect $O(32\%)$ to fall outside of this interval by chance, even when the density model is equal to the ground truth. Extending this idea, we expect $O(5\%)$ of bins to fall outside the interval $[-2, +2]$ and $O(0.3\%)$ outside $[-3, +3]$ due to random fluctuations. If the density model is over-trained, we expect to observe an excess of bins with small pulls. Where mis-modeling occurs, we expect to observe a systematic trend of large pulls.

This allows us to study the agreement between the density model and training events in the following way. All bins with pulls less than 1 in magnitude are shown in green in figure 8. These are bins where the agreement between the density model and training data is better than the estimated statistical uncertainty. Bins with pulls above $+1$ (below -1) are shown in increasingly dark shades of red (blue). Since we expect

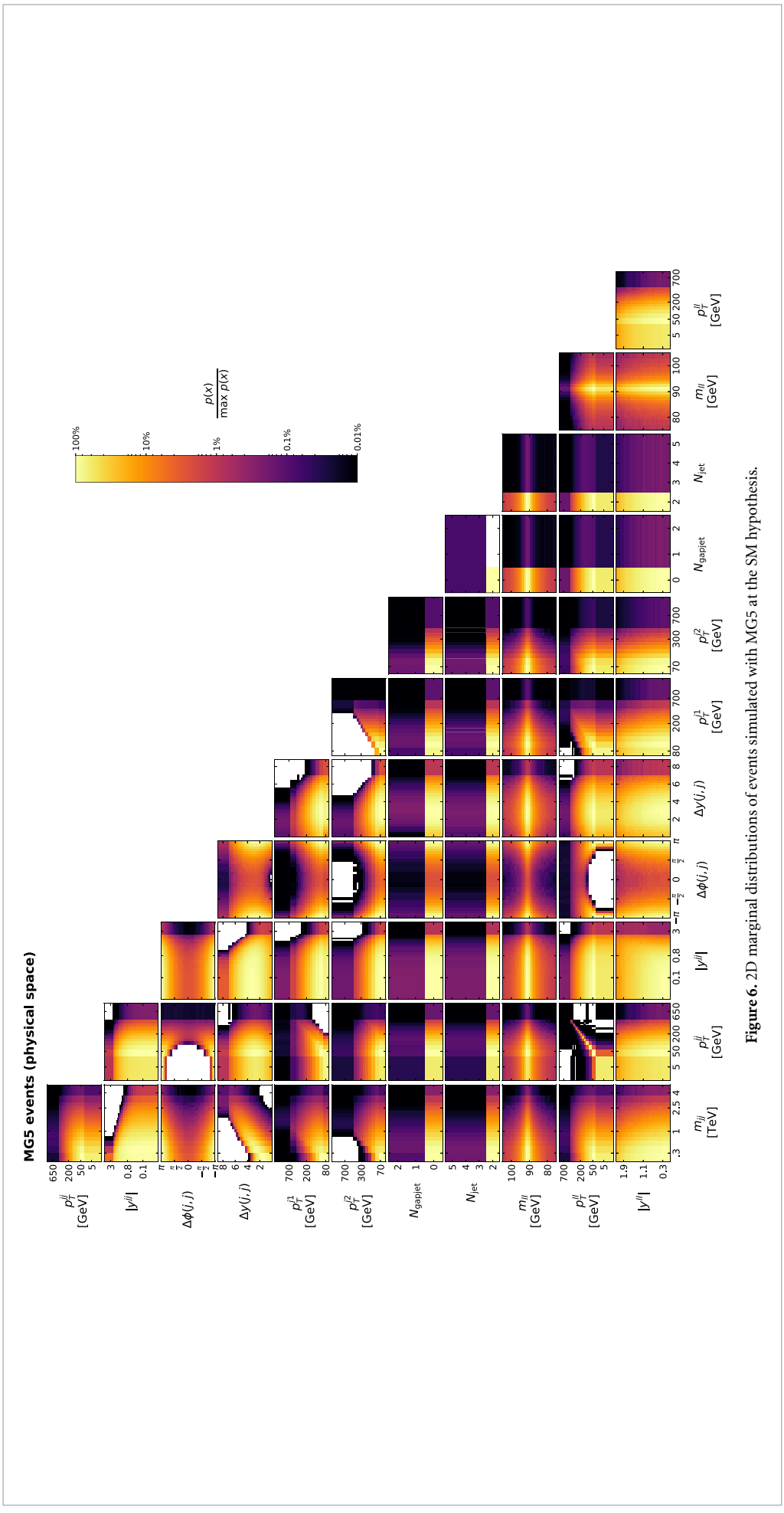


Figure 6. 2D marginal distributions of events simulated with MG5 at the SM hypothesis.

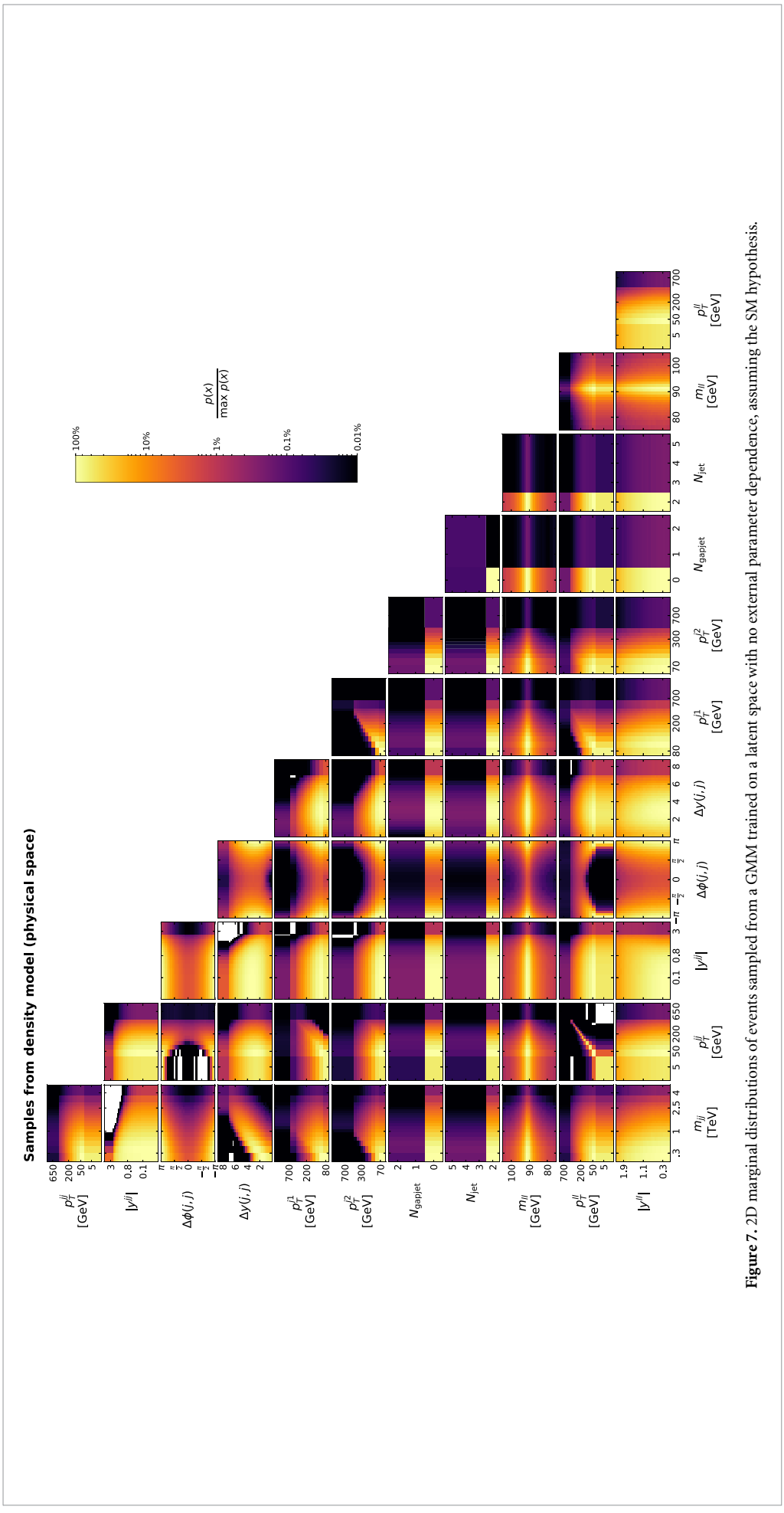


Figure 7. 2D marginal distributions of events sampled from a GMM trained on a latent space with no external parameter dependence, assuming the SM hypothesis.

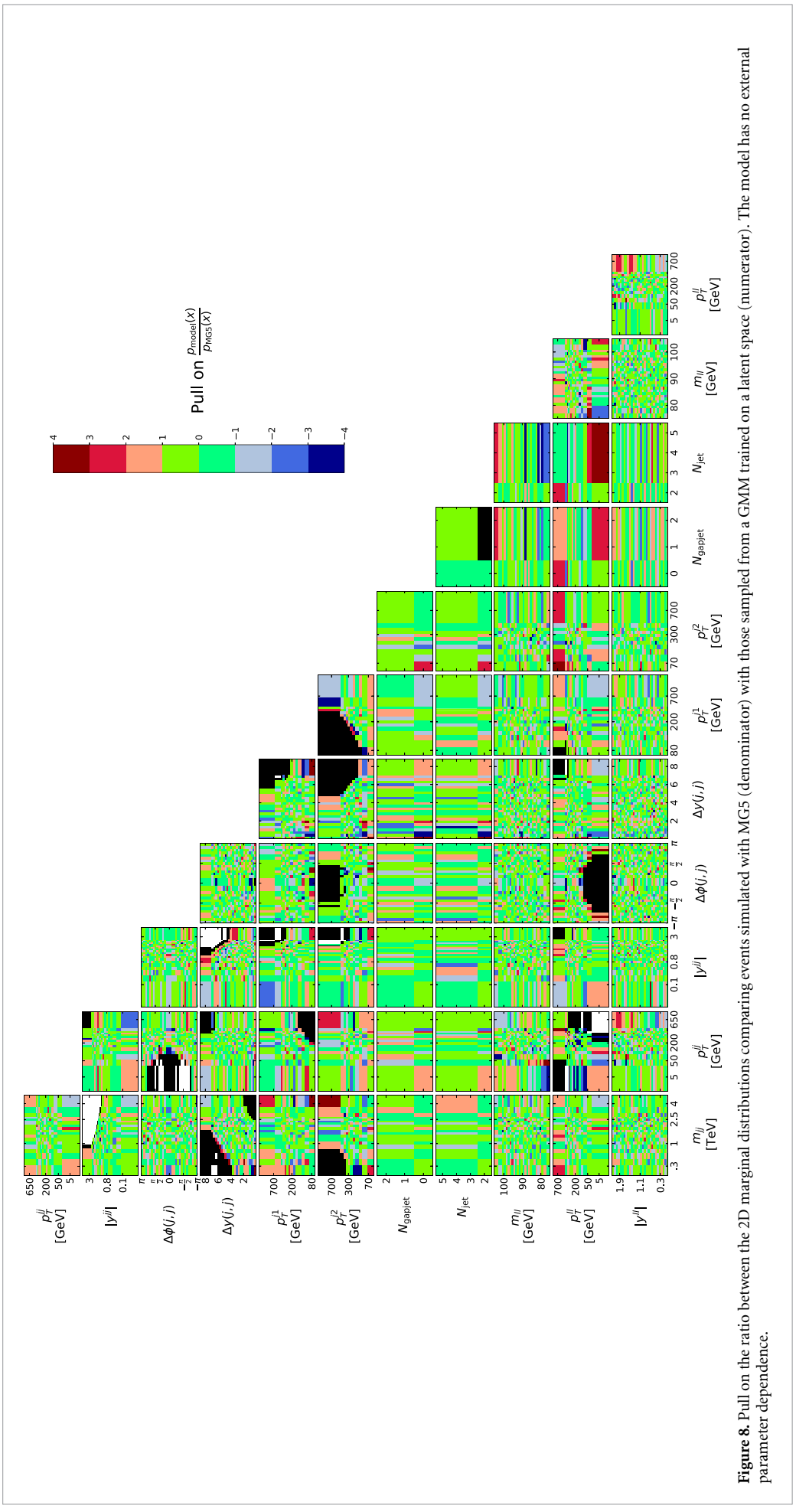


Figure 8. Pull on the ratio between the 2D marginal distributions comparing events simulated with MG5 (denominator) with those sampled from a GMM trained on a latent space (numerator). The model has no external parameter dependence.

Table 4. Frequencies of pulls observed in figure 8, ignoring black and white bins which contain zero sampled events.

Pull range	Observed frequency	Expected frequency
Below −4	$\ll 0.003\%$	$\mathcal{O}(0.003\%)$
−4 to −3	0.58%	$\mathcal{O}(0.13\%)$
−3 to −2	3.4%	$\mathcal{O}(2.1\%)$
−2 to −1	13.4%	$\mathcal{O}(13.6\%)$
−1 to 0	31.6%	$\mathcal{O}(34.1\%)$
0 to +1	35.0%	$\mathcal{O}(34.1\%)$
+1 to +2	13.7%	$\mathcal{O}(13.6\%)$
+2 to +3	1.8%	$\mathcal{O}(2.1\%)$
+3 to +4	0.17%	$\mathcal{O}(0.13\%)$
Above +4	$\ll 0.003\%$	$\mathcal{O}(0.003\%)$

that $O(32\%)$ of bins will be coloured red or blue, the presence of these bins does not indicate mis-modeling. Instead we search for the following signatures:

- Adjacent dark red or blue bins indicate that the difference between the density model and training data is unlikely to occur by chance. These regions are likely mis-modeled.
- Multiple adjacent bins which are all coloured red or blue suggest an effect of systematic mis-modeling rather than statistical fluctuation.
- More bins shaded in red or blue than expected, indicating that more bins than expected exceed the statistical variance due to sampling, suggesting that mis-modeling is present in some of these regions.

Since most bins in figure 8 are coloured green or light red/blue, we observe that most of the space is well-described within ±2 standard deviations. This indicates that, in general, the model is able to describe the high-dimensional distribution of the data at a level comparable with the statistical precision of the training data.

Some red or blue bands are observed, for example (i) in the steeply falling tail of the $\Delta y(j,j)$ distribution when projected along with m_{jj} and $|y^{jj}|$, and (ii) in the region $p_{\mathrm{T}}^{j1} \approx p_{\mathrm{T}}^{j2}$ in the projection of the two. This suggests some systematic mis-modeling in these regions, and scope for tuning using the optimization methods suggested in section 3.

White regions indicate that no density is present, whilst black regions indicate that events are present when sampling the density model but not `MG5`, repeating the observations discussed above. Table 4 summarizes the total frequency with which pulls are observed in the different colour bins.

5. EW Zjj with 4 observables and 2 external parameters

We now train a model which captures the dependence of EW *Zjj* data on the external parameters $\vec{c} = \{c_{\mathrm{HWB}}, \tilde{c}_W\}$. Such a model may be used to perform maximum likelihood estimation or derive exclusion limits on the space of $\vec{c}$ based on an observed dataset[6].

In this case, two SMEFT coefficients would be profiled with all others assumed to be 0. This is consistent with experimental analyses in the Higgs and electroweak sectors, in which only one or two parameters are usually profiled at a time. In general, it is not possible to constrain many more parameters. This is because we must simulate training data at regular intervals in all directions of $\vec{c}$. The number of required simulations therefore grows exponentially with the number of parameters profiled, which quickly becomes computationally intractable[7].

We note that the external parameters also impact the *rate* $\sigma_{\mathrm{fid}}(\vec{c})$ at which signal is expected to be produced within the observable phase space. When performing an experiment with a fixed exposure (rather than a fixed number of events), we expect to observe events at a point x in phase space at a rate of:

$$\frac{\mathrm{d}\sigma(x|\vec{c})}{\mathrm{d}x} = \sigma_{\mathrm{fid}}(\vec{c}) \cdot p(x|\vec{c}). \tag{14}$$

[6] We emphasize that detector effects have not been applied to our training data, but would be for such an analysis.

[7] We note that global fits of SMEFT parameters are possible when using binned measurements [27, 44, 45]. This is because the prediction for a given $\vec{c}$ may be decomposed into a parametric relationship between a number of pure-SM, pure-SMEFT and interference terms. In this case, since the number of unique terms rises slower than exponentially with the number of parameters, all parameters which impact EW *Zjj* events may be profiled together. However, for general new physics models where no such parameterization exists, the number of parameters profiled will be limited by the curse-of-dimensionality. Exploiting this special case is not possible using our method because we cannot express negative event densities which may arise in the interference term.

Table 5. Constants used to construct and train a density model describing EW *Zjj* data with 4 observables and 2 external parameters.

$N_G = 30$	$A_1 = 50$	$A_2 = 0$	$B_1 = 50$	$B_2 = 20$
$C = 2$	$D = 3$	$s_f = 0.125$	$s_\mu = 0.125$	$s_\sigma = 0.125$
$f_\sigma = 0.25$	batch size = 5k	$\lambda_{\text{lr}} = 0.001$	$\lambda_{\text{lr}}^{\text{update factor}} = 0.5$	$\lambda_{\text{lr}}^{\text{patience}} = 3$

In this work we consider the modeling of $p(x|\vec{c})$. We note that $\sigma_{\text{fid}}(\vec{c})$ may typically be modelled using a simple feed-forward neural network, allowing the event rate to be used as a discriminating observable if desired.

We also note that we are not modeling any backgrounds to the EW *Zjj* process. This is because we wish to test our ability to model multi-dimensional data with a non-trivial parameter dependence. This is best achieved by isolating the signal component, since in general background processes will not depend on the same parameters. However, we note that background modeling must be considered when performing parameter inference using detector-level data, and in particular a large irreducible background from non-electroweak *Zjj* production would exist in a 'real-world' EW *Zjj* analysis. For such an analysis, a statistical model combining individual components $p_{\text{sig}}(x|\vec{c})$ and $p_{\text{bkg}}(x)$ with expected cross-sections $\sigma_{\text{sig}}(\vec{c})$ and σ_{bkg} may be constructed as:

$$p(x|\vec{c}) = \frac{\sigma_{\text{sig}}(\vec{c}) \cdot p_{\text{sig}}(x|\vec{c}) + \sigma_{\text{bkg}} \cdot p_{\text{bkg}}(x)}{\sigma_{\text{sig}}(\vec{c}) + \sigma_{\text{bkg}}}, \tag{15}$$

assuming that interference is either small or absorbed into the background model.

For simplicity we select four observables to model, in the sequential order p_T^{ll}, p_T^{j1}, m_{jj} and finally $\Delta\phi(j,j)$, excluding the other eight from consideration. All four observables are expected to depend on the external parameters, and we aim to capture this dependence within our model.

The projection onto the latent space is performed using the same f-values as presented in table 2 and used in the previous section. Table 5 presents the constants used to configure the neural networks which contain 18k–85k trainable parameters. Compared with those in table 3, we note that larger values of s_f, s_μ and s_σ are used. This initializes the model such that external parameter variations deform the kinematic spectra, and so impact the log-likelihood, significantly enough that we find an improved parameter dependence to be learned during training. However, we note that large values may excessively enhance fluctuations and lead to an unstable initial state, and the final constants are chosen to balance these effects. The constant f_σ is tuned to ensure that the initial Gaussian width is not much larger than the scale of latent space features which are deformed by parameter variations.

Each neural network is trained for up to 200 epochs, stopping early if the log-likelihood does not improve by an amount greater than 10^{-10} over a period of 15 epochs. Figure 9 shows the 1D marginal distributions evaluated at the SM hypothesis of $\vec{c} = (0,0)$, obtained by sampling $4\,M$ events from the density model. Figure 10 shows the corresponding pulls on the 2D marginal spectra. Replicating the results of the previous section, these demonstrate that the model describes the 1D distributions to within $\pm 5\%$ at this point in parameter space, and without significant pulls in the 2D projections.

To investigate whether the parameter dependence has been learned, we scan across all hypotheses in the $\vec{c}$-plane and study the *ratio* of the 1D marginal distributions when compared with the SM. To reduce sampling variance when studying the density model, we form this ratio using importance sampling. We first sample 100k events from the model assuming the SM hypothesis. We then use the density model to evaluate the probability density of every datapoint under both the SM and $\vec{c}$ hypotheses, labelled p_{SM} and p_c respectively. The distribution under the $\vec{c}$ hypothesis is then obtained by assigning a weight of $\frac{p_c}{p_{\text{SM}}}$ to every datapoint. This approach assumes that the probability distribution under the SM hypothesis fully spans the support of that of the $\vec{c}$ hypothesis. The result is that the distributions obtained under the SM and $\vec{c}$ hypotheses have strongly correlated statistical fluctuations. These largely cancel when we take the ratio, which can be estimated using fewer samples than if the hypotheses were sampled independently.

Figure 11 shows how the p_{T}^{ll} PDF, expressed as a ratio with respect to the SM, varies as a function of the $\vec{c}$ hypothesis which is indicated by the green box in every panel. Events generated with `MG5` are shown in red, and those sampled from the density model are shown in black. We observe a significant enhancement of the high energy tail when $\tilde{c}_W$ is large in magnitude, approximately independent of its sign. We observe that negative values of c_{HWB} lead to a modest enhancement of the tail, whilst positive values suppress the tail by a comparable factor. The combination of these effects, plus any interference between them, manifests as a non-trivial structure throughout the plane of $\vec{c}$. We observe that the density model has captured this external parameter dependence well, since it is able to describe the deformations with an accuracy significantly better

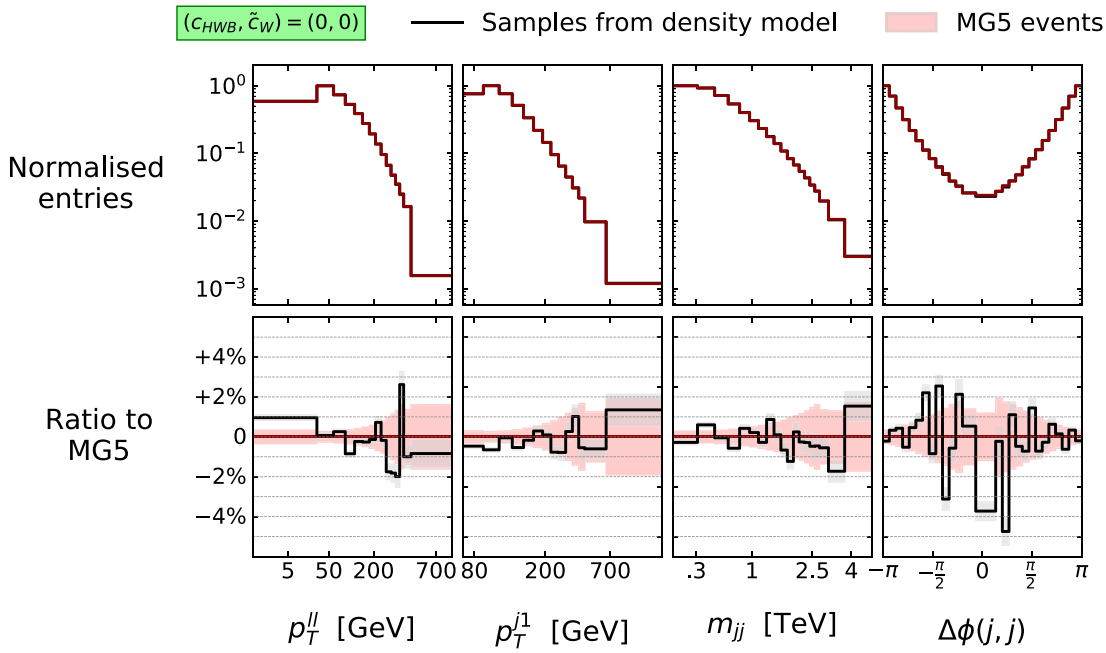


Figure 9. Marginal distributions of events sampled using the density model (black) compared with those generated using MG5 (red) for a value of $(c_{\mathrm{HWB}}, \tilde{c}_W) = (0, 0)$. Shaded areas show sampling uncertainties. Note that the two spectra are not statistically independent, since the density model was trained using the MG5 events.

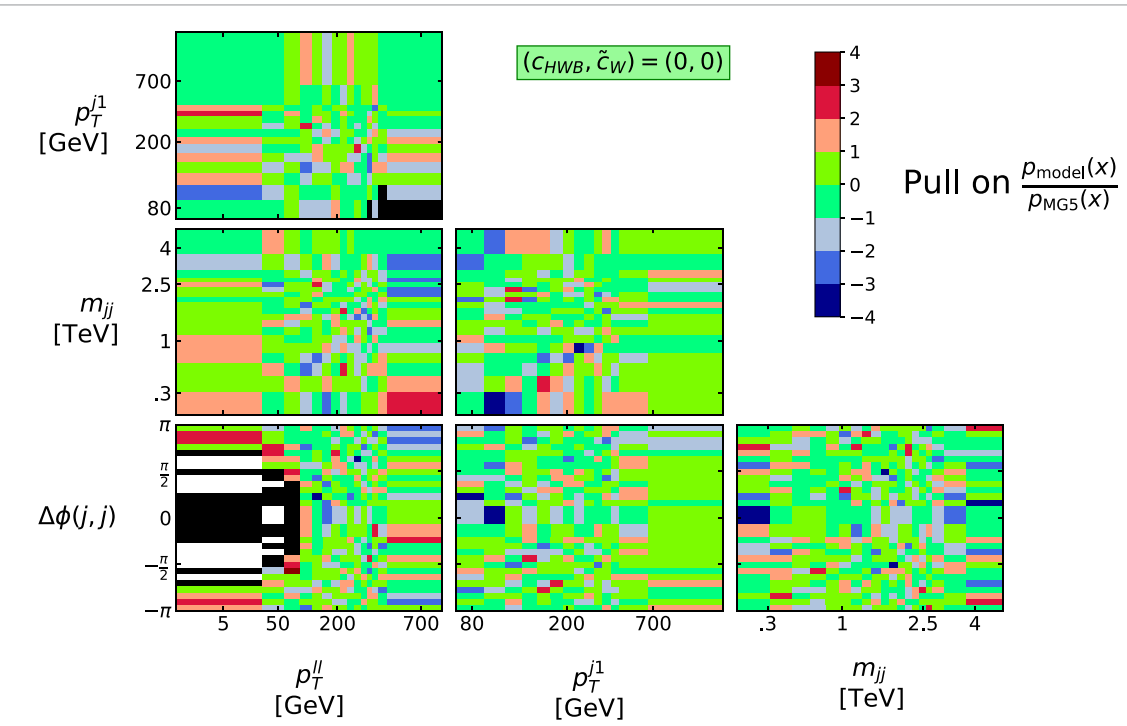


Figure 10. Pull on the ratio between the 2D marginal distributions comparing events simulated with MG5 (denominator) with those sampled from a GMM trained on a latent space (numerator), both assuming the SM hypothesis of $(c_{\mathrm{HWB}}, \tilde{c}_W) = (0, 0)$. The model accepts c_{HWB} and $\tilde{c}_W$ as input parameters.

than the size of the deformations themselves. The double ratio, quantitatively comparing the two histograms, is presented in figure A1 of appendix A.

Figure 12 shows how the p_{T}^{j1} PDF varies as a function of $\vec{c}$. We observe an enhancement of the high-energy tail when $\tilde{c}_W$ is large in magnitude. We also observe a low-energy enhancement when c_{HWB} is highly negative, resulting in another non-trivial structure as we scan the plane of $\vec{c}$. Once again, we find that the density model has captured this external parameter dependence well. The double ratio, quantitatively comparing the two histograms, is presented in figure A2 of appendix A.

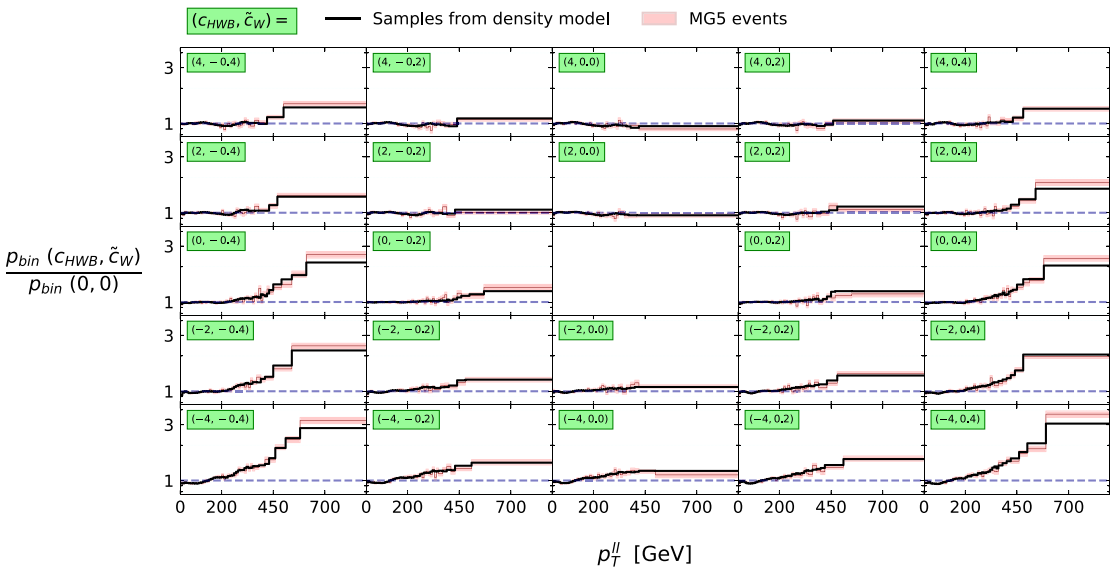


Figure 11. Evolution of the p_{T}^{ll} PDF as a function of $(c_{\mathrm{HWB}}, \tilde{c}_W)$, presented as a ratio with respect to the SM hypothesis. The dependence is well captured by the density model. The double ratio comparing the two histograms is shown in figure A1.

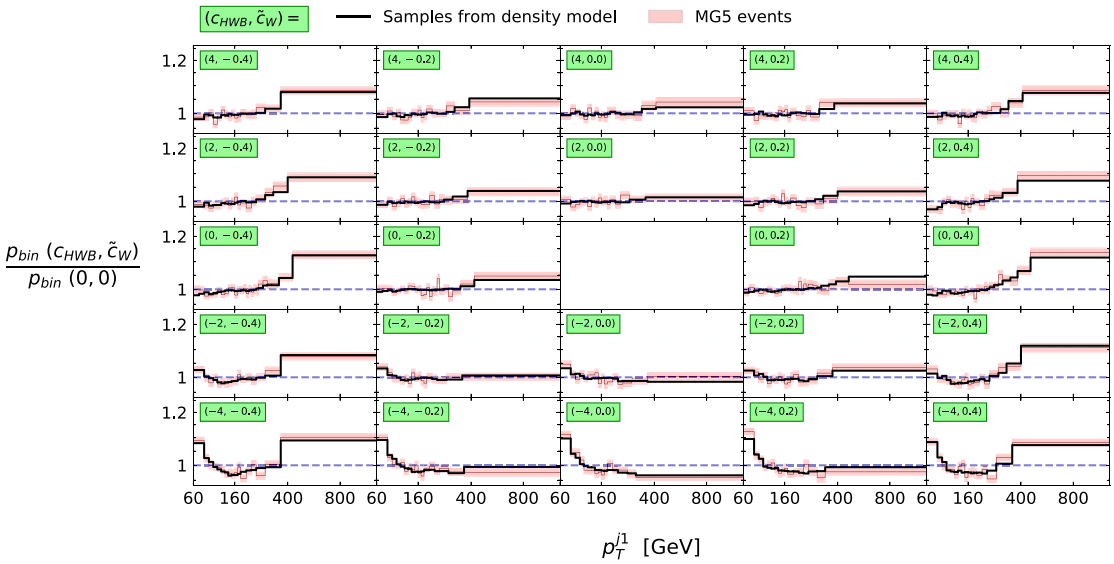


Figure 12. Evolution of the p_{T}^{j1} PDF as a function of $(c_{\mathrm{HWB}}, \tilde{c}_W)$, presented as a ratio with respect to the SM hypothesis. The dependence is well captured by the density model. The double ratio comparing the two histograms is shown in figure A2.

Figure 13 shows how the m_{jj} PDF varies as a function of $\vec{c}$. We observe that highly negative values of c_{HWB} lead to significant structure at $m_{jj} \sim 0.15$ TeV. As shown in figure 9, this is also where the bulk of the data is expected to be measured. When measuring other observables, experimental analyses typically apply pre-selection criteria requiring m_{jj} to exceed $\mathcal{O}(1\ \mathrm{TeV})$ in order to preferentially reject non-electroweak processes. By instead modelling an inclusive range of m_{jj} simultaneously with all other observables and performing a high-dimensional unbinned analysis, such a restrictive requirement would not be required, provided that all backgrounds can also be sufficiently well modelled. The double ratio, quantitatively comparing the two histograms, is presented in figure A3 of appendix A.

Figure 14 shows how the $\Delta\phi(j,j)$ PDF varies as a function of $\vec{c}$. We observe that $\tilde{c}_W$ modulates the amplitude of an approximately sinusoidal oscillation introduced into the $\Delta\phi(j,j)$ spectrum. We observe that negative values of c_{HWB} modulate an enhancement at $\Delta\phi(j,j) \sim 0$, whereas positive values of c_{W} cause a suppression. This observable is therefore sensitive to the sign of both parameters. Once again we note that

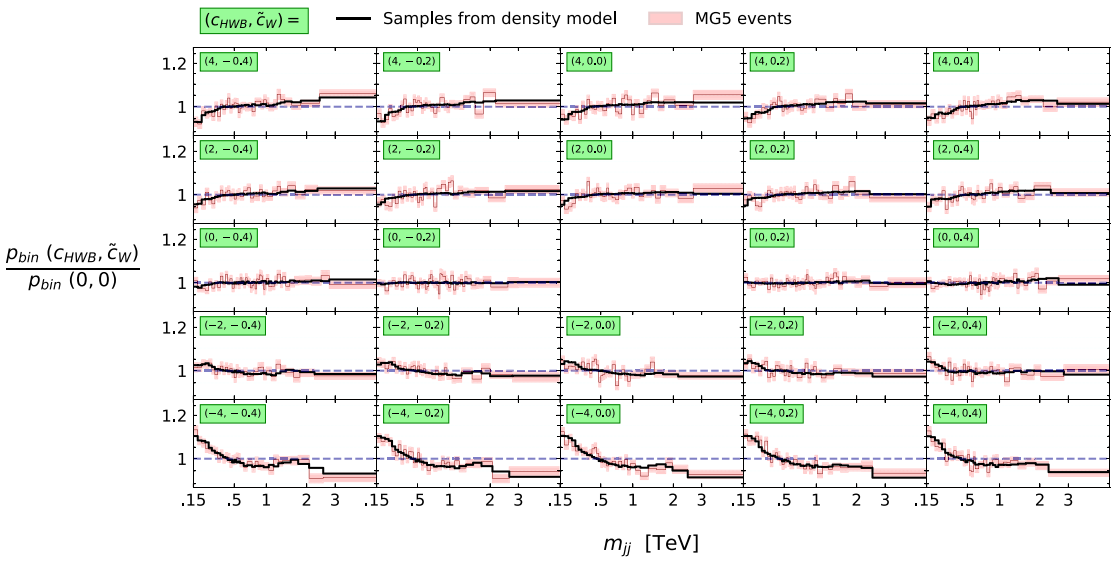


Figure 13. Evolution of the m_{jj} PDF as a function of $(c_{\mathrm{HWB}}, \tilde{c}_W)$, presented as a ratio with respect to the SM hypothesis. The dependence is well captured by the density model. The double ratio comparing the two histograms is shown in figure A3.

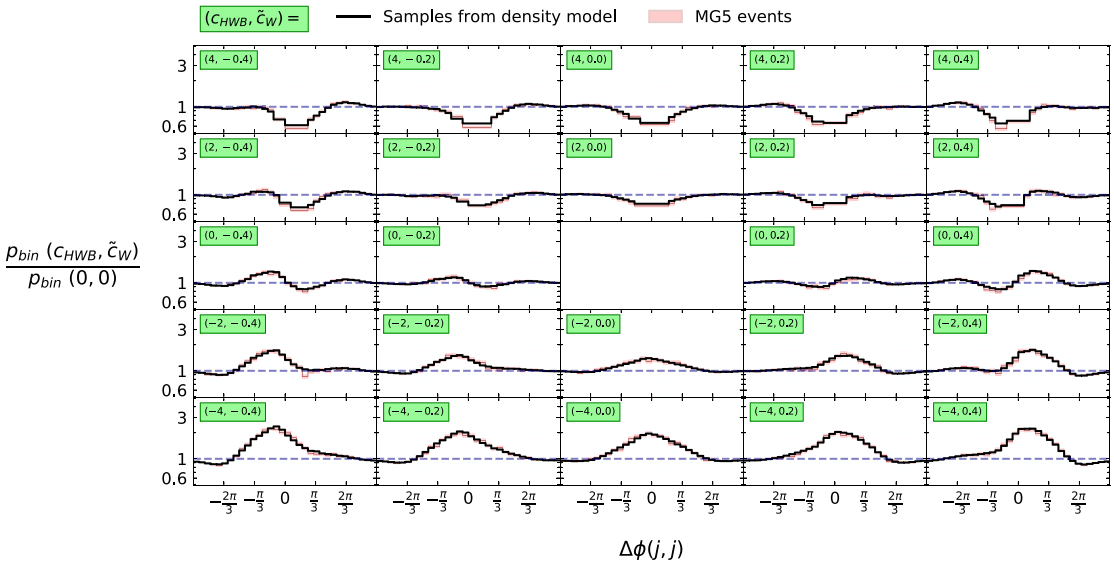


Figure 14. Evolution of the $\Delta\phi\,(j,j)$ PDF as a function of $(c_{\mathrm{HWB}}, \tilde{c}_W)$, presented as a ratio with respect to the SM hypothesis. The dependence is well captured by the density model. The double ratio comparing the two histograms is shown in figure A4.

the distribution shows a significantly non-trivial dependence as a function of $\vec{c}$, and that this dependence is captured well by the model. The double ratio, quantitatively comparing the two histograms, is presented in figure A4 of appendix A.

6. Demonstration of statistical interpretation using a toy model

In the previous two sections we have demonstrated that we can construct density models which replicate the behaviour of simulated training data when sampled. Whilst this implies that good behaviour should also be obtained when performing inference tasks at the trained points in parameter space, this cannot be demonstrated because we are not able to evaluate the ground truth PDF for any given datapoint.

Nonetheless, we consider such a demonstration to be important. This is because the quality of inference is impacted not only by the ability to fit the training data but by (1) the degree of under- or over-training and (2) the way in which the probability distribution is interpolated between training points, hereafter referred to as the inductive bias. Whilst the probability distribution may be learned with arbitrarily high accuracy *at* the training points, depending on the complexity of the model configuration and number of training samples

provided, it is likely that the interpolation between training points will not exactly match the true behaviour, which is unobserved. We aim to show that the approximate behaviour of the model can work sufficiently well for inference tasks, provided that training data are provided at dense enough points in parameter space.

To achieve this, we construct a toy model from which to sample ground truth training data. This is projected onto a latent space and used to train a density model using the method proposed in this paper. The toy contains four observables which vary according to two external parameters. Several pseudo-datasets are sampled from the true model assuming different parameter hypotheses. For each dataset, the density model is used to compute exclusion bounds on the latent space, and the results are compared with ground truth exclusion bounds computed using the true PDF on the data space. The level of agreement is then analyzed. Use of a toy model allows us to compute these ground truth bounds, which are typically intractable for real simulations.

We define a toy model with four observables $x = \{x_0,\ x_1,\ x_2,\ x_3\}$ and two external parameters $\vec{c} = \{c_x,\ c_y\}$. These observables are defined over the intervals $x_0 \in [100,\ 800]$, $x_1 \in [100,\ 800]$, $x_2 \in [-\pi,\ \pi]$ and $x_3 \in [-\infty,\ \infty]$. Appendix B defines the ground truth PDF and documents how samples are drawn. 50 k datapoints are sampled at each of the 49 parameter points in a two-dimensional grid spanning all permutations with $c_x \in [-1.5, -1, -0.5, 0, 0.5, 1, 1.5]$ and $c_y \in [-1.5, -1, -0.5, 0, 0.5, 1, 1.5]$.

Figure 15 (top) shows the 1D marginal distributions at the null hypothesis $\vec{c} = (0, 0)$ as well as several alternative hypotheses in the $\vec{c}$-plane. Observables x_0 and x_1 are highly correlated falling distributions, where variations of c_x away from 0 enhance the amplitude in the tail. These observables are insensitive to c_y as well as the sign of c_x. Observable x_2 is an angular observable for which c_x and c_y induce sinusoidal oscillations with a phase difference of $\frac{\pi}{2}$. This observable is sensitive to the sign and amplitude of both external parameters. Observable x_3 follows a smooth-peak distribution with no physical limits, and is correlated with all observables and external parameters.

Data are projected onto the latent space using values of $f = 0.5$ for all observables. Neural networks are configured using the constants presented in table 6 and contain 18k–85k trainable parameters. Each network is trained on 60% of the available data until the log-likelihood evaluated over the other 40% no longer improves by an amount greater than 10^{-6} over a period of 8 consecutive epochs, after which the solution with the least-positive (or most-negative) validation loss is chosen. Training is found to terminate after 33–46 epochs. Figure 15 (bottom) shows the latent space distributions. A third panel compares the 1D marginal distributions obtained from the ground truth data and from drawing 50 k samples from the resulting density model. The level of agreement is found to be comparable with the statistical precision of the data.

We now test the accuracy of inference performed using the density model. We select nine different 'true' hypotheses $\vec{c}_{\text{true}}$ in a 2D grid with edges at $c_x \in [-0.8, 0, 0.8]$ and $c_y \in [-0.8, 0, 0.8]$. For each value of $\vec{c}_{\text{true}}$, a pseudo-dataset with a size of 400 events is created by sampling the true PDF. We assume that the expected number of observed events is identical for every value of $\vec{c}$. Figure 16(a) shows nine panels in which the different $\vec{c}_{\text{true}}$ hypotheses are presented as black dots. Open circles show the points in parameter space $\vec{c}_{\text{trained}}$ at which the model was trained, excluding those which lie outside of the axis range.

The true PDF is used to profile the likelihood of the dataset. Using this method we evaluate (1) the true maximum likelihood estimate (MLE) and (2) the frequentist 68% and 95% confidence limits, assuming that the expected distribution of the profile likelihood ratio follows the asymptotic approximation described by Wilks' theorem [46, 47]. In figure 16(a), orange crosses present the MLE evaluated using the true PDF, whilst orange contours present the confidence limits. We note that, since the pseudo-datasets are stochastically sampled from the true PDF, we expect each MLE to fluctuate away from $\vec{c}_{\text{true}}$ as observed. The datasets are then transformed onto the latent space, and the same analysis is performed using the density model to evaluate the likelihood. Blue crosses present the MLE evaluated using the density model, whilst blue contours present the confidence limits.

Figure 16(a) demonstrates generally good agreement between the exclusions bounds evaluated using the density model and ground truth PDF, although we observe a mild over-coverage when $c_x \sim 0$ or $c_y \sim 0$. We expect that this is because these axes represent turning points in the function $p\,(x|\vec{c})$, the form of which is only approximated by the inductive bias of the density model. To test this, we train a second model which contains additional training data at $c_x = \pm 0.2$ and $c_y = \pm 0.2$. The resulting contours are shown in figure 16(b). We observe that the additional training data have constrained the model at $|c_x|, |c_y| \sim 0$, resulting in an improved agreement with the ground truth. We conclude that the most reliable results will be achieved when the spacing of $\vec{c}_{\text{trained}}$ points is smaller than the size of the expected exclusion bounds.

In both cases, figure 16 shows that accurate MLEs and exclusion contours have been estimated using density models on the latent space. Reliable results could therefore be obtained in this example without having access to the true PDF.

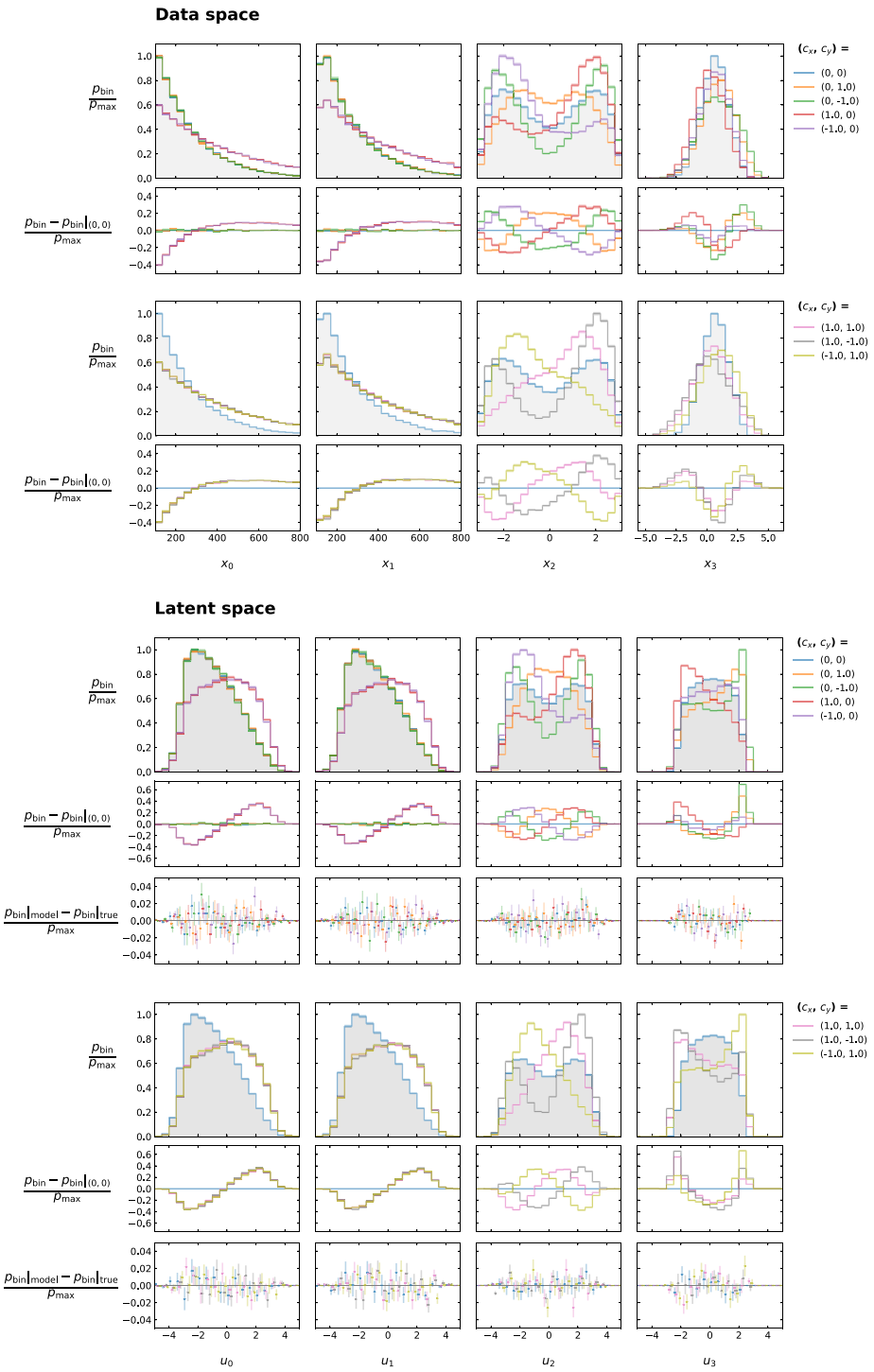
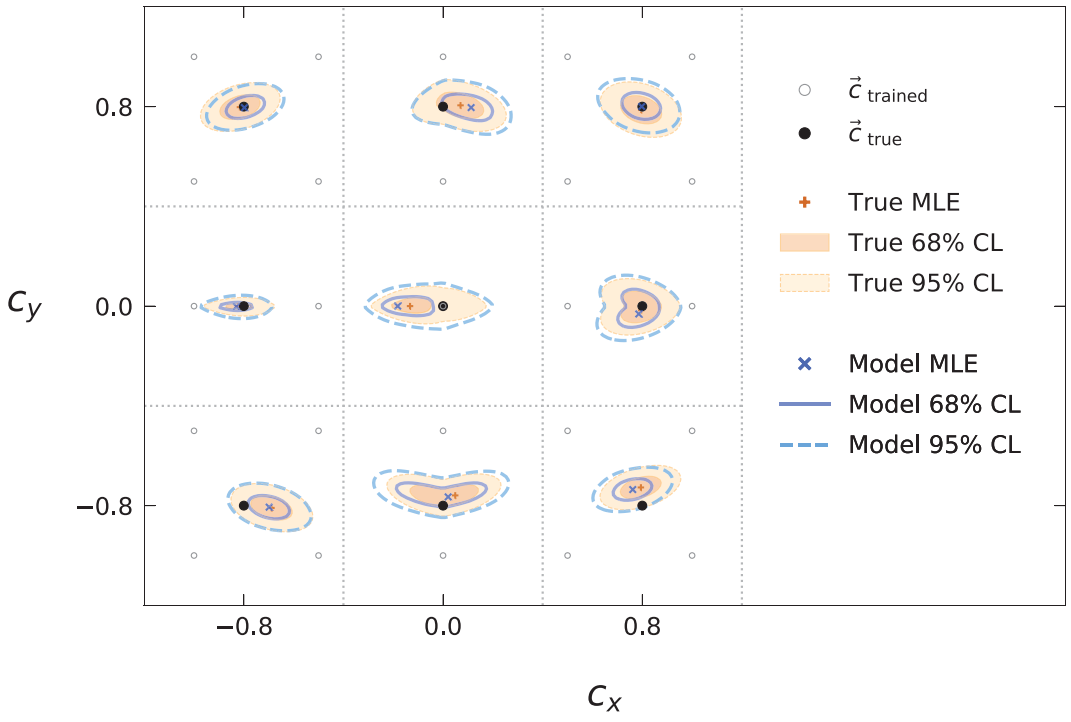


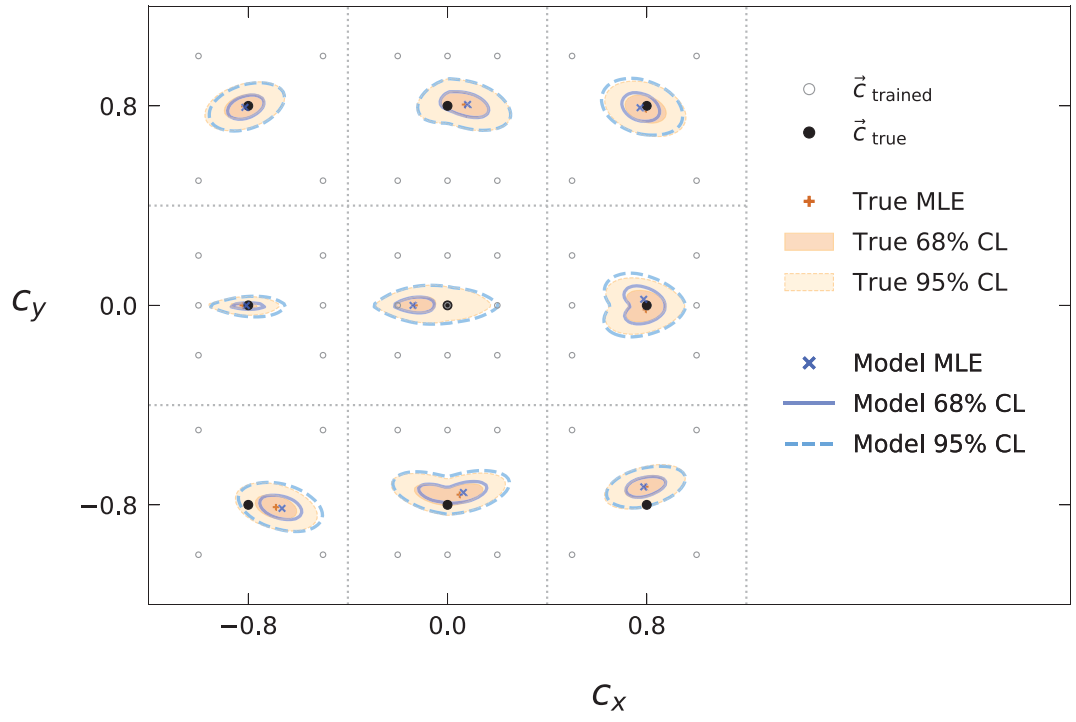
Figure 15. Kinematic distributions of toy model data before (top: 'Data space') and after (bottom: 'Latent space') projecting onto the latent space. Secondary panels highlight how these are modified by variations of the conditional parameters $\vec{c} = (c_x, c_y)$. On the latent space, a third panel compares ground truth events with those sampled from the learned density model, demonstrating agreement within the statistical precision of the training data for all values of $\vec{c}$.

Table 6. Constants used to construct and train a density model describing toy data with 4 observables and 2 external parameters.

$N_G = 20$	$A_1 = 50$	$A_2 = 0$	$B_1 = 50$	$B_2 = 20$
$C = 2$	$D = 3$	$s_f = 0.01$	$s_\mu = 0.01$	$s_\sigma = 0.01$
$f_\sigma = 0.25$	batch size $= 500$	$\lambda_{\text{lr}} = 0.001$	$\lambda_{\text{lr}}^{\text{update factor}} = 0.5$	$\lambda_{\text{lr}}^{\text{patience}} = 2$



(a) Model trained with nominal $\vec{c}_{\text{trained}}$.



(b) Model trained with additional $\vec{c}_{\text{trained}}$ points.

Figure 16. 68% and 95% confidence level contours in the $\vec{c}$-plane for nine separate datasets of size $N = 400$ randomly sampled around the hypotheses $\vec{c}_{\text{true}}$ shown in black. Contours are evaluated on the data space using the true probability model (orange) and on the latent space using the density model (blue). Crossed markers show the corresponding maximum likelihood estimators (MLEs). Good agreement is observed. Open circles show the points in parameter space $\vec{c}_{\text{trained}}$ at which the model is trained.

7. Conclusion

We present a method for modelling probability distributions over a high-dimensional space of observables with dependence on external parameters, a dataset type which is common within the physical sciences. The method uses a novel transformation of input data and a targeted network architecture to improve the

expressive power of GMMs. It is designed to capture smooth deformations of the probability density induced by external parameter variations, and respects strict boundaries on the observables. The model may be used to perform inference on observed data, or sampled to act as a stochastic generator.

We demonstrate the power of the method by applying it to two high-energy particle physics datasets: one which contains twelve highly correlated observables, and one which depends on two external parameters. We then use a toy model to demonstrate that fast and accurate inference may be performed from experimental data. We demonstrate that the problem-of-interest may also contain discrete observables, which are modelled with a relatively simple categorical model. Whilst the method enables interpretations to be performed using unbinned multi-dimensional data, it may also be used within the experimental design of binned measurements (which are intended to characterize observed data with minimal physical model assumptions). Such an analysis may proceed as follows. An experimenter may assign benchmark hypotheses to which a planned measurement should have reasonably optimized sensitivity. We expect that a near-optimal classifier[8] for a given parameter hypothesis may be created using the ratio of the PDFs evaluated at the null and alternative hypotheses. By isolating the regions of the high-dimensional space which provide the most discrimination power, they may ensure that these regions are targeted by dedicated bins.

The method presented is not domain-specific, and may be used to model any dataset of continuous observables which follow a smooth PDF, and to subsequently perform statistical inference from experimental data for the purposes of scientific discovery.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://dx.doi.org/10.48420/17136839.

Data and code availability

The code implementing the methodology developed in this paper and which can be used to reproduce these results is available at [49]. All corresponding simulated data and neural network model files are openly available at [50].

Acknowledgments

Darren Price is supported by a Turing Fellowship from the Alan Turing Institute, London, UK, by the Science and Technology Facilities Council (STFC) under Grant ST/N000374/1, and by the University of Manchester. Stephen Menary is supported through a grant from the Alan Turing Institute and STFC Grant No. ST/N000374/1.

Appendix A. Double ratio plots comparing MG5 events with those sampled from the density model constructed used in section 5

This appendix presents further results concerning the experiments shown in section 5. In that section, figures 11–14 present the single-ratios comparing $p_{bin}\left(c_{\mathrm{HWB}}, \tilde{c}_W\right)$ with $p_{bin}\left(0,0\right)$, visually demonstrating that the density model is able to capture deformations to the four observable spectra as c_{HWB} and $\tilde{c}_W$ are varied. Here, figures A1–A4 show the double-ratios which quantitatively compare the single-ratios evaluated using `MG5` events with the single-ratios evaluated using events sampled from the density model.

In figures A1–A4 we observe that the double-ratio is consistent with unity at a level comparable with the estimated statistical uncertainty on the training data, which is presented as the red shaded area.

Appendix B. Ground truth probability density and sampling for the toy model used in section 6

For observables $\vec{x} = \{x_0,\ x_1,\ x_2,\ x_3\}$ and external parameters $\vec{c} = \{c_x,\ c_y\}$, the toy model described in section 6 is defined by a probability density:

[8] The Neyman–Pearson lemma states that the PDF ratio is the test-statistic with the highest fake rejection rate for a given true positive rate [48].

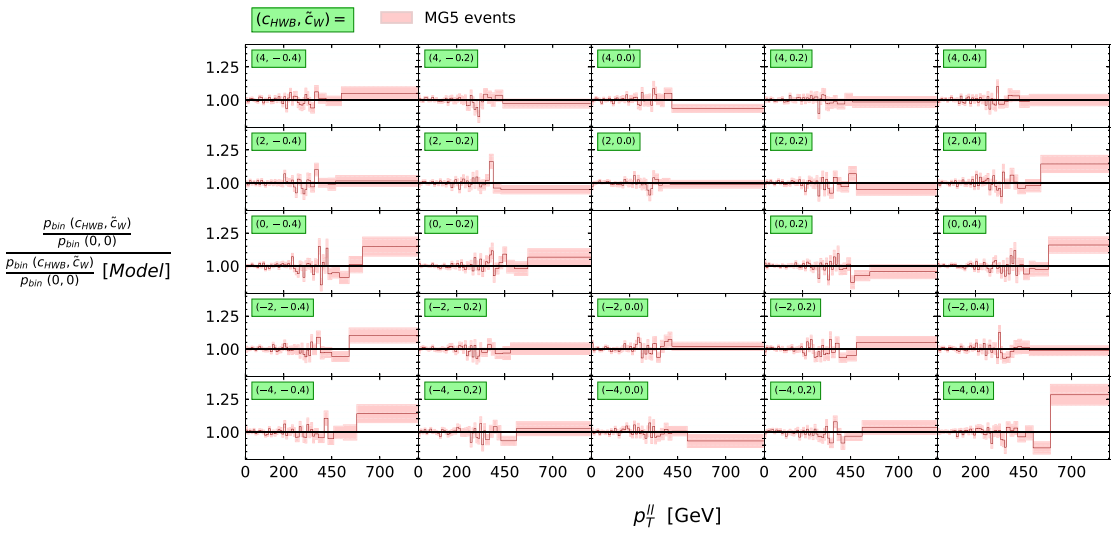


Figure A1. Double ratio comparing $p_{bin}(c_{HWB}, \tilde{c}_W)/p_{bin}(0,0)$ estimated using MG5 events with those sampled from the density model, shown as a function of p_T^{ll}. The corresponding single ratios are shown in figure 11.

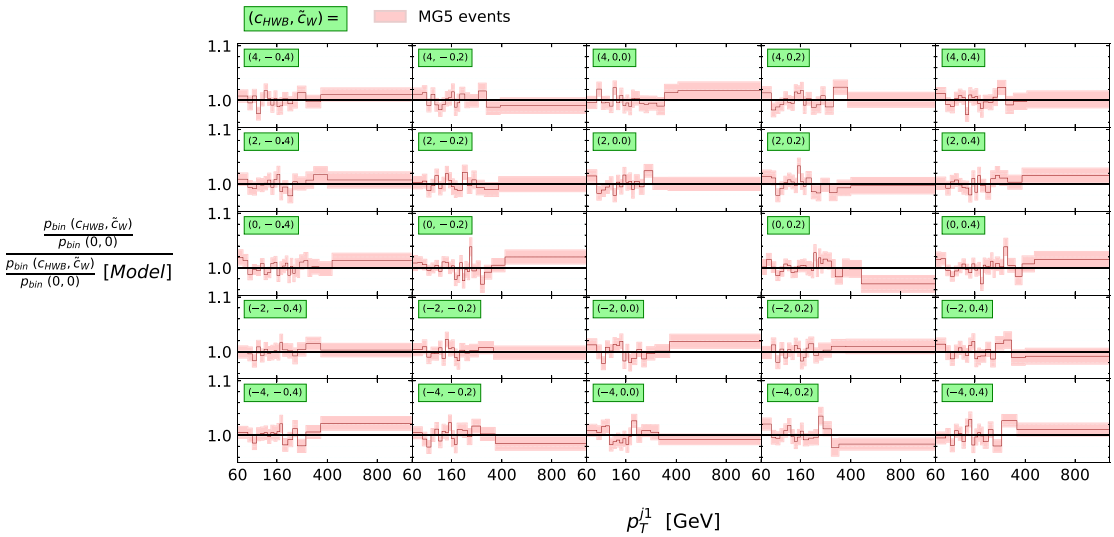


Figure A2. Double ratio comparing $p_{bin}(c_{HWB}, \tilde{c}_W)/p_{bin}(0,0)$ estimated using MG5 events with those sampled from the density model, shown as a function of p_T^{j1}. The corresponding single ratios are shown in figure 12.

$$p_{\text{true}}(\vec{x}|\vec{c}) = p^{(0)}_{\text{true}}(x_0|c_x) \cdot p^{(1)}_{\text{true}}(x_1|x_0) \cdot p^{(2)}_{\text{true}}(x_2|\vec{c}) \cdot p^{(3)}_{\text{true}}(x_3|\vec{c},x_1,x_2)\,, \tag{B.1}$$

with the conditional probability densities:

$$\begin{aligned}
p^{(0)}_{\text{true}}(x_0|\,c_x) &= \frac{1}{700}\cdot\frac{2(2-|c_x|)}{\left(1-e^{-2(2-|c_x|)}\right)}\cdot e^{-2(2-|c_x|)\cdot x_0'}\\
p^{(1)}_{\text{true}}(x_1|\,x_0) &= \frac{1}{700}\cdot\frac{1}{\sqrt{\frac{\pi}{2}}\cdot\sigma_1\cdot\left(\text{erf}\frac{x_0'}{\sqrt{2}\sigma_1}-\text{erf}\frac{x_0'-1}{\sqrt{2}\sigma_1}\right)}\cdot e^{-\frac{(x_1'-x_0')^2}{2\cdot\sigma_1^2}}\\
p^{(2)}_{\text{true}}(x_2|\,\vec{c}) &= \frac{\left(\alpha_2+\beta_2 x_2^2+\gamma_2 x_2^4\right)\cdot\left(1+\delta_2(c_x)\sin x_2+\epsilon_2(c_y)\cos x_2\right)}{f_2(\vec{c},\pi)-f_2(\vec{c},-\pi)}\\
p^{(3)}_{\text{true}}(x_3|\,\vec{c},x_1,x_2) &= q_3\left(x_3+\frac{3}{5}\left(\sqrt{4+|c_x|}+|c_y|\right)\,(x_1'+x_2')\right),
\end{aligned} \tag{B.2}$$

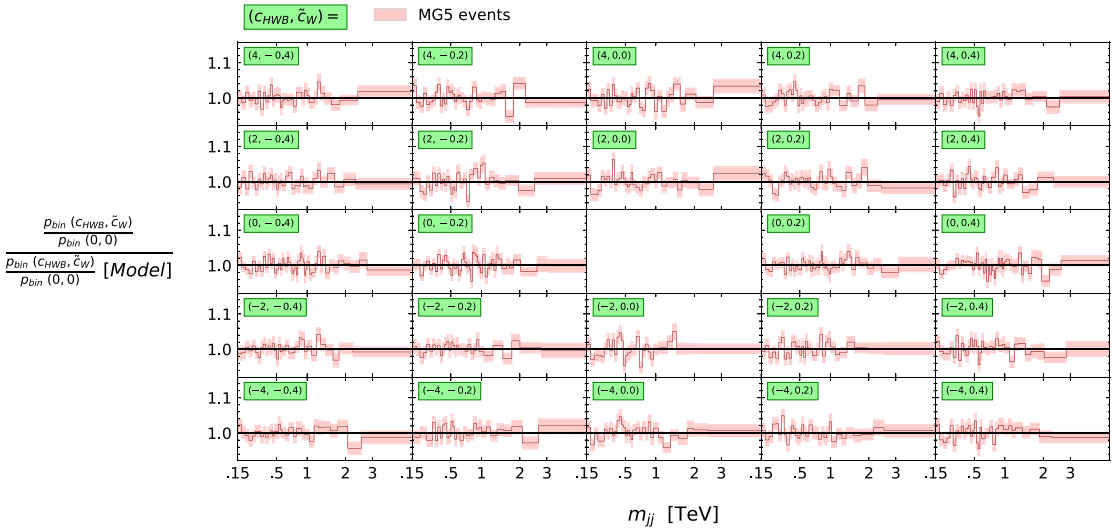


Figure A3. Double ratio comparing $p_{\mathrm{bin}}(c_{\mathrm{HWB}}, \tilde{c}_W)/p_{\mathrm{bin}}(0,0)$ estimated using MG5 events with those sampled from the density model, shown as a function of m_{jj}. The corresponding single ratios are shown in figure 13.

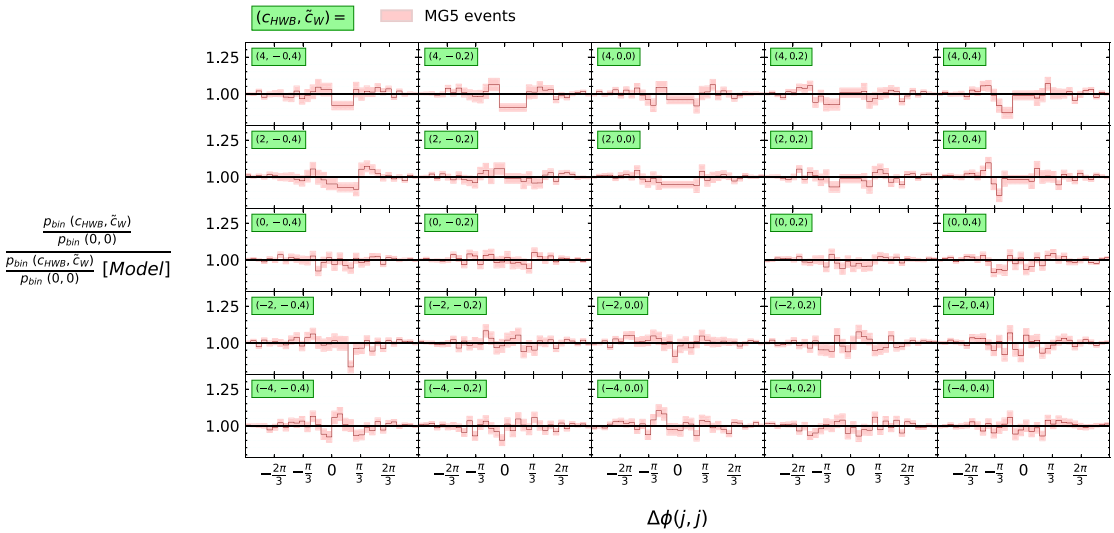


Figure A4. Double ratio comparing $p_{\mathrm{bin}}(c_{\mathrm{HWB}}, \tilde{c}_W)/p_{\mathrm{bin}}(0,0)$ estimated using MG5 events with those sampled from the density model, shown as a function of $\Delta\phi(j,j)$. The corresponding single ratios are shown in figure 14.

defined over the intervals:

$$\begin{aligned} x_0 &\in [100,\ 800] \\ x_1 &\in [100,\ 800] \\ x_2 &\in [-\pi,\ \pi] \\ x_3 &\in [-\infty,\ \infty], \end{aligned} \tag{B.3}$$

where

$$x_0' = 2\,\frac{x_0 - 100}{700} - 1, \quad x_1' = 2\,\frac{x_1 - 100}{700} - 1, \quad x_2' = \frac{x_2 + \pi}{\pi} - 1, \tag{B.4}$$

with $\alpha_2 = 1$, $\beta_2 = \frac{4}{\pi^2}$, $\gamma_2 = -\frac{5}{\pi^4}$, $\delta_2(c_x) = \frac{2}{5}c_x$, $\epsilon(c_y) = \frac{1}{2}c_y$, $\alpha_3 = 10$, $\beta_3 = 1$, $\gamma_3 = 1$ and

$$
\begin{aligned}
f_2\left(\vec{c},\, x\right) = \alpha_2 x + \frac{\beta_2}{3}x^3 + \frac{\gamma_2}{5}x^5 \\
&+ \left[\alpha_2\epsilon_2 + 2\beta_2\delta_2 x + \beta_2\epsilon_2\left(x^2-2\right) + 4\gamma_2\delta_2 x\left(x^2-6\right)\right. \\
&\quad \left. + \gamma_2\epsilon_2\left(x^4 - 12x^2 + 24\right)\right]\sin x \\
&+ \left[-\alpha_2\delta_2 + 2\beta_2\epsilon_2 x - \beta_2\delta_2\left(x^2-2\right) + 4\gamma_2\epsilon_2 x\left(x^2-6\right)\right. \\
&\quad \left. - \gamma_2\delta_2\left(x^4 - 12x^2 + 24\right)\right]\cos x, \\
q_3\ (x) = \frac{1}{\left(1+\exp[\alpha_3\left(x-\beta_3\right)-\gamma_3]\right)} \cdot \frac{1}{\left(1+\exp[-\alpha_3\left(x-\beta_3\right)-\gamma_3]\right)} \cdot \frac{1}{2\left(\alpha_3\beta_3+\gamma_3\right)f_3}, \\
f_3 = \frac{1}{\alpha_3} \cdot \frac{\exp[2\left(\alpha_3\beta_3+\gamma_3\right)]}{\exp[2\left(\alpha_3\beta_3+\gamma_3\right)]-1}, \\
g_3 = f_3\ \cdot\left(\alpha_3\beta_3+\gamma_3\right), \\
h_3\ (x) = \exp\left[\frac{g_3\left(2x-1\right)}{f_3}\right].
\end{aligned}
\tag{B.5}
$$

Samples are drawn according to:

$$
\begin{aligned}
x_0^* &= 100 - 700\ \cdot \frac{1}{2\left(2-c_x\right)} \cdot \log\left(1 - i_0^*\left(1 - e^{-2\left(2-|c_x|\right)}\right)\right) \\
x_1^* &= 100 + 700\ \left[x_0' - \sqrt{2}\sigma_1\ \mathrm{erf}^{-1}\left(\left(1-i_1^*\right)\mathrm{erf}\left(\frac{x_0'}{\sqrt{2}\sigma_1}\right) + \mathrm{erf}\left(\frac{x_0'-1}{\sqrt{2}\sigma_1}\right) i_1^*\right)\right] \\
x_2^* &= I_2^{-1}\left(\vec{c},\, i_2^*\right) \\
x_3^* &= I_3^{-1}\left(i_3^*\right) - \frac{3}{5}\left(\sqrt{4+|c_x|} + |c_y|\right)\left(x_1^* + x_2^*\right),
\end{aligned}
\tag{B.6}
$$

where I_2^{-1} is evaluated numerically as the inverse function of:

$$
I_2\left(\vec{c},\, x\right) = \frac{f_2\left(\vec{c},\, x\right) - f_2\left(\vec{c},\, -\pi\right)}{f_2\left(\vec{c},\, \pi\right) - f_2\left(\vec{c},\, -\pi\right)},
\tag{B.7}
$$

and,

$$
I_3^{-1}\left(i_3\right) = \frac{1}{\alpha_3}\log\frac{h_3\left(i_3\right)\exp\left[\alpha_3\beta_3+\gamma_3\right]-1}{\exp\left[\alpha_3\beta_3+\gamma_3\right]-h_3\left(i_3\right)}.
\tag{B.8}
$$

ORCID iDs

Stephen B Menary https://orcid.org/0000-0003-1244-2802
Darren D Price https://orcid.org/0000-0003-2750-9977

References

[1] Brehmer J, Cranmer K, Louppe G and Pavez J 2018 A guide to constraining effective field theories with machine learning *Phys. Rev.* D **98** 052004
[2] Brehmer J, Kling F, Espejo I and Cranmer K 2020 MadMiner: machine learning-based inference for particle physics *Comput. Softw. Big Sci.* **4** 3
[3] Brehmer J, Louppe G, Pavez J and Cranmer K 2020 Mining gold from implicit models to improve likelihood-free inference *Proc. Natl Acad. Sci.* **117** 5242–9
[4] Cranmer K, Pavez J and Louppe G 2015 Approximating likelihood ratios with calibrated discriminative classifiers (arXiv:1506.02169 [stat.AP])
[5] Papamakarios G, Pavlakou T and Murray I 2018 Masked autoregressive flow for density estimation (arXiv:1705.07057 [stat.ML])
[6] Uria B, Côté M-A, Gregor K, Murray I and Larochelle H 2016 Neural autoregressive distribution estimation (arXiv:1605.02226 [cs.LG])
[7] Alsing J, Charnock T, Feeney S and Wandelt B 2019 Fast likelihood-free cosmology with neural density estimators and active learning *Mon. Not. R. Astron. Soc.* **488** 4440–58
[8] Dinh L, Sohl-Dickstein J and Bengio S 2017 Density estimation using real NVP (arXiv:1605.08803 [cs.LG])
[9] Štěpánek M, Franc J and Kůs V 2015 Modification of Gaussian mixture models for data classification in high energy physics *J. Phys.: Conf. Ser.* **574** 012150
[10] Barron J, Curtin D, Kasieczka G, Plehn T and Spourdalakis A 2021 Unsupervised hadronic SUEP at the LHC (arXiv:2107.12379 [hep-ph])
[11] Freitas F F, Khosa C K and Sanz V 2019 Exploring the standard model EFT in *VH* production with machine learning *Phys. Rev.* D **100** 035040
[12] Kasieczka G *et al* 2021 The LHC olympics 2020: a community challenge for anomaly detection in high energy physics (arXiv:2101.08320 [hep-ph])

[13] Goodfellow I J, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A and Bengio Y 2014 Generative adversarial networks (arXiv:1406.2661 [stat.ML])
[14] Kingma D P and Welling M 2014 Auto-encoding variational Bayes (arXiv:1312.6114 [stat.ML])
[15] Kingma D P and Welling M 2019 An introduction to variational autoencoders *Found. Trends Mach. Learn.* **12** 307–92
[16] Sipio R Di, Giannelli M F, Haghighat S K and Palazzo S 2019 DijetGAN: a generative-adversarial network approach for the simulation of QCD dijet events at the LHC *J. High Energy Phys.* **08** 110
[17] Butter A, Plehn T and Winterhalder R 2019 How to GAN LHC events *SciPost Phys.* **7** 075
[18] Butter A and Plehn T 2020 Generative networks for LHC events *Artificial Intelligence for Particle Physics* (arXiv:2008.08558 [hep-ph])
[19] ATLAS Collaboration 2018 Deep generative models for fast shower simulation in ATLAS *Technical Report* ATL-SOFT-PUB-2018-001 (Geneva: CERN) (available at: https://cds.cern.ch/record/2630433)
[20] Bishop C M 1994 Mixture density networks *Technical Report* NCRG/94/004
[21] Variani E, McDermott E and Heigold G 2015 A Gaussian mixture model layer jointly optimized with discriminative features within a deep neural network architecture *2015 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)* pp 4270–4
[22] ATLAS Collaboration 2021 Differential cross-section measurements for the electroweak production of dijets in association with a Z boson in proton–proton collisions at ATLAS *Eur. Phys. J.* C **81** 163
[23] ATLAS Collaboration 2017 Measurement of the cross-section for electroweak production of dijets in association with a Z boson in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector *Phys. Lett.* B **775** 206–28
[24] ATLAS Collaboration 2008 The ATLAS experiment at the CERN Large Hadron Collider *J. Instrum.* **3** S08003
[25] Grzadkowski B, Iskrzynski M, Misiak M and Rosiek J 2010 Dimension-six terms in the Standard Model Lagrangian *J. High Energy Phys.* **10** 085
[26] Brivio I and Trott M 2019 The standard model as an effective field theory *Phys. Rep.* **793** 1–98
[27] Ellis J, Murphy C W, Sanz V and You T 2018 Updated global SMEFT fit to Higgs, Diboson and electroweak data *J. High Energy Phys.* **06** 146
[28] Alwall J *et al* 2014 The automated computation of tree-level and next-to-leading order differential cross sections and their matching to parton shower simulations *J. High Energy Phys.* **07** 079
[29] Buckley A *et al* 2011 General-purpose event generators for LHC physics *Phys. Rep.* **504** 145–233
[30] Zyla P A *et al* (Particle Data Group) 2020 Review of particle physics *Prog. Theor. Exp. Phys.* **2020** 083C01
[31] Sjöstrand T *et al* 2015 An introduction to PYTHIA 8.2 *Comput. Phys. Commun.* **191** 159–77
[32] Sjostrand T, Mrenna S and Skands P Z 2008 A brief introduction to PYTHIA 8.1 *Comput. Phys. Commun.* **178** 852–67
[33] Bierlich C *et al* 2020 Robust independent validation of experiment and theory: rivet version 3 *SciPost Phys.* **8** 026
[34] Abadi M *et al* 2015 TensorFlow: large-scale machine learning on heterogeneous systems (available at: www.tensorflow.org)
[35] Chollet F *et al* 2015 Keras (available at: https://keras.io)
[36] Brivio I, Jiang Y and Trott M 2017 The SMEFTsim package, theory and tools *J. High Energy Phys.* **12** 070
[37] ATLAS Collaboration 2015 Proposal for truth particle observable definitions in physics measurements *Technical Report* ATL-PHYS-PUB-2015-013 (Geneva: CERN) (available at: https://cds.cern.ch/record/2022743)
[38] Cacciari M, Salam G P and Soyez G 2012 FastJet user manual *Eur. Phys. J.* C **72** 1896
[39] Cacciari M, Salam G P and Soyez G 2008 The anti-k_t jet clustering algorithm *J. High Energy Phys.* **04** 063
[40] McLachlan G J, Lee S X and Rathnayake S I 2019 Finite mixture models *Annu. Rev. Stat. Appl.* **6** 355–78
[41] Maas A L, Hannun A Y and Ng A Y 2013 Rectifier nonlinearities improve neural network acoustic models *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*
[42] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization *3rd Int. Conf. for Learning Representations (San Diego, CA, USA)* (arXiv:1412.6980 [cs.LG])
[43] Wadia N S, Duckworth D, Schoenholz S S, Dyer E and Sohl-Dickstein J 2021 Whitening and second order optimization both make information in the dataset unusable during training, and can reduce or prevent generalization (arXiv:2008.07545 [cs.LG])
[44] Ellis J, Madigan M, Mimasu K, Sanz V and You T 2021 Top, Higgs, Diboson and electroweak fit to the standard model effective field theory *J. High Energy Phys.* **04** 279
[45] Brivio I, Bruggisser S, Geoffray E, Kilian W, Krämer M, Luchmann M, Plehn T and Summ B 2021 From models to SMEFT and back? (arXiv:2108.01094 [hep-ph])
[46] Wilks S S 1938 The large-sample distribution of the likelihood ratio for testing composite hypotheses *Ann. Math. Stat.* **9** 60–62
[47] Wald A 1943 Tests of statistical hypotheses concerning several parameters when the number of observations is large *Trans. Am. Math. Soc.* **54** 426–82
[48] Neyman J, Pearson E S and Pearson K 1933 IX. On the problem of the most efficient tests of statistical hypotheses *Phil. Trans. R. Soc.* A **231** 289–337
[49] Menary S B and Price D D 2021 Expressive Gaussian mixture model implementation code (available at: https://github.com/darrendavidprice/science-discovery/tree/master/expressive_gaussian_mixture_models)
[50] Menary S B and Price D D 2021 Expressive Gaussian mixture models for high-dimensional statistical modelling: simulated data and neural network model files (available at: https://dx.doi.org/10.48420/17136839)