Eur. Phys. J. C (2023) 83:1158
https://doi.org/10.1140/epjc/s10052-023-12314-z

THE EUROPEAN
PHYSICAL JOURNAL C

Regular Article - Theoretical Physics

# Machine-learned exclusion limits without binning

**Ernesto Arganda**[1,2,3,a], **Andres D. Perez**[1,2,3,b], **Martín de los Rios**[1,2,c], **Rosa María Sandá Seoane**[2,d]

[1] Departamento de Física Teórica, Universidad Autónoma de Madrid, Cantoblanco, 28049 Madrid, Spain
[2] Instituto de Física Teórica UAM-CSIC, C/ Nicolás Cabrera 13-15, Campus de Cantoblanco, 28049 Madrid, Spain
[3] IFLP, CONICET-Dpto. de Física, Universidad Nacional de La Plata, C.C. 67, 1900 La Plata, Argentina

Received: 14 November 2022 / Accepted: 1 December 2023 / Published online: 19 December 2023
© The Author(s) 2023

**Abstract** Machine-learned likelihoods (MLL) combines machine-learning classification techniques with likelihood-based inference tests to estimate the experimental sensitivity of high-dimensional data sets. We extend the MLL method by including kernel density estimators (KDE) to avoid binning the classifier output to extract the resulting one-dimensional signal and background probability density functions. We first test our method on toy models generated with multivariate Gaussian distributions, where the true probability distribution functions are known. Later, we apply the method to two cases of interest at the LHC: a search for exotic Higgs bosons, and a $Z'$ boson decaying into lepton pairs. In contrast to physical-based quantities, the typical fluctuations of the ML outputs give non-smooth probability distributions for pure-signal and pure-background samples. The non-smoothness is propagated into the density estimation due to the good performance and flexibility of the KDE method. We study its impact on the final significance computation, and we compare the results using the average of several independent ML output realizations, which allows us to obtain smoother distributions. We conclude that the significance estimation turns out to be not sensible to this issue.

## Contents

## 1 Introduction

Modern machine learning (ML) has become a fundamental tool in experimental and phenomenological analyses of high-energy physics (for reviews see, for instance, [1–9] and for pioneer papers see [10–12]). The ML algorithms can be applied not only to event-by-event collider analyses but also used at the event-ensemble level [13–20]. In order to estimate the experimental sensitivity to potential new-physics signals at colliders, several studies have recently appeared that combine the use of ML classifiers with traditional statistical tests [21–36]. More specifically, it was shown in [21] that the calibration of classifiers trained to distinguish signal and background samples under the relevant hypotheses ensures to proper estimate the likelihood ratio and consequently can be used to compute a statistical significance.

Recently, a simplification of [21] has been proposed in [36], the so-called machine-learned likelihoods (MLL), which computes the expected experimental sensitivity through the use of ML classifiers, utilizing the entire discriminant output. A single ML classifier estimates the individual probability densities and subsequently one can calculate the statistical significance for a given number of signal and background events ($S$ and $B$, respectively) with traditional hypothesis tests. By construction, the output of the classifier is always one-dimensional, so we reduce the hypothesis test to a single parameter of interest, the signal strength $\mu$. On the one hand, it is simply and reliably applicable to any high-dimensional

[a] e-mail: ernesto.arganda@uam.es (corresponding author)

[b] e-mail: andresd.perez@uam.es

[c] e-mail: martin.delosrios@uam.es

[d] e-mail: r.sanda@csic.es

problem. On the other hand, using all the information available from the ML classifier does not require defining working points like traditional cut-based analyses. The ATLAS and CMS Collaborations incorporate similar methods in their experimental analyses but consider only the classifier output as a good variable to bin and fit the binned likelihood formula (see, for instance, Refs. [37–44]).

The MLL code [45] developed in [36] only includes the calculation of the discovery hypothesis test, although the expressions needed to calculate the exclusion limits were provided. In [46] we extend the MLL method by adding the exclusion hypothesis test. It is well-known that unbinned methods could provide a better performance than binned ones since the loss of information is minimized. In that sense, in this work we improve the MLL method with the use of kernel density estimators (KDE) [47,48], in order to avoid binning the ML classifier output for extracting the resulting one-dimensional signal and background probability density functions (PDFs), as proposed in [36,46]. The implementation of unbinned methods to the ML output space has intrinsic difficulties that are usually not present if one considers physical based features, specifically the stochasticity of the machine learning training introduces fluctuations, even when the classifier approaches its optimal limit. These fluctuations translate to non-smooth distribution functions, that in turn, are propagated by the KDE into the density estimation given the plasticity of this consistent non-parametric method [49]. Therefore, it is necessary to analyze the impact of the lack of smoothness in the statistical analysis. We propose to tackle this issue by working with a variable build from the average of several independent machine-learning realizations, that gives smoother PDFs.

We would like to highlight that binned methods are commonly used since one can usually optimize the binning to extract nearly all of the benefits of the unbinned approach, but this optimization can be a highly non-trivial and scenario-dependent task. The incorporation of KDE within our framework allows to automatically elude any binning optimization and outperform some of the most common binning schemes. For illustration, we compare the results of our unbinned MLL method with the results obtained by doing linear and non-linear binnings in the toy examples used to validate our setup, where the true PDFs are known.

The structure of the paper is the following: Sect. 2 is devoted to summarizing the main features of the MLL method with the relevant expressions for the calculation of exclusion limits and the implementation of KDE in it. In Sect. 3 we show the performance of the MLL method with KDE and analyze the application of this unbinned method to the ML output space in different examples: in Sect. 3.1 a case where the true probability density functions (PDFs) are known, through a toy model generated with multivariate Gaussian distributions; in Sect. 3.2 we present an LHC analysis for the search for new heavy neutral Higgs bosons at $\sqrt{s}$ = 8 TeV and luminosity of 20 fb$^{-1}$, estimating not only exclusion limits, but also comparing our results with those report in [12]; and in Sect. 3.3 we present an HL-LHC study for Sequential Standard Model (SSM) [50] $Z'$ bosons decaying into lepton pairs, comparing the MLL+KDE performance for estimating 95% CL exclusion limits with the results obtained applying a binned likelihood to the machine learning classifier output and also with respect to the projections reported by the ATLAS Collaboration for an LHC center-of-mass energy of $\sqrt{s} = 14$ TeV with a total integrated luminosity of $\mathcal{L} = 3$ ab$^{-1}$ [51]. Finally, Sect. 4 summarizes our more important results and conclusions.

## 2 Method

In this section, we present the corresponding formulae for the estimation of exclusion sensitivities with the MLL method, first introduced in [36,46]. We summarize the main features of the method which allows dealing with data of arbitrarily high dimension through a simple ML classifier while using the traditional inference tests to compare a null hypothesis (the signal-plus-background one) against an alternative one (the background-only one). We also present the details of the implementation of KDE to obtain the unbinned posterior probability distributions from the classifier output, needed to compute the corresponding likelihood functions.

Following the statistical model in [52], we can define the likelihood $\mathcal{L}$ of $N$ independent measurements with an arbitrarily high-dimensional set of observables $x$ as

$$\mathcal{L}(\mu, s, b) = \text{Poiss}\big(N|\mu S + B\big) \prod_{i=1}^{N} p(x_i|\mu, s, b), \tag{1}$$

where $S$ ($B$) is the expected total signal (background) yield, Poiss stands for a Poisson probability mass function, and $p(x|\mu, s, b)$ is the probability density for a single measurement $x$, where $\mu$ defines the hypothesis we are testing for.

We can model the probability density containing the event-by-event information as a mixture of signal and background densities

$$p(x|\mu, s, b) = \frac{B}{\mu S + B}\, p_b(x) + \frac{\mu S}{\mu S + B}\, p_s(x), \tag{2}$$

where $p_s(x) = p(x|s)$ and $p_b(x) = p(x|b)$ are, respectively, the signal and background probability density functions (PDFs) for a single measurement $x$, and $\frac{\mu S}{\mu S+B}$ and $\frac{B}{\mu S+B}$ are the probabilities of an event being sampled from the corresponding probability distributions.

To derive upper limits on $\mu$, and in particular considering additive new physics scenarios ($\mu \geq 0$), we need to consider the following test statistic for exclusion limits [53]:

$$\tilde{q}_\mu = \begin{cases} 0 & \text{if } \hat{\mu} > \mu, \\ -2 \text{ Ln } \frac{\mathcal{L}(\mu,s,b)}{\mathcal{L}(\hat{\mu},s,b)} & \text{if } 0 \leq \hat{\mu} \leq \mu, \\ -2 \text{ Ln } \frac{\mathcal{L}(\mu,s,b)}{\mathcal{L}(0,s,b)} & \text{if } \hat{\mu} < 0, \end{cases} \tag{3}$$

where $\hat{\mu}$ is the parameter that maximizes the likelihood in Eq. (1)

$$\sum_{i=1}^{N} \frac{p_s(x_i)}{\hat{\mu} S \, p_s(x_i) + B \, p_b(x_i)} = 1. \tag{4}$$

Considering our choice for the statistical model in Eq. (1), $\tilde{q}_\mu$ turns out

$$\tilde{q}_\mu = \begin{cases} 0 & \text{if } \hat{\mu} > \mu \\ 2(\mu - \hat{\mu})S \\ -2 \sum_{i=1}^{N} \\ \text{Ln } \left( \frac{B p_b(x_i) + \mu S p_s(x_i)}{B p_b(x_i) + \hat{\mu} S p_s(x_i)} \right) & \text{if } 0 \leq \hat{\mu} \leq \mu \\ 2\mu S - 2 \sum_{i=1}^{N} \text{ Ln } \left(1 + \frac{\mu S p_s(x_i)}{B p_b(x_i)}\right) & \text{if } \hat{\mu} < 0; \end{cases} \tag{5}$$

Since $p_{s,b}(x)$ are typically not known, the base idea of our method in [36] is to replace these densities for the one-dimensional manifolds that can be obtained for signal and background from a machine-learning classifier. After training the classifier with a large and balanced data set of signal and background events, it can be obtained the classification score $o(x)$ that maximizes the binary cross-entropy (BCE) and thus approaches [21,54]

$$o(x) = \frac{p_s(x)}{p_s(x) + p_b(x)}, \tag{6}$$

as the classifier approaches its optimal performance. The dimensionality reduction can be done by dealing with $o(x)$ instead of $x$, using

$$p_s(x) \to \tilde{p}_s(o(x)), \quad \text{and} \quad p_b(x) \to \tilde{p}_b(o(x)), \tag{7}$$

where $\tilde{p}_{s,b}(o(x))$ are the distributions of $o(x)$ for signal and background, obtained by evaluating the classifier on a set of pure signal or background events, respectively. Notice that this allows us to approximate both signal and background distributions individually, retaining the full information contained in both densities, without introducing any working point. These distributions are one-dimensional, and therefore can always be easily handled and incorporated into the test statistic in Eq. (5)

$$\tilde{q}_\mu = \begin{cases} 0 & \text{if } \hat{\mu} > \mu \\ 2(\mu - \hat{\mu})S \\ -2 \sum_{i=1}^{N} \text{ Ln} \\ \left( \frac{B \tilde{p}_b(o(x_i)) + \mu S \tilde{p}_s(o(x_i))}{B \tilde{p}_b(o(x_i)) + \hat{\mu} S \tilde{p}_s(o(x_i))} \right) & \text{if } 0 \leq \hat{\mu} \leq \mu \\ 2\mu S - 2 \sum_{i=1}^{N} \text{ Ln } \left(1 + \frac{\mu S \tilde{p}_s(o(x_i))}{B \tilde{p}_b(o(x_i))}\right) & \text{if } \hat{\mu} < 0; \end{cases} \tag{8}$$

as well as into the condition on $\hat{\mu}$ from Eq. (4)

$$\sum_{i=1}^{N} \frac{\tilde{p}_s(o(x_i))}{\hat{\mu} S \, \tilde{p}_s(o(x_i)) + B \, \tilde{p}_b(o(x_i))} = 1. \tag{9}$$

The test statistic in Eq. (8) is estimated through a finite data set of $N$ events and thus has a probability distribution conditioned on the true unknown signal strength $\mu'$. For a given hypothesis described by the $\mu'$ value, we can estimate numerically the $\tilde{q}_\mu$ distribution. When the true hypothesis is assumed to be the background-only one ($\mu' = 0$), the median expected exclusion significance med $[Z_\mu|0]$ is defined as

$$\text{med } \left[Z_\mu|0\right] = \sqrt{\text{med } \left[\tilde{q}_\mu|0\right]}, \tag{10}$$

where we estimate the $\tilde{q}_\mu$ distribution by generating a set of pseudo-experiments with background-only events. Then, to set upper limits to a certain confidence level, we select the lowest $\mu$ which achieves the required median expected significance.

It is worth remarking that the output of the machine learning classifier, for a given set of events, gives us a sample of the desired PDFs $\tilde{p}_{s,b}(o(x))$. Hence, to apply Eq. (8) we first need to extract the classifier posteriors. As these samples are one-dimensional, we can always compute binned PDFs, as was done in [36]. Binning the output variable is a typical procedure when using ML tools. Nevertheless, it is also possible to compute the PDFs through other parametric (such as mixture models [55]) or non-parametric methods (such as kernel density estimation (KDE) [47,48] or neural density estimation [49]). In comparison with other density-estimation methods, KDE has the advantage of not assuming any functional form for the PDF, in contrast with the mixture of Gaussian methods, while keeping the computation and the interpretation simple, as opposed to neural density estimation methods. For this reason, in this work, we made extensive use of the KDE method,[1] through its `scikit-learn` implementation [57].

Given a set of $N$ events that were previously classified by the machine learning as signal (background) events, the PDF estimated by the KDE method is defined as

$$p_{s,b}(o(x)) = \frac{1}{N} \sum_{i}^{N} \kappa_\epsilon \left[o(x) - o(x_i)\right] \tag{11}$$

where $\kappa_\epsilon$ is a kernel function that depends on the "smoothing" scale, or bandwidth parameter $\epsilon$. There are several different options for the kernel function. In this work, we used

[1] During the completion of this work, Ref. [56] has appeared, in which data-driven methods are used for dealing with unbinned multivariate EFT observables. Although this work is similar to ours in the spirit of avoiding the binning of multidimensional variables, there are significant differences in the application of ML and KDE algorithms between our MLL+KDE method and the ML4EFT framework of [56].

the *Epanechnikov kernel* [58] as it is known to be the most efficient kernel [49]. This kernel is defined as

$$\kappa_\epsilon(u) = \begin{cases} \frac{1}{\epsilon}\frac{3}{4}\left(1-(u/\epsilon)^2\right), & \text{if } |u| \leq \epsilon \\ 0, & \text{otherwise} \end{cases}$$

It is important to remark that the "bandwidth" parameter $\epsilon$ censors the degree of smoothness. Hence, a very low $\epsilon$ will overfit the data, whereas a very high $\epsilon$ will underfit it. In all our examples the $\epsilon$ was selected through a grid search done using the `GridSearchCV` function inside the `sklearn.model_selection` python package. Given a value for $\epsilon$, this function estimates the log-likelihood of the data using a 5-fold cross-validation strategy, i.e. the data set is split into 5 smaller sets, 4 are used to fit the KDE which is then validated on the remaining part of the data. Finally, the function gives as an output the $\epsilon$ which maximizes the data likelihood. Also is worth remarking that although KDEs method suffers from the curse of dimensionality, we are applying such technique to the one-dimensional output of the machine learning classifier to avoid this problem.

Notice that the machine learning training (and hence the machine learning predictions) is a stochastic process that introduces small fluctuations around the optimal limit. These in turn could translate to non-smooth PDFs. To tackle this issue, the same procedure described above can be done when using an ensemble of $N$ base classifiers trained on random subsets of the original data set, that average their individual predictions to form a final prediction. In this case, $o(x)$ can simply be replaced by $< o(x) >= \frac{1}{N}\sum_i^N o_i(x)$, which in turns gives smoother PDFs $\tilde{p}_{s,b}(< o(x) >)$.

For completeness, we also introduce here the median exclusion significance estimation for the traditional binned likelihood (BL) method and the use of Asimov data sets [53], which will be used to compare our technique

$$\text{med}\left[Z_\mu|0\right] = \sqrt{\tilde{q}_\mu} = \left[2\sum_{d=1}^{D}\left(B_d \,\text{Ln}\left(\frac{B_d}{S_d+B_d}\right)+S_d\right)\right]^{1/2}, \tag{12}$$

where $S_d$ and $B_d$ are the expected number of signal and background events in each bin $d$. This approximation is very effective but runs into trouble when the dimension of the data grows, which is known as the curse of dimensionality since the number of data points required to reliably populate the bins scales exponentially with the dimension of the observables $x$. This is a non-existent problem in our method, which always reduces the original dimension to one as stated in Eq. (7), allowing the application of the BL method to the classifier output, as also done by experimental collaborations when using ML methods, as mentioned in Sect. 1.

## 3 Application examples

### 3.1 Known true PDFs: multivariate Gaussian distributions

To show the performance of the MLL method with KDE we first analyze toy models generated with multivariate Gaussian distributions of different dimensions,

$$\mathcal{N}_{dim}(\boldsymbol{m}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \, exp\left(-\frac{1}{2}(x-m)^T\boldsymbol{\Sigma}^{-1}(x-m)\right), \tag{13}$$

with mean $\boldsymbol{m}$, and covariance matrix $\boldsymbol{\Sigma}$.

We start with the simplest case, consisting of an abstract two-dimensional space $(x_1, x_2)$. Events are generated by Gaussian distributions $\mathcal{N}_2(\boldsymbol{m}, \boldsymbol{\Sigma})$, with $\boldsymbol{m} = +0.3(-0.3)$ and no correlation, i.e., covariance matrices $\boldsymbol{\Sigma} = \mathbb{I}_{2\times 2}$ for $S$ ($B$). We trained supervised per-event classifiers, `XGBoost`, with 1 M events per class (balanced data set), to distinguish $S$ from $B$. The PDFs obtained from the classifier output, $o(x)$, can be found in the top left panel of Fig. 1, for two new independent data sets of pure signal (blue) and pure background (red) events.

Since in this example we know the true underlying distributions in the original multidimensional space, we can test Eq. (6). In the right panel of Fig. 1 we show, in green dots, the output of one machine learning realization vs. the right-hand-side of Eq. (6) estimated with the real signal and background probability functions. We can observe that the classifier approaches the optimal limit, although there are some small fluctuations around the 1-to-1 line. These fluctuations are independent of the sampling of the data and come from the stochasticity inherent to any machine learning training process. In turn, these fluctuations translate to non-smooth PDFs for the machine learning output of background and signal events, as can be seen in the red and blue shadow histograms in the top left panel of Fig. 1.

As explained before, to solve this issue we can take advantage of ensembles, and build a variable from the average output of ten independent machine learning realizations, define as $< o(x) >= \frac{1}{10}\sum_i^{10} o_i(x)$. It can be seen in the red and blue shadow histograms of the bottom left panel of Fig. 1 that, with this definition, the small fluctuations are washed out resulting in smoother PDFs. For completeness on both left panels, we also present the estimations of $\tilde{p}_{s,b}(o(x))$ using the true PDFs (orange and purple solid lines), the KDE over the machine learning output $o(x)$ (red and blue dashed curves of top left panel), and KDE over the average variable $< o(x) >$ (red and blue dashed lines of the bottom left panel). On the one hand, it can be seen that, due to the flexibility of the KDE method, when fitting the machine learning output $o(x)$ the resulting distributions follows the fluctuations around the true PDFs. On the other hand, it can be seen

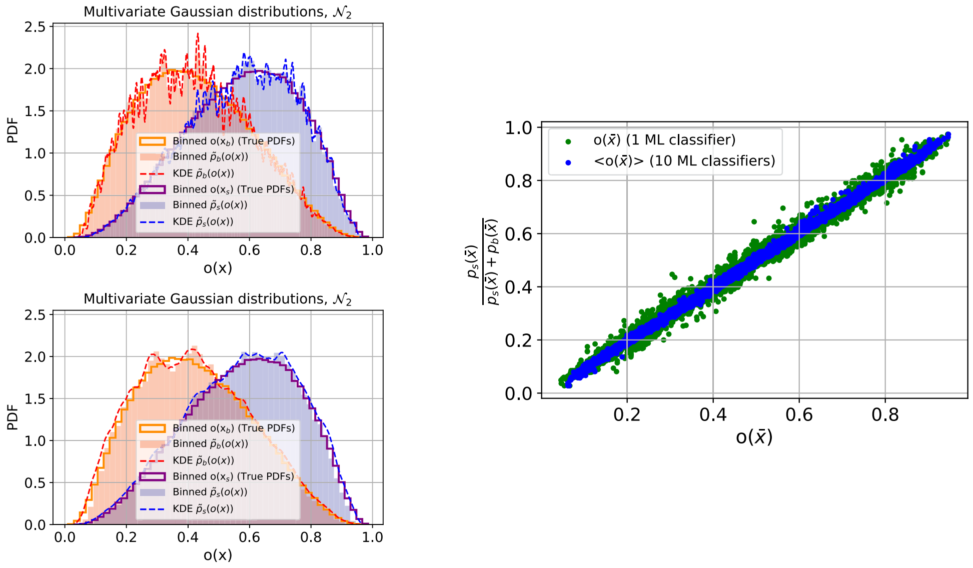


**Fig. 1** Results for the $\mathcal{N}_2(\boldsymbol{m}, \boldsymbol{\Sigma})$ case. Top left panel: output of a single `XGBoost` classifier. Bottom left panel: averaged output of 10 `XGBoost` classifiers, defined as $o(x) = \frac{1}{10}\sum_i^{10} o_i(x)$. Right panel: comparison between our trained classifier output and the mathematically optimal performance defined in Eq. (6)

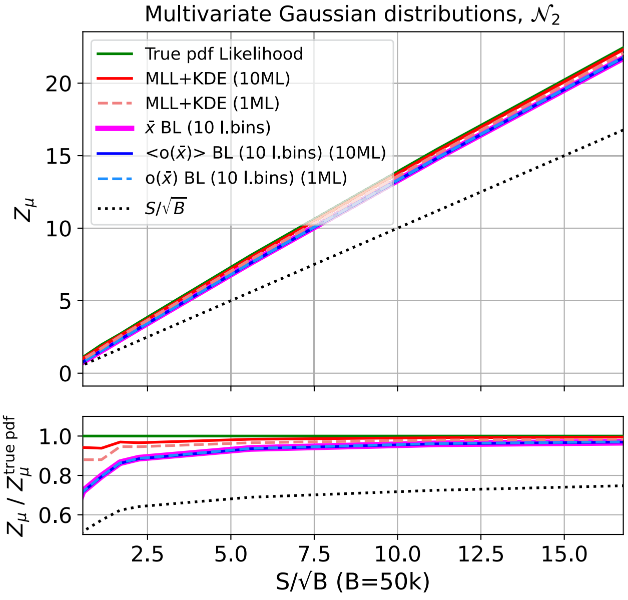


**Fig. 2** Exclusion-limit significance for $\mathcal{N}_2(\boldsymbol{m}, \boldsymbol{\Sigma})$ with $\boldsymbol{m} = +0.3(-0.3)$ for $S$ $(B)$ and no correlation, for fixed $\langle B \rangle = 50$ k, and different signal strengths $\langle S \rangle$. The red curves show the result of implementing the MLL+KDE method, while the blue and magenta curves represent the results obtained by applying the BL method to the classifier's one-dimensional output and the original two-dimensional space, respectively. Dashed curves use the output of a single classifier, while solid lines use the averaged output of 10 classifiers. For comparison, we include the green solid curve with the results obtained using the true PDFs

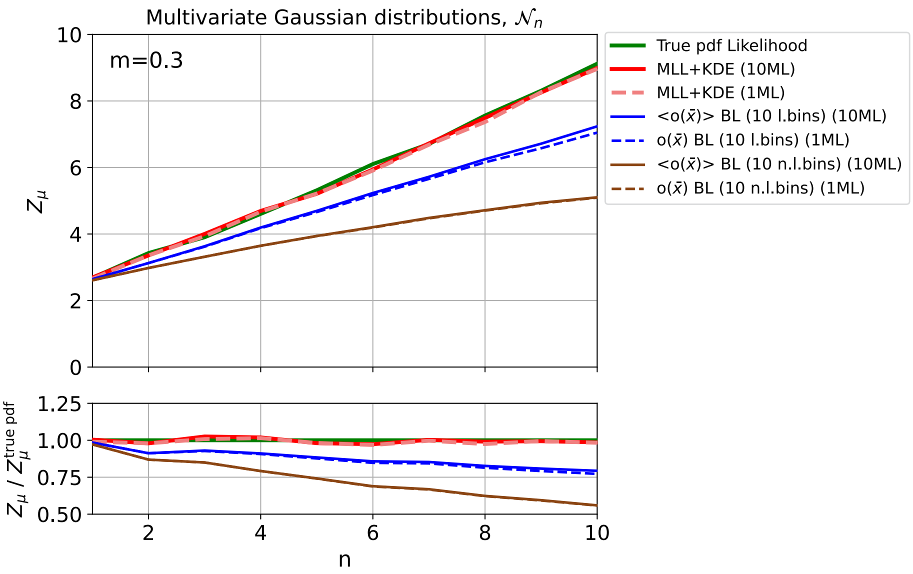
that, the KDE distributions obtained when fitting the average variable are smooth and closely approach the true PDFs.

In Fig. 2 we show the results for the MLL exclusion significance with KDE considering an example with a fixed background of $\langle B \rangle = 50$ k and different signal strengths. We also include the significance calculated using the true probability density functions in Eq. (3), and the results employing a binned Poisson log-likelihood of the original two-dimensional space $(x_1, x_2)$ with Eq. (12), which is possible to compute in this simple scenario. For completeness, we also include the results binning the one-dimensional ML output variable for obtaining the PDFs as in [36,46]. As can be seen, since we are analyzing a simple example, the significances estimated with all the methods are indistinguishable from the ones estimated with the true PDFs, which is expected given the low dimensionality of the space.

We would like to highlight that the significance does not change significantly if we employ either $o(x)$ computed with a single ML classifier, or with the averaged variable $< o(x) >$ calculated ensembling several ML trainings. In addition, for both the MLL+KDE and the true PDF methods, the significance is estimated by generating a set of pseudo-experiments with a finite-size number of events. This introduces a small statistical fluctuation due to the randomness of the sample.

The advantage of the MLL+KDE method against traditional approaches appears when dealing with $dim = n$, with $n > 2$. In Fig. 3 we present the exclusion significance for higher dimensional data generated with $\mathcal{N}_n(\boldsymbol{m}, \boldsymbol{\Sigma})$, no correlation $\boldsymbol{\Sigma} = \mathbb{I}_{n\times n}$, and $\boldsymbol{m} = +0.3(-0.3)$ for $S$ $(B)$.

It is worth reminding that the binned-poisson likelihood method becomes intractable in the original high-dimensional space. Also, it is interesting to note that, the results with the MLL+KDE method approach the ones with the true generative functions for all the analyzed dimensions. It is important to highlight that the ML output is always one-dimensional regardless of the dimension of the input data and, hence, can always be easily binned. For completeness, we show in Fig. 3 the significances obtained by applying a BL method to the machine learning output with two different types of binning: a linear binning where all bins have the same size (in the one-dimensional output space), and a standard non-linear approach where all bins have the same number of background



s

**Fig. 3** Exclusion-limit significance for $\mathcal{N}_{dim}(\boldsymbol{m}, \boldsymbol{\Sigma})$ with $\boldsymbol{m} = +0.3(-0.3)$ for $S$ $(B)$ and no correlation, as a function of the $dim$, for fixed $\langle B \rangle = 50$ k and $\langle S \rangle = 500$. The red curves show the result of implementing the MLL+KDE method, while the blue and brown curves represent the results obtained by applying the BL method to the classifier one-dimensional output for 10 linear and no-linear bins, respectively. Dashed curves use the output of a single classifier, while solid lines use the averaged output of 10 classifiers. For comparison, we include the green solid curve with the results obtained using the true PDFs

events (a binning strategy typically used by experimental collaborations since it avoids the presence of low-statistic bins in the background estimation, which in turns constraints systematic uncertainties). As can be seen, binning the output of the machine learning results in a non-negligible drop in significance. This can be understood as the binning introduce a loss of information due to a resolution effect. For this example, the linear binning turns out to be more effective for the BL method. In addition, and as in the $n = 2$ example, for MLL+KDE the use of an ensemble of machine learning realizations to obtain smoother PDFs does not change the results obtained with one single classifier. The same is verified when using the BL method over the average variable, although this behavior is expected since this method creates histograms from the distributions.

In the left and right panels of Fig. 4 we show the impact in the previous example of increasing the number of bins when applying BL to the classifier output, both for linear and non-linear bins, respectively. As stated before, linear binning proves to be a better sampling choice since its result approaches the ones obtained with the MLL+KDE method and with the true PDFs, when increasing the number of bins. Regarding the bins with the same number of background events, even though performance improves with more bins, the results are worst than its linear binning counterpart. This example shows the difficulties arising when trying to find an optimal binning that is not known a priori, and this highlights the advantage of using MLL+KDE, which although computationally expensive (when tuning the bandwidth parameter), sets an upper limit in the significance that can be achieved. It is also possible to automatically choose the optimal number of bins for histograms, as well as to tune the width of each bin, in a similar fashion as done for the $\epsilon$ parameter in the KDE method. The results of this analysis can be found in Appendix A, where we show that the significances obtained optimizing the bin widths is similar to the ones assuming equal-sized bins, and hence, the MLL-KDE method still offers the best significances when compared to different binned multivariate approaches.

Finally, in the right panel of Fig. 5 we show a case with correlation, $\mathcal{N}_n(\boldsymbol{m}, \boldsymbol{\Sigma})$, with $\boldsymbol{m} = +0.7(-0.7)$ for $S$ $(B)$, and $\boldsymbol{\Sigma}_{ij} = 1$ if $i = j$ and 0.5 if $i \neq j$. Comparing with the same example without correlation in the left panel of Fig. 5, the correlation makes the signal and background more difficult to distinguish, hence we obtain lower significance values, with MLL+KDE still offering the best performance.

Although these are toy models they allow us to understand the performance of MLL with KDE method over problems of different complexity and demonstrate its improvement with respect to the BL method applied to the classifier output. Particularly the MLL+KDE has a stable behavior when increasing the dimensionality of the input space, as well as when increasing the separation of the signal and background distributions on the original abstract variables. On the other hand, the BL method applied to the classifier output departs from the results obtained with the true PDFs as the number of

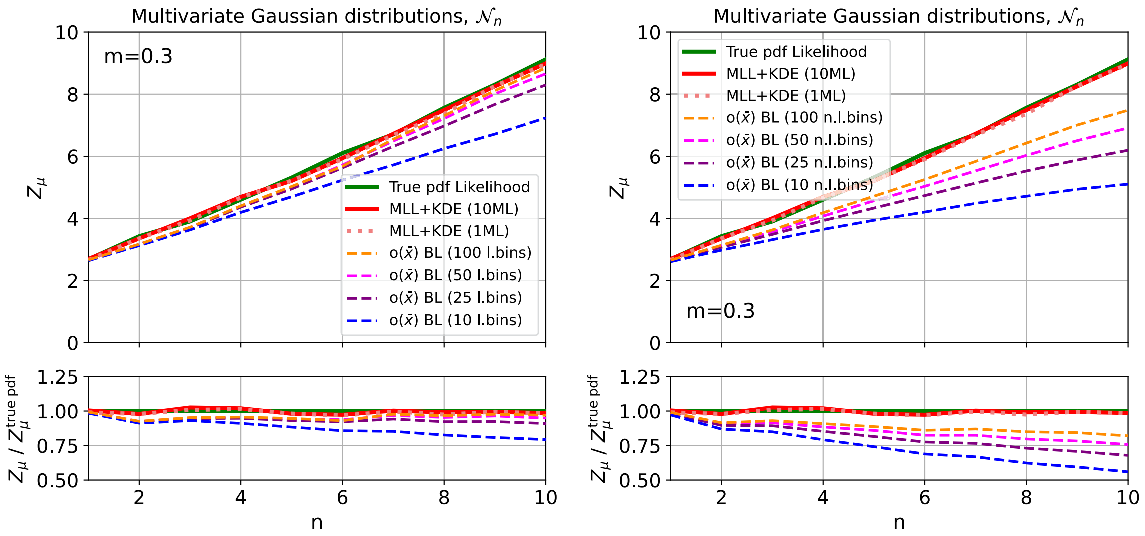


**Fig. 4** Exclusion-limit significance for $\mathcal{N}_{dim}(\boldsymbol{m}, \boldsymbol{\Sigma})$ with $\boldsymbol{m} = +0.3(-0.3)$ for $S$ $(B)$ and no correlation, as a function of the $dim$, for fixed $\langle B \rangle = 50$ k and $\langle S \rangle = 500$. The red solid curve shows the result of implementing the MLL+KDE method, while the green curve shows the results obtained using the true PDFs. Dashed color curves represent the results obtained by applying the BL method to the classifier's one-dimensional output for different bin numbers. Left panel: linear binning. Right panel: non-linear binning (same number of B events per bin)

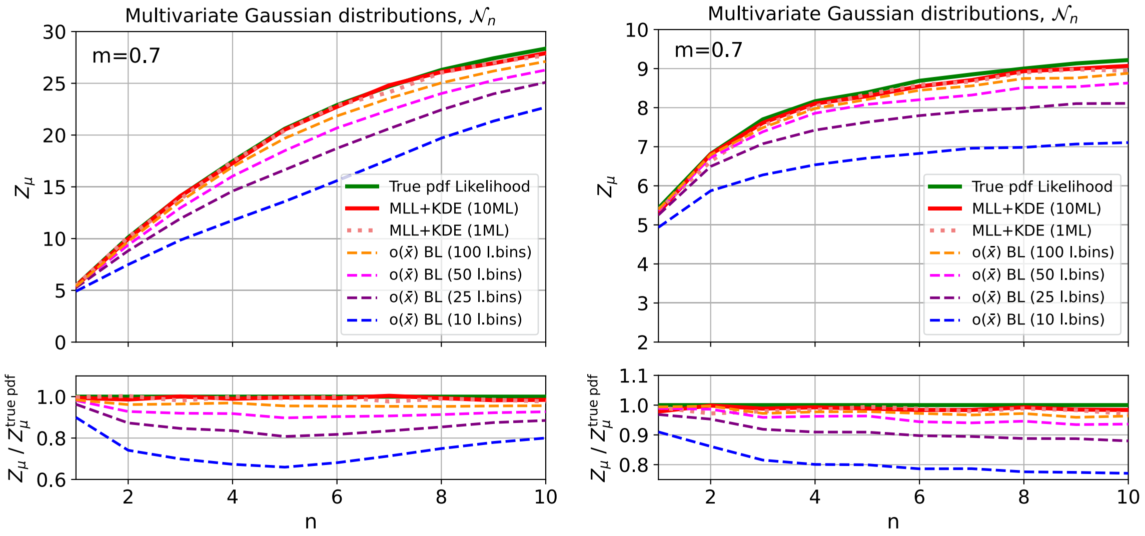


**Fig. 5** Exclusion-limit significance for $\mathcal{N}_{dim}(\boldsymbol{m}, \boldsymbol{\Sigma})$ with $\boldsymbol{m} = +0.7(-0.7)$ for $S$ $(B)$, as a function of the $dim$, for fixed $\langle B \rangle = 50$ k and $\langle S \rangle = 500$. The red solid curve shows the result of implementing the MLL+KDE method, while the green curve shows the results obtained using the true PDFs. Dashed color curves represent the results obtained by applying the BL method (with linear bins) to the classifier one-dimensional output for different bin numbers. Left panel: covariance matrices $\boldsymbol{\Sigma} = \mathbb{I}_{2\times 2}$ (no correlation). Right panel: covariance matrices $\boldsymbol{\Sigma}_{ij} = 1$ if $i = j$ and 0.5 if $i \neq j$

dimensions and separation of signal and background samples increases. The number of bins to use is another limitation, non-existent in our method that uses a non-parametric technique for PDF extraction. We also tested that although the KDE method is sensible to the fluctuations inherent to the machine learning classifier output, the lack of smoothness of the extracted PDFs does not affect the estimation of the significance within our framework.

## 3.2 New exotic Higgs bosons at the LHC

In this section, we apply our method in the search for an exotic electrically-neutral heavy Higgs boson ($H^0$) at the

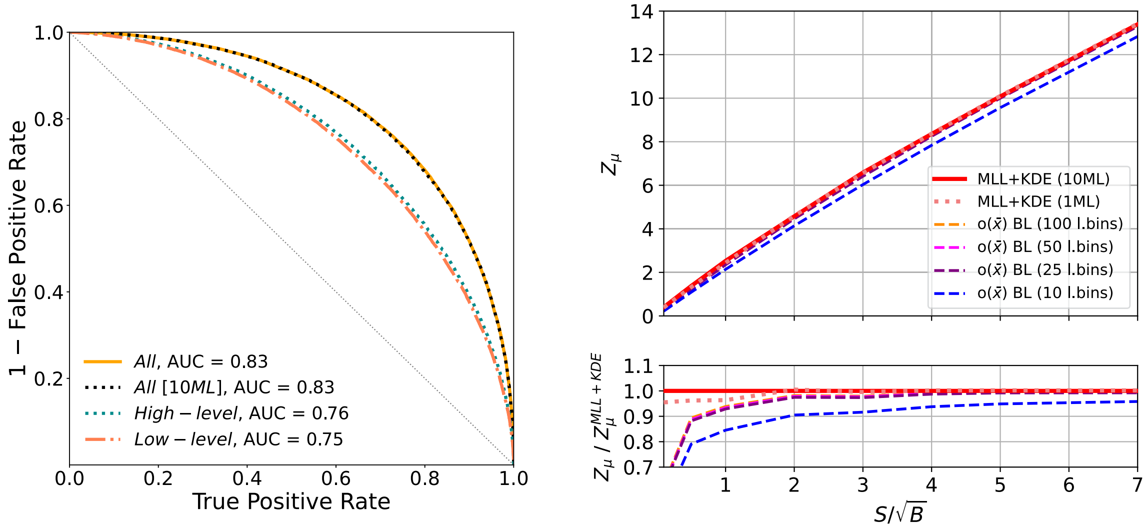


**Fig. 6** Left panel: ROC curves for the `XGBoost` classifiers associated with each possible data representation, trained to discriminate between $H^0$ and $t\bar{t}$ productions. Right panel: Exclusion limits for the search for a heavy neutral $H^0$ with MLL+KDE method (red solid line corresponds to the averaged variable of 10 ML, and dotted orange line corresponds to 1 ML), and with the BL fit for different number of linear bins (dashed curves), for fixed $\langle B \rangle = 86$ k, and different signal strengths $\langle S \rangle$

LHC, which subsequently decays to a $W$ boson and a heavy electrically-charged Higgs boson ($H^\pm$). This example was first analyzed with machine learning methods in Ref. [12]. The exotic $H^\pm$ decays to another $W$ boson and the SM Higgs boson ($h$). Taking into account only the dominant decay of the Higgs boson, the signal process is defined as

$$gg \to H^0 \to W^\mp H^\pm \to W^\mp W^\pm h \to W^\mp W^\pm b\bar{b}. \qquad (14)$$

The background is therefore dominated by top-pair production, which also gives $W^\mp W^\pm b\bar{b}$.

For our analysis, we use the same data presented in [12] that is publicly available at [59], which focus on the semi-leptonic decay mode of both background and signal events (one $W$ boson decaying leptonically and the other one decaying hadronically), giving as final state $\ell \nu j j b\bar{b}$. The data set consists of low-level variables (twenty-one in total, considering the momentum of each visible particle, the $b$-tagged information of all jets, and the reconstruction of the missing energy) and seven high-level variables ($m_{jj}$, $m_{jjj}$, $m_{\ell\nu}$, $m_{j\ell\nu}$, $m_{b\bar{b}}$, $m_{Wb\bar{b}}$ and $m_{WWb\bar{b}}$), expected to have higher discrimination power between signal and background (see [12] for more details). The signal benchmark case corresponds to a $m_{H^0} = 425$ GeV and $m_{H^\pm} = 325$ GeV.

For this example, we have trained three `XGBoost` classifiers with three different data representations: only low-level variables, only high-level variables, and combining both low and high-level features. For completeness we also add the result obtained when using an average variable obtained after ensembling 10 ML classifiers with all the input variables. In the left panel of Fig. 6 we show the ROC curves for the analysis, and as expected, the best performance was achieved using both low and high-level features (for both the averaged and non-averaged variable). These results are in agreement with the analysis performed in [12], obtained with different ML algorithms. In the following, we will work with the latter data representation to estimate the expected significance for the search for heavy Higgs.

To compute the expected background yield at the ATLAS detector at $\sqrt{s} = 8$ TeV and luminosity of 20 fb$^{-1}$, $B \simeq 86$ k, we simulated background events with `MadGraph5_aMC@NLO 2.6` [60], using `PYTHIA 8` [61] for showering and hadronization, and `Delphes 3` [62] for fast detector simulation. We applied all the selection cuts in [12], and checked that the different kinematic distributions from our simulation are in agreement with the ones from the public data set. With the expected background prediction, we scan over the expected signal yield, $S$, to be agnostic regarding the coupling values of the model.

The exclusion significance for different $\frac{S}{\sqrt{B}}$ ratios are shown in the right panel of Fig. 6. The results for the MLL+KDE methods do not yield significant differences and are shown as the red solid curve for the averaged variable of 10 ML and as dotted orange curve correspond for 1 ML. We also present as dashed curves the significance binning the one-dimensional ML output for different numbers of bins. We would like to remark that binning the original feature space is not possible due to its high dimensionality (twenty-one and seven low and high-level variables, respectively). Also for this collider example, it can be seen that the

**Table 1** Expected cross-section upper limit at 95% C.L., considering ATLAS detector, $\sqrt{s}$ = 8 TeV and luminosity of 20 fb$^{-1}$ ($B$ = 86 k). Background process and cuts as discussed in the main text

| Method | $\sigma_{\text{fid}}$ (pb) (95% C.L. upper limit) |
|---|---|
| MLL+KDE | $8.94 \times 10^{-3}$ |
| MLL+KDE (10ML) | $8.84 \times 10^{-3}$ |
| o($\bar{x}$) BL (100 bins) | $9.91 \times 10^{-3}$ |
| o($\bar{x}$) BL (50 bins) | $9.97 \times 10^{-3}$ |
| o($\bar{x}$) BL (25 bins) | $10.03 \times 10^{-3}$ |
| o($\bar{x}$) BL (10 bins) | $11.15 \times 10^{-3}$ |

**Table 2** Discovery significances assuming $B = 1000$ and $S = 100$. For comparison, we also include the results for the same case shown in [12] using a shallow neural network (NN) and a deep neural network (DN)

| Method | Z |
|---|---|
| NN [12] | $3.7\sigma$ |
| DN [12] | $5.0\sigma$ |
| MLL+KDE | $6.61\sigma$ |
| MLL+KDE (10ML) | $6.65\sigma$ |
| o($\bar{x}$) BL (100 bins) | $6.53\sigma$ |
| o($\bar{x}$) BL (50 bins) | $6.52\sigma$ |
| o($\bar{x}$) BL (25 bins) | $6.43\sigma$ |
| o($\bar{x}$) BL (10 bins) | $6.14\sigma$ |

MLL+KDE method outperforms the results obtained with the binned likelihood procedure.

Since no excess has been found, we can compute the expected cross-section upper limit at 95%C.L. for the new exotic Higgs bosons search, which corresponds to the value of $\frac{S}{\sqrt{B}}$ that gives $Z = 1.64$. The results are presented in Table 1.

For completeness, and to compare with the results of [12], we show in Table 2 the discovery significance for MLL+KDE and BL methods. Notice that for this calculation we artificially set $B = 1000$ and $S = 100$ to directly compare our results with the ones in [12]. The significant improvement in this case is due to the use of the full ML output in both MLL+KDE and BL methods, while in Ref. [12] only a fraction of $o(x)$ is used to define a signal enriched region with a working point.

### 3.3 SSM $Z'$ boson decaying into lepton pairs at the HL-LHC

In this section, we analyzed the performance of our method on a simple collider example, namely the search for an SSM $Z'$ boson decaying into lepton pairs at the HL-LHC. We generated sample events for signal and background with `MadGraph5_aMC@NLO 2.6` [60], the showering and clustering were performed with `PYTHIA 8` [61], and finally, the detector simulation was done with `Delphes 3` [62]. For the SM background, we considered the Drell-Yan production of $Z/\gamma* \rightarrow \ell\ell$, with $\ell = e, \mu$, as in [51]. As in the previous examples, we trained a `XGBoost` classifier, with 1 M events per class, to distinguish $S$ from $B$, for each $Z'$ mass value, $m_{Z'}$ = [2.5, 3.5, 4.5, 5, 5.5, 6.5, 7.5, 8.5] TeV, and final state (dielectron and dimuon). We use as input parameters the transverse momentum $|p_T|$, the azimuthal angle $\phi$, and the pseudorapidity $\eta$ of the final state leptons in each channel, the kinematic variables that can be extracted directly from the `Delphes 3` output file. Considering the expected background prediction, for each parameter point we scan over $S$ to obtain the expected signal yield upper limit at 95% C.L., corresponding to the value that gives $Z = 1.64$. Finally, we convert this yield to a cross section-upper limit that can be compared with the theoretical prediction.

We are employing the same setup and detector level cuts as in the work presented by the ATLAS Collaboration at $\sqrt{s} = 14$ TeV and 3 ab$^{-1}$ [51], but we only generated signal and background events with dielectron and dimuon invariant masses above 1.8 TeV, and since we are dealing with a signal-enriched region and not the entire spectrum, the direct comparison with ATLAS projections for 95% CL exclusion limits is not strictly fair. This may enhance the performance of our classifier, since a $Z'$ signal would appear as an excess at high dilepton invariant masses. However, the power of our method can be shown in the left (right) panel of Fig. 7 for the dielectron (dimuon) channel when compared to the BL fit of the ML classifier output, which is on equal footing with the results for our method since it uses the same ML classifier. Unbinning signal and background posteriors provide more constraining exclusion limits for both final states, and as in the previous examples, there is no significant difference between MLL+KDE using the output of 1 ML or the averaged 10 ML.

## 4 Conclusions

The Machine-Learned Likelihoods method can be used to obtain discovery significances and exclusion limits for additive new physics scenarios. It uses a single classifier and its full one-dimensional output, which allows the estimation of the signal and background PDFs needed for statistical inference. In this paper, we extend the MLL method to obtain exclusion significances and improve its performance by using the KDE method to extract the corresponding PDFs from the ML output. We found that the small fluctuations of the machine learning output around the optimal value translate into non-smooth PDFs. We verify that this problem can be handled by averaging the output of several indepen-

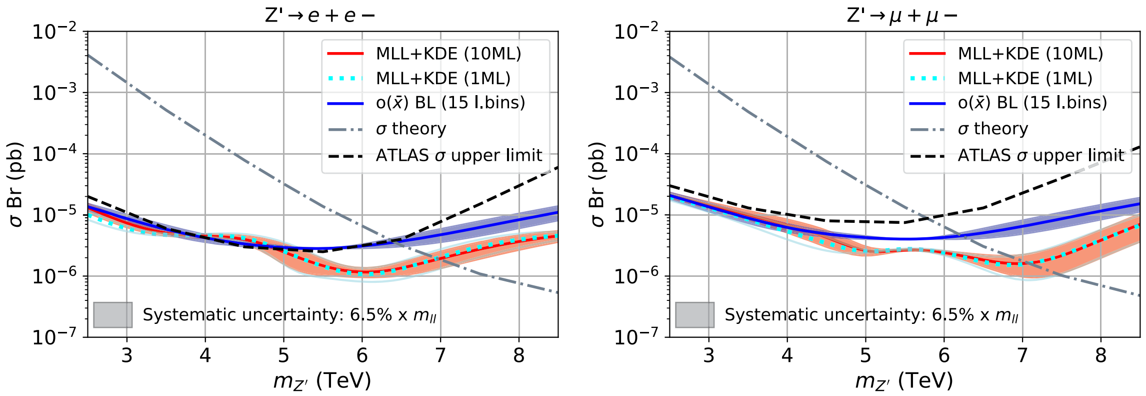


**Fig. 7** Exclusion limits for the $Z'_{SSM}$ with MLL+KDE method (red solid curve corresponds to the averaged variable of 10 ML, and dotted cyan curve to 1 ML), and with the BL fit of the ML output using 15 linear bins (blue curve). The shaded area in each case includes a naive estimation of the significance uncertainty caused by the mass variation on each point, according to the systematic uncertainty for the invariant mass estimated by ATLAS in [51]. Left panel: dielectron channel. Right panel: dimuon channel

dent machine-learning realizations. But mostly, we show that these small fluctuations do not have a major impact on the final significance.

Although the binning of the classifier output is always possible, irrespective of the dimensionality of the original variables, we verify that computing the PDFs with a non-parametric method such as KDE to avoid the binning, enhances the performance. By analyzing toy models generated with Gaussian distribution of different dimensions (with and without correlation between signal and background), we showed that MLL with KDE outperforms the BL method (with both linear and non-linear bins) when dealing with high-dimensional data, while for low-dimensional data all the methods converge to the results obtained with the true PDFs. Although it is a well-known fact that almost all the benefits of unbinned approaches can be obtained with optimal binning, avoiding such a (usually cumbersome) process is one of the main advantages of our work, providing an automatic way of estimating the probability density distributions through the KDE implementation.

Finally, we test the MLL framework in two physical examples. We found that, as expected, MLL also improves the exclusion-limits results obtained in a realistic $Z'$ analysis as well as in the search for exotic Higgs bosons at the LHC, surpassing the ones computed with the simple BL fit of the ML one-dimensional output.

Last but not least, we would like to remark that this new version of MLL with KDE does not include systematic uncertainties in the likelihood fit, which is necessary for any realistic search. As this is a highly non-trivial issue for unbinned methods, we leave the inclusion of nuisance parameters to the MLL framework for future work. Nevertheless, we also highlight that even though likelihoods without uncertainties can not be used in most experimental setups, it could be useful in specific scenarios where the nuisance parameters can be considered small, and in phenomenological analyses as proofs of concept.

**Acknowledgements** We thank Xabier Marcano, Manuel Szewc, Alejandro Szynkman, and Hernán Wahlberg for fruitful discussions on the implementation of KDE in the MLL method. This work is partially supported by the “Atracción de Talento” program (Modalidad 1) of the Comunidad de Madrid (Spain) under the Grant number 2019-T1/TIC-14019 (EA, RMSS), by the Spanish Research Agency (Agencia Estatal de Investigación) through the Grant IFT Centro de Excelencia Severo Ochoa No CEX2020-001007-S (EA, MdlR, AP, RMSS) and the Grant PID2021-124704NB-I00 (EA, RMSS), funded by MCIN/AEI/10.13039/ 501100011033, and by CONICET and ANPCyT under projects PICT 2017-2751 and PICT 2018-03682 (EA, AP). MdlR also acknowledges financial support from the Comunidad Autónoma de Madrid through the Grant SI2/PBG/2020-00005.

**Data Availability Statement** This manuscript has associated data in a data repository. [Authors’ comment: The MLL code is available in the following open access link: https://github.com/AndresDanielPerez/2211.04806-ML-Likelihood-with-KDE.]

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Funded by SCOAP$^3$. SCOAP$^3$ supports the goals of the International Year of Basic Sciences for Sustainable Development.

## A Automatic tune of binning

In the KDE method presented in this work, we are automatically tuning the bandwidths of the background and signal distributions. On the other hand, for the histograms, assuming equal-sized bins, there is only one parameter to tune: the bin width (or equivalently, the number of bins). It is important to remark that although we have two samples (background and signal) the formula for significance in Eq. (8) requires the same binning for both distributions. In that sense, we have chosen to tune the histogram associated with the background sample, since it limits the validity of the binned likelihood method in statistical terms. There are different methods that can be used to tune such parameter (Stone, Freedman Diaconis or FD, Auto, Doane, Scott, Sturges, etc.). Nevertheless none of these methods is general for every possible ML output as each one is optimized taking into account the functional form of the data (size, shape, etc), which in our case is the output of our classifier that depends on the specific physical scenario. For example, the Sturges method is robust for Gaussian data, while FD method is good for large data samples.

In Table 3 we show the optimal number of bins found by three different methods[2] for $\mathcal{N}_{dim}(\boldsymbol{m}, \boldsymbol{\Sigma})$, with $\boldsymbol{m} = +0.3(-0.3)$ for $S$ ($B$), no correlation, and fixed $\langle B \rangle = 50$ k and $\langle S \rangle = 500$, corresponding to the Gaussian example introduced in Sect. 3.1. As stated before, since there is no general method to choose $N$, we decided to compare three methods whose assumptions fit some of our data set properties: FD due to our large sample, Sturges because at low dimensions the ML output resembles a normal distribution, and Doane to account the skew of the data for high dimensions. Additionally, in Fig. 8 we show the significances obtained with these 3 methods for Gaussian distributions of different dimensions. It is important to highlight that all these methods assume equal size bin widths, and hence they show the same tendency already presented in the left panel of Fig. 4, the significance increases with the number of bins.

If we do not assume equal-size bins, we must tune $(N-1)$ parameters (i.e. the width of each bin) where $N$ is the number of bins. Unlike in the KDE method, now we are dealing with a high dimensional space. To perform a full exploration of this space is computationally expensive, therefore we performed a data-driven procedure to cross-validate the selection of the number of bins and each bin size as follows:

1. We fix the number of bins ($N$) with one of the previously described methods (FD, Doane and Sturges) applied to the background data set.
2. We randomly select the width of each bin.

**Table 3** Estimation of the optimal number of bins to describe the background data of the example shown in Fig. 8, using FD, Doane and Sturges methods

| Dim | Number of bins (N) | | |
|---|---|---|---|
| | FD | Doane | Sturges |
| 1 | 74 | 21 | 18 |
| 2 | 62 | 23 | 18 |
| 3 | 57 | 23 | 18 |
| 4 | 54 | 24 | 18 |
| 5 | 52 | 24 | 18 |
| 6 | 52 | 24 | 18 |
| 7 | 52 | 24 | 18 |
| 8 | 54 | 25 | 18 |
| 9 | 55 | 25 | 18 |
| 10 | 57 | 25 | 18 |

3. We divide the background data set into 5 k-folds (as done in the KDE optimization). For each bin $d$, we compute the mean number of events per bin, $\mu_d = \frac{1}{4}\sum_{k=1}^{4} N_d^{(k)}$, where $N_d^{(k)}$ is the number of events in each bin for the sample $k$. Notice that we used 4 k-folds.
4. For the last k-fold, $k = 5$, assuming a Poissonian distribution for each bin and the Stirling approximation for $\log(N_d!)$, we calculate

$$q_{\text{poiss}} = -\log\mathcal{L} = \sum_{d}^{N_{bin}} \left( \mu_d - N_d^{(k=5)} \log(\mu_d) \right. \\ \left. + N_d^{(k=5)} \log\left(N_d^{(k=5)}\right) - N_d^{(k=5)} \right).$$

5. We repeat steps 2. to 4. 5000 times.
6. Finally, we select the binning that provides the minimum value of $q_{\text{poiss}}$ from among the 5000 iterations and compute the significance using the signal and an independent background sample.

This procedure provides a good trade-off between optimization and computational cost. In Fig. 8 we show the obtained exclusion significance ($Z$). For each dimension, the significance found optimizing the bin widths is similar to the result assuming equal-sized bins (for the same number of bins). It is worth to remark that the MLL-KDE method consistently outperforms the binned multivariate analysis in terms of significance. For completeness, we have also check that similar conclusions can be drawn for alternative likelihood assumptions (for example, assuming a Gaussian distribution for each bin).

[2] We use the python implementation available in the library numpy.histogram_bin_edges.

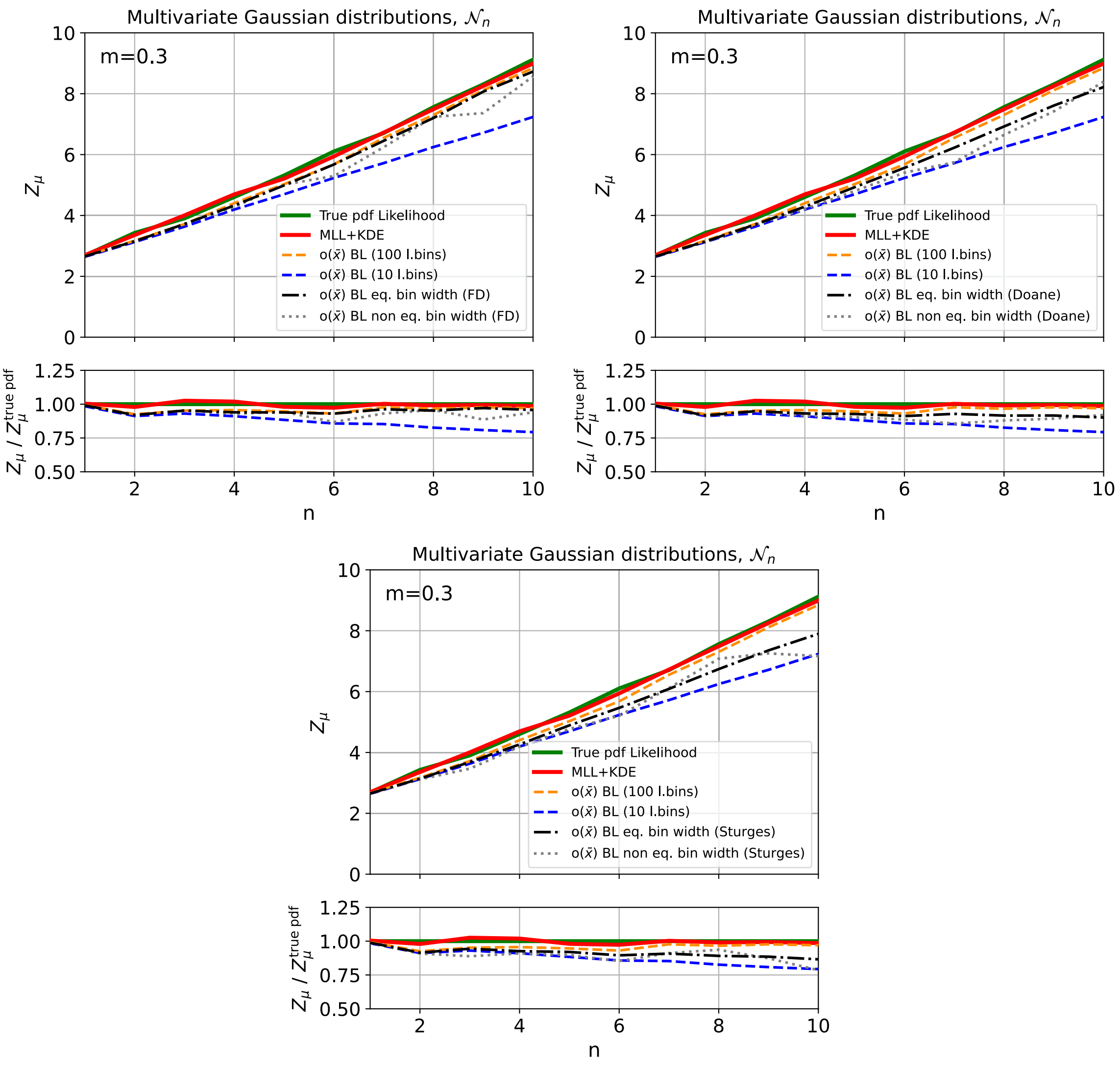


**Fig. 8** Exclusion-limit significance for $\mathcal{N}_{dim}(\boldsymbol{m}, \boldsymbol{\Sigma})$ with $\boldsymbol{m} = +0.3(-0.3)$ for $S$ ($B$) and no correlation, as a function of the $dim$, for fixed $\langle B\rangle = 50$ k and $\langle S\rangle = 500$. The red solid curve shows the result of implementing the MLL+KDE method, while the green curve shows the results obtained using the true PDFs. The orange dashed curve represents the results obtained by applying the BL method to the classifier's one-dimensional output for 100 equal-width bins, and the blue dashed curve for 10 equal-width bins. The black dot-dashed curve represents the BL method to the classifier's one-dimensional output for N equal bin width with N determined for each dimension by FD (top left panel), Doane (top right panel), and Sturges (bottom panel), see Table 3. The gray dotted curve shows the significance using BL method with N bins, assuming non-equal bin widths

## References

1. A.J. Larkoski, I. Moult, B. Nachman, Jet substructure at the large hadron collider: a review of recent advances in theory and machine learning. Phys. Rep. **841**, 1 (2020). https://doi.org/10.1016/j.physrep.2019.11.001. arXiv:1709.04464
2. D. Guest, K. Cranmer, D. Whiteson, Deep learning and its application to LHC physics. Ann. Rev. Nucl. Part. Sci. **68**, 161 (2018). https://doi.org/10.1146/annurev-nucl-101917-021019. arXiv:1806.11484
3. K. Albertsson et al., Machine learning in high energy physics community white paper. J. Phys. Conf. Ser. **1085**, 022008 (2018). https://doi.org/10.1088/1742-6596/1085/2/022008. arXiv:1807.02876
4. A. Radovic, M. Williams, D. Rousseau, M. Kagan, D. Bonacorsi, A. Himmel et al., Machine learning at the energy and intensity frontiers of particle physics. Nature **560**, 41 (2018). https://doi.org/10.1038/s41586-018-0361-2
5. G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby et al., Machine learning and the physical sciences. Rev. Mod. Phys. **91**, 045002 (2019). https://doi.org/10.1103/RevModPhys.91.045002. arXiv:1903.10563
6. D. Bourilkov, Machine and deep learning applications in particle physics. Int. J. Mod. Phys. A **34**, 1930019 (2020). https://doi.org/10.1142/S0217751X19300199. arXiv:1912.08245
7. G. Karagiorgi, G. Kasieczka, S. Kravitz, B. Nachman, D. Shih, Machine learning in the search for new fundamental physics. arXiv:2112.03769

8. M. Feickert, B. Nachman, A living review of machine learning for particle physics. arXiv:2102.02770
9. T. Plehn, A. Butter, B. Dillon, C. Krause, Modern machine learning for LHC physicists. arXiv:2211.01421
10. B.H. Denby, Neural networks and cellular automata in experimental high-energy physics. Comput. Phys. Commun. **49**, 429 (1988). https://doi.org/10.1016/0010-4655(88)90004-5
11. L. Lonnblad, C. Peterson, T. Rognvaldsson, Finding gluon jets with a neural trigger. Phys. Rev. Lett. **65**, 1321 (1990). https://doi.org/10.1103/PhysRevLett.65.1321
12. P. Baldi, P. Sadowski, D. Whiteson, Searching for exotic particles in high-energy physics with deep learning. Nat. Commun. **5**, 4308 (2014). https://doi.org/10.1038/ncomms5308. arXiv:1402.4735
13. E.M. Metodiev, B. Nachman, J. Thaler, Classification without labels: Learning from mixed samples in high energy physics. JHEP **10**, 174 (2017). https://doi.org/10.1007/JHEP10(2017)174. arXiv:1708.02949
14. C.K. Khosa, V. Sanz, M. Soughton, Using machine learning to disentangle LHC signatures of dark matter candidates. SciPost Phys. **10**, 151 (2021). https://doi.org/10.21468/SciPostPhys.10.6.151. arXiv:1910.06058
15. A. Mullin, S. Nicholls, H. Pacey, M. Parker, M. White, S. Williams, Does SUSY have friends? A new approach for LHC event analysis. JHEP **02**, 160 (2021). https://doi.org/10.1007/JHEP02(2021)160. arXiv:1912.10625
16. S. Chang, T.-K. Chen, C.-W. Chiang, Distinguishing $W'$ signals at hadron colliders using neural networks. Phys. Rev. D **103**, 036016 (2021). https://doi.org/10.1103/PhysRevD.103.036016. arXiv:2007.14586
17. F. Flesher, K. Fraser, C. Hutchison, B. Ostdiek, M.D. Schwartz, Parameter inference from event ensembles and the top-quark mass. JHEP **09**, 058 (2021). https://doi.org/10.1007/JHEP09(2021)058. arXiv:2011.04666
18. Y.S. Lai, D. Neill, M. Płoskoń, F. Ringer, Explainable machine learning of the underlying physics of high-energy particle collisions. arXiv:2012.06582
19. B. Nachman, J. Thaler, Learning from many collider events at once. Phys. Rev. D **103**, 116013 (2021). https://doi.org/10.1103/PhysRevD.103.116013. arXiv:2101.07263
20. E. Arganda, A.D. Medina, A.D. Perez, A. Szynkman, Towards a method to anticipate dark matter signals with deep learning at the LHC. SciPost Phys. **12**, 063 (2022). https://doi.org/10.21468/SciPostPhys.12.2.063. arXiv:2105.12018
21. K. Cranmer, J. Pavez, G. Louppe, Approximating likelihood ratios with calibrated discriminative classifiers. arXiv:1506.02169
22. A. Elwood, D. Krücker, Direct optimisation of the discovery significance when training neural networks to search for new physics in particle colliders. arXiv:1806.00322
23. R.T. D'Agnolo, A. Wulzer, Learning new physics from a machine. Phys. Rev. D **99**, 015014 (2019). https://doi.org/10.1103/PhysRevD.99.015014. arXiv:1806.02350
24. B. Nachman, A guide for deploying deep learning in LHC searches: how to achieve optimality and account for uncertainty. SciPost Phys. **8**, 090 (2020). https://doi.org/10.21468/SciPostPhys.8.6.090. arXiv:1909.03081
25. R.T. D'Agnolo, G. Grosso, M. Pierini, A. Wulzer, M. Zanetti, Learning multivariate new physics. Eur. Phys. J. C **81**, 89 (2021). https://doi.org/10.1140/epjc/s10052-021-08853-y. arXiv:1912.12155
26. J. Hollingsworth, D. Whiteson, Resonance searches with machine learned likelihood ratios. arXiv:2002.04699
27. K.T. Matchev, P. Shyamsundar, J. Smolinsky, A quantum algorithm for model independent searches for new physics. arXiv:2003.02181
28. A.S. Cornell, W. Doorsamy, B. Fuks, G. Harmsen, L. Mason, Boosted decision trees in the era of new physics: a smuon analysis case study. JHEP **04**, 015 (2022) https://doi.org/10.1007/JHEP04(2022)015
29. J.A. Aguilar-Saavedra, Anomaly detection from mass unspecific jet tagging. Eur. Phys. J. C **82**, 130 (2022). https://doi.org/10.1140/epjc/s10052-022-10058-w. arXiv:2111.02647
30. R.T. d'Agnolo, G. Grosso, M. Pierini, A. Wulzer, M. Zanetti, Learning new physics from an imperfect machine. arXiv:2111.13633
31. V. Mikuni, B. Nachman, D. Shih, Online-compatible unsupervised nonresonant anomaly detection. Phys. Rev. D **105**, 055006 (2022). https://doi.org/10.1103/PhysRevD.105.055006. arXiv:2111.06417
32. C.K. Khosa, V. Sanz, M. Soughton, A simple guide from machine learning outputs to statistical criteria. arXiv:2203.03669
33. M. Letizia, G. Losapio, M. Rando, G. Grosso, A. Wulzer, M. Pierini et al., Learning new physics efficiently with nonparametric methods. arXiv:2204.02317
34. T. Finke, M. Krämer, M. Lipp, A. Mück, Boosting mono-jet searches with model-agnostic machine learning. JHEP **08**, 015 (2022). https://doi.org/10.1007/JHEP08(2022)015
35. F.F. Freitas, J.a. Gonçalves, A.P. Morais, R. Pasechnik, Phenomenology at the large hadron collider with deep learning: the case of vector-like quarks decaying to light jets. arXiv:2204.12542
36. E. Arganda, X. Marcano, V.M. Lozano, A.D. Medina, A.D. Perez, M. Szewc et al., A method for approximating optimal statistical significances with machine-learned likelihoods. Eur. Phys. J. C **82**, 993 (2022). https://doi.org/10.1140/epjc/s10052-022-10944-3. arXiv:2205.05952
37. ATLAS collaboration, Measurement of the $t$-channel single top-quark production cross section in $pp$ collisions at $\sqrt{s} = 7$ TeV with the ATLAS detector. Phys. Lett. B **717**, 330 (2012). https://doi.org/10.1016/j.physletb.2012.09.031. arXiv:1205.3130
38. ATLAS collaboration, Search for the electroweak production of supersymmetric particles in $\sqrt{s}$=8 TeV $pp$ collisions with the ATLAS detector. Phys. Rev. D **93**, 052002 (2016). https://doi.org/10.1103/PhysRevD.93.052002. arXiv:1509.07152
39. CMS collaboration, Search for new phenomena in final states with two opposite-charge, same-flavor leptons, jets, and missing transverse momentum in pp collisions at $\sqrt{s} = 13$ TeV. JHEP **03**, 076 (2018). https://doi.org/10.1007/s13130-018-7845-2. arXiv:1709.08908
40. CMS collaboration, Evidence for associated production of a Higgs boson with a top quark pair in final states with electrons, muons, and hadronically decaying $\tau$ leptons at $\sqrt{s} = 13$ TeV. JHEP **08**, 066 (2018). https://doi.org/10.1007/JHEP08(2018)066. arXiv:1803.05485
41. CMS collaboration, Measurement of the production cross section for single top quarks in association with W bosons in proton-proton collisions at $\sqrt{s} = 13$ TeV. JHEP **10**, 117 (2018). https://doi.org/10.1007/JHEP10(2018)117. arXiv:1805.07399
42. CMS collaboration, Search for supersymmetry in final states with two oppositely charged same-flavor leptons and missing transverse momentum in proton-proton collisions at $\sqrt{s} = 13$ TeV. JHEP **04**, 123 (2021). https://doi.org/10.1007/JHEP04(2021)123. arXiv:2012.08600
43. ATLAS collaboration, Measurement of the $t\bar{t}t\bar{t}$ production cross section in $pp$ collisions at $\sqrt{s}$ = 13 TeV with the ATLAS detector. JHEP **11**, 118 (2021). https://doi.org/10.1007/JHEP11(2021)118. arXiv:2106.11683
44. ATLAS collaboration, Measurements of Higgs boson production by gluon−gluon fusion and vector-boson fusion using $H \to WW^* \to e\nu\mu\nu$ decays in $pp$ collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector. arXiv:2207.00338
45. A method for approximating optimal statistical significances with machine-learned likelihoods. (2021). https://github.com/AndresDanielPerez/2205.05952-ML-Likelihood-method

46. E. Arganda, M. de los Rios, A.D. Perez, R.M. Sandá Seoane, Imposing exclusion limits on new physics with machine-learned likelihoods. PoS **ICHEP2022**, 1226 (2022). https://doi.org/10.22323/1.414.1226
47. M. Rosenblatt, Remarks on some nonparametric estimates of a density function. Ann. Math. Stat. **27**, 832 (1956). https://doi.org/10.1214/aoms/1177728190
48. E. Parzen, On estimation of a probability density function and mode. Ann. Math. Stat. **33**, 1065 (1962). https://doi.org/10.1214/aoms/1177704472
49. G. Papamakarios, Neural density estimation and likelihood-free inference. Ph.D. thesis, Edinburgh U., p. 4 (2019). arXiv:1910.13233
50. G. Altarelli, B. Mele, M. Ruiz-Altaba, Searching for new heavy vector bosons in $p\bar{p}$ colliders. Z. Phys. C **45**, 109 (1989). https://doi.org/10.1007/BF01556677
51. ATLAS collaboration, Prospects for searches for heavy $Z'$ and $W'$ bosons in fermionic final states with the ATLAS experiment at the HL-LHC, ATL-PHYS-PUB-2018-044 (2018)
52. K. Cranmer et al., Publishing statistical models: getting the most out of particle physics experiments. SciPost Phys. **12**, 037 (2022). https://doi.org/10.21468/SciPostPhys.12.1.037. arXiv:2109.04981
53. G. Cowan, K. Cranmer, E. Gross, O. Vitells, Asymptotic formulae for likelihood-based tests of new physics. Eur. Phys. J. C **71**, 1554 (2011). https://doi.org/10.1140/epjc/s10052-011-1554-0. arXiv:1007.1727
54. J. Neyman, E.S. Pearson, On the problem of the most efficient tests of statistical hypotheses. Philos. Trans. Roy. Soc. Lond. A **231**, 289 (1933)
55. G.J. McLachlan, K.E. Basford, Mixture models : inference and applications to clustering. J. Roy. Stat. Soc. SerC: Appl. Stat. **38**(2), 384 (2018). https://doi.org/10.2307/2348072
56. R. Gomez Ambrosio, J. ter Hoeve, M. Madigan, J. Rojo, V. Sanz, Unbinned multivariate observables for global SMEFT analyses from machine learning. arXiv:2211.02058
57. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel et al., Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825 (2011)
58. V.A. Epanechnikov, Non-parametric estimation of a multivariate probability density. Theory Probab. Appl. **14**, 153–158 (1969). https://doi.org/10.1137/1114019
59. Higgs data set. http://archive.ics.uci.edu/ml/datasets/HIGGS
60. J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer 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**, 079 (2014). https://doi.org/10.1007/JHEP07(2014)079. arXiv:1405.0301
61. T. Sjöstrand, S. Ask, J.R. Christiansen, R. Corke, N. Desai, P. Ilten et al., An introduction to PYTHIA 8.2. Comput. Phys. Commun. **191**, 159 (2015). https://doi.org/10.1016/j.cpc.2015.01.024. arXiv:1410.3012
62. DELPHES 3 collaboration, DELPHES 3, A modular framework for fast simulation of a generic collider experiment. JHEP **02**, 057 (2014). https://doi.org/10.1007/JHEP02(2014)057. arXiv:1307.6346