Effective LHC measurements with matrix elements and machine learning

J. Brehmer[1], **K. Cranmer**[1], **I. Espejo**[1], **F. Kling**[2], **G. Louppe**[3], **and J. Pavez**[4]

[1] New York University, USA; [2] University of California, Irvine, USA; [3] University of Liège, Belgium; [4] Federico Santa María Technical University, Chile

E-mail: johann.brehmer@nyu.edu

Abstract. One major challenge for the legacy measurements at the LHC is that the likelihood function is not tractable when the collected data is high-dimensional and the detector response has to be modeled. We review how different analysis strategies solve this issue, including the traditional histogram approach used in most particle physics analyses, the Matrix Element Method, Optimal Observables, and modern techniques based on neural density estimation. We then discuss powerful new inference methods that use a combination of matrix element information and machine learning to accurately estimate the likelihood function. The MadMiner package automates all necessary data-processing steps. In first studies we find that these new techniques have the potential to substantially improve the sensitivity of the LHC legacy measurements.

1. LHC measurements as a likelihood-free inference problem

The legacy measurements of the Large Hadron Collider (LHC) experiments aim to put precise constraints on indirect effects of physics beyond the Standard Model, for instance parameterized in an effective field theory (EFT) [1]. This requires a careful analysis of high-dimensional data. The relation between data x and physics parameters θ is most fundamentally described by the likelihood function or normalized fully differential cross section $p(x|\theta) = \mathrm{d}\sigma(x|\theta)/\sigma(\theta)$. In fact, this likelihood function is the basis for most established statistical methods in high-energy physics, including maximum likelihood estimation, hypothesis testing, and exclusion limits based on the profile likelihood ratio [2].

Typically, LHC processes are most accurately described by a suite of complex computer simulations that describe parton density functions, hard process, parton shower, hadronization, detector response, sensor readout, and construction of observables with impressive precision. These tools take values of the parameters θ as input and use Monte-Carlo techniques to sample from the many different ways in which an event can develop, ultimately leading to simulated samples of observations $x \sim p(x|\theta)$. The likelihood that these simulators implicitly define can be symbolically written as [3, 4]

$$p(x|\theta) = \int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p \underbrace{p(x|z_s)\, p(z_s|z_p)\, p(z_p|\theta)}_{p(x,z|\theta)}, \tag{1}$$

where z_d are the variables characterizing the detector interactions in one simulated event, z_s describes the parton shower and hadronization, and z_p are the properties of the elementary particles in the hard interaction (four-momenta, helicities, charges, and flavours). These latent variables form an extremely high-dimensional space: with state-of-the-art simulators including GEANT4 [5] for the detector simulation, one simulated event can easily involve tens of millions of random numbers! Explicitly calculating the integral over this huge space is clearly impossible: the likelihood function $p(x|\theta)$ is intractable.

This is not a problem unique to particle physics. Phenomena that are modeled by a forward simulation that does not admit a tractable likelihood are common in fields as diverse as cosmology, epidemiology, and systems biology. This has given rise to the development of many different methods of "likelihood-free inference" that allow us to constrain parameter values even without a computable likelihood function. Phrasing particle physics measurements in this language allows us to tap into recent developments in these fields as well as in statistics and computer science.

2. Established methods

We will now briefly review six established types of inference methods that can be used for particle physics measurements.

2.1. Histograms of summary statistics

Typically, not all the tens or hundreds of observables that can be calculated for an event collision are equally informative on a given physics question. Often it is enough to analyze one or two summary statistics $v(x)$, hand-picked kinematic variables such as reconstructed invariant masses, momenta, or angles. The likelihood function $p(v(x)|\theta)$ in the space of these summary statistics can then be computed with simple density estimation techniques such as one-dimensional or two-dimensional histograms, kernel density estimation techniques, or Gaussian processes, and used instead of the likelihood function of the high-dimensional event data. This approach discards any information in the other phase-space directions.

This is by far the most common inference technique in high-energy physics. It is fast and transparent. The disadvantage is, of course, that choosing the summary statistics is a difficult and problem-specific task. While there are obvious candidates for some problems, like the invariant mass in the case of the search for narrow resonances, the indirect effects of EFT operators are typically not captured well by any single observable [6, 7]. Histograms suffer from the "curse of dimensionality": since the required number of samples grows exponentially with the dimension of v, they do not scale to more than a few observables.

2.2. Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) [8–11] is a family of Bayesian sampling techniques for likelihood-free inference. Rather than providing a tractable surrogate of the likelihood, ABC returns a set of parameter points $\theta \sim \hat{p}(\theta|x)$ sampled from an approximate version of the posterior. In its simplest form, this is based on accepting or rejecting individual simulated samples x based on some notion of distance to the observed data and an distance threshold ϵ.

As in the histogram case, this requires compressing the data x to summary statistics $v(x)$, again discarding any information in other variables. In addition, the posterior is only exact in the limit $\epsilon \to 0$, but in this limit the number of required simulations explodes.

2.3. Neural density estimation

In recent years, several machine learning techniques have been developed that train a neural network to estimate the likelihood $\hat{p}(x|\theta)$ based only on samples from the simulator [12–31]. Two particularly promising classes are autoregressive models, which model a high-dimensional

probability distribution as a product of one-dimensional conditionals, and normalizing flows, which model the distribution of x as a simple base density followed by several invertible transformations.

These neural density estimation techniques scale well to high-dimensional observables, so they do not require a compression to summary statistics. However, these algorithms can be less transparent than other approaches, requiring careful cross-checks. In addition, they are agnostic about the physics processes they are modeling: to the neural network, it does not make a difference whether it estimates the density of smiling cats in the space of all images or the density of LHC collisions for a particular EFT hypothesis. Often a large number of training samples is necessary for the network to an accurate estimate the true likelihood.

2.4. Likelihood ratios from classifiers

Other machine learning techniques are based on training neural networks to classify between samples drawn from a given parameter point $x \sim p(x|\theta)$ and samples drawn from a reference parameter point $x \sim p(x|\theta_{\text{ref}})$. If trained succesfully, the classifier decision function can be converted into an estimator $\hat{r}(x|\theta)$ of the likelihood ratio

$$r(x|\theta) = \frac{p(x|\theta)}{p(x|\theta_{\text{ref}})} \,. \tag{2}$$

In practice, this is just as powerful as an estimator for the likelihood itself. This "likelhood ratio trick" is widely appreciated [32, 33] and can be made much more accurate by adding a calibration stage, resulting in the CARL inference technique [34].

Similar to the neural density estimation techniques, this approach is well-suited to high-dimensional data and does not require choosing summary statistics, at the cost of some (perceived) intransparency and a potentially large number of required training samples.

2.5. The Matrix Element Method

An important insight is that while the integral in Eq. (1) is intractable, some parts of the integrand can in fact be calculated. In particular, the parton-level likelihood function $p(z_p|\theta)$ is given by a combination of phase-space factors, parton density functions, and matrix elements, all of which we can compute. On the other hand, in many processes the combined effect of parton shower and detector response is just a "smearing" of the true parton-level momenta z_p into the observed reconstructed particle momenta x. If these effects can be approximated with a simple tractable density $\hat{p}(x|z_p)$, the "transfer function", the likelihood is approximately

$$\hat{p}(x|\theta) = \int \mathrm{d}z_p \, \hat{p}(x|z_p) \, p(z_p|\theta) \sim \frac{1}{\sigma(\theta)} \int \mathrm{d}z_p \, \hat{p}(x|z_p) \, |\mathcal{M}(z_p|\theta)|^2 \,. \tag{3}$$

In the last part we have left out parton densities as well as phase-space factors for simplicity. This integral is much lower-dimensional than the one in Eq. (1), and we can indeed often calculate this approximate likelihood function! In the simplest case, the observed particle momenta are identified with the parton-level momenta, the transfer function becomes $\hat{p}(x|z_p) = \prod_i \delta^4(x_i - z_{p\,i})$, and the integration is trivial.

This is the essential idea behind the Matrix Element Method (MEM) [35–49]. Recently, this approach has been extended to include an explicit calculation of leading parton-shower effects ("shower deconstruction", "event deconstruction") [50–53].

Unlike the neural network methods discussed above, the MEM explicitly uses our knowledge about the physics structure of the hard process and does not rely on the correct training of neural networks. However, there are two significant downsides. The approximation of shower and detector effects with the transfer function $\hat{p}(x|z_p)$ is not always accurate. Even if many resolution effects can be modeled well with such an ad-hoc function, other physical effects such as additional

QCD radiation are very difficult or even impossible to describe in this framework. At the same time, the integral over z_p can still be very expensive, and has to be calculated for every single event — which can sum up to an immense computational cost when large numbers of events are considered.

2.6. Optimal Observables

The matrix element information and the approximation in Eq. (3) can also be used to define observables

$$O_i(x|\theta_{\text{ref}}) = \frac{\partial}{\partial\theta_i} \log \left(\int \mathrm{d}z_p \; \hat{p}(x|z_p) \, p(z_p|\theta) \right) \Bigg|_{\theta=\theta_{\text{ref}}} = \frac{\int \mathrm{d}z_p \; \hat{p}(x|z_p) \, \partial_i p(z_p|\theta_{\text{ref}})}{\int \mathrm{d}z_p \; \hat{p}(x|z_p) \, p(z_p|\theta_{\text{ref}})} \tag{4}$$

with one component per theory parameter θ_i. θ_{ref} is a reference parameter point, often chosen to be $\theta_{\text{ref}} = 0$. In the literature this approach is typically used with the identification of observed and true momenta $\hat{p}(x|z_p) = \prod_i \delta^4(x_i - z_{p\,i})$, but the extension to non-trivial transfer functions is straightforward. These observables can be used like any kinematic variable. In particular, the likelihood in the space of the $O_i(x)$ can be estimated using histograms or other density estimation techniques.

In the approximation of Eq. (3) and as long as only parameter points close the θ_{ref} are considered, the $O_i(x)$ are the sufficient statistics: they fully characterize the high-dimensional event data x, and an analysis based on $p(O_i(x)|\theta)$ will lead to the same conclusions as an analysis of the intractable $p(x|\theta)$. The O_i are therefore known as Optimal Observables [54–56].

The Optimal Observable approach shares the underlying approximation, strengths, and disadvantages of the Matrix Element Method. While the immediate use of matrix element information is beautiful, it requires stark approximations, and any treatment of detector effects incurs a large computational cost for each analyzed event.

3. "Mining gold": New inference techniques

In a series of recent publications, a family of new techniques for likelihood-free inference based on a combination of matrix element information and machine learning was introduced [3, 4, 57, 58]. They fall in two categories: a first class of methods trains a neural network to estimate the full likelihood function, while a second class is motivated by an expansion of the likelihood around a reference parameter point and trains a neural network to provide optimal observables.

3.1. Learning the high-dimensional likelihood

The starting point for the new methods is the same as for the Matrix Element Method: while the likelihood in Eq. (1) is intractable because of a high-dimensional integral, we can in fact compute the parton-level likelihood function $p(z_p|\theta)$, given by a combination of phase-space factors, parton density functions, and matrix elements. This means for each simulated event we can also calculate the *joint likelihood ratio*

$$r(x,z|\theta) = \frac{p(x,z|\theta)}{p(x,z|\theta_{\text{ref}})} = \frac{p(z_p|\theta)}{p(z_p|\theta_{\text{ref}})} \sim \frac{|\mathcal{M}|^2(z_p|\theta)}{|\mathcal{M}|^2(z_p|\theta_{\text{ref}})} \, \frac{\sigma(\theta_{\text{ref}})}{\sigma(\theta)} \tag{5}$$

and the *joint score*

$$t(x,z|\theta) = \nabla_\theta \log p(x,z|\theta) = \frac{\nabla_\theta p(z_p|\theta)}{p(z_p|\theta)} \sim \frac{\nabla_\theta |\mathcal{M}|^2(z_p|\theta)}{|\mathcal{M}|^2(z_p|\theta)} - \frac{\nabla_\theta \sigma(\theta)}{\sigma(\theta)} \tag{6}$$

These two quantities define how much more or less likely one particular evolution of an event (fixing all the latent variables z) would be if we changes the theory parameters.

Why are these quantities useful? They are conditional on unobservable variables z, most notably the parton-level momenta of the particles. But the key insights behind the new methods is that the joint likelihood ratio and joint score can be used to construct functionals $L[g(x,\theta)]$ that *are minimized by the true likelihood or likelihood ratio function!* In practice, we can implement this minimization with machine learning: a neural network $g(x,\theta)$ that takes as input the observables x and the parameters θ [see 34, 59] is trained by minimizing a loss function that involves both the joint likelihood ratio $r(x,z|\theta)$ and the joint score $t(x,z|\theta)$ via stochastic gradient descent (or some other numerical optimizer). Assuming sufficient network capacity, efficient minimization, and enough training samples, the network will then converge towards the true likelihood function

$$g(x,\theta) \to \arg\min_g L_p[g(x,\theta)] = p(x|\theta) \tag{7}$$

or, equivalently, the true likelihood ratio function

$$g(x,\theta) \to \arg\min_g L_r[g(x,\theta)] = r(x|\theta)\,! \tag{8}$$

These loss functions thus allow us to turn tractable quantities based on the matrix element into an estimator for the intractable likelihood function.

There are several loss functionals with this property, named with the acronyms Rolr, Rascal, Cascal, Scandal, Alice, and Alices. They are individually discussed and compared in Refs. [3, 4, 57, 58]. In first experiments, the Alices loss [58] provides the best approximation of the likelihood ratio, while the Scandal loss [57] allows to directly estimate the likelihood function and uses state-of-the-art neural density estimation techniques in combination with the matrix element information.

Once a neural network is trained to estimate the likelihood (ratio) function, established statistical techniques such as profile likelihood ratio tests [2] can be used to construct confidence limits in the parameter space.

This approach can be seen as a generalization of the MEM technique that supports state-of-the-art shower and detector simulations as opposed to simple transfer functions. While it requires an upfront training phase, the evaluation of the likelihood for each event is extremely fast ("amortized inference").

3.2. Learning locally optimal observables

Rather than learning the full likelihood function, the joint score can also be used to define statistically optimal observables. This approach is motivated by an expansion of the log likelihood in theory space around a reference parameter point θ_{ref} (such as the SM):

$$\log p(x|\theta) = \log p(x|\theta_{\text{ref}}) + t(x|\theta_{\text{ref}}) \cdot (\theta - \theta_{\text{ref}}) + \mathcal{O}\left((\theta - \theta_{\text{ref}})^2\right) \tag{9}$$

where we have introduced the *score*

$$t(x|\theta) = \nabla_\theta \log p(x|\theta)\,. \tag{10}$$

As long as we are considering parameter points close enough to θ_{ref}, we can neglect the higher orders, and the score vector fully characterizes the likelihood function up to θ-independent constants. In fact, in this local approximation the likelihood is in the exponential family and the score components are the sufficient statistics: for measuring θ, knowing the $t(x|\theta_{\text{ref}})$ is just as powerful as knowing the full likelihood function [3, 4, 10, 57, 60]. The score at θ_{ref} is thus a vector of the statistically most powerful observables! Further away from θ_{ref}, the higher-order terms become important, and the score loses this optimality property.

Method	Estimates	Approximations				$\lvert\mathcal{M}\rvert^2$	Comp. cost
		summaries	PL/TF	local	functional		
Histograms of observables	$\hat{p}(v(x)\lvert\theta)$	✓			binning		low
Approximate Bayesian Computation	$\theta \sim p(\theta\lvert x)$	✓			ϵ-kernel		high (small ϵ)
Neural density estimation	$\hat{p}(x\lvert\theta)$				NN		amortized
Carl	$\hat{r}(x\lvert\theta)$				NN		amortized
Matrix Element Method	$\hat{p}(x\lvert\theta)$		✓		integral	✓	high (TF)
Optimal Observables	$\hat{t}(x)$		✓	✓	integral	✓	high (TF)
Scandal	$\hat{p}(x\lvert\theta)$				NN	✓	amortized
Alice, Alices, Cascal, Rascal, Rolr	$\hat{r}(x\lvert\theta)$				NN	✓	amortized
Sally, Sallino	$\hat{t}(x)$			✓	NN	✓	amortized

Table 1: Classes of established (top part) and novel (bottom half) inference techniques. We classify them by the key quantity that is estimated in the different approaches, by whether they rely on the choice of summary statistics, are based on a parton-level or transfer-function approximation ("PL/TF"), whether their optimality depends on a local approximation ("local"), by whether they use any other functional approximations such as a histogram binning or a neural network ("NN"), whether they leverage matrix-element information ("$|\mathcal{M}|^2$"), and by the computational evaluation cost.

Unfortunately, the score itself is defined through the intractable likelihood function and cannot be calculated explicitly. But it is possible to compute the joint score of Eq. (6) for each simulated event. Similarly to the approach discussed above, we can train a neural network $g(x)$ on a suitable loss function and show that it will converge to

$$g(x) \to \arg\min_g L_t[g(x)] = t(x|\theta)\,. \tag{11}$$

In this way, we can train a neural network to define the most powerful observables. In a next step, the likelihood can be determined for instance with simple histograms of the score components. This is the basic idea behind the Sally and Sallino inference techniques of Refs. [3, 4, 57].

This approach is particularly robust and requires only minor changes to established analysis pipelines. Note that the score vector is almost the same as the Optimal Observables O_i of Eq. (4), but replaces the parton-level (or transfer-function) approximation with a neural network estimation of the full statistical model, including state-of-the-art shower and detector simulations.

3.3. Discussion

In Table 1 we compare the different approaches roughly. First, all techniques rely on some approximations or simplifications to make the likelihood tractable. For the traditional histogram and ABC techniques, this is the restriction to one or a few summary statistics, which potentially throws away a substantial amount of information. For the matrix element method and Optimal Observables, neglecting or approximating the shower and detector response plays this role, which is expected to have a larger impact the less clean a final state is. For neural density estimation techniques, likelihood ratio estimation based on classifiers (Carl), or the new techniques presented in this section, the key approximation is the reliance on neural networks to minimize a functional. Several diagnostic tools and calibration procedures have been proposed [3, 34] that make the network predictions more trustworthy and can guarantee statistically correct (if potentially suboptimal) limits. In addition, both the traditional Optimal Observable method and the new Sally and Sallino techniques are based on a local approximation: the summary statistics they define are statistically optimal only within a small region in parameter space.

Second, the different methods have very different computational costs and scale differently to high-dimensional problems. The classical histogram approach can be relatively cheap: if samples

are generated for every tested parameter point on a grid, the number of required simulator runs scales as e^{n_θ}, where n_θ is the dimension of the parameter space. But a morphing technique can typically be used to make the number of required runs scale as n_θ^2 or n_θ^4 [3, 61]. ABC can be expensive for small ϵ or when many parameter points from the posterior are required. Since they are based on explicitly calculating integrals over latent variables, the Matrix Element Method and Optimal Observable approaches scale differently from the other methods: the run time scales approximately exponential in the number of unobservable directions in phase space n_z. In terms of n_θ, the calculation time scales as n_θ in the case of Optimal Observables and e^{n_θ} in the case of the MEM evaluated on a parameter grid. A major disadvantage of these methods is that the calculation of the integral has to be repeated for every new event, resulting in an overall $e^{n_z} n_\theta n_x$ (OO) or $e^{n_z} e^{n_\theta} n_x$ (MEM) scaling.

The methods based on machine learning, on the other hand, allow for *amortized inference*: after an upfront training phase, the evaluation of the approximate likelihood, likelihood ratio, or score is extremely fast. The key question is therefore how much training data is required for an accurately trained model. For the methods that estimate the likelihood or likelihood ratio, this scaling depends on the specific problem: in the worst case, when distributions vary strongly over the parameter space, the number of Monte-Carlo samples is expected to scale as e^{n_θ}. In practice, distributions change smoothly over parameter space, and parameterized neural networks can discover these patterns with little data. Once the network is trained, inference is fast (unless large toy measurement samples are required for calibration, which scales with e^{n_θ} again). How much the augmented data improves sample efficiency depends again on the specific problem: a larger effect of shower and detector on observables increases the variance of the joint likelihood ratio and joint score around the true likelihood ratio and true score and finally the number of required samples.

In the SALLY / SALLINO approach, the networks do not have to learn a function of both x and θ, but rather an n_θ-dimensional vector as a function of x — this turns out to be a substantially simpler problem and requires much less training data. The main difference between them is in the number of simulations required to calculate the likelihood for a single parameter point. SALLY requires filling an n_θ-dimensional histogram, for which the number of required samples scales like e^{n_θ}, while SALLINO is based on one-dimensional histograms.

4. MadMiner: A sustainable software environment

We are developing MADMINER [62], a Python package that automates all steps of the new inference techniques from the running of the Monte-Carlo simulations, extracting the augmented data, training neural network with suitable loss functions, and calculating expected and observed limits on parameters. The current version v0.4.0 wraps around MADGRAPH5_AMC [63], PYTHIA 8 [64], and DELPHES 3 [65], providing all tools for a phenomenological analysis. Its modular interface allows the extension to state-of-the-art tools used by the experimental collaborations; such an extension would mostly require book-keeping of event weights.

MADMINER is open source, available on GitHub [66], and can be installed with a simple `pip install madminer`. Its documentation is published on ReadTheDocs [67]. We provide interactive tutorials [66], a Docker container with a working software environment [68], and deploy MADMINER with a reusable REANA workflow [69].

5. In the wild

The new inference techniques are used in several real-life projects. The original publications in Refs. [3, 4, 57, 58] contained a proof-of-concept EFT analysis of Higgs production in weak boson fusion (WBF) in the four-lepton decay mode. In a somewhat simplified setup, the new methods could substantially improve the expected sensitivity to two dimension-six operators compared to

histogram-based analyses. They also required more than two orders of magnitude less training samples than the CARL method.

After these encouraging first results, the new methods and MADMINER are now being used in several realistic analyses, including Wh production [70], $W\gamma$ production [71], and $t\bar{t}h$ production [62]. Other ongoing projects include a comparison of the new techniques with the Matrix Element Method and applications of the new techniques to problems in other fields.

6. Conclusions

The LHC legacy measurements will require picking out (or excluding) subtle signatures in high-dimensional data to put limits on an also high-dimensional parameter space. While Monte-Carlo simulations provide an excellent description of this process, they do not allow us to explicitly calculate the corresponding likelihood function. We have reviewed how different analysis strategies address this issue, including the traditional histogram approach, the Matrix Element Method, Optimal Observables, and methods based on machine learning.

We then discussed a new family of multivariate inference techniques that combine matrix element information with machine learning. This new paradigm brings together the strengths of different existing methods: the methods do not require the choice of a summary statistic, utilize matrix element information efficiently, but unlike the Matrix Element Method or Optimal Observables they support state-of-the-art shower and detector simulations without approximations on the underlying physics. After an upfront training phase, they can also be evaluated very fast.

The new Python package MADMINER automates all steps of the analysis chain. It currently supports all tools of a typical phenomenological analysis — MADGRAPH5_AMC, PYTHIA 8, DELPHES 3 — but can be scaled up to experimental analyses.

A first analysis of EFT operators in WBF Higgs production showed that the new techniques led to stronger bounds with less training data compared to established methods. Several studies of other channels are now underway. If they confirm the initial results, these new techniques have the potential to substantially improve the precision of the LHC legacy measurements.

Acknowledgements

We would like to thank Zubair Bhatti, Pablo de Castro, Lukas Heinrich, and Samuel Homiller for great discussions, and we are grateful to the ACAT organizers for a wonderful workshop. This work was supported by the National Science Foundation under the awards ACI-1450310, OAC-1836650, and OAC-1841471. It was also supported through the NYU IT High Performance Computing resources, services, and staff expertise. JB, KC, and GL are grateful for the support of the Moore-Sloan data science environment at NYU. KC is also supported through the NSF grant PHY-1505463, FK is supported by NSF grant PHY-1620638, while JP is partially supported by the Scientific and Technological Center of Valparaíso (CCTVal) under Fondecyt grant BASAL FB0821.

References

[1] W. Buchmuller and D. Wyler: 'Effective Lagrangian Analysis of New Interactions and Flavor Conservation'. Nucl. Phys. B268, p. 621, 1986.

[2] G. Cowan, K. Cranmer, E. Gross, and O. Vitells: 'Asymptotic formulae for likelihood-based tests of new physics'. Eur. Phys. J. C71, p. 1554, 2011. [Erratum: Eur. Phys. J. C73, p. 2501, 2013]. arXiv:1007.1727.

[3] J. Brehmer, K. Cranmer, G. Louppe, and J. Pavez: 'Constraining Effective Field Theories with Machine Learning'. Phys. Rev. Lett. 121 (11), p. 111801, 2018. arXiv:1805.00013.

[4] J. Brehmer, K. Cranmer, G. Louppe, and J. Pavez: 'A Guide to Constraining Effective Field Theories with Machine Learning'. Phys. Rev. D98 (5), p. 052004, 2018. arXiv:1805.00020.

[5] S. Agostinelli et al. (GEANT4): 'GEANT4: A Simulation toolkit'. Nucl. Instrum. Meth. A506, p. 250, 2003.

[6] J. Brehmer, K. Cranmer, F. Kling, and T. Plehn: 'Better Higgs boson measurements through information geometry'. Phys. Rev. D95 (7), p. 073002, 2017. arXiv:1612.05261.

[7] J. Brehmer, F. Kling, T. Plehn, and T. M. P. Tait: 'Better Higgs-CP Tests Through Information Geometry'. Phys. Rev. D97 (9), p. 095017, 2018. arXiv:1712.02350.

[8] D. B. Rubin: 'Bayesianly justifiable and relevant frequency calculations for the applied statistician'. Ann. Statist. 12 (4), p. 1151, 1984. URL `https://doi.org/10.1214/aos/1176346785`.

[9] M. A. Beaumont, W. Zhang, and D. J. Balding: 'Approximate bayesian computation in population genetics'. Genetics 162 (4), p. 2025, 2002.

[10] J. Alsing, B. Wandelt, and S. Feeney: 'Massive optimal data compression and density estimation for scalable, likelihood-free inference in cosmology' , 2018. arXiv:1801.01497.

[11] T. Charnock, G. Lavaux, and B. D. Wandelt: 'Automatic physical inference with information maximizing neural networks'. Phys. Rev. D97 (8), p. 083004, 2018. arXiv:1802.03537.

[12] Y. Fan, D. J. Nott, and S. A. Sisson: 'Approximate Bayesian Computation via Regression Density Estimation'. ArXiv e-prints , 2012. arXiv:1212.1479.

[13] L. Dinh, D. Krueger, and Y. Bengio: 'NICE: Non-linear Independent Components Estimation'. ArXiv e-prints , 2014. arXiv:1410.8516.

[14] M. Germain, K. Gregor, I. Murray, and H. Larochelle: 'MADE: Masked Autoencoder for Distribution Estimation'. ArXiv e-prints , 2015. arXiv:1502.03509.

[15] D. Jimenez Rezende and S. Mohamed: 'Variational Inference with Normalizing Flows'. ArXiv e-prints , 2015. arXiv:1505.05770.

[16] K. Cranmer and G. Louppe: 'Unifying generative models and exact likelihood-free inference with conditional bijections'. J. Brief Ideas , 2016.

[17] L. Dinh, J. Sohl-Dickstein, and S. Bengio: 'Density estimation using Real NVP'. ArXiv e-prints , 2016. arXiv:1605.08803.

[18] G. Papamakarios and I. Murray: 'Fast ϵ-free inference of simulation models with bayesian conditional density estimation'. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), 'Advances in Neural Information Processing Systems 29', Curran Associates, Inc., p. 1028–1036, 2016. URL `http://papers.nips.cc/paper/6084-fast-free-inference-of-simulation-models-with-bayesian-conditional-density-estimation.pdf`.

[19] B. Paige and F. Wood: 'Inference Networks for Sequential Monte Carlo in Graphical Models'. ArXiv e-prints , 2016. arXiv:1602.06701.

[20] R. Dutta, J. Corander, S. Kaski, and M. U. Gutmann: 'Likelihood-free inference by ratio estimation'. ArXiv e-prints , 2016. arXiv:1611.10242.

[21] B. Uria, M.-A. Côté, K. Gregor, I. Murray, and H. Larochelle: 'Neural Autoregressive Distribution Estimation'. ArXiv e-prints , 2016. arXiv:1605.02226.

[22] A. van den Oord, S. Dieleman, H. Zen, et al.: 'WaveNet: A Generative Model for Raw Audio'. ArXiv e-prints , 2016. arXiv:1609.03499.

[23] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu: 'Conditional Image Generation with PixelCNN Decoders'. ArXiv e-prints , 2016. arXiv:1606.05328.

[24] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu: 'Pixel Recurrent Neural Networks'. ArXiv e-prints , 2016. arXiv:1601.06759.

[25] M. U. Gutmann, R. Dutta, S. Kaski, and J. Corander: 'Likelihood-free inference via classification'. Statistics and Computing p. 1–15, 2017.

[26] D. Tran, R. Ranganath, and D. M. Blei: 'Hierarchical Implicit Models and Likelihood-Free Variational Inference'. ArXiv e-prints , 2017. arXiv:1702.08896.

[27] G. Louppe and K. Cranmer: 'Adversarial Variational Optimization of Non-Differentiable Simulators'. ArXiv e-prints , 2017. arXiv:1707.07113.

[28] G. Papamakarios, T. Pavlakou, and I. Murray: 'Masked Autoregressive Flow for Density Estimation'. ArXiv e-prints , 2017. arXiv:1705.07057.

[29] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville: 'Neural Autoregressive Flows'. ArXiv e-prints , 2018. arXiv:1804.00779.

[30] G. Papamakarios, D. C. Sterratt, and I. Murray: 'Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows'. ArXiv e-prints , 2018. arXiv:1805.07226.

[31] W. Grathwohl, R. T. Q. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud: 'FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models'. ArXiv e-prints , 2018. arXiv:1810.01367.

[32] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, et al.: 'Generative Adversarial Networks'. ArXiv e-prints , 2014. arXiv:1406.2661.

[33] S. Mohamed and B. Lakshminarayanan: 'Learning in Implicit Generative Models'. ArXiv e-prints , 2016. arXiv:1610.03483.

[34] K. Cranmer, J. Pavez, and G. Louppe: 'Approximating Likelihood Ratios with Calibrated Discriminative Classifiers' , 2015. arXiv:1506.02169.

[35] K. Kondo: 'Dynamical Likelihood Method for Reconstruction of Events With Missing Momentum. 1: Method and Toy Models'. J. Phys. Soc. Jap. 57, p. 4126, 1988.

[36] V. M. Abazov et al. (D0): 'A precision measurement of the mass of the top quark'. Nature 429, p. 638, 2004. arXiv:hep-ex/0406031.

[37] P. Artoisenet and O. Mattelaer: 'MadWeight: Automatic event reweighting with matrix elements'. PoS CHARGED2008, p. 025, 2008.

[38] Y. Gao, A. V. Gritsan, Z. Guo, K. Melnikov, M. Schulze, and N. V. Tran: 'Spin determination of single-produced resonances at hadron colliders'. Phys. Rev. D81, p. 075022, 2010. arXiv:1001.3396.

[39] J. Alwall, A. Freitas, and O. Mattelaer: 'The Matrix Element Method and QCD Radiation'. Phys. Rev. D83, p. 074010, 2011. arXiv:1010.2263.

[40] S. Bolognesi, Y. Gao, A. V. Gritsan, et al.: 'On the spin and parity of a single-produced resonance at the LHC'. Phys. Rev. D86, p. 095031, 2012. arXiv:1208.4018.

[41] P. Avery et al.: 'Precision studies of the Higgs boson decay channel $H \to ZZ \to 4l$ with MEKD'. Phys. Rev. D87 (5), p. 055006, 2013. arXiv:1210.0896.

[42] J. R. Andersen, C. Englert, and M. Spannowsky: 'Extracting precise Higgs couplings by using the matrix element method'. Phys. Rev. D87 (1), p. 015019, 2013. arXiv:1211.3011.

[43] J. M. Campbell, R. K. Ellis, W. T. Giele, and C. Williams: 'Finding the Higgs boson in decays to $Z\gamma$ using the matrix element method at Next-to-Leading Order'. Phys. Rev. D87 (7), p. 073005, 2013. arXiv:1301.7086.

[44] P. Artoisenet, P. de Aquino, F. Maltoni, and O. Mattelaer: 'Unravelling $t\bar{t}h$ via the Matrix Element Method'. Phys. Rev. Lett. 111 (9), p. 091802, 2013. arXiv:1304.6414.

[45] J. S. Gainer, J. Lykken, K. T. Matchev, S. Mrenna, and M. Park: 'The Matrix Element Method: Past, Present, and Future'. In 'Proceedings, 2013 Community Summer Study on the Future of U.S. Particle Physics: Snowmass on the Mississippi (CSS2013): Minneapolis,

MN, USA, July 29-August 6, 2013', , 2013. arXiv:1307.3546, URL http://inspirehep.net/record/1242444/files/arXiv:1307.3546.pdf.

[46] D. Schouten, A. DeAbreu, and B. Stelzer: 'Accelerated Matrix Element Method with Parallel Computing'. Comput. Phys. Commun. 192, p. 54, 2015. arXiv:1407.7595.

[47] T. Martini and P. Uwer: 'Extending the Matrix Element Method beyond the Born approximation: Calculating event weights at next-to-leading order accuracy'. JHEP 09, p. 083, 2015. arXiv:1506.08798.

[48] A. V. Gritsan, R. Röntsch, M. Schulze, and M. Xiao: 'Constraining anomalous Higgs boson couplings to the heavy flavor fermions using matrix element techniques'. Phys. Rev. D94 (5), p. 055023, 2016. arXiv:1606.03107.

[49] T. Martini and P. Uwer: 'The Matrix Element Method at next-to-leading order QCD for hadronic collisions: Single top-quark production at the LHC as an example application' , 2017. arXiv:1712.04527.

[50] D. E. Soper and M. Spannowsky: 'Finding physics signals with shower deconstruction'. Phys. Rev. D84, p. 074002, 2011. arXiv:1102.3480.

[51] D. E. Soper and M. Spannowsky: 'Finding top quarks with shower deconstruction'. Phys. Rev. D87, p. 054012, 2013. arXiv:1211.3140.

[52] D. E. Soper and M. Spannowsky: 'Finding physics signals with event deconstruction'. Phys. Rev. D89 (9), p. 094005, 2014. arXiv:1402.1189.

[53] C. Englert, O. Mattelaer, and M. Spannowsky: 'Measuring the Higgs-bottom coupling in weak boson fusion'. Phys. Lett. B756, p. 103, 2016. arXiv:1512.03429.

[54] D. Atwood and A. Soni: 'Analysis for magnetic moment and electric dipole moment form-factors of the top quark via $e^+e^- \to t\bar{t}$'. Phys. Rev. D45, p. 2405, 1992.

[55] M. Davier, L. Duflot, F. Le Diberder, and A. Rouge: 'The Optimal method for the measurement of tau polarization'. Phys. Lett. B306, p. 411, 1993.

[56] M. Diehl and O. Nachtmann: 'Optimal observables for the measurement of three gauge boson couplings in $e^+e^- \to W^+W^-$'. Z. Phys. C62, p. 397, 1994.

[57] J. Brehmer, G. Louppe, J. Pavez, and K. Cranmer: 'Mining gold from implicit models to improve likelihood-free inference' , 2018. arXiv:1805.12244.

[58] M. Stoye, J. Brehmer, G. Louppe, J. Pavez, and K. Cranmer: 'Likelihood-free inference with an improved cross-entropy estimator' , 2018. arXiv:1808.00973.

[59] P. Baldi, K. Cranmer, T. Faucett, P. Sadowski, and D. Whiteson: 'Parameterized neural networks for high-energy physics'. Eur. Phys. J. C76 (5), p. 235, 2016. arXiv:1601.07913.

[60] J. Alsing and B. Wandelt: 'Generalized massive optimal data compression'. Mon. Not. Roy. Astron. Soc. 476 (1), p. L60, 2018. arXiv:1712.00012.

[61] G. Aad et al. (ATLAS): 'A morphing technique for signal modelling in a multidimensional space of coupling parameters', 2015. Physics note ATL-PHYS-PUB-2015-047. URL http://cds.cern.ch/record/2066980.

[62] J. Brehmer, F. Kling, I. Espejo, and K. Cranmer: In progress.

[63] J. Alwall, R. Frederix, S. Frixione, et al.: 'The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations'. JHEP 07, p. 079, 2014. arXiv:1405.0301.

[64] T. Sjostrand, S. Mrenna, and P. Z. Skands: 'A Brief Introduction to PYTHIA 8.1'. Comput. Phys. Commun. 178, p. 852, 2008. arXiv:0710.3820.

[65] J. de Favereau, C. Delaere, P. Demin, et al. (DELPHES 3): 'DELPHES 3, A modular framework for fast simulation of a generic collider experiment'. JHEP 02, p. 057, 2014. arXiv:1307.6346.

[66] J. Brehmer, F. Kling, I. Espejo, and K. Cranmer: 'MadMiner code repository'. URL `https://github.com/johannbrehmer/madminer`.

[67] J. Brehmer, F. Kling, I. Espejo, and K. Cranmer: 'MadMiner technical documentation'. URL `https://madminer.readthedocs.io/en/latest/`.

[68] I. Espejo, J. Brehmer, and K. Cranmer: 'MadMiner Docker repositories'. URL `https://hub.docker.com/u/madminertool`.

[69] I. Espejo, J. Brehmer, F. Kling, and K. Cranmer: 'MadMiner Reana deployment'. URL `https://github.com/irinaespejo/workflow-madminer`.

[70] J. Brehmer, S. Dawson, S. Homiller, F. Kling, and T. Plehn: In progress.

[71] J. Brehmer, K. Cranmer, M. Farina, F. Kling, D. Pappadopulo, and J. Ruderman: In progress.