ATLAS CONF Note

ATLAS-CONF-2024-015

28th October 2024

An implementation of Neural Simulation-Based Inference for Parameter Estimation in ATLAS

The ATLAS Collaboration

Neural Simulation-Based Inference (NSBI) is a powerful class of machine learning (ML)-based methods for statistical inference that naturally handles high-dimensional parameter estimation without the need to bin data into low-dimensional summary histograms. Such methods are promising for a range of measurements, including at the Large Hadron Collider (LHC), where no single observable may be optimal to scan over the entire theoretical phase space under consideration, or where binning data into histograms could result in a loss of sensitivity. This work develops an NSBI framework for statistical inference, using neural networks to estimate probability density ratios, which enables the application of NSBI to a full-scale LHC analysis. It incorporates a large number of systematic uncertainties, quantifies the uncertainty coming from finite training statistics, develops a method to construct confidence intervals, and demonstrates a series of intermediate diagnostic checks that can be performed to validate the robustness of the method. As an example, the power and feasibility of the method are demonstrated on simulated data for a simplified version of an off-shell Higgs boson couplings measurement in the four-leptons final states. This NSBI framework is an extension of the standard statistical framework used by LHC experiments and can benefit a large number of physics analyses.

ATLAS-CONF-2024-015
28 October 2024

Contents

1 Introduction

The precision measurement of theoretical parameters is a core element of the scientific program of experiments at the Large Hadron Collider (LHC). Such measurements typically rely on the method of

maximal likelihood, which assesses the likelihood of the observed data for a range of parameter values [1, 2]. While the likelihood cannot be analytically calculated, high-fidelity simulations of data under varying parameter values in conjunction with density estimation methods allow for estimation of the likelihood [3]. As the space of observational measurements grows to higher dimensionality, density estimation becomes very challenging and is often preceded by data reduction, which compresses the relevant information into a low-dimensional summary statistic, often a single observable, allowing for simple density estimation such as histogramming [1]. While significant effort goes into both the design of this summary observable and the choice of histogram binning, these simplifications may nonetheless result in a loss of sensitivity. This is of particular concern for problems where the kinematic distributions (the differential cross-sections) of different physics processes have a non-linear dependence on the parameter of interest,[1] as opposed to several signal strength measurements where the parameter of interest simply scales the distributions of the signal processes linearly. In these non-linear cases, no single observable may contain all the information required to maximise the sensitivity of the analysis over the full range of the theory parameter under consideration [4–6]. Examples of analyses for which this can have significant consequences for the sensitivity include the off-shell Higgs boson production measurements and effective field theory (EFT) measurements, where quantum interference introduces these non-linearities.

While histograms cannot effectively scale to high dimensions, neural networks have been shown to perform high-dimensional, unbinned density estimation in the context of parameter estimation at the LHC [4, 7–10] without the need to collapse information to a single observable. Referred to as *neural simulation-based inference (NSBI)*, these methods can dramatically enhance sensitivity in analyses where the histogram-related simplifications are unwarranted. However, for the application of NSBI at a particle physics experiment, crucial questions remain unanswered. How can a large number of nuisance parameters be incorporated? How can the uncertainty from limited Monte-Carlo (MC) simulated samples be quantified? Can neural networks produce robust likelihood ratios and confidence intervals when applied in a realistic experimental context, and how can their reliability be effectively tested? This note answers these questions and, therefore, enables the construction of a complete NSBI framework, together with diagnostic tools to address these questions.

The developed framework is an extension of the established statistical method at the LHC [1, 2], to an unbinned, multi-dimensional setting, where neural networks are used to estimate likelihood ratios between hypotheses. It accounts for linear as well as non-linear dependence of physics observables as a function of theory parameters, which is crucial to building optimal test statistics. The challenge of *model-misspecification* in NSBI, where the simulations have systematic differences from real data, is addressed by the introduction of nuisance parameters representing systematic uncertainties, and by testing the modelling of these systematic uncertainties themselves, in a similar way to how it is done for traditional analysis techniques in particle physics [11, 12]. This method can leverage an analytical factorisation of contributions from different physics processes to the full likelihood, which is possible in a majority of analyses at the LHC. To demonstrate the power and applicability of the method, an example use case is shown on samples describing the gluon-gluon fusion processes simulated for the ATLAS off-shell Higgs boson production measurement in the four leptons final states [13]. The improvement in sensitivity compared to a histogram-based method, while accounting for systematic uncertainties, comes from the optimisation of the analysis over the entire range of the theory parameter, which cannot be achieved with the use of only a single observable for all regions of the theory space, and an additional improvement

[1] In particular, there exists no representation of the parameter of interest (such as by taking the square or the square root) under which the probability densities corresponding to each physics process in the analysis have either a constant or linear dependence on the parameter of interest.

comes from the unbinned nature of the method. The robustness tests that are needed to build a reliable likelihood ratio model using neural networks are also demonstrated.

Since this note builds upon the established statistical methods at the LHC, it focuses on the tools and concepts necessary to extend this for the high-dimensional and unbinned NSBI analysis. The note is organised as follows. Section 2 reviews the concepts of neural simulation-based inference as well as modifications that are developed for a practical application at the LHC. Section 3 introduces the context of the off-shell Higgs boson production measurement, which is the example analysis used to demonstrate the developed method. Section 4 then describes the diagnostic tools used to validate the trained models, Section 5 extends the method to incorporate systematic uncertainties, Section 6 describes how to build confidence intervals for NSBI, and Sec 7 demonstrates the gain in sensitivity. The conclusion is presented in Section 8 with a discussion of opportunities and challenges.

2 Neural simulation-based inference

Neural simulation-based inference techniques are of interest to a wide range of scientific fields for parameter estimation in cases where likelihoods are either intractable or computationally expensive to evaluate. When high-fidelity simulators can provide samples drawn from these likelihoods, neural networks are capable of learning the underlying density of these simulated samples and can be used to approximate a likelihood ratio [7], the likelihood itself [14], or, in the context of Bayesian inference, the posterior [15]. These techniques have several potential applications in the physical sciences [3, 4] and can be used, for instance, to study galaxy clustering [16], probe the interior of neutron stars from telescope data [17], or analyse data from gravitational wave detectors [18].

This section reviews the core principles of classifier-based NSBI and subsequently discusses a framework in which the method can be made robust and numerically stable. Nuisance parameters will be introduced to this framework in Section 5.

2.1 Classifiers as probability density ratio estimators

Neural network classifiers can be used to discriminate between two hypotheses μ_0 and μ_1 by minimizing the binary cross-entropy loss function,

$$\mathcal{L}[s] = -\frac{1}{(\sum_{i=1}^{N} w_i)} \sum_{i=1}^{N} w_i \cdot [y_i \log s(x_i) + (1 - y_i) \log(1 - s(x_i))] \tag{1}$$

where the sum is over N events x_i sampled from probability density functions $p(x_i|\mu_0)$ or $p(x_i|\mu_1)$ with weights w_i and assigned labels $y_i = 0$ or $y_i = 1$, respectively, and $s(x_i)$ is the classifier decision function. The event x_i is described by a vector of observables.

The optimal decision function (in the infinite sample limit, i.e. as $N \to \infty$), which minimizes the binary cross entropy function, is given by [7, 19]

$$s(x_i) = \frac{p(x_i|\mu_1) \cdot \nu(\mu_1)}{p(x_i|\mu_0) \cdot \nu(\mu_0) + p(x_i|\mu_1) \cdot \nu(\mu_1)}, \tag{2}$$

where $\nu(\mu_0)$ and $\nu(\mu_1)$ are the expected number of events for each hypothesis.

In high-energy physics, training datasets are usually taken from simulated Monte Carlo (MC) samples generated according to the two hypotheses. These events are weighted, and the weights may take negative values. Typically, the weights are scaled to perform the training with *balanced samples*, i.e., $\sum_{y=0} w_i = \sum_{y=1} w_i$, which tends to improve the convergence of the neural network to the optimal classifier. For the case of training a classifier between two hypotheses, this choice simplifies the optimal classifier to

$$s(x_i) = \frac{p(x_i|\mu_1)}{p(x_i|\mu_0) + p(x_i|\mu_1)}. \tag{3}$$

The *likelihood-ratio trick* [7] can be used to write the probability density ratio between the hypotheses μ_0 and μ_1 for a single event x_i as:

$$r(x_i; \mu_1, \mu_0) = \frac{p(x_i|\mu_1)}{p(x_i|\mu_0)} = \frac{s(x_i)}{1 - s(x_i)}. \tag{4}$$

The estimator $\hat{r}(x_i; \mu_1, \mu_0)$ is obtained from a neural network estimator $\hat{s}(x_i)$ of the optimal decision function. This relation enables the estimation of the probability density ratio between two values of a parameter(s) of interest for individual events without the need for dimensionality reduction or histograms. The probability density ratio for the dataset is constructed by taking the product of probability density ratios for individual events, which can be combined with the total rate information to compute the likelihood ratio and the likelihood ratio can be used to build a test statistic comparing the two hypotheses μ_0 vs μ_1. This trick has, for instance, been used to obtain probability density ratios per event, in data-driven background estimation [20] and unfolding [21] in the ATLAS experiment.

The task of parameter estimation is a composite hypothesis test, but can be performed by comparing the likelihood for two values of the parameter at a time. While it may appear that parameter estimation would require training a separate classifier for each pair of hypotheses being compared, in practice there are more elegant solutions. A single *parameterised network* may be trained to learn a conditional decision function that varies with the hypothesis under consideration (i.e. the different values of the theory parameter) [7, 22]. However, the parametric dependence of a test statistic can often be analytically expressed in terms of the parameter(s) of interest and a finite number of likelihood ratios estimated from binary classifiers. Such a formalism eliminates the need for a network to learn the parametric dependence.

For example, consider the case of a search where μ is the signal strength to be measured, in an analysis with no interference between the signal and background processes,

$$p(x_i|\mu) = \frac{\mu \cdot \nu_S \; p(x_i|S) + \nu_B \; p(x_i|B)}{\mu \cdot \nu_S + \nu_B}, \tag{5}$$

with S representing the signal processes, B the background processes, ν_S the total expected signal rate and ν_B the total expected background rate. One can train a classifier to estimate a decision function separating signal from background events (using balanced class weights),

$$s(x_i) = \frac{p(x_i|S)}{p(x_i|B) + p(x_i|S)}, \tag{6}$$

and then compute the per-event probability density ratio,

$$r(x_i; S, B) = \frac{p(x_i|S)}{p(x_i|B)} = \frac{s(x_i)}{1 - s(x_i)}. \tag{7}$$

This can subsequently be scaled as required to construct the likelihood ratio,

$$\frac{p(x_i|\mu)}{p(x_i|\mu = 0)} = \frac{\mu \cdot \nu_S \; p(x_i|S) + \nu_B \; p(x_i|B)}{(\mu \cdot \nu_S + \nu_B) \; p(x_i|B)} = \frac{1}{(\mu \cdot \nu_S + \nu_B)} \Big(\mu \cdot \nu_S \; r(x_i; S, B) + \nu_B\Big), \tag{8}$$

where ν_S and ν_B are estimated from simulation. The output of a single, unparameterised classifier, therefore, is a *sufficient summary statistic*, meaning that it contains all the information necessary to perform hypothesis tests over a range of μ, for problems where μ linearly scales the distributions. This is guaranteed by the Neyman-Person lemma, which states that the likelihood ratio is the optimal observable when comparing two hypotheses. To use this classifier output directly as an estimate of the probability density ratio, stringent requirements would need to be placed on the quality of this estimation. Alternatively, the output of the classifier can be treated as a high-level observable particularly sensitive to μ. For this reason, it is often used as the final observable in histogram-based signal strength measurements. In such analyses, the likelihood is traditionally computed in each bin of a histogram using an analytically known Poisson probability model, where the expected number of events is obtained from simulation and the observed number of events from data [1]. This prescription for the design of a single observable that acts as a sufficient summary statistic only works for problems linear in μ. The rest of this section develops a more general framework to build a test statistic that captures information available in the higher-dimensional view of the data, using the output of a few classifiers. It describes a method to factorise the problem of estimating likelihood ratios into a set of simpler estimation tasks and improve the robustness of the estimation.

2.2 Factorisation into a search-oriented mixture model

When the hypotheses being tested can be decomposed into several components, the learning task can be factorised into a series of simpler classification tasks [7]. Further, if the only free parameters to be measured can be written as coefficients of the mixture model, the individual classifiers no longer need to be parameterised in the parameter(s) of interest (e.g. a signal strength μ), since the relation is explicitly known. This reduces the burden of validating the interpolation capabilities of the likelihood ratio estimation from validating over the entire theory parameter space, to simply validating the performance of the small number of classifiers. If every classifier is well-trained and well-calibrated, then their combination, too, may be expected to remain well-behaved, although this must be explicitly verified.

For an LHC analysis, this decomposition can use different physics processes that give rise to the same final state, each with a coefficient that is some function of the parameter(s) of interest. If the decomposition is into C different components, representing C different physics processes, then the probability density is,

$$p(x_i|\mu) = \frac{1}{\nu(\mu)} \sum_j^C f_j(\mu) \cdot \nu_j \; p_j(x_i), \tag{9}$$

where $p_j(x_i)$ is the probability density for the event x_i corresponding to the process j, ν_j the inclusive rate for that process with μ at the Standard Model value, and $\nu(\mu) = \sum f_j(\mu) \cdot \nu_j$. Here μ could represent multiple theory parameters, and this formalism accommodates multiple independent parameters of interest. For most LHC analyses, the full dependence on μ can be captured using only the coefficients $f_j(\mu)$ and

the total rate $\nu(\mu)$, where the coefficients $f_j(\mu)$ are known from theory [4]. If the dependence on the parameter(s) of interest is not analytically known[2], a parameterised network can be trained instead to directly estimate $p_j(x_i|\mu)$ [7]. This note defines a *search-oriented* mixture model, which is the probability density ratio between a hypothesis and a reference,

$$\frac{p(x_i|\mu)}{p_{\text{ref}}(x_i)} = \frac{1}{\nu(\mu)} \sum_{j}^{C} f_j(\mu) \cdot \nu_j \, \frac{p_j(x_i)}{p_{\text{ref}}(x_i)}, \tag{10}$$

expressed using only a finite number of μ-independent probability density ratios, $p_j(x_i)/p_{\text{ref}}(x_i)$, which are estimated using classifiers. While there is freedom to make any choice for the reference, this note defines it as a combination of signal processes,

$$p_{\text{ref}}(x_i) = \frac{1}{\sum_k \nu_k} \sum_{k}^{C_{\text{signals}}} \nu_k \; p_k(x_i), \tag{11}$$

with C_{signals} as the number of signal processes. This definition ensures that the denominators in Eq. 10 have support over the entire signal region of an analysis, which is the region that is sensitive to the signal processes. Here, p_{ref} is defined to be independent of μ, which allows the construction of the final profile likelihood ratio that is independent of p_{ref} (see Section 5). The term p_{ref} contributes only as a constant offset towards $\log p(x_i|\mu)$, which can be ignored in the maximisation of the (log-)likelihood.

The search-oriented mixture model overcomes issues of numerical instability that may arise in alternative mixture model formulations. Additionally, the pre-selected region for the analysis must be defined to ensure $p_{\text{ref}}(x_i) > 0$ throughout the region. This definition of p_{ref} ensures that no signal-sensitive parts of the phase space need to be removed. Further, this choice of p_{ref} also aids in the sample-efficient training of the individual classifiers. Finally, it may be convenient to define p_{ref} such that it can be represented using simulated samples with only positive weighted events. This simplifies the procedure to construct confidence intervals, which will be described in Section 6.

2.3 Robust Estimators with Ensembling

In a traditional analysis where a classifier is employed solely for constructing a sensitive observable and where probability density estimation is performed with a histogram, an imperfect training leads to a suboptimal observable and a slightly less sensitive analysis. However, it does not lead to an ill-behaved test statistic, introduce inaccuracies in the measured confidence intervals or introduce biases in the maximum likelihood estimate of the parameter(s) of interest. This is because the likelihood of event counts per bin in a histogram can be computed exactly using the Poisson probability density function. In NSBI, the probability density ratios are instead estimated using networks, and therefore, ensuring the high quality of these estimates is imperative. Since an individual classifier may not perfectly estimate the decision function $s(x_i)$, a series of steps is described to ensure that the estimator $\hat{s}(x_i)$ is well-behaved (as determined by the diagnostic tests described in Sec. 4). One possibility is to calibrate $\hat{s}(x_i)$ using simulated samples [7]; however, achieving accurate and continuous calibration in practice can be technically challenging. Instead, an ensemble of networks may be trained [23] on bootstrapped samples of the training data, and their average response used to construct a robust estimation of likelihood ratios. The bootstrapping can be

[2] Factorising out the dependence on the parameter of interest(s) is possible for signal strength measurements but may not be possible in certain other instances, such as mass measurement.

implemented either through resampling or using Poission perturbations to the event weights that correspond to statistical fluctuations [24]. This approach helps account for the variance between individual networks, originating from the random initialisation of weights and the finite statistics of the training samples. A similar method has previously been used for neural-network-based data-driven background estimation [20] and unfolding of differential cross-sections [21] in ATLAS. Examining classifier and ensemble performance across different parts of the observable phase space can guide decisions about neural network architecture optimisation and data pre-processing. Iterative optimisation is essential to achieve a high level of accuracy in likelihood-ratio estimation. Multiple diagnostic tests help determine whether the level of precision desired from the ensembles has been achieved, which are discussed in Section 4. Ultimately, the full framework must be tested on simulated samples at different values of the parameter(s) of interest to ensure that reliable results with the desired precision are consistently produced over the entire parameter range. Since the ensembles are trained on bootstrapped samples, it is natural to use the spread in their predictions to quantify the uncertainty due to the finite training data.

3 Example use case: ggF off-shell Higgs boson production

The developed NSBI framework is demonstrated using a subset of simulated samples originally generated for an off-shell Higgs boson production measurement in the $H \to ZZ \to 4\ell$ decay channel. The full context of the analysis is described in Ref. [13], only the details relevant to NSBI will be summarised below. Only a subset of the physics processes and systematic uncertainties from the original analysis are considered for this demonstration.

When the quantum interference between signal and background processes is negligible, a single observable that optimally separates signal from background contains all the information necessary to perform optimal hypothesis tests over the full range of signal strength values (see Eq. 6). However, this is no longer true when the interference cannot be ignored, and therefore does not apply to the off-shell Higgs boson analysis, where there is considerable destructive interference between the signal and background processes. In this case, the kinematic distributions change non-linearly with the signal strength μ, and Ref. [5] demonstrates that the use of NSBI can fully account for these non-linear effects.

The simulated samples used in the study by ATLAS in Ref. [13] include those for the $gg \to H \to ZZ \to 4\ell$ signal-only (S) process, $gg \to ZZ \to 4\ell$ background-only (B) process, and the combined simulation including interference effects $gg \to (H) \to ZZ \to 4\ell$ (SBI_1, where the subscript indicates that μ was set to 1 for the simulation). These samples from the gluon-gluon fusion (ggF) production channel will be re-used for the demonstrations in this note.The full ggF probability model can be expressed as[3]

$$p_{\text{ggF}}(x|\mu) = \frac{1}{\nu_{\text{ggF}}(\mu)} \left[(\mu - \sqrt{\mu})\, \nu_S\, p_{\text{S}}(x) + \sqrt{\mu}\, \nu_{\text{SBI}_1}\, p_{\text{SBI}_1}(x) + (1 - \sqrt{\mu}) \nu_{\text{B}}\, p_{\text{B}}(x) \right], \tag{12}$$

where $\nu_{\text{ggF}}(\mu) = (\mu - \sqrt{\mu})\, \nu_S + \sqrt{\mu}\, \nu_{\text{SBI}_1} + (1 - \sqrt{\mu}) \nu_{\text{B}}$. The contribution from the interference (I) is defined as $p_{\text{I}} = p_{\text{SBI}_1} - p_{\text{B}} - p_{\text{S}}$, and it is this inference effect that introduces the non-linearity in μ. This formulation of the probability model follows from Ref. [13]. The terms $p_{\text{ggF}}(x|\mu)$ and $\nu_{\text{ggF}}(\mu)$ are

[3] In principle, a coupling modifier parameter that scales the signal amplitude could be a complex number, which would lead to a phase contributing to the interference term in the cross-section computation. This would require the measurement of two independent parameters of interest, which can be done with NSBI. In this analysis however, the modifier $\sqrt{\mu}$ is assumed to be a positive real number, and therefore only the inference of one parameter of interest μ is required.

functions of μ while $p_{\mathrm{SBI}_1}(x)$ and ν_{SBI_1} are terms where μ is fixed to 1. For simplicity, the ggF subscripts will be suppressed henceforth. The definition for the reference in Eq. 11 leads to $p_{\mathrm{ref}} = p_{\mathrm{S}}$ for this example, and the search-oriented mixture model from Eq. 10 becomes

$$\frac{p(x|\mu)}{p_{\mathrm{S}}(x)} = \frac{1}{\nu(\mu)} \left[(\mu - \sqrt{\mu})\, \nu_S + \sqrt{\mu}\, \nu_{\mathrm{SBI}_1} \frac{p_{\mathrm{SBI}_1}(x)}{p_{\mathrm{S}}(x)} + (1 - \sqrt{\mu}) \nu_{\mathrm{B}} \frac{p_{\mathrm{B}}(x)}{p_{\mathrm{S}}(x)} \right]. \tag{13}$$

This can be constructed using two ensembles, the first to estimate $p_{\mathrm{SBI}_1}(x)/p_{\mathrm{S}}(x)$ and the second $p_{\mathrm{B}}(x)/p_{\mathrm{S}}(x)$. The event section strategy follows Ref. [13] and additionally uses a multi-variate-analysis-based discriminant, similar to the discriminant designed in Ref. [13], used here only to define the signal and control regions. The control region is a part of the data without sensitivity to the signal which can be used to validate the background model and potentially fit background-related nuisance parameters. The rest of this section will describe input features and architecture for the networks trained for these tasks, and the systematics model considered in this demonstration.

3.1 Input features

With sufficient training statistics, deep neural networks can learn only from low-level input features such as the four-momenta of all observed final state particles. They can then automatically capture all higher-level correlations.However, in the regime of limited simulated samples, as is often the case at LHC experiments, there is a benefit to using a set of physics-motivated high-level observables that completely describe the observed final state.

The set of observables used in this demonstration to train the networks are described in Table 1. The Higgs decay to Z bosons is described in the data with seven kinematic observables $\cos\theta^*$, $\cos\theta_1$, $\cos\theta_2$, ϕ_1, ϕ, m_{Z1} and m_{Z2}. These have traditionally been used as inputs to construct a discriminant based on matrix-element calculations, and are known to contain all relevant information to distinguish the Higgs boson signal process from the background [25]. Combined with the production kinematic observables $m_{4\ell}$, $p_T^{4\ell}$ and $\eta_{4\ell}$, these observables can be used to calculate the four-momenta of all final-state leptons in the $ZZ \to 4\ell$ decay channel. Further details on the observables and event selection can be found in Ref. [13].

Table 1: List of input variables for the neural network. For additional details, see Ref. [13].

Variable	Definition
Production Kinematics	
$m_{4\ell}$	Four-lepton invariant mass
$p_T^{4\ell}$	Four-lepton transverse momentum
$\eta^{4\ell}$	Four-lepton pseudo-rapidity
Decay Kinematics	
m_{Z1}	Z_1 mass
m_{Z2}	Z_2 mass
$\cos\theta^*$	Higgs boson decay angle
$\cos\theta_1$	Z_1 boson decay angle
$\cos\theta_2$	Z_2 boson decay angle
ϕ	Angle between Z_1, Z_2 bosons decay planes
ϕ_1	Z_1 decay plane angle

3.2 Network architecture and training

The classifiers trained in this demonstration are all feed-forward dense networks and comprise 5 hidden layers with 1000 nodes each and a swish activation [26], followed by an output layer with a single node and a sigmoid activation. The events were split into train and test sets using the k-fold method with $k = 10$, and a bootstrapped sample was generated from the training set to train each network in an ensemble. A weighted binary cross-entropy loss function that accounts for event weights is used to train the networks with the Nadam optimiser [27] in TensorFlow [28]. The training required large-scale GPU infrastructure [29], consisting of several hundred Nvidia T4 and Nvidia A100 GPUs. The final networks used in this note, with 500 networks used in an ensemble, required approximately 4000 GPU hours to train.

3.3 Systematic uncertainties

At the LHC systematic uncertainties on the modelling of physics processes are often considered in terms of their effect on the shape of distributions and on the expected inclusive rates (overall normalisation). This convention can be carried forward to NSBI.[4] Two systematic uncertainties from the original study in Ref. [13] are considered to showcase the handling of nuisance parameters that either modify both the shape $p(x|\mu)$ and inclusive rate $\nu(\mu)$ of distributions or only the inclusive rate $\nu(\mu)$. These are:

- **ggF higher-order QCD uncertainty**: the uncertainty on the missing QCD higher-order corrections to the ggF processes in perturbation theory, which modifies both the shapes of the kinematic distributions and the inclusive rates.
- **Luminosity uncertainty**: the uncertainty on the integrated luminosity measurement of ATLAS. This affects only the inclusive rates.

This setup is used to demonstrate an NSBI analysis in Section 7.

4 Diagnostics

The precise estimation of likelihood ratios is crucial for a robust final result, and therefore the classifiers used in the framework described in Section 2 require additional scrutiny compared to classifiers used in traditional histogram-based analyses. In addition to traditional visualisations of classifier performance, such as the receiver operating characteristic (ROC) curve and the distribution of the classifier's output, this section describes a list of additional diagnostic tools that are essential for the validation of the likelihood ratio estimation at the level of detail required for NSBI analysis.

4.1 Reweighting closures

If an ensemble has estimated the likelihood ratio between two classes a and b correctly, it can be used to reweight samples from one class to another. Since $p(x_i|a) = p(x_i|b)r(x_i; a, b)$, the distribution of samples

[4] Although existing conventions such as the factorisation of systematic effects into shape and inclusive rate have been carried over to NSBI in this work, any future change in the conventions may also be carried over, as these choices are not fundamental components of the method.

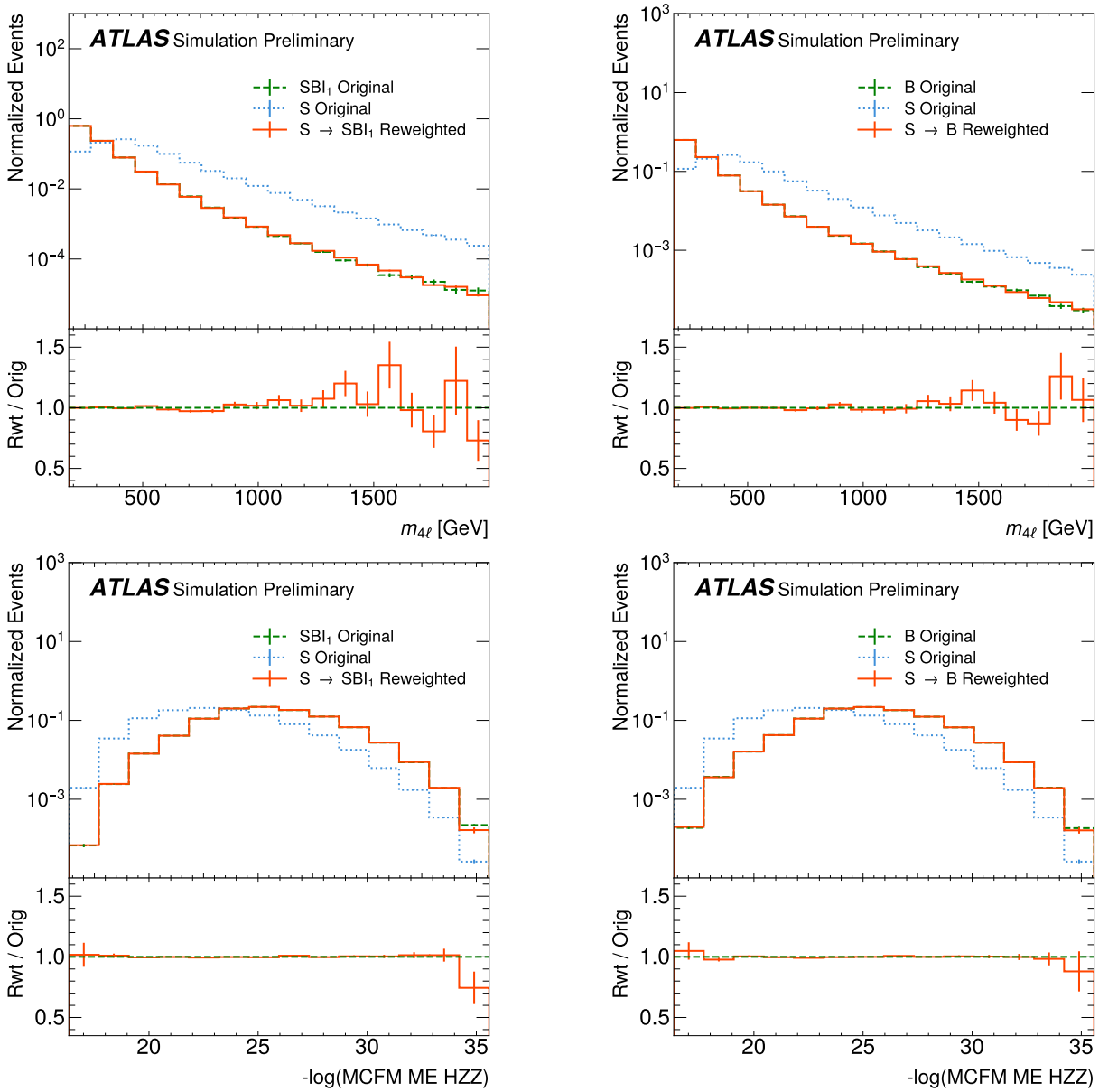

Figure 1: One-dimensional reweight closure diagnostic with $m_{4\ell}$ and a high-level observable that represents the squared matrix-element for the $gg \to H \to ZZ \to 4l$ process from reconstructed quantities computed using MCFM [30]. The former is an example diagnostic for an observable directly used in the network training, and the latter is an example diagnostic of the network's ability to learn high-level physics observables that are not used directly for training. The original reference sample (blue, dashed), is reweighted (orange, solid) using the likelihood ratio estimated with ensembles to match the target (green, dashed). The lower panel shows the ratio between the reweighted reference sample and the target sample.

from b reweighted as $w_i \cdot \hat{r}(x_i; a, b)$, where w_i indicates the normalised simulation weight of an event from class b, should match the distribution of samples from a,

$$\hat{r}(x_i; a, b) \cdot p(x_i|b) \sim p(x_i|a). \tag{14}$$

Any discrepancies indicate a failure of the ensemble to correctly estimate the likelihood ratio in a given part of the phase space. The normalisation of the weights is necessary to focus on the differences in the shape of the distributions. These comparisons can be made by taking one- or two-dimensional projections of the full input phase space. Examples of good reweight closure are shown in Figure 1 to validate the p_{SBI_1}/p_S and p_B/p_S, using a one-dimensional histogram of the $m_{4\ell}$ observable. The closure is also shown using high-level observables that were not explicitly used in the training, in this case, a matrix-element-based observable that is known to be good summary statistic [31].

An independent classifier (such as a deep neural network or a boosted decision tree) can be trained to separate events from class a and the reweighted events from b to identify any high-dimensional mismatches between the distributions [7]. A perfect reweighting would lead to the failure of this independent classifier, indicated by an area under the ROC curve (AUC) of 0.5. Such *classifier tests* have previously been used to assess the performance of generative models in HEP [32, 33].

A related tool, the *normalisation closure*,

$$\sum_{i \in a} w_i \frac{p_b(x_i)}{p_a(x_i)} = \sum_{i \in b} w_i, \tag{15}$$

should also be explicitly verified. This simple test can fail if the numerical precision of the training and inference are not enough to correctly describe events with $s(x_i) \approx 0$ or $s(x_i) \approx 1$.

4.2 Calibration closure

Another useful visualisation is the calibration curve. If the predicted relative probability $\hat{s}(x_i)$ from the ensembles is binned, then the fraction of events in each bin from the first class provides an empirical MC estimate of the mean $s(x_i)$ in that bin. In the ideal case, the binned estimate would match the ensemble estimate in each bin, therefore a well-calibrated classifier produces a diagonal line along $y = x$. Figure 2 shows the calibration curves for the estimators of $p_{\text{SBI}_1}/\left(p_S + p_{\text{SBI}_1}\right)$ and $p_{\text{SBI}_1}/\left(p_S + p_{\text{SBI}_1}\right)$ using ensemble predictions. The binned estimates are compared to the estimates from neural network ensembles.

4.3 Spread in ensemble predictions

An ensemble of networks is trained for each classification task, as discussed in Section 2.3. The spread in their predictions for the same event reveals the type of events for which the limitations of training statistics come into play, and this can inform the optimisation of the training strategy. Examples of this spread are shown in Figure 3. A wider spread indicates a larger ensemble uncertainty. The propagation of these uncertainties on the estimated probability density ratios, however, requires careful consideration of their correlated impact on the final parameter estimation. This is described in Section 5.3.

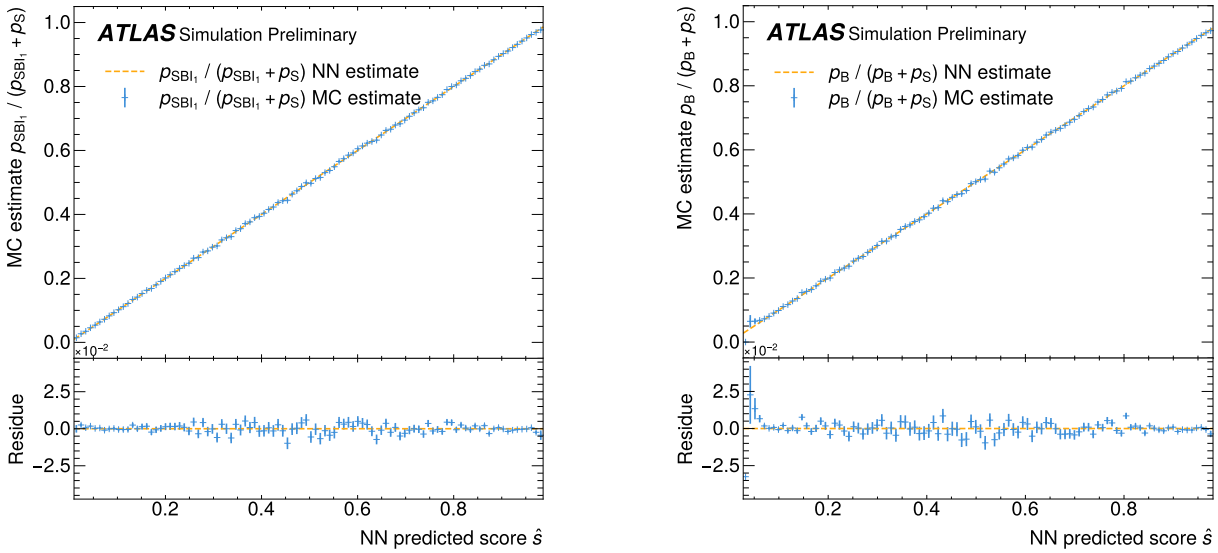

Figure 2: Calibration curve comparing ensemble estimated $\hat{s}(x_i)$ with the expected value from binned MC simulated samples, for the validation of the $p_{\mathrm{SBI_1}}/p_{\mathrm{ref}}$ (left) and $p_{\mathrm{B}}/p_{\mathrm{ref}}$ (right) probability density ratio estimations. The absolute residuals are shown in the bottom panel.

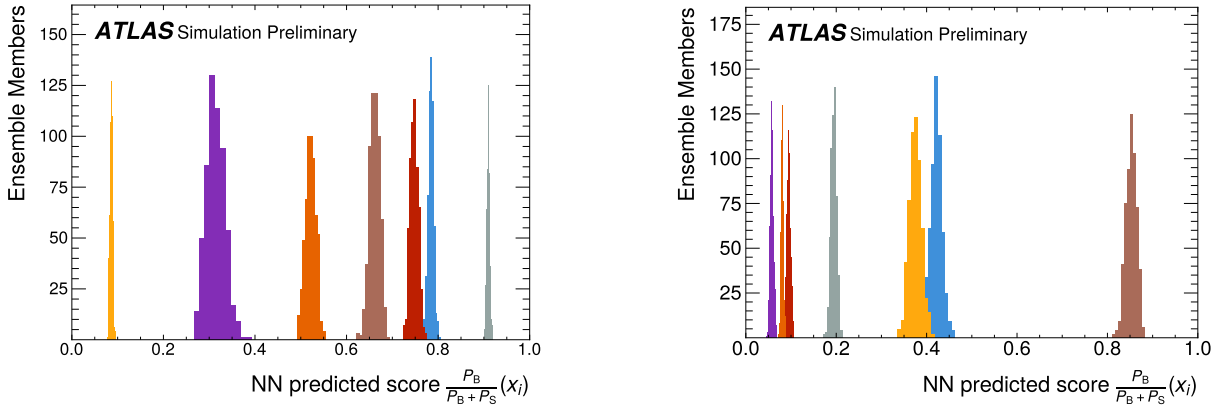

Figure 3: The distribution of neural network output for example events (in different colours) from an ensemble of classifiers trained to separate B from S samples, evaluated on seven example events from B (left) and seven example events from S (right). A wider spread indicates a larger uncertainty on that event from the ensemble.

4.4 Additional diagnostics

Additional diagnostic plots may be used to explore the performance of the method, motivated by analysis-specific considerations. In addition to validating the individual estimated probability density ratios $p_j(x_i)/p_{\mathrm{ref}}(x_i)$ that form the mixture model, the combined probability density ratio $p(x_i|\mu)/p_{\mathrm{ref}}(x_i)$ can also be validated using the discussed diagnostic tools. The inference can in addition be validated on independent samples simulated at values of μ that were not used for training. The response of these ensembles in data and in MC simulation must also be compared in the control regions. The final performance of the analysis method can also be verified on simulated datasets across a range of the parameter of interest to ensure that the correct maximum likelihood estimate is consistently obtained.

As a cross-check, the analysis method can be tested on samples simulated with a different event generator, on samples simulated with a shifted value of a nuisance parameter, or with signal injections. In addition, this method provides interpretable, per-event quantities to examine. These are the estimated probability density ratios between different theory hypotheses for a given event. These quantities can be studied in detail as functions of several observables to understand the sub-category of events that influence the overall test statistic in favour of one hypothesis over another. A few examples are discussed in Section 7.

5 Systematic uncertainties

A major challenge in applying NSBI to LHC data has been addressing systematic uncertainties. In a traditional histogram-based analysis, these are typically parameterised by nuisance parameters denoted collectively as a vector α and the nominal point as a vector α^0. Each individual nuisance parameter is represented by α_k (where k spans all the nuisance parameters in an analysis), their nominal values α_k^0. These values may be constrained by an auxiliary measurement, providing, in addition, an uncertainty δ_k on the nominal value.

In principle, classifiers can be conditioned on nuisance parameters in an analysis to propagate uncertainties through to the final inference step [7, 34], but in practice, this is not feasible for all nuisance parameters in an analysis. First, due to computational costs, samples are typically generated by varying a single nuisance parameter at a time, with no training samples available where multiple parameters vary simultaneously. Second, only three sets of samples are typically available per nuisance parameter: one at the nominal value α_k^0 and the others at variations $\alpha_k^- = \alpha_k^0 - \delta_k$ and $\alpha_k^+ = \alpha_k^0 + \delta_k$. These sets are insufficient for a network to learn the full parametric dependence. Finally, validating the interpolation capabilities of a classifier across all regions of this high-dimensional space of nuisance parameters would be challenging even if the classifier were parameterised on all of them.

Instead, this note extends the systematics framework already in place for histogram-based analyses to an unbinned multi-dimensional setting, and incorporates it into NSBI. While in a histogram-based analysis the impact of a nuisance parameter is estimated per bin, for NSBI it is estimated per event, and the interpolation between nuisance parameter values is also performed using traditional methods, rather than relying on the networks to learn it. Moreover, the impact of systematic uncertainties from independent sources is treated independently, following the standard practice at the LHC.

5.1 Nuisance parameters in the likelihood ratio function

In a histogram-based analysis at the LHC, the impact of systematic uncertainties is typically propagated into the likelihood using *vertical interpolation* [2]. Here the impacts of different nuisance parameters on the measured cross-section are considered to be independent and also to be independent of the parameter(s) of interest,

$$\nu_j(\alpha) = \nu_j(\alpha^0) \prod_k^{N_{\text{syst}}} G_j(\alpha_k) \tag{16}$$

for N_{syst} nuisance parameters with $G_j(\alpha_k) = \nu_j(\alpha_k)/\nu_j(\alpha_k^0)$. The functions $g_j(\alpha_k)$ are chosen to smoothly interpolate between their three known values at the points α_k^-, α_k^0 and α_k^+, which are determined from the

available simulations [2]. The choice of a differentiable interpolation function facilitates the computation of pulls and impacts, detailed in Section 5.4.

Extending this formalism and the corresponding assumptions to a per-event approach, Eq. 10 can be updated to incorporate nuisance parameters as

$$\frac{p(x_i|\mu,\alpha)}{p_{\text{ref}}(x_i)} = \frac{1}{\nu(\mu,\alpha)} \sum_j^C f_j(\mu) \cdot \nu_j \, \frac{p_j(x_i)}{p_{\text{ref}}(x_i)} \prod_k^{N_{\text{syst}}} G_j(\alpha_k) \; g_j(x_i,\alpha_k) \tag{17}$$

with $\nu(\mu,\alpha) = \sum_j^C f_j(\mu) \cdot \nu_j \cdot \prod_k^{N_{\text{syst}}} \cdot G_j(\alpha_k)$. The contribution to the per-event probability density ratio from each nuisance parameter comes from $g_j(x_i,\alpha_k) = p_j(x_i,\alpha_k)/p_j(x_i)$, where $p_j(x_i)$, $f_j(\mu)$, ν_j and $p_{\text{ref}}(x_i)$ are defined at α^0.

As with the functions $G_j(\alpha_k)$, the functions $g_j(x_i,\alpha_k)$ are chosen to interpolate between the three known values at α_k^0 and $\alpha_k^\pm$ for each event, using the same interpolation strategy. For the nominal case $g_j(x_i,\alpha_k^0) = 1$, and for $\alpha_k^\pm$, the probability density ratios $g_j(x_i,\alpha_k^\pm)$ are estimated per event by training ensembles of classifiers. These classifiers are trained to separate nominal samples $p_j(x_i)$ from systematic variation samples $p_j(x,\alpha_k^\pm)$, with one ensemble per physics process, per nuisance parameter, and per variation. Once the functions $g_j(x_i,\alpha_k^\pm)$ are determined, these can even be used to replace the alternative simulations altogether in an analysis [13]. The diagnostic tests described in Section 4 are also useful tools to validate these networks, although they can be less illuminating if the systematic variation is very small (leading to $s(x_i) \approx 0.5$).

NSBI not only constructs a more sensitive analysis in the entire phase space of μ, but also in the space of α [34]. As with histogram analyses, it is important to ensure that an NSBI analysis does not overconstrain a nuisance parameter. This might indicate that the modelling of the systematic uncertainty is oversimplified or the fit is exploiting aspects of the systematic uncertainty model that are not known well, for instance in the case of two-point theory uncertainties [35]. An analysis of the pulls on the nuisance parameters and impacts (described further in Section 5.4), and the use of alternative modelling of the systematic uncertainties (such as splitting the nuisance parameter into independent sub-components) can reveal such issues, or the use of more recently developed methods to analyse the effect of systematic uncertainties [36]. Further, LHC experiments often quantify the uncertainties on the systematic uncertainties themselves, and on models of correlation between different components of systematic uncertainties [11, 12]. Such challenges are often discussed in the context of model misspecification in ML literature.

5.2 The profile log-likelihood ratio

The full test statistic based on a profile log-likelihood ratio [37] can be constructed from Eq. 17 by considering all events in the observed data, adding a Poisson term corresponding to the total rate and Gaussian constraint factors for the nuisance parameters. If N_{data} is the number of events in observed data $\mathcal{D}$,

$$\frac{L_{\text{full}}(\mu,\alpha|\mathcal{D})}{L_{\text{ref}}(\mathcal{D})} = \text{Pois}(N_{\text{data}}|\nu(\mu,\alpha)) \prod_i^{N_{\text{data}}} \frac{p(x_i|\mu,\alpha)}{p_{\text{ref}}(x_i)} \prod_k \text{Gaus}(a_k|\alpha_k,\delta_k), \tag{18}$$

where the global observables a_k and δ_k are the values of the auxiliary measurements and their associated uncertainty, which are used to constrain the source of systematic uncertainty associated with the nuisance parameter α_k. $L_{\text{ref}}(\mathcal{D}) = \prod_i^{N_{\text{data}}} p_{\text{ref}}(x_i)$.

If the nuisance parameter is unconstrained, the corresponding constraint factor is suppressed. An important case of unconstrained nuisance parameters is data-driven normalisation parameters.

The profiling step involves an unconditional and a conditional maximum likelihood estimation of Eq. 18 (keeping the dependence on $\mathcal{D}$ implicit),

$$(\widehat{\mu}, \widehat{\alpha}) = \underset{\mu,\alpha}{\text{argmax}} \frac{L_{\text{full}}(\mu, \alpha)}{L_{\text{ref}}}$$
$$\widehat{\widehat{\alpha}}(\mu) = \underset{\alpha}{\text{argmax}} \frac{L_{\text{full}}(\mu, \alpha)}{L_{\text{ref}}}.$$

Note that since L_{ref} has been defined without any dependence on μ or α, it does not affect the position of the maxima. The test statistic is constructed by taking the ratio of Eq. 18 at these two points. The dependency on L_{ref} cancels out and the traditional profile log-likelihood ratio is recovered,

$$t_\mu = -2 \ln \left(\frac{L_{\text{full}}(\mu, \widehat{\widehat{\alpha}}(\mu))}{L_{\text{full}}(\widehat{\mu}, \widehat{\alpha})} \right). \tag{19}$$

Further, the use of likelihood ratios instead of likelihoods does not prevent the combination of NSBI and histogram-based analyses. The combination can be written as

$$\frac{L_{\text{comb}}(\mu, \alpha)}{L_{\text{ref}}} = \frac{L_{\text{full}}(\mu, \alpha)}{L_{\text{ref}}} \, L_{\text{hist}}(\mu, \alpha). \tag{20}$$

The test statistic is again independent of L_{ref}, which appears as a constant offset in the log-likelihood.

5.3 Effects from finite Monte Carlo samples

When likelihood ratios are estimated with neural networks, an uncertainty may be introduced to account not only for the limited number of simulated training samples, but also for the stochastic nature of the training algorithm. Training ensembles on bootstrapped versions of the training data, as described in Section 2.3 provides a natural way to describe both of these effects.

Since the estimator for the density ratio is computed as the mean[5] prediction from an ensemble of networks, the variance of that mean can be estimated using the bootstrapping technique. The mean of each bootstrapped ensemble is used to estimate a best-fit value of the parameter(s) of interest $\hat{\mu}$, and the standard deviation of these estimates determines the variation of the mean $\Delta\hat{\mu}$ due the finite number of events in the training sample. The variance can be determined at different values of μ using different Asimov datasets.[6] Such datasets at any value of the parameter(s) of interest can often be constructed from a set of simulations at few *basis points* in this parameter, using various morphing techniques [8, 38]. The estimated $\Delta\hat{\mu}$ is an uncertainty on the modelling of the expected probability density of the physics processes, and therefore,

[5] The median, known to be unbiased and robust to outliers, could also be used.

[6] An Asimov dataset is one for which the application of any unbiased estimator for all parameters will provide the true values [37]. In practice, an approximation of such a dataset can be constructed using a sufficiently large number of simulated samples with appropriate event weights.

it can be introduced as a systematic uncertainty following the spurious signal approach [39] frequently employed in unbinned LHC analyses. The nuisance parameter α_{stat} with a Gaus$(0, 1)$ constraint term is introduced to Eq. 17 with the modification

$$f_j(\mu) \to f_j(\mu + \alpha_{\text{stat}} \cdot \Delta\hat{\mu}(\mu)). \tag{21}$$

5.4 Calculation of pulls and impacts

While the unbinned nature of NSBI poses computational challenges to traditional statistical tools for evaluating and analysing the profile likelihood ratio, this framework enables the direct application of modern computational tools that simplify calculations. The full likelihood ratio (Eq. 18) and the test statistic (Eq. 19) are differentiable functions. Their dependence on the parameters of interest μ and nuisance parameters α is introduced through differentiable functions, and the probability density ratios are built from neural networks which are themselves differentiable. It is natural to leverage auto-differentiation techniques [40] to perform the profiling and to calculate the Hessian matrix of $L_{\text{full}}(\mu, \alpha)$.

The estimation of pulls and impacts relies on the calculation of the covariance matrix (we identify the parameter of interest with index 0 to simplify the notation),

$$C_{nm} = \left[\frac{1}{2}\frac{\partial^2\lambda}{\partial\alpha_n\partial\alpha_m}(\widehat{\mu}, \widehat{\alpha})\right]^{-1}, \tag{22}$$

using the inverse of the Hessian matrix at the maximum likelihood estimate $(\widehat{\mu}, \widehat{\alpha})$, and where $\lambda(\mu, \alpha) = -2\ln(L_{\text{full}}(\mu, \alpha)/L_{\text{ref}})$. The calculation of the Hessian matrix can be parallelised on computing clusters [41]. The pull of the NP α is calculated as

$$\frac{\widehat{\alpha}_k - \alpha_k^0}{\sqrt{C_{kk}}}. \tag{23}$$

This is the definition often adopted in histogram-based analysis with the *MINOS* procedure [42–44], which defines pulls based on approximate profile likelihood ratio confidence intervals, with the exact computation reserved only for pathological cases.

The impact of a nuisance parameter on a measurement is traditionally computed by re-running the entire likelihood minimisation after fixing the nuisance parameter at a few values. This calculation is more expensive since it requires multiple minimisations of the log-likelihood ratio. Here, the maximum likelihood estimate of μ is re-computed for different fixed values of α_k to estimate $\Gamma_k^{\text{NP}} = \widehat{\mu}(\widehat{\alpha}_k \pm \sqrt{C_{kk}}) - \widehat{\mu}(\widehat{\alpha}_k)$. With auto-differentiation, a local estimate of the *post-fit* impact can be estimated as

$$\Gamma_k^{\text{NP}} = \frac{\partial\widehat{\mu}}{\partial\alpha_k} \cdot \sqrt{C_{kk}} = \left[\frac{\partial^2\lambda}{\partial^2\mu}(\widehat{\mu}, \widehat{\alpha})\right]^{-1} \frac{\partial^2\lambda}{\partial\mu\partial\alpha_k}(\widehat{\mu}, \widehat{\alpha}) \cdot \sqrt{C_{kk}}, \tag{24}$$

considerably simplifying the analysis of the profile likelihood ratio, and reserving the finite-difference estimate to pathological cases. The *pre-fit* impact can be calculated by replacing $(\widehat{\mu}, \widehat{\alpha}_j) \to (\mu_0, \alpha_j^0)$ and $\sqrt{C_{kk}} \to \delta_k$. A similar definition, but based on variations of global observables a_k, has been suggested for a consistent separation between statistical and systematic uncertainties in Ref. [36]. In this case, the

impact is estimated without fixing any nuisance parameter, but calculating $\Gamma_k^{\text{GO}} = \widehat{\mu}(a_k \pm \delta_k) - \widehat{\mu}(a_k)$. The local estimate of the variation based on global observables, given simply by $\Gamma_k^{\text{GO}} = C_{0k}(\widehat{\mu}, \widehat{\alpha})$ (where the covariance matrix is defined in Eq. 22), can also be calculated using auto-differentiation. The local definition also avoids ambiguities that exist in models with multiple local minima. Further details about these calculations for NSBI using auto-differentiation techniques are described in Ref. [41].

6 Neyman construction

In frequentist statistics, a confidence interval derived from a measurement is expected to cover the true value with a specified probability (e.g., in 68% or 95% of experiments). The procedure for building such confidence intervals, referred to as the *Neyman construction*, involves the inversion of the hypothesis tests with the help of a large number of pseudo-experiments generated based on simulated samples [45]. This step is crucial when the test statistic cannot be assumed to follow a chi-squared distribution, such as when the analysis has few pre-selected data events (e.g. low-background searches) and non-linear problems (e.g. due to quantum-interference effects) [37]. In the case of NSBI, any residual bias in the estimated probability density ratios may produce a test statistic that does not follow a chi-squared distribution, making this procedure all the more crucial.

The procedure for producing such pseudo-experiments, often referred to as *throwing toys*, is well established for histogram-based analyses, where the probability density can be sampled as individual Poisson distributions in each bin. This approach can be extended to an unbinned, multi-dimensional NSBI analysis.

6.1 Generating pseudo-experiments

Similar to events measured in an actual experiment, pseudo-experiments consist of unweighted events. These can be generated by sampling simulated events with replacement, with the probability of sampling an event determined by its original weight in the Asimov dataset, w_i^{Asimov}. Since the same simulated event can be chosen multiple times in a pseudo-experiment, this count can be represented by a new whole-number event weight, w_i^{toy}.[7] For a computationally efficient generation of these pseudo-experiments, each simulated event is assigned a w_i^{toy} sampled from a Poisson random number generator with a mean corresponding to the Asimov weight of the event,

$$w_i^{\text{Asimov}} \rightarrow w_i^{\text{toy}} = \text{Poisson}(w_i^{\text{Asimov}}). \tag{25}$$

The generated weights w_i^{toy} are whole numbers by construction. Since w_i^{Asimov} represents fractional weights (on the order of $O(10^{-3})$ for the example described in Section 3), the majority of events are assigned a weight of zero, and a smaller subset is assigned integer weights. A very small fraction of events may be represented multiple times in a single pseudo-experiment ($w_i^{\text{toy}} \geq 2$), similar to the case of generating samples via bootstrapping. To generate such pseudo-experiments from a simulated sample, the original number of simulated samples needs to be much larger than the number of events in an individual pseudo-experiment.

[7] While the whole-number weights w_i^{toy} are used for convenience, the constructed pseudo-experiment still behaves effectively like an unweighted dataset.

6.2 Overcoming negative weights

The above prescription for generating unweighted pseudo-experiments requires the original weights of the simulated events to be non-negative, $w_i^{\text{Asimov}} \geq 0$, since the Poisson distribution is only defined for non-negative values. When the MC simulation sample at a given value of the parameter(s) of interest includes events with negative weights, an alternate sample may be used which consists only of positive weights and covers the support of the original sample. The alternate sample, henceforth referred to as the *reweight reference* sample, will have to first be reweighted to the desired value of the parameter(s) of interest. The samples corresponding to the reference defined in Sec 2 may be a convenient choice for the reweight reference sample because it already covers the entire pre-selection region and can be defined to comprise only positive-weighted events. Since the reference sample does not need to correspond to a physical process, a very large sample can be simulated at leading-order in perturbation theory and, therefore, without negative weights. A large reference sample is not only ideal for the network training but also to allow the generation of large number of pseudo-experiments following the methods described here. The reweight reference can be reweighted to the desired value of the theory parameter (using Eq. 10) as

$$w_i^{\text{rwt-ref}} \to w_i^{\text{Asimov}}(\mu) = \frac{\nu(\mu)}{\nu_{\text{rwt-ref}}} \cdot \frac{p(x_i|\mu)}{p_{\text{rwt-ref}}(x_i)} \cdot w_i^{\text{rwt-ref}}, \tag{26}$$

where $p_{\text{rwt-ref}}(x_i)$ is the probability density and $\nu_{\text{rwt-ref}}$ the rate for the reweight-reference sample. The probability density ratio $p(x_i|\mu)/p_{\text{rwt-ref}}(x_i)$ can be obtained from ensembles specifically trained for the reweighting procedure, following the same prescription as the networks used for inference. The estimation can be validated using the same diagnostics described in Section 4, and the new samples are thereby verified to have the same asymptotic properties as the original MC simulation samples. There are also certain other methods that could be explored to handle negative weighted events [46–48].

6.3 Confidence intervals

Once the pseudo-experiments are generated, the confidence intervals can be constructed following the standard method [45]. For the analysis described in Section 3, the distribution of $p(t_\mu|\mu)$, representing the test statistic t_μ for pseudo-experiments generated at a fixed value of μ, is used to determine the one and two standard-deviation confidence intervals as functions of μ. In the presence of systematic uncertainties, the values of the global observables a_k can be sampled from the constraint density. The distribution of test statistics over many pseudo-experiments is shown in Figure 4 with a μ of 1. This procedure is repeated over the range of μ to construct complete confidence bands as shown in Fig 5. The shapes of these bands deviate slightly from the asymptotic χ^2 distribution because of the non-linear parameterisation used in the off-shell Higgs boson production measurement [13], and are not specifically a feature of NSBI.

The formalism discussed in this section lends itself to further tests for robustness on toy samples generated by shifting multiple nuisance parameters simultaneously and verifying that the confidence bands remain well-behaved in such scenarios. Such samples can be generated by a reweighting procedure similar to the one described in Section 6.2, this time using the probability density ratio that includes nuisance parameters (Eq. 17),

$$w_i^{\text{rwt-ref}} \to w_i^{\text{Asimov}}(\mu, \alpha) = \frac{\nu(\mu, \alpha)}{\nu_{\text{rwt-ref}}} \cdot \frac{p(x_i|\mu, \alpha)}{p_{\text{rwt-ref}}(x_i)} \cdot w_i^{\text{rwt-ref}}. \tag{27}$$

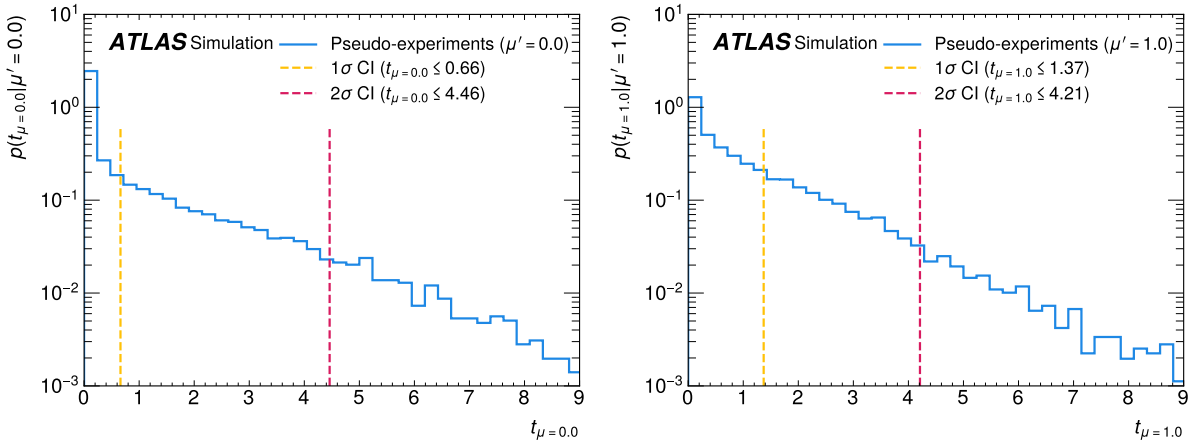

Figure 4: Distribution of the test statistic for pseudo-experiments with a μ of 0 (left) and 1 (right). The 1σ and 2σ confidence intervals are built using a Neyman construction by integrating up to 68.27% (yellow vertical dashed line) and 95.45% (red vertical dashed line) of the 15,000 pseudo-experiments, respectively.

7 Comparison of sensitivity

This section demonstrates the sensitivity of the NSBI method and the impact of systematic uncertainties on the result. The demonstration is performed for a simplified version of an off-shell Higgs boson signal strength measurement on simulated samples and considers a subset of the physics processes and systematic uncertainties that are releveant to a full physics analysis.

7.1 Comparison to histogram-based methods

The NSBI method is compared to two histogram-based analysis strategies on a simulated Asimov dataset, to demonstrate the gains coming from the parameterised and unbinned nature of the method. The first histogram method employs a single observable, a signal vs. full process discriminant, that is commonly used for LHC analyses with quantum interference,

$$\mathrm{O}_{\mathrm{fixed}} = \log \frac{p_{\mathrm{S}}(x_i)}{p_{\mathrm{SBI}}(x_i)}. \tag{28}$$

Since this ratio is already estimated with ensembles for the NSBI method, no additional networks need to be trained. This observable is subsequently used to construct a histogram (with 15 bins), and a Poisson likelihood fit is performed with it, analogous to what would be done in traditional analysis. The likelihood ratio is used as the test statistic. This serves as the baseline for comparison of sensitivities to a traditional analysis using the same data. The improvement from NSBI can be seen in Figure 5.

To demonstrate the power of the parameterisation nature of NSBI, it is also compared to a parameterised but binned method, which may not always be practical to use in analysis but is useful for this demonstration. The second method uses an observable that is parameterised in μ,

$$\mathrm{O}_{\mu} = \frac{p(x_i|\mu)}{p(x_i|\mu = 1)}, \tag{29}$$

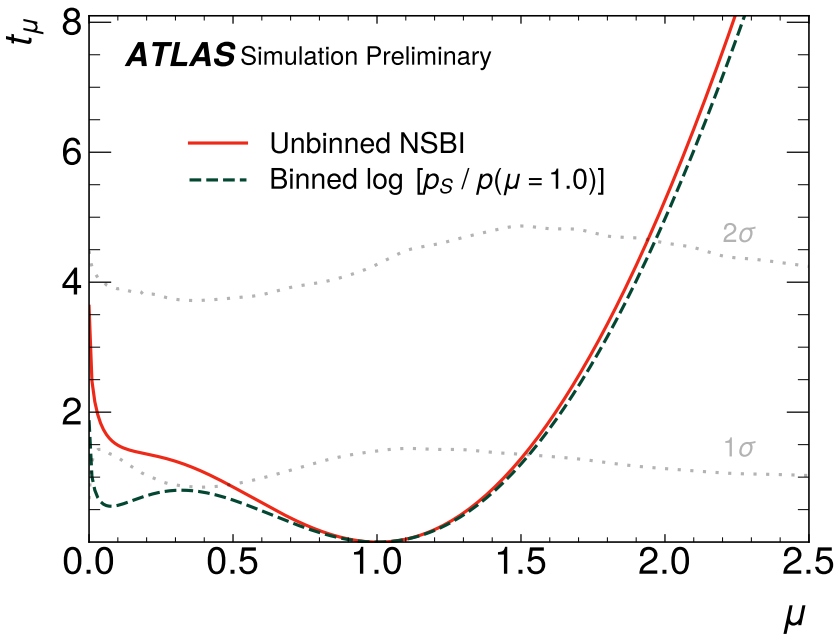

Figure 5: A comparison of expected sensitivity of NSBI to a typical histogram-based analysis, not including systematic uncertainties. The evaluation is performed on an Asimov dataset generated with $\mu = 1$. The test statistic, the log-likelihood ratio t_μ, is shown as a function of signal strength μ. The 1σ and 2σ confidence bands in grey are determined for NSBI using the Neyman construction procedure outlined in Section 6.

which is subsequently binned and used to perform a Poisson likelihood fit. The log-likelihood ratio is computed for each value of μ using a histogram of the corresponding version of O_μ, similar to the method described in Ref. [22]. The improvement shown in Figure 6 for O_μ over O_{fixed} illustrates the power of a parameterised method. The traditional analysis (with the fixed observable) exhibits two prominent minima, which is typical in analyses with non-linear effects from, for example, quantum interference. However, the minimum at the incorrect value of μ is far less prominent for the analysis using a parameterised observable. Since the observable is optimised for each value of the parameter of interest, the method is able to more confidently reject the incorrect values of μ. The further improvement coming from NSBI is due to the unbinned nature of the method. As the number of bins increases, O_μ can approach the sensitivity of NSBI; however, this may introduce numerical instability, requiring careful bin width optimisation, and make sufficiently fine binning untenable across the full range of μ. If the number of bins in a histogram-based analysis is limited by statistics, then leveraging the power of unbinned fits may be desirable.

An additional tool to interpret the results is shown in Figure 7, where the the per-event contribution to the test statsitic, $-2\log(p(x_i|\mu')/p(x_i|\hat{\mu}))$, is shown as a function of $m_{4\ell}$ for two different hypotheses μ' and the maximum likelihood estimate $\hat{\mu} = 1$ on an Asimov dataset generated at $\mu = 1$. Events in regions with this term greater than zero indicate a better compatibility with a $\mu = \mu'$ hypthesis over a $\mu = \hat{\mu}$ hypothesis, while regions with this term less than zero indicate less compatibility. However, these one-dimensional distributions marginalise over the rest of the high-dimensional phase space, and compare only two hypotheses at a time. Therefore, a single distribution is not sufficient to draw conclusions about the phase space responsible for the enhanced sensitivity of this high-dimensional analysis.

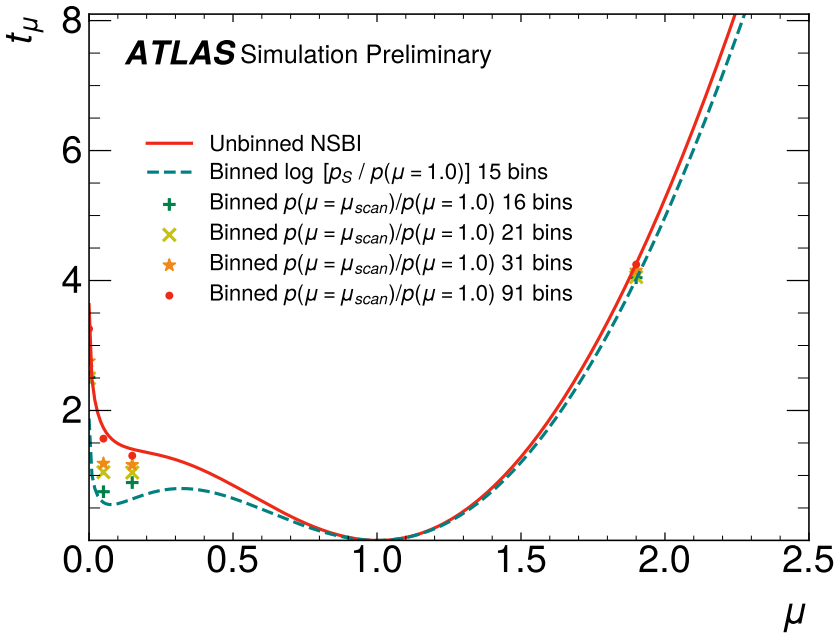

Figure 6: A comparison of expected sensitivity from various analysis strategies using the log-likelihood ratio test statistic t_μ, as a function of μ. The evaluation is performed on an Asimov dataset generated with $\mu = 1$. The red curve represents NSBI. The green curve represents a typical histogram analysis that uses a fixed observable, $\log p_s/p(x|\mu = 1)$, as a discriminant, with 15 bins. The markers show the sensitivity for various histogram analyses that use specific discriminants, $p(x_i|\mu)/p(x_i|\mu = 1)$, for specific values of $\mu(= 0.0, 0.05, 0.15, 1.9)$, with 15 (green pluses), 20 (yellow crosses), 30 (orange stars) or 90 (red dots) bins. The improved sensitivity of the green dots over the green curve (both using 15 bins) is due to the use of a parameterised observable.

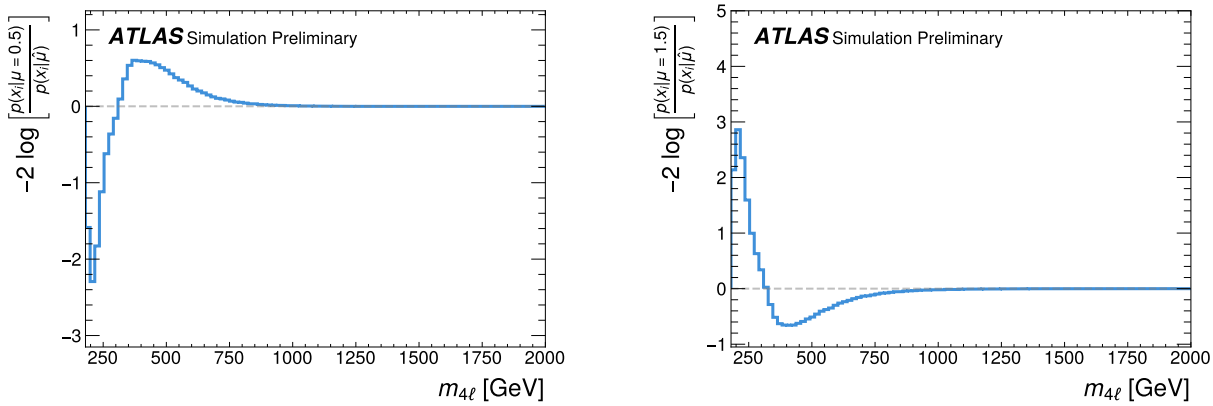

Figure 7: The sum of log density-ratios $-2\log(p(x_i|\mu')/p(x_i|\hat{\mu}))$ for events in bins of $m_{4\ell}$, for a hypothesis $\mu' = 0.5$ (left) or a hypothesis $\mu' = 1.5$ (right), with $\hat{\mu} = 1$ as the maximum likelihood estimate on an Asimov dataset generated at $\mu = 1$. This represents the per-event contribution to the test statistic for a given hypothesis, as a function of $m_{4\ell}$. Events in regions with a sum greater than zero are collectively more consistent with a $\mu = \mu'$ hypthesis over a $\mu = \hat{\mu}$ hypothesis, while regions with a sum less than zero are collectively less consistent. The very high mass region ($m_{4\ell} > 1000$ GeV) is equally consistent with both hypotheses and provides no additional sensitivity.

7.2 Impact of systematic uncertainties

The systematic uncertainties considered in this demonstration are described in Section 3.3, and their impact is taken into account following the formalism developed in Section 5. The $g_j(x_i, \alpha_k)$ term in Eq. 17 accounts for the impact on the shape of the distributions and the $G_j(\alpha_k)$ term accounts for the impact on the inclusive rate. The interpolation functions used are described in Appendix A. In the case of uncertainties that affect the inclusive rate, but not the shape of distributions, the term $g_j(x_i, \alpha_k)$ in Eq. 17 is fixed to 1 over the full range of α_k. This way, the impact of the nuisance parameter on the test statistic pertains only to the overall yields but not to the per-event probability density ratios. The profile (log-)likelihood is shown in Figure 8 and compared to a histogram analysis using the O_{fixed} observable. The systematic uncertainties reduce the sensitivity of the measurement, as is expected.

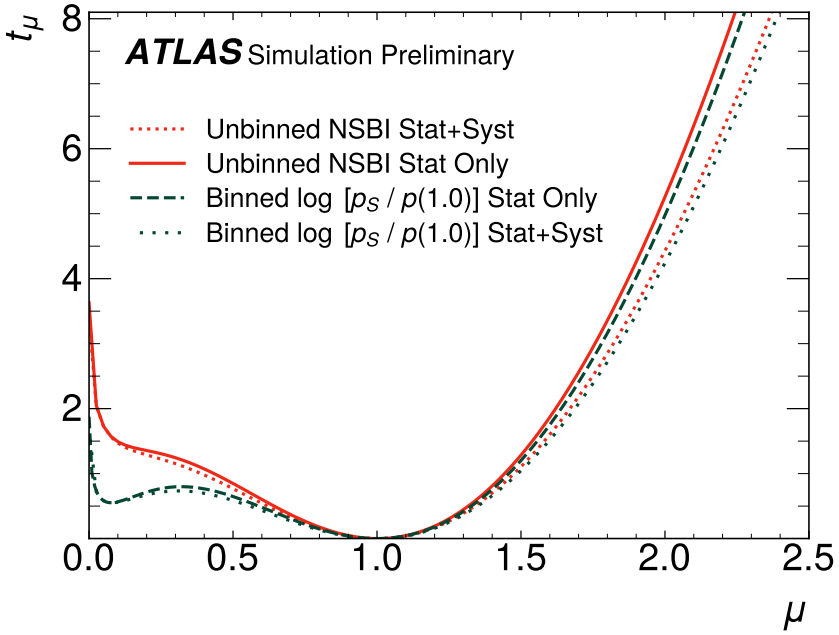

Figure 8: The log-likelihood ratio as a function of signal strength μ, representing only statistical uncertainties (solid red for NSBI, dashed green for histogram analysis), compared to the profile log-likelihood ratio, representing both statistical and systematic uncertainties (dotted red for NSBI, dotted green for histogram analysis), evaluated on Asimov data generated with $\mu = 1$. The histogram analysis is performed with a fixed observable, $\log p_s/p(x|\mu = 1)$. The two nuisance parameters in this study are described in Section 3.3.

8 Conclusion and outlook

While neural simulation-based inference methods have drawn interest for their potential to dramatically improve the sensitivity of key analyses at the LHC [4, 7], several open questions have remained regarding their application in a full-scale LHC analysis. This work develops the necessary tools and concepts required to have a complete statistical framework for NSBI at the LHC and addresses these open questions. The power and feasibility of the method is demonstrated through an example use case: the measurement of the off-shell Higgs boson couplings in the four-lepton final states. This is an analysis with destructive quantum

interference between the signal and background processes, which makes the likelihood model non-linear in the signal strength parameter and, therefore, benefits from the power of NSBI methods [5]. Comparisons with two histogram-based methods illustrate the gains from the unbinned and parameterised nature of the NSBI method. Since this demonstration was performed on a simplified version of the analysis that does not include all the relevant physics processes and systematic uncertainties, the expected sensitivity shown does not reflect the expected sensitivity of a full physics result.

The framework extends the standard statistical methodology employed at the LHC, transitioning to an unbinned, multi-dimensional setting, capable of accommodating a large number of systematic uncertainties. The note also provides a list of diagnostics that can be used to understand and validate the performance of the neural network classifiers and describes a method to build a robust test statistic needed for hypothesis tests. It also describes the procedure to construct confidence intervals for unbinned analyses such as those using NSBI. Computational challenges in evaluating and analysing the test statistic are overcome with the use of auto-differentiation techniques, which facilitate profiling and computing of pulls and impacts [29]. This setup is also conducive to analysing the effect of systematic uncertainties using more recently proposed methods [36].

This method can be applied for parameter estimation in various particle physics analyses, making optimal use of both the available data and simulated samples. It is particularly advantageous for analyses with non-linear likelihood models, large quantum interference, or limited statistics in the observed data, or those requiring complex analysis observables. While "optimal observables" have previously been used to measure theory parameters in EFT analyses [49], these observables are close to optimal only for small regions of the parameter space, and often optimised for regions near the Standard Model. They typically do not account for detector effects. In contrast, NSBI is designed to achieve close to optimal sensitivity throughout the phase space under consideration, accounting for detector effects and systematic uncertainties. The ATLAS experiment has also demonstrated the ability to unfold differential cross-sections in a high-dimensions and without binning [21] and this technique [50] also relies on the ability of classifiers to estimate probability density ratios. The goals of unfolding are different from parameter estimation from experimental data, and the two approaches are complementary.

Since this method inherits formalisms from the standard statistical methods used in LHC experiments, it also inherits the challenges. These include the challenge of model misspecification, for instance, if the simulation has systematic differences from data and the systematic uncertainties have not been well modelled. These issues can be diagnosed by looking at pulls, impacts and other diagnostics described in Section 5.4 in the same manner as in traditional histogram-based analyses. This method can also be used in conjunction with data-driven background estimation when the background simulations are not reliable and incorporate other techniques used in traditional analyses to mitigate systematic uncertainties. One potential technical challenge to an NSBI analysis over a histogram-based one is the need for sufficient training data to optimise precise probability density ratio estimators. These could be overcome by pre-training the networks first on larger datasets such as fast simulated samples [51]. Another technical challenge lies in the computational cost of training such a large number of neural networks; however, the increasing availability of large scientific computing facilities may mitigate this concern in the near future.

Appendix

A Interpolation Function

Section 5 discusses the use of interpolation methods for systematic uncertainties. A common choice for the interpolation function to parameterise the impact of nuisance parameters at the LHC is [2]

$$G_j(\alpha_k) = \begin{cases} \left(\frac{\nu_j(\alpha_k^+)}{\nu_j(\alpha_k^0)}\right)^{\alpha_k} & \alpha_k > 1 \\ 1 + \sum_{n=1}^{6} c_n \alpha_k^n & -1 \leq \alpha_k \leq 1 \\ \left(\frac{\nu_j(\alpha_k^-)}{\nu_j(\alpha_k^0)}\right)^{-\alpha_k} & \alpha_k < -1 \end{cases}, \tag{30}$$

where the six coefficients c_n of the polynomial in α_k are determined uniquely from the requirements that $G_j(\alpha_k)$ be continuous and its first and second derivatives be continuous at $\alpha_k = \pm 1$. The same interpolation strategy and continuity requirements can be used to interpolate $g_j(x_i, \alpha_k)$,

$$g_j(x_i, \alpha_k) = \begin{cases} \left(g_j(x_i, \alpha_k^+)\right)^{\alpha_k} & \alpha_k > 1 \\ 1 + \sum_{n=1}^{6} c_n \alpha_k^n & -1 \leq \alpha_k \leq 1 \\ \left(g_j(x_i, \alpha_k^-)\right)^{-\alpha_k} & \alpha_k < -1 \end{cases}. \tag{31}$$

References

[1] G. Cowan, *Statistical data analysis*, Oxford University Press, USA, 1998 (cit. on pp. 3, 6).

[2] K. Cranmer, 'Practical Statistics for the LHC', *2011 European School of High-Energy Physics*, 2016, arXiv: 1503.07622 [physics.data-an] (cit. on pp. 3, 14, 15, 25).

[3] K. Cranmer, J. Brehmer and G. Louppe, *The frontier of simulation-based inference*, Proceedings of the National Academy of Sciences **117** (2020) 30055 (cit. on pp. 3, 4).

[4] J. Brehmer, K. Cranmer, G. Louppe and J. Pavez, *A Guide to Constraining Effective Field Theories with Machine Learning*, Phys. Rev. D **98** (2018) 052004, arXiv: 1805.00020 [hep-ph] (cit. on pp. 3, 4, 7, 23).

[5] A. Ghosh, 'Measuring quantum interference in the off-shell Higgs to four leptons process with Machine Learning', *Journées de Rencontre des Jeunes Chercheurs 2019 (JRJC 2019)*, 2020 171, URL: https://hal.archives-ouvertes.fr/hal-02971995 (cit. on pp. 3, 8, 24).

[6] A. Butter, T. Plehn, N. Soybelman and J. Brehmer, *Back to the Formula – LHC Edition*, (2024), arXiv: 2109.10414 [hep-ph] (cit. on p. 3).

[7] K. Cranmer, J. Pavez and G. Louppe, *Approximating Likelihood Ratios with Calibrated Discriminative Classifiers*, (2016), arXiv: 1506.02169 [stat.AP] (cit. on pp. 3–7, 12, 14, 23).

[8] J. Brehmer, F. Kling, I. Espejo and K. Cranmer, *MadMiner: Machine learning-based inference for particle physics*, Comput. Softw. Big Sci. **4** (2020) 3, arXiv: 1907.10621 [hep-ph] (cit. on pp. 3, 16).

[9] L. Heinrich, *Learning Optimal Test Statistics in the Presence of Nuisance Parameters*, (2022), arXiv: 2203.13079 [stat.ME] (cit. on p. 3).

[10] L. Heinrich, S. Mishra-Sharma, C. Pollard and P. Windischhofer, *Hierarchical Neural Simulation-Based Inference Over Event Ensembles*, (2023), arXiv: 2306.12584 [stat.ML] (cit. on p. 3).

[11] ATLAS Collaboration, *Determination of jet calibration and energy resolution in proton–proton collisions at $\sqrt{s} = 8$ TeV using the ATLAS detector*, Eur. Phys. J. C **80** (2020) 1104, arXiv: 1910.04482 [hep-ex] (cit. on pp. 3, 15).

[12] ATLAS Collaboration, *Measurement of the inclusive jet cross-sections in proton–proton collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector*, JHEP **09** (2017) 020, arXiv: 1706.03192 [hep-ex] (cit. on pp. 3, 15).

[13] ATLAS Collaboration, *Evidence of off-shell Higgs boson production from ZZ leptonic decay channels and constraints on its total width with the ATLAS detector*, Phys. Lett. B **846** (2023) 138223, arXiv: 2304.01532 [hep-ex] (cit. on pp. 3, 8–10, 15, 19), Erratum: Phys. Lett. B **854** (2024) 138734.

[14] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed and B. Lakshminarayanan, *Normalizing Flows for Probabilistic Modeling and Inference*, 2021, arXiv: 1912.02762 [stat.ML] (cit. on p. 4).

[15] G. Papamakarios and I. Murray, *Fast ϵ-free Inference of Simulation Models with Bayesian Conditional Density Estimation*, 2018, arXiv: 1605.06376 [stat.ML] (cit. on p. 4).

[16] C. Modi et al., *Sensitivity Analysis of Simulation-Based Inference for Galaxy Clustering*, (2023), arXiv: 2309.15071 [astro-ph.CO] (cit. on p. 4).

[17] L. Brandes et al., *Neural simulation-based inference of the neutron star equation of state directly from telescope spectra*, JCAP **09** (2024) 009, arXiv: 2403.00287 [astro-ph.HE] (cit. on p. 4).

[18] M. Dax et al., *Real-Time Gravitational Wave Science with Neural Posterior Estimation*, Phys. Rev. Lett. **127** (2021) 241103, arXiv: 2106.12594 [gr-qc] (cit. on p. 4).

[19] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning*, Springer Series in Statistics, New York, NY, USA: Springer New York Inc., 2001 (cit. on p. 4).

[20] ATLAS Collaboration, *Search for nonresonant pair production of Higgs bosons in the $b\bar{b}b\bar{b}$ final state in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector*, Phys. Rev. D **108** (2023) 052003, arXiv: 2301.03212 [hep-ex] (cit. on pp. 5, 8).

[21] ATLAS Collaboration, *A simultaneous unbinned differential cross section measurement of twenty-four Z+jets kinematic observables with the ATLAS detector*, (2024), arXiv: 2405.20041 [hep-ex] (cit. on pp. 5, 8, 24).

[22] P. Baldi, K. Cranmer, T. Faucett, P. Sadowski and D. Whiteson, *Parameterized neural networks for high-energy physics*, Eur. Phys. J. C **76** (2016) 235, arXiv: 1601.07913 [hep-ex] (cit. on pp. 5, 21).

[23] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*, 1st, Chapman & Hall/CRC, 2012 (cit. on p. 7).

[24] ATLAS Collaboration, *Evaluating statistical uncertainties and correlations using the bootstrap method*, ATL-PHYS-PUB-2021-011, 2021, URL: https://cds.cern.ch/record/2759945 (cit. on p. 8).

[25] S. Bolognesi et al., *Spin and parity of a single-produced resonance at the LHC*, Phys. Rev. D **86** (2012) 095031, arXiv: 1208.4018 [hep-ph] (cit. on p. 9).

[26] P. Ramachandran, B. Zoph and Q. V. Le, *Searching for Activation Functions*, (2017), arXiv: 1710.05941 (cit. on p. 10).

[27] S. Li, D. Li and Y. Zhang, 'Incorporating Nesterov's Momentum into Distributed Adaptive Gradient Method for Online Optimization', *2021 China Automation Congress (CAC)*, 2021 7338 (cit. on p. 10).

[28] TensorFlow Developers, *TensorFlow*, version v2.9.0-rc2, 2022, URL: https://doi.org/10.5281/zenodo.6519082 (cit. on p. 10).

[29] F. B. Megino et al., *Operational Experience and R&D results using the Google Cloud for High Energy Physics in the ATLAS experiment*, (2024), arXiv: 2403.15873 [hep-ex] (cit. on pp. 10, 24).

[30] J. M. Campbell and R. K. Ellis, *Update on vector boson pair production at hadron colliders*, Phys. Rev. D **60** (1999) 113006, arXiv: hep-ph/9905386 (cit. on p. 11).

[31] A. M. Sirunyan et al., *Measurements of the Higgs boson width and anomalous HVV couplings from on-shell and off-shell production in the four-lepton final state*, Phys. Rev. D **99** (2019) 112003, arXiv: 1901.00174 [hep-ex] (cit. on p. 12).

[32] C. Krause and D. Shih, *Fast and accurate simulations of calorimeter showers with normalizing flows*, Phys. Rev. D **107** (2023) 113003, arXiv: 2106.05285 [physics.ins-det] (cit. on p. 12).

[33] R. Kansal et al., *Evaluating generative models in high energy physics*, Phys. Rev. D **107** (2023) 076017, arXiv: 2211.10295 [hep-ex] (cit. on p. 12).

[34] A. Ghosh, B. Nachman and D. Whiteson, *Uncertainty-aware machine learning for high energy physics*, Phys. Rev. D **104** (2021) 056026, arXiv: 2105.08742 [physics.data-an] (cit. on pp. 14, 15).

[35] A. Ghosh and B. Nachman, *A cautionary tale of decorrelating theory uncertainties*, Eur. Phys. J. C **82** (2022) 46, arXiv: 2109.08159 [hep-ph] (cit. on p. 15).

[36] A. Pinto et al., *Uncertainty components in profile likelihood fits*, (2023), arXiv: 2307.04007 [physics.data-an] (cit. on pp. 15, 17, 24).

[37] G. Cowan, K. Cranmer, E. Gross and O. Vitells, *Asymptotic formulae for likelihood-based tests of new physics*, Eur. Phys. J. C **71** (2011) 1554, arXiv: 1007.1727 [physics.data-an] (cit. on pp. 15, 16, 18), Erratum: Eur. Phys. J. C **73** (2013) 2501.

[38] ATLAS Collaboration, *A morphing technique for signal modelling in a multidimensional space of coupling parameters*, ATL-PHYS-PUB-2015-047, 2015, URL: https://cds.cern.ch/record/2066980 (cit. on p. 16).

[39] ATLAS Collaboration, *Recommendations for the Modeling of Smooth Backgrounds*, ATL-PHYS-PUB-2020-028, 2020, URL: https://cds.cern.ch/record/2743717 (cit. on p. 17).

[40] A. G. Baydin, B. A. Pearlmutter, A. A. Radul and J. M. Siskind, *Automatic Differentiation in Machine Learning: a Survey*, Journal of Machine Learning Research **18** (2018) 1 (cit. on p. 17).

[41] Sandesara, Jay et al.,
ATLAS Data Analysis using a Parallel Workflow on Distributed Cloud-based Services with GPUs,
EPJ Web of Conf. **295** (2024) 04007 (cit. on pp. 17, 18).

[42] F. James, *MINUIT: Function Minimization and Error Analysis Reference Manual*,
(1998), CERN Program Library Long Writeups, URL: https://cds.cern.ch/record/2296388
(cit. on p. 17).

[43] W. Verkerke and D. Kirkby, 'The RooFit toolkit for data modeling',
ed. by L. Lyons and M. Karagoz, vol. C0303241, 2003 MOLT007, arXiv: physics/0306116
(cit. on p. 17).

[44] L. Moneta et al., *The RooStats Project*, 2011, arXiv: 1009.1003 [physics.data-an]
(cit. on p. 17).

[45] J. Neyman,
Outline of a Theory of Statistical Estimation Based on the Classical Theory of Probability,
Phil. Trans. Roy. Soc. Lond. **236** (1937) 333 (cit. on pp. 18, 19).

[46] M. Backes, A. Butter, T. Plehn and R. Winterhalder, *How to GAN Event Unweighting*,
SciPost Phys. **10** (2021) 089, arXiv: 2012.07873 [hep-ph] (cit. on p. 19).

[47] B. Nachman and J. Thaler,
Neural resampler for Monte Carlo reweighting with preserved uncertainties,
Phys. Rev. D **102** (2020) 076004, arXiv: 2007.11586 [hep-ph] (cit. on p. 19).

[48] M. Drnevich, S. Jiggins, J. Katzy and K. Cranmer,
Neural Quasiprobabilistic Likelihood Ratio Estimation with Negatively Weighted Data, (2024),
arXiv: 2410.10216 [stat.ML] (cit. on p. 19).

[49] N. Castro et al., *LHC EFT WG Report: Experimental Measurements and Observables*, 2022,
arXiv: 2211.08353, URL: https://cds.cern.ch/record/2809469 (cit. on p. 24).

[50] A. Andreassen, P. T. Komiske, E. M. Metodiev, B. Nachman and J. Thaler,
OmniFold: A Method to Simultaneously Unfold All Observables,
Phys. Rev. Lett. **124** (2020) 182001, arXiv: 1911.09107 [hep-ph] (cit. on p. 24).

[51] Z. Zhao, L. Alzubaidi, J. Zhang, Y. Duan and Y. Gu, *A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations*,
Expert Systems with Applications **242** (2024) 122807 (cit. on p. 24).