




Learning new physics efficiently with nonparametric methods

Marco Letizia^{1,2,a} , Gianvito Losapio¹, Marco Rando¹, Gaia Grosso^{3,4,5}, Andrea Wulzer³, Maurizio Pierini⁵, Marco Zanetti^{3,4}, Lorenzo Rosasco^{1,6,7}

¹ MaLGA-DIBRIS, Università di Genova, Genoa, Italy

² INFN, Sezione di Genova, Genoa, Italy

³ Dipartimento di Fisica e Astronomia, Università di Padova, Padua, Italy

⁴ INFN, Sezione di Padova, Padua, Italy

⁵ Experimental Physics Department, CERN, Geneva, Switzerland

⁶ CBMM, Massachusetts Institute of Technology, Cambridge, MA, USA

⁷ Istituto Italiano di Tecnologia, Genoa, Italy

Received: 13 April 2022 / Accepted: 21 September 2022 / Published online: 5 October 2022
© The Author(s) 2022

Abstract We present a machine learning approach for model-independent new physics searches. The corresponding algorithm is powered by recent large-scale implementations of kernel methods, nonparametric learning algorithms that can approximate any continuous function given enough data. Based on the original proposal by D’Agnolo and Wulzer (Phys Rev D 99(1):015014, 2019, [arXiv:1806.02350](https://arxiv.org/abs/1806.02350) [hep-ph]), the model evaluates the compatibility between experimental data and a reference model, by implementing a hypothesis testing procedure based on the likelihood ratio. Model-independence is enforced by avoiding any prior assumption about the presence or shape of new physics components in the measurements. We show that our approach has dramatic advantages compared to neural network implementations in terms of training times and computational resources, while maintaining comparable performances. In particular, we conduct our tests on higher dimensional datasets, a step forward with respect to previous studies.

1 Introduction

Experimental observations and convincing conceptual arguments indicate that the present understanding of fundamental physics is not complete. Our theoretical formulation of the fundamental laws of Nature, the Standard Model, has been predicting with extremely high precision an impressive amount of data collected at past and ongoing experiments. On the other hand, the Standard Model does not provide

answer to a multitude of questions including the origin of the electroweak scale, the mass of neutrinos, the flavour structure in the quark, lepton and neutrino sectors, and is unable to account for observed phenomena like the origin and the composition of the dark matter of the baryon asymmetry in the Universe. Further, it does not provide a microscopic description of gravity. These considerations guarantee the existence of more fundamental laws of Nature waiting to be unveiled. In order to access these laws, we must search the experimental data for phenomena that depart from the Standard Model predictions.

Currently, the most common searching strategy is to test the data for the presence of specific new physics models, one at the time. Each search is then optimized to be sensitive to the features specific of the considered new physics scenario. This approach is in general insensitive to sources of discrepancy that differ from those considered. There is therefore a strong effort in developing analysis strategies that are agnostic about the nature of potential new physics and thus complementary to the model-dependent approaches described above [2–13]. Ideally, this type of analysis should be sensitive to generic departures from a given reference model. In practice, this is a challenge given the complexity of the experimental data in modern experiments and the fact that the new physics signal is expected to be “small” and/or located in a region of the input features which is already populated by events predicted by the reference model. Recently, there has been a strong push towards developing solutions based on machine learning for (partial or full) model-independent searches in high energy physics [1, 14–39].

In this work we present a novel machine learning implementation of the analysis strategy proposed by D’Agnolo

^a e-mail: marco.letizia@edu.unige.it (corresponding author)

et al. [1, 16]. The aim of this strategy is to compute the log-likelihood-ratio test statistics without specifying the alternative new physics hypothesis a priori. Towards this end, a neural network model was used in [1, 16] to learn the alternative hypothesis directly from the data while the log-likelihood-ratio was maximized to get an optimal test statistics. The strategy assumes that a sample of events representing the Standard Model hypothesis (“reference” sample) is available and that its size is much larger than the one of the experimental data, so that the only relevant statistical uncertainties are those of the data themselves. In the new implementation presented here, neural networks are replaced by kernel methods, specifically a modern and efficient implementation of kernel logistic regression [40]. Kernel methods are non-parametric algorithms that can approximate any continuous function given enough data. Recent large-scale implementations [40] provides fast and efficient solvers even with very large data-sets. This is relevant since a key bottleneck of the neural network model used in Refs. [1, 16] is the extremely long training time, even on low dimensional problems, due to regularization considerations that limit the types of viable architectures and training strategies. The solution we propose solves this issue by delivering comparable performances with orders of magnitude gain in training times, see Table 1 for the average training times needed to process a single toy experiment. We demonstrate the viability of the framework by testing on particle physics datasets of increasing dimensionality, a further step forward with respect to previous studies.

We note that the ideas recently proposed in Ref. [37] share some similarities to our approach. Indeed, the authors of Ref. [37] developed a model-independent strategy based on classifiers to perform hypothesis testing on Standard Model samples and experimental measurements. However, they implement a train-test split of the data for the reconstruction of the test statistics and for inference. This is a major difference with respect to our approach, where the distribution employed for the evaluation of the test statistics is the one that best fits the very same set of data on which the test has to be performed, in accordance with the maximum likelihood philosophy. Moreover, while their approach permits to estimate the distribution of the test statistics with a single training of a classifier, only half of the experimental data is used for new physics detection. A in-depth comparison with this and other unsupervised and semi-supervised approaches will be explored in future works.

The rest of the paper is organized as follows. In Sect. 2 we introduce the main statistical framework at the basis of this work, elaborating on the discussion in Ref. [1]. In Sect. 3, we discuss the different aspects of the proposed model, in particular the underlying machine learning algorithm. In Sect. 4, we test the algorithm on realistic simulated datasets in various dimensions and we explicitly compare our proposal with the neural network models in Refs. [1, 16]. Finally in Sect. 5,

we lay out our conclusions and discuss future developments. In the appendices, we review some background material and present other complementary experiments.

2 Statistical foundations

In this section, we reprise and elaborate the main ideas in Refs. [1, 16], tackling the problem of testing the data for the presence of new physics with tools from statistics and machine learning.

We start by assuming that an experiment is performed and its outcome can be described by a multivariate random variable x . A physical model corresponds to an ensemble of mathematical laws characterizing a distribution for x . In this view, we denote by $p(x|0)$ the distribution of the measurements as described by the Standard Model and by $p(x|1)$ the unknown true distribution of the data. Discovering new physics will be cast as the problem of *testing* whether the latter coincides with the former or not.

The distribution $p(x|0)$ is essentially known. Although not analytically computable in most high energy physics applications, it can be sampled via Monte Carlo simulations or extracted using control regions with data driven techniques. In the following, we denote one such set of independent and identically distributed random variables (*i.i.d.*) by

$$S_0 = \{x_i\}_{i=1}^{\mathcal{N}_0}, \text{ with } x_i \stackrel{i.i.d.}{\sim} p(x|0), \quad (1)$$

and the actual measured data by,

$$S_1 = \{x_i\}_{i=1}^{\mathcal{N}_1}, \text{ with } x_i \stackrel{i.i.d.}{\sim} p(x|1). \quad (2)$$

It should be pointed out that in real applications one would also consider the uncertainties affecting the knowledge of the reference model. Similarly to Refs. [1, 16], we will assume $\mathcal{N}_0 \gg \mathcal{N}_1$ so that the statistical uncertainties on the reference sample can be neglected. It should be possible to include systematic uncertainties as nuisance parameters, as shown in Ref. [41] for the neural network implementation. However, we assume that the systematic uncertainties are negligible in what follows and leave this aspect to future works.

The idea in Ref. [1] is to translate the maximization of the log-likelihood-ratio test into a machine learning problem, where the null hypothesis characterising one of the likelihood terms is the reference hypothesis (namely the Standard Model) and the alternative hypothesis characterising the other likelihood term is unspecified a priori and learnt from the data themselves during the training. The test statistic obtained in this way is therefore a good approximation of the optimal test statistic according to the Neyman–Pearson lemma.

We define the likelihood of the data S_1 under a generic hypothesis H as

$$\begin{aligned} \mathcal{L}(S_1, H) &= \frac{e^{-N(H)} N(H)^{\mathcal{N}_1}}{\mathcal{N}_1!} \prod_{x=1}^{\mathcal{N}_1} p(x|H) \\ &= \frac{e^{-N(H)}}{\mathcal{N}_1!} \prod_x n(x|H), \end{aligned} \tag{3}$$

where

$$n(x|H) = N(H)p(x|H) \tag{4}$$

is the data distribution normalized to the expected number of events

$$N(H) = \int n(x|H) dx. \tag{5}$$

As already said, $p(x|0)$ is essentially known and well represented by the reference sample while $p(x|1)$ is not and thus its exact form must be replaced by a family of distributions $p_w(x|1)$, parametrized by a set of trainable variables w . We can write the likelihood ratio test statistics as,

$$\begin{aligned} t_w(S_1) &= -2 \log \frac{\mathcal{L}_w(S_1, 0)}{\mathcal{L}(S_1, 1)} \\ &= -2 \log \left[e^{N_w(1)-N(0)} \prod_{x=1}^{\mathcal{N}_1} \frac{n(x|0)}{n_w(x|1)} \right] \\ &= -2 \left[N_w(1) - N(0) - \sum_{x=1}^{\mathcal{N}_1} \log \frac{n_w(x|1)}{n(x|0)} \right]. \end{aligned} \tag{6}$$

and optimize it by maximizing over the set of parameters w . The original proposal in Ref. [1] suggested to exploit the ability of neural networks as universal approximators to define a family of functions describing the log-ratio of the density distributions in Eq. (6)

$$f_w(x) = \log \frac{n_w(x|1)}{n(x|0)}. \tag{7}$$

As discussed below the same approach can be taken replacing neural networks with other machine learning approaches, e.g. kernel methods. Following the above reasoning, the maximum of the test statistic could then be rewritten as the minimum of a loss function $L(S_1, f_w)$

$$\begin{aligned} t_{\hat{w}}(S_1) &= \max_w t_w(S_1) = -2 \min_w L(S_1, f_w) \\ &= -2 \min_w \left[\sum_{S_0} \frac{N(0)}{\mathcal{N}_0} (e^{f_w(x)} - 1) - \sum_{S_1} f_w(x) \right] \end{aligned} \tag{8}$$

and the set of parameters \hat{w} which maximizes $t_w(S_1)$

$$n_{\hat{w}}(x|1) = n(x|0)e^{f_{\hat{w}}(x)} \approx n(x|1) \tag{9}$$

provides also the best approximation of the true underlying data distribution and with it a first insight on the source and shape of the discrepancy, if present. To obtain Eq. (8) from Eq. (6), one needs to estimate the number of expected events in the alternative hypothesis. This can be done using Eq. (7) and by the Monte Carlo method, namely

$$\begin{aligned} N_w(1) &= \int n_w(x|1) dx = \int n(x|0)e^{f_w(x)} dx \\ &\approx \sum_{S_0} \frac{N(0)}{\mathcal{N}_0} e^{f_w(x)}. \end{aligned} \tag{10}$$

Note that the loss defined by Eq. (8) is unbounded from below. In Ref. [1] a regularization parameter is introduced as a hard upper bound (weight clipping) on the magnitude of the parameters w .

2.1 Designing a classifier for hypothesis testing

In this work we develop the above ideas considering a different loss function, namely a weighted cross-entropy (logistic) loss function. This was a possibility mentioned as a viable alternative in Ref. [1] that we indeed show to yields several advantages. To estimate the ratio in Eq. (9) we train a binary classifier on $S = S_0 \cup S_1$ using a weighted cross-entropy loss

$$\begin{aligned} \ell(y, f(x)) &= a_0(1 - y) \log \left(1 + e^{f(x)} \right) \\ &\quad + a_1 y \log \left(1 + e^{-f(x)} \right). \end{aligned} \tag{11}$$

where y is the class label and takes value zero for S_0 and one for S_1 . $a_y \in \mathbb{R}$ are arbitrary weights assigned to the examples from the two classes that will be specified later. The classifier is obtained minimizing an empirical criterion

$$\hat{L}(f_w) = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \ell(y, f_w(x)), \tag{12}$$

over a suitable class of machine learning models f_w . If such a models class is sufficiently rich, in the large sample limit we would recover a minimizer of the expected risk

$$L(f) = \int \ell(y, f(x)) dp(x, y), \tag{13}$$

where $p(x, y)$ is the joint data distribution. By a standard computation (see Appendix A), the function minimizing the expected risk in Eq. (13) can be shown to be

$$f^*(x) = \log \left(\frac{p(1|x) a_1}{p(0|x) a_0} \right), \tag{14}$$

that, by Bayes theorem and Eq. (4), we can rewrite as

$$\begin{aligned} f^*(x) &= \log \left(\frac{p(x|1) p(1) a_1}{p(x|0) p(0) a_0} \right) \\ &= \log \left(\frac{n(x|1) N(0) p(1) a_1}{n(x|0) N(1) p(0) a_0} \right). \end{aligned} \tag{15}$$

From the above expression and choosing the weights so that

$$\frac{a_1}{a_0} = \frac{N(1) p(0)}{N(0) p(1)}, \quad (16)$$

Eq. (14) reduces to Eq. (9), as desired. In practice, the above condition can be satisfied only approximately, since it depends on quantities we do not know. Hence, we first estimate the class priors using the empirical class frequencies, $p(y) \approx \mathcal{N}_y/\mathcal{N}$ with $\mathcal{N} = \mathcal{N}_0 + \mathcal{N}_1$ and obtain

$$\frac{a_1}{a_0} \approx \frac{\hat{a}_1}{\hat{a}_0} = \frac{N(1) \mathcal{N}_0}{N(0) \mathcal{N}_1}. \quad (17)$$

Then, we approximate the number of expected events in the alternative hypothesis with the actual number of experimental measurements $N(1) \approx \mathcal{N}_1$.¹ The following expression of the weights can then be used in practice,

$$\frac{\hat{a}_1}{\hat{a}_0} = \frac{\mathcal{N}_0}{N(0)}. \quad (18)$$

To reconstruct the test statistics in Eq. (6), the number of expected events in the alternative hypothesis needs to be computed. Using the density ratio in Eq. (9), we have that

$$N_w(1) = \int n_w(x|1) dx = \int n(x|0) e^{f_w(x)} dx, \quad (19)$$

with $\frac{n_w(x|1)}{n(x|0)} = e^{f_w(x)}.$

Since the reference distribution $n(x|0)$ is not known analytically, we can estimate the above expression using a Monte Carlo approximation considering

$$N_w(1) \approx \frac{N(0)}{\mathcal{N}_0} \sum_{x \in \mathcal{S}_0} e^{f_w(x)}. \quad (20)$$

Using Eq. (20), the test statistics in Eq. (6) can be written as

$$t_{\hat{w}}(S_1) = -2 \left[\frac{N(0)}{\mathcal{N}_0} \sum_{x \in \mathcal{S}_0} \left(e^{f_{\hat{w}}(x)} - 1 \right) - \sum_{x \in S_1} f_{\hat{w}}(x) \right] \quad (21)$$

recovering the original result from Ref. [1].

The main conceptual difference with respect to the original solution in Ref. [1] lays on the computation of the test statistic. When using the loss in Eq. (8) the test statistic can be directly obtained from the value of the loss function at the end of the training. When using the cross-entropy loss, each term of the log-likelihood-ratio test is calculated separately and then combined, see Eq. (21). This could be a problem if the optimality of the minimization procedure is not ensured. More precisely, in the first case the minimum found at the end of the training is by construction the one maximizing the log-likelihood-ratio test, while this is guaranteed only in the asymptotic limit in the second case. On the other hand, as

noted before, the loss function in Eq. (8) is less well behaved from a mathematical standpoint, making optimization during training less trivial. Interestingly, both loss functions are designed to estimate the same density ratio, and in practice we show that they obtain comparable performances in terms of sensitivity to new physics.

We conclude noting that the value of the test statistic $t_{\hat{w}}(S_1)$ is a random variable itself following a distribution $p(t|H)$. The level of significance associated to a value of the test statistic is computed as a p -value of the test statistic with respect to its distribution under the null hypothesis

$$p_{S_1} = \int_{t(S_1)}^{\infty} p(t|0) dt. \quad (22)$$

This can be further rewritten as a Z-score

$$Z_{obs}(S_1) = \Phi^{-1}(1 - p_{S_1}), \quad (23)$$

where Φ^{-1} is the quantile of a Normal distribution. In this way Z_{obs} is expressed in units of standard deviations. Following Ref. [1], by leveraging the possibility to sample from the reference distribution, we choose to reconstruct $p(t|0)$ by estimating the likelihood ratio test statistics on a number N_{toy} of toy experiments run on pseudo datasets extracted from the reference sample. The latter have the same statistics of the actual data but do not have any genuine new physics component.

Class imbalance. To accurately represent the reference distribution, it is preferable to consider a large reference sample, while the number of experimental samples is determined by the parameters of the experiment, specifically its luminosity. This leads to an imbalanced classification problem and a natural approach is to re-weight the loss using the inverse class frequencies \mathcal{N}_y (see for instance, Ref. [42]). The true number of expected events differ from the number of events in the reference hypothesis by the number of expected new physics events, i.e., $N(1) = N(0) + N(S)$. Then, one has that $\mathcal{N}_1 \sim \text{Pois}(N(0) + N(S))$. From both the experimental and theoretical points of view, it is reasonable to assume that $N(S) \ll N(0)$. Therefore, one has that $\mathcal{N}_1 \approx N(0)$. Hence, by using the weight in Eq. (18), besides recovering the desired target function, we solve potential issues related with an imbalanced dataset, while keeping the statistical advantage of having a large reference sample.

2.2 Analysis strategy

The complete analysis strategy can be summarized in three steps:

- the test statistic distribution is empirically built by running the training on $N_{toy} = \mathcal{O}(100)$ toy experiments for

¹ This is exact on average, since $\mathcal{N}_1 \sim \text{Pois}(N(1))$.

which both the training sample S_1 and S_0 are generated according to the null hypothesis.

- One last training is performed on the dataset of interest S_1 for which the true underlying hypothesis is unknown and the test statistic value $t(S_1)$ is evaluated.
- The p -value corresponding to $t(S_1)$ is computed with respect to the test statistic distribution under the null hypothesis, studied at step 1.

If a statistically significant deviation from the reference data is found, the nature of the discrepancy can be further characterized by inspecting the learned density ratio in Eq. (9). This quantity is expected to be approximately zero if no disagreement is found and it can be inspected as a function of the input features or their combinations.

Asymptotic formula Typically, for an accurate estimation of $p(t|0)$, the empirical distribution of the test statistic under the reference hypothesis has to be reconstructed using a large number of toy experiments and this might be practically unfeasible. If the value of $t(S_1)$ falls outside of the range of the empirical distribution the p -value cannot be computed and only a lower bound can be set. Inspired by the results by Wald and Wilks [43–45] characterizing the asymptotic behavior of the log-likelihood test statistics, we approximate the null distribution with a χ^2 distribution. We use the toy-based empirical estimate to determine the degrees of freedom of the χ^2 distribution and we test the compatibility of the empirical test statistic distribution with the χ^2 hypothesis using a Kolmogorov–Smirnov test. This approximation holds well in almost all instances of our model. We did not explore this aspect in details but we present a counterexample towards the end of Sect. 4. The same approximation is also used in the neural network model of [1, 16]. It is worth specifying that, in real-life scenarios, if the p -value computed in this way would imply a discovery, one would run additional toys to obtain an accurate empirical estimation by brute-force exploitation of the large-scale computing resources typically accessible by the LHC collaborations.

3 Scalable nonparametric learning with kernels

As mentioned before, a rich model class is needed to effectively detect new physics clues in the data. In this work, we consider kernel methods [40,46] of the form

$$f_w(x) = \sum_{i=1}^{\mathcal{N}} w_i k_\gamma(x, x_i). \tag{24}$$

Here $k_\gamma(x, x_i)$ is the kernel function and γ some hyper-parameter. In our experiments, we consider the Gaussian ker-

nel

$$k_\sigma(x, x') = e^{-\|x-x'\|^2/2\sigma^2}, \tag{25}$$

so that f_w corresponds to a linear combination of Gaussians of prescribed width γ , centered at the input points. Such an approach is called nonparametric because the number of parameters corresponds to the number of data points: the more the data, the more the parameters. Indeed, this makes kernel methods universal in the large sample limit, in the sense that they can recover any continuous function [47,48]. The computational complexity to determine a function as in Eq. (24) is typically cubic in time and quadratic in space with respect to the number of points, since it requires handling the kernel matrix $K_{NN} \in \mathbb{R}^{N \times N}$ with entries $k_\gamma(x_i, x_j)$ (see Refs. [40,49] for further details). These costs prevent the application of basic solvers in large-scale setting, and some approximation is needed. Towards this end we consider Falkon [40], which replaces Eq. (24) by

$$f_w(x) = \sum_{i=1}^M w_i k_\sigma(x, \tilde{x}_i), \tag{26}$$

where $\{\tilde{x}_1, \dots, \tilde{x}_M\} \subset \{x_1, \dots, x_N\}$ are called Nyström centers and are sampled uniformly at random from the input data, with M an hyper-parameter to be chosen. Notably, the corresponding solution can be shown to be with high probability as accurate as the original exact one while computable with only a small fraction of computational resources [50–55]. We defer further details to the appendices.

Algorithm training The model’s weights in Eq. (26) are computed to minimize the empirical error (12) defined by the weighted cross-entropy loss introduced before. Since, the kernel model can be very rich, the search of the best model is done considering

$$\hat{L}(f_w) + \lambda R(f_w), \tag{27}$$

where the first term is the empirical risk, while $R(f)$ is a regularization term

$$R(f_w) = \sum_{ij} w_i w_j k_\sigma(x_i, x_j). \tag{28}$$

constraining the complexity of the model [56]. Problem (27) is then solved by an approximate Newton iteration [40].

Hyper-parameters tuning The number of Nyström centers (M), the bandwidth of the Gaussian kernel σ and the regularization parameter λ are the main hyper-parameters of the model. The number of centers M determines the number of Gaussians, hence it has an impact on the accuracy and on the computational cost; studies suggests that optimal statistical bounds can be achieved already with $M = \mathcal{O}(\sqrt{N})$ [53,57]. On the other hand, by varying the hyper-parameters σ and λ , more or less complex functions can be selected. For large

λ or σ the model simplifies and tends to be linear, while for small values it tends to fit the statistical fluctuations in the data.

The values of M , σ and λ affect the distribution of the test statistic under the reference hypothesis. In particular we observe that the test statistic distribution obtained with different choices of the hyper-parameters always fits a χ^2 distribution with a number of degrees of freedom determined empirically as explained in Sect. 2.2. More complex functions cause the distribution of the test statistic to move to higher values (see Fig. 10a).

On the M direction, a stable configuration is eventually reached and this information can be used to select a proper trade-off value for M (see for instance Fig. 10); conversely there is not clear indication on how to choose the values of σ and λ . The bandwidth σ is related to the resolution of the model and its ability to fit statistical fluctuations in the data. To estimate the relevant scales of the problem and find a good trade-off between complexity and smoothness, we look at the distribution of the pairwise (Euclidean) distance in the reference data. We then fix σ approximately as the 90th percentile (see Appendix C and Fig. 14 for further details). Finally, λ determines the weight of the penalty term in the loss function, which constraint the magnitude of the trainable weights, and avoid instabilities during the training. We take λ as small as possible so that the impact on the weight magnitude is minimum, while maintaining the algorithm numerically stable.

Summarizing, the hyper-parameter tuning protocol is composed by the following three steps:

- We consider a number of centers greater or equal to \sqrt{N} , with the criteria that more centers could improve accuracy but at the cost of losing efficiency.
- We then fix σ approximately as its 90th percentile of the pairwise distance distribution.
- We take λ as small as possible while maintaining a numerically stable algorithm.

Note that we consider the algorithm numerically unstable either when the training fails to converge or when the test statistics evaluates to NaN. Similarly to the tuning procedure introduced in Ref. [16] for the neural networks, the outlined directives for hyper-parameters selection rely on the reference data alone, preserving model-independence.

We tested this heuristic performing several experiments on the toy scenario presented in Appendix C. In particular, we verified that it gives rise to instances that demonstrate good performances, in terms of sensitivity to new physics clues, across different types of signal. We also verified that the results are robust against small variations of the chosen hyper-parameters. When applied to the final experiments presented in the following section, we followed the prescription

given above without any fine tuning that might introduce a bias that favors the specific dataset considered.

Assessing the algorithm performances Following Ref. [1], in order to evaluate different models on benchmark cases it is useful to introduce the ideal significance Z_{id} , i.e., the value of the median Z-score that is obtained by using the exact (ideal) likelihood ratio test statistics:

$$t_{id}(\cdot) = -2 \log \frac{\mathcal{L}(\cdot, 1)}{\mathcal{L}(\cdot, 0)}. \quad (29)$$

Typically, this quantity cannot be computed exactly since the likelihoods are not known analytically. We can however obtain an accurate estimate \hat{Z}_{id} using simulated data and model-dependent analyses that leverage what is known about the type of new physics in the data. We will report how \hat{Z}_{id} has been computed for every experiment.

4 Experiments

In this section, we apply the proposed approach to three realistic simulated high energy physics datasets with an increasing number of dimensions. Each dataset is made of two classes: a reference class, containing events following the Standard Model, and a data class, made of reference events with the injection of a new physics signal. Each case includes a set of features given by kinematical variables as measured by the particle detectors (plus additional quantities when available, such as reconstructed missing momenta and b-tagging information) that we call *low-level features*. From the knowledge of the intermediate physics processes, one can compute additional *high-level features* that are functions of low-level ones and possess a higher discriminative power.² The different features are used to test the flexibility of the model. The pipeline for training and tuning our method is that described in Sect. 3.

4.1 Datasets

Here, we briefly review some properties of the datasets, how \hat{Z}_{id} is computed and the parameters chosen for the experiments. We refer the reader to Ref. [16, 58] for further details.

DIMUON This is a five dimensional simulated dataset that was introduced in Ref. [16] and it is composed of simulated LHC collision events producing two muons in the final state $pp \rightarrow \mu^+ \mu^-$, at a center-of-mass energy of 13 TeV.³ The low-level features are the transverse momenta and pseudorapidities of the two muons and their relative azimuthal angle, i.e., $x = [p_{T1}, p_{T2}, \eta_1, \eta_2, \Delta\phi]$. We consider two

² We borrow this nomenclature from Ref. [58].

³ Data available at <https://zenodo.org/record/4442665>.

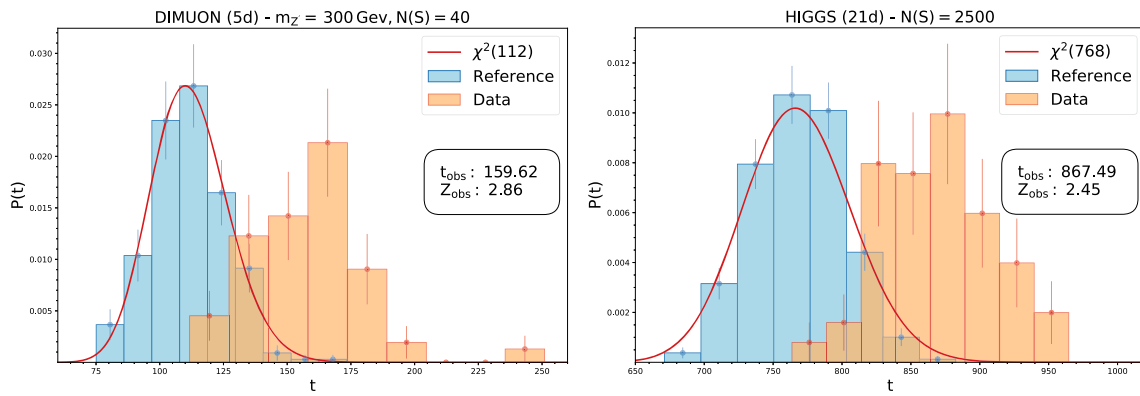


Fig. 1 Distribution of the test statistics under the null and alternative hypotheses for the DIMUON (left) and HIGGS (right) datasets

types of new physics contributions: the first one is a new vector boson (Z') for which we study different mass values ($m_{Z'} = 200, 300$ and 600 GeV); the second one is instead a non-resonant signal obtained by adding a four fermions contact interaction to the Standard Model lagrangian

$$\frac{c_W}{\Lambda} J_{L\mu}^a J_{La}^\mu \tag{30}$$

where J_{La}^μ is the $SU(2)_L$ Standard Model current, the energy scale Λ is fixed at 1 TeV and the Wilson coefficient c_W determining the coupling strength can be chosen between three values ($c_W = 1, 1.2$ and 1.5 TeV^{-2}). For both types of signal the invariant mass of the two muons is the most discriminant non trivial combination of the kinematic variables describing the system so we consider it as a high-level feature. We fix $N(0) = 2 \times 10^4$ expected events in the reference hypothesis and the size of the reference sample is $N_0 = 10^5$, unless specified otherwise. We vary the number of expected signal events in the range $N(S) \in [6, 80]$. We selected the following hyper-parameters: $(M, \sigma, \lambda) = (2 \times 10^4, 3, 10^{-6})$.

Following Ref. [16], the ideal significance is estimated via a cut-and-count strategy in the invariant mass $m_{\ell\ell}$ distribution around $m_{Z'}$ for the resonant signal, while a likelihood ratio test on the binned $m_{\ell\ell}$ distribution is used for the non-resonant case.

SUSY The SUSY dataset [58] is composed of simulated LHC collision events in which the final state is made of two charged leptons $\ell\ell$ and missing momentum.⁴ The latter is given, in the Standard Model, by two neutrinos coming from the fully leptonic decay of the two W bosons. The new physics scenario also includes the decay of a pair of electrically charged supersymmetric particles $\tilde{\chi}^\pm$ in two neutral supersymmetric particles $\tilde{\chi}^0 \tilde{\chi}^0$, undetectable and thus contributing to the missing transverse momentum, and two W bosons. The dataset has 8 raw features and 10 high-level features.

Unless specified differently, we take $N(0) = 10^5$ and $N_0 = 5 \times 10^5$ and we vary the signal component in $N(S) \in [200, 650]$. We selected the following hyper-parameters: $(M, \sigma, \lambda) = (10^4, 4.5, 10^{-6})$ when using the raw features, increasing σ to 5 when the high-level features are included.

The ideal significance is estimated by training a supervised classifier to discriminate between background and signal with a total of 2M examples, following the approach in Ref. [58]. The significance is then estimated by a cut-and-count analysis on the classifier output.

HIGGS The HIGGS dataset [58] is made of simulated events in which the signal is given by the production of heavy Higgs bosons H .⁵ The final state is given by a pair of vector bosons $W^\pm W^\mp$ and two bottom quarks $b\bar{b}$ for both the reference and the signal components. The dataset has 21 raw features and 7 high-level features.

Unless specified differently, we choose $N(0) = 10^5$, $N_0 = 5 \times 10^5$ and we vary the signal component in $N(S) \in [1000, 2500]$. We take the following hyper-parameters: $(M, \sigma, \lambda) = (10^4, 7, 10^{-6})$ when using the raw features and $\sigma = 7.5$ when the high-level features are included.

The ideal significance is estimated as in the previous case by using the output of a supervised classifier trained to separate signal from background (again, following the approach in Ref. [58]).

4.2 Results

Sensitivity to new physics We discuss here the sensitivity of the model to the presence of new physics signals in the data. The test statistic distribution under the reference hypothesis is empirically reconstructed using 300 toy experiments while 100 toys are used to reconstruct the distribution of the test statistic under the alternative new physics scenarios (see two instances in Fig. 1). We show in Fig. 2 the median observed significance against the estimated ideal significance with

⁴ Data available at <https://archive.ics.uci.edu/ml/datasets/SUSY>.

⁵ Data available at <https://archive.ics.uci.edu/ml/datasets/HIGGS>.

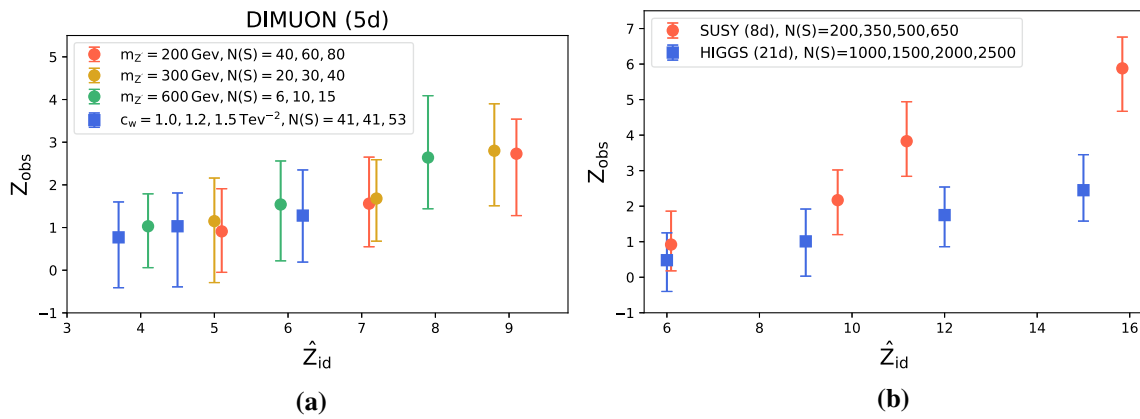


Fig. 2 Observed significance against estimated ideal significance with low-level input features

Falkon trained on low-level features only. These experiments were performed by varying the signal fraction $N(S)/N(0)$ (at fixed luminosity) and the type of signal (the latter in the DIMOUN case only). The error bars represent the 68% confidence interval. As expected for a model-independent strategy, the observed significance is always lower than what obtainable with a model-dependent approach. The loss of sensitivity is more pronounced in higher dimensions. Nevertheless, we observe in all cases a correlation between the observed and the ideal significance. In the DIMOUN case, the observe significance seems to depend weakly on the type of new physics signal. In Fig. 3, we show explicitly, for the Z' new physics with $m_{Z'} = 300$ GeV, the estimated probabilities to find a discrepancy of at least α for a given value of \hat{Z}_{id} . Similar results are obtained with the other types of signal. To test the ability of the kernel-based approach to extract useful information from data, we show in Fig. 4 that adding the high-level features does not significantly improve the results, especially in higher dimensions. The plot includes

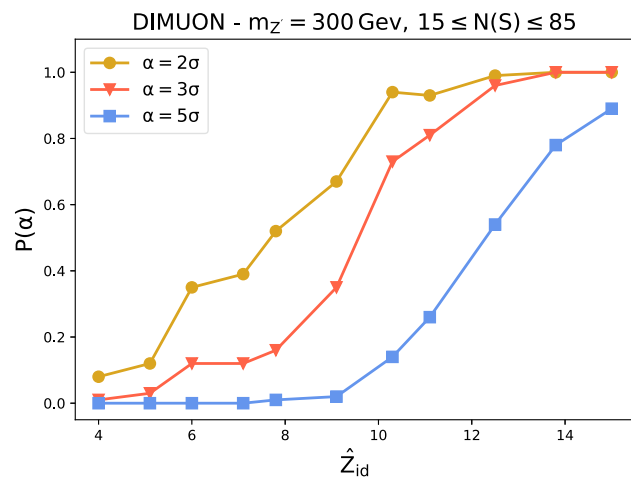


Fig. 3 Probability of finding a $\alpha = 2\sigma, 3\sigma, 5\sigma$ evidence for new physics as a function of the ideal significance

the observed significance, with the bar showing the 68% confidence interval and the grey area representing the region $Z_{obs}^{(all)} = Z_{obs}^{(low-level)} \pm \sigma$.

Comparison with neural networks To compare the kernel-based approach with the neural network implementation, we considered the results from Ref. [16] for the DIMOUN dataset, while we trained the latter on the SUSY and the HIGGS datasets. The considered neural network has 2 hidden layer with 10 neurons each and a weight clipping of $w_{clip} = 0.87$ for SUSY, while it has 5 layers with 6 neurons each layer and $w_{clip} = 0.65$ for HIGGS. Training is stopped after 3×10^5 epochs. The results are summarized in Figs. 4 and 5. We see that, overall, the two approaches give similar results and the degradation of the sensitivity in high dimensions affects both. We notice that in the DIMOUN case, the kernel approach is slightly less sensitive, as it can be seen from the results presented in Section 5 of Ref. [16] against Figs. 2a and 3. However, by looking at Fig. 5 we see that, for the HIGGS dataset, the kernel approach gives a higher observed significance while, for the SUSY dataset, the two methods give almost identical results. On the other hand, the average training times, summarized in Table 1, demonstrate an advantage in favor of the kernel approach of orders of magnitude. This also allows efficient training on single GPU machines and ensures high scalability for multi-GPU systems, as shown in Ref. [40].

As an example, one would need approximately 10^3 toys for a 3 sigma assessment purely based on toys. Considering the worst case for the kernel approach, which is the DIMOUN dataset, this requires about 4.5×10^4 sec of computing time for the kernel approach, if toys are processed in series. On a cluster of CPUs that permits the simultaneous training of 300 toys, the neural network model requires about 5×10^4 sec. This estimate does not take into account the multiple trainings needed for hyper-parameter tuning and model selection. Alternatively, one can assess the compatibility of the distribution of the test statistics with a χ^2 asymptotic distribution, in

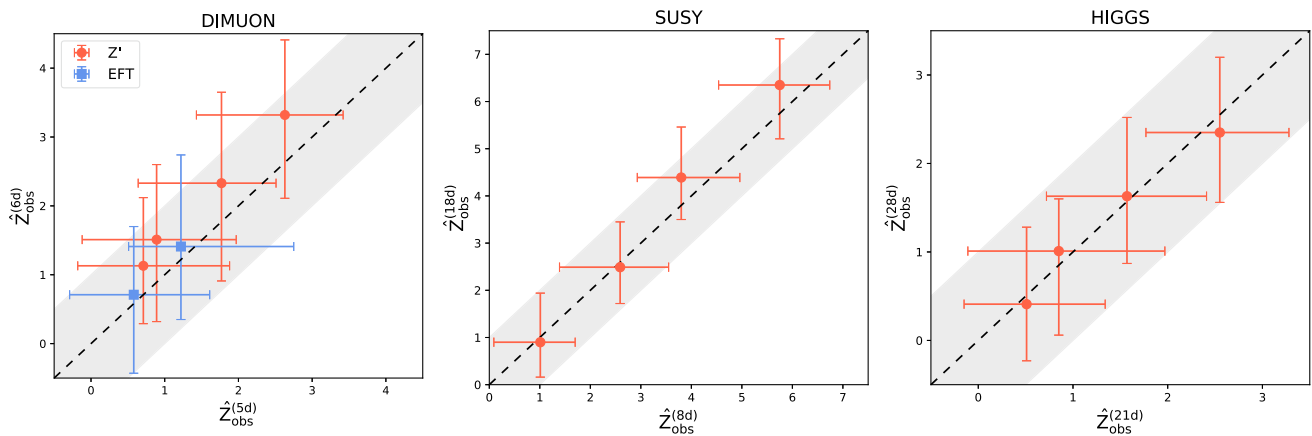
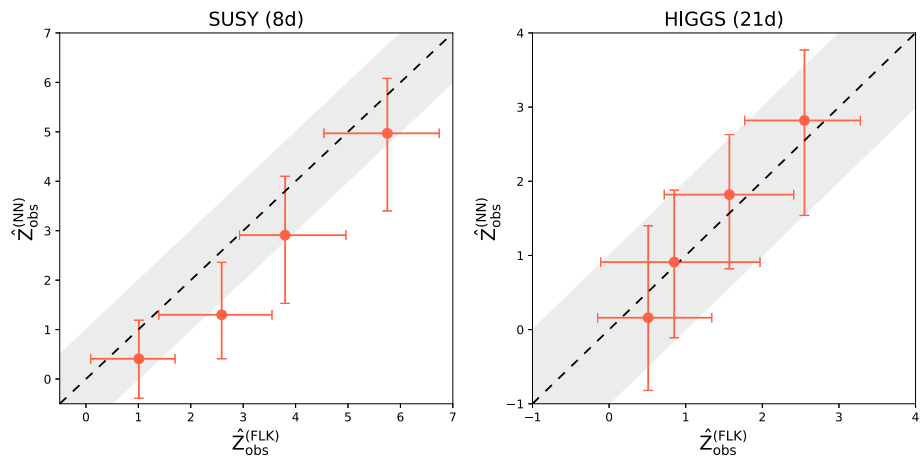


Fig. 4 Comparison of the observed significance obtained with Falkon using low level features only and all the features

Fig. 5 Observed significance with the Falkon implementation against neural networks



analogy to what is done at the LHC in the asymptotic regime of the LHC test statistics, with a smaller number of toys, say 100. In this case the kernel approach takes about 4.5×10^2 sec while the neural network model requires about 1.4×10^4 sec. It is also worth noticing that in our experiments, although the training of neural networks is faster on GPUs, the speedup obtained by processing toys in parallel on a large number of CPUs wins over the slowdown factor. Ideally, training on a large number of GPUs would be better, however this option was not available at the computing facility where this study was performed.

Learned density ratio As discussed in Sect. 2, the function approximated by using the weighted cross-entropy loss is the density ratio given in Eq. (9). The latter can be directly inspected to characterize the nature of the “anomalies” in the experimental data, if found significant. We report in Fig. 6 examples of the reconstructed density ratios as functions of certain high-level features (not given as inputs) together with estimates of the true ratios and extrapolations from the data used for training. The learned density ratio is constructed by re-weighting the relevant high-level feature of the reference sample by $e^{f_{\tilde{w}}(x)}$ (evaluated on the reference training data), binning it and taking the ratio with the same binned reference

sample (unweighted). The toy density ratio is computed by replacing the numerator with the binned distribution of the high-level feature of the toy data sample. The ideal case is obtained in the same way but using a large ($\geq 1M$) data sample instead.

Size of the reference sample A larger reference sample yields a better representation of the reference model, which is crucial for a model-independent search. In Fig. 7a, we see that as long as $\mathcal{N}_0/N(0) \gtrsim 1$, the median observed significance is indeed stable. On the other hand, when the reference sample is too small ($\mathcal{N}_0/N(0) < 1$), we observe that the correspondence between the distribution of the test statistics and the χ^2 distribution breaks down, see Fig. 7b. We observe this behavior for all the datasets. Then, it is in general a good approach to take a reference sample as large as possible keeping in consideration the computational cost of training on a possibly very large dataset.

Resources The models based on Falkon have been trained on a server with the specifications reported in Table 2. The NN experiments have been performed on a CPU farm with 32 computing nodes of Intel 64 bit dual processors, for a total amount of 712 core. The codes to reproduce the experiments

Table 1 Average training times per single run with standard deviations (low level features and reference toys). Note that time measured in hours (for NN) and seconds (for Falcon)

Model	DIMUON	SUSY	HIGGS
FLK	(44.9 ± 3.4) s	(18.2 ± 1.2) s	(22.7 ± 0.4) s
NN	(4.23 ± 0.73) h	(73.1 ± 10) h	(112 ± 9) h

Bold values indicate the lowest for each column (lower is better)

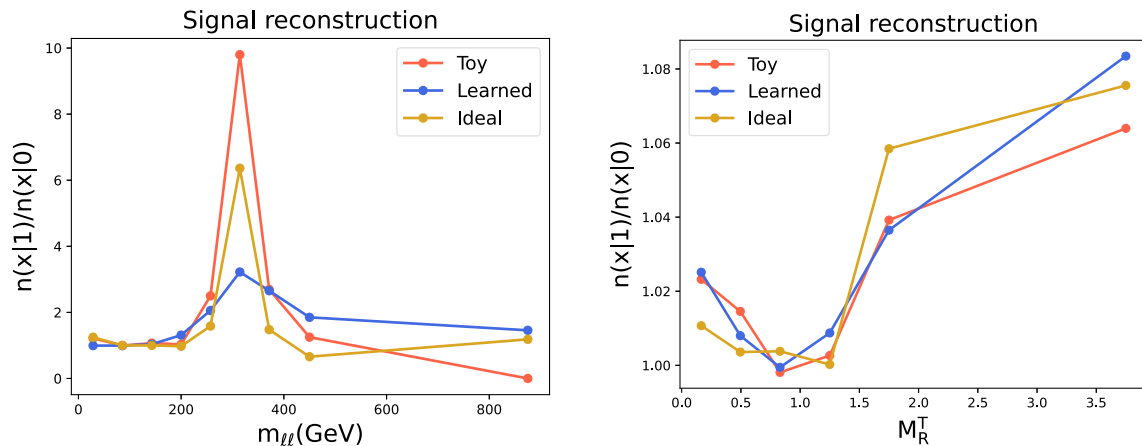


Fig. 6 Examples of reconstructed density ratios as a functions of high-level features (not given as inputs) for the DIMUON (left) and SUSY (right) datasets with new physics components in the data. Note that the SUSY dataset is normalized

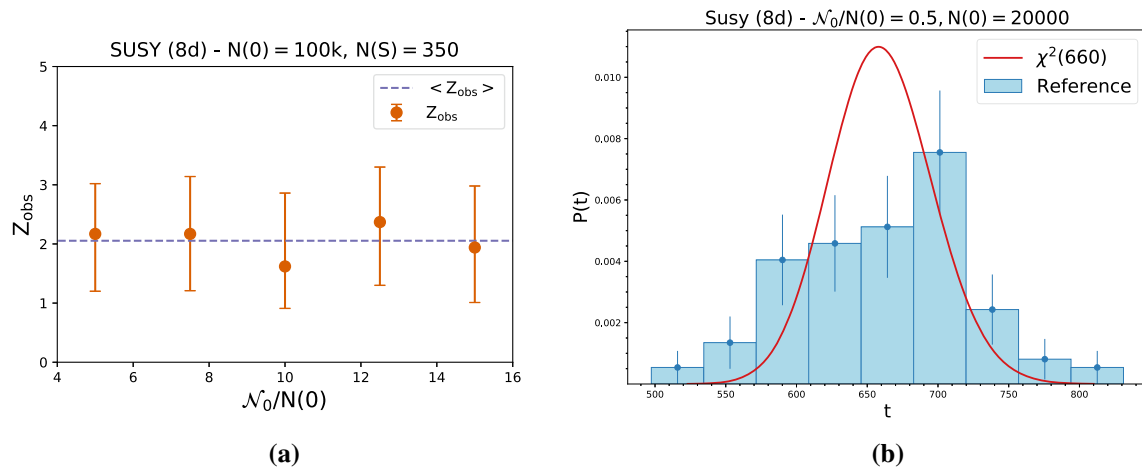


Fig. 7 Observed significance as a function of the size of the reference sample (left). Example of distribution of the test statistics given a small reference sample (right)

are available at https://github.com/GaiaGrosso/Learning_NP and <https://github.com/FalconHEP/falconhep>.

5 Conclusions

In this work we have presented a machine learning approach for model-independent searches applying kernel-based machine learning models to the ideas introduced in Ref. [1, 16]. Our approach is powered at its core by Falcon, a recent library

developed for large scale applications of kernel methods. The focus of our work is on computational efficiency. Indeed, the original neural network proposal suffers from long training times which, combined with a toy-based hypothesis testing framework, makes the use of the algorithm challenging in high dimensional cases. Our model delivers comparable performances with a dramatic reduction in training times, as shown in Table 1. As a consequence, the model can be efficiently trained on single GPU machines while possessing high scalability for multi-GPU systems [40]. In contrast, the

Table 2 Specifications of the machine used to perform the experiments with Falkon

OS	Ubuntu 18.04.1
CPU(s)	2× Intel(R) Xeon(R) Silver 4116 CPU
RAM	256GB
GPU(s)	2× NVIDIA Titan Xp (12 GB RAM)
CUDA version	10.2

neural network implementation crucially relies on per toy parallelization, hence the need for large scale resources such as CPU/GPU clusters.

Similarly to Ref. [16], the applicability of the proposed method relies on a heuristic procedure to tune the algorithm hyper-parameters. A more in-depth understanding of the interplay between the expressibility of the model, its complexity and the topology of the input dataset could lead to more performant and better motivated alternatives to the current hyper-parameter selection. Further investigations are left for future work. One possibility would be to find a more principled way to relate Falkon hyper-parameters to physical quantities. This could also allow the introduction of explicit quantities to be optimized, opening to the possibility of applying modern optimization techniques for the selection of the hyper-parameters.

Besides the challenges related to the algorithm optimization and regularization, an essential development for the application to realistic data analysis concerns the treatment of systematic uncertainties which has not been considered in the present work. This aspect was successfully addressed on a recent work [41] in the context of the neural network implementation.

A crucial step towards the understanding of this approach, both in its neural network and kernel-based implementations, would be an in-depth comparison with similar approaches based on typical classifiers' metrics, such as classification accuracy and AUC. Preliminary results suggest that the strategy based on the maximum likelihood-ratio test, presented in Ref. [1] and further explored in this work, delivers better performances across several types of new physics scenarios. An extended analysis will be presented elsewhere.

Finally, the boost in efficiency provided by the model developed in this work could extend the landscape of applicability of this analysis strategy to other use cases, beyond the search for new physics, and to other domains. In particular, the application to multivariate data quality monitoring in real time is currently under study.

Acknowledgements L.R., M.L. and M.R. acknowledge the financial support of the European Research Council (grant SLING 819789). L.R. acknowledges the financial support of the AFOSR projects FA9550-18-1-7009, FA9550-17-1-0390 and BAA-AFRL-AFOSR-2016-0007 (European Office of Aerospace Research and Development) and the EU H2020-MSCA-RISE project NoMADS-DLV-777826. We grate-

fully acknowledge the support of NVIDIA Corporation for the donation of the Titan Xp GPUs and the Tesla k40 GPU used for this research. M.P. and G.G. are supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement no 772369). A.W. acknowledges support from the Swiss National Science Foundation under contract 200021-178999 and PRIN grant 2017FMJFMW.

Data Availability Statement This manuscript has associated data in a data repository. [Authors' comment: The datasets used in this work can be found at <https://zenodo.org/record/4442665#.YAGiaC9h23I> (DIMUON), <https://archive.ics.uci.edu/ml/datasets/SUSY> (SUSY) and <https://archive.ics.uci.edu/ml/datasets/HIGGS> (HIGGS).]

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 SCOAP³. SCOAP³ supports the goals of the International Year of Basic Sciences for Sustainable Development.

Appendices

A Loss functions and target functions

Different loss functions determine different goals via an associated target function f^* . This is the function learned by the model in the large-sample limit and it can be computed, given a loss function $\ell(y, f(x))$, by considering the expected risk

$$L(f) = \int \ell(y, f(x)) dp(x, y), \quad (31)$$

where $p(x, y)$ is the true joint distribution. It can be further rewritten as

$$\begin{aligned} L(f) &= \int \ell(y, f(x)) p(x, y) dx dy \\ &= \int p(x) dx \int \ell(y, f(x)) p(y|x) dy. \end{aligned} \quad (32)$$

One can then find the minimizer simply as

$$f^* = \arg \min_{f \in \mathbb{R}} \int \ell(y, f(x)) p(y|x) dy, \quad \forall x, \quad (33)$$

with $p(x)p(y|x) = p(x, y)$. In the case of the weighted cross-entropy loss, one has

$$\ell(y, f(x)) = a_0 (1 - y) \log(1 + e^{f(x)}) + a_1 y \log(1 + e^{-f(x)}), \quad (34)$$

with $y = \{0, 1\}$. One then simply takes the derivative and sets it equal to zero, obtaining the following minimizer

$$f^* = \log \frac{p(1|x) a_1}{p(0|x) a_0}. \quad (35)$$

B Falkon

In this appendix, we provide more details on Falkon [40], the algorithm powering our model. The original library includes an implementation based on the square loss, which we do not discuss here. The core ideas from a theoretical and algorithmic viewpoint are developed in Ref. [49, 57, 59].

The problem of minimizing the regularized empirical risk in Eq. (27) is formulated in terms of an approximate Newton method (see Algorithm 2 of Ref. [40]) The model is based on the Nyström approximation, which is used twice. First to reduce the size of the problem, by considering solutions of the form shown in Eq. (26). Then, it is again used to derive an approximate Newton step. At every step, preconditioned conjugate gradient descent is run for a limited number of iterations with a decreasing sequence of regularization parameters λ_k , down to the desired regularization level. We choose $k = 1$ in our experiments, as we did not observe any benefit in selecting more values. The preconditioner plays here the role of approximate Hessian. Given $(x_i, y_i)_{i=1}^m$ selected uniformly at random from the dataset and let T be the Cholesky decomposition of K_{mm} then the approximate Hessian \tilde{H} has the form

$$\tilde{H} = \frac{1}{m} T \tilde{D}_k T^T + \lambda_k I, \quad (36)$$

where $\tilde{D}_k \in \mathbb{R}^{m \times m}$ is a diagonal matrix s.t. the i -th element is the second derivative of the loss $\ell''(y_i, f(x_i), \cdot)$ with respect to the first variable. To preserve efficiency, this matrix is never built explicitly but we build it in terms of Cholesky decomposition: let A be the Cholesky decomposition of Eq. (36), we compute

$$P = T^{-1} A^{-1} \quad \tilde{H}^{-1} = P P^T.$$

Then conjugate gradient is applied to solve the preconditioned problem at time k

$$P^T (K_{nm}^T D_k K_{nm} + \lambda_k I) P \beta = P^T K_{nm}^T g_k,$$

where $g_k \in \mathbb{R}^n$ such that $(g_k)_i = l'(f(x_i), y_i)$. With this strategy, the overall computational cost to achieve optimal statistical bounds is $\mathcal{O}(n\sqrt{n} \log n)$ in time, and in $\mathcal{O}(n)$ in memory, making it suitable for large scale problems.

C 1D example

We consider here a simple univariate toy scenario taken from Ref. [1]. We use this example to present explicitly all the steps discussed in Sects. 2 and 3.

Data. We know here explicitly both the reference and the true data generating distributions. The former is an exponential

$$n(x|0) = N(0) 8 e^{-8x}. \quad (37)$$

The latter is given by the reference distribution combined with a signal component and reads as

$$n(x|1) = n(x|0) + n(x|S) = N(0)p(x|0) + N(S)p(x|S), \quad (38)$$

where $p(x|S)$ is the distribution of the signal alone and $N(S)$ is the expected number of signal events. We consider three types of signals, two that are localized in a region of the input feature (resonant) and one that is not (non-resonant). They are given by the following expressions (see also Fig. 8):

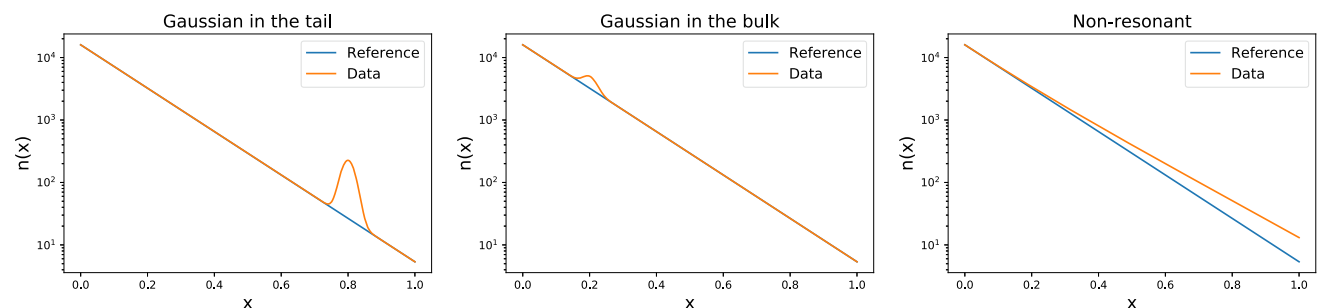


Fig. 8 True univariate densities

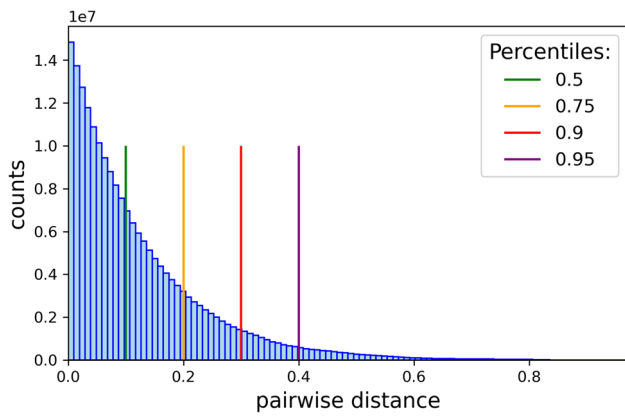


Fig. 9 Euclidean pairwise distance

- A Gaussian distribution centered in the tail of the exponential background

$$n(x|S_1) = N(S_1) \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu_1)^2/2\sigma^2}, \quad (39)$$

$$\mu_1 = 0.8, \quad \sigma = 0.02, \quad N(S_1) = 10,$$

- A Gaussian distribution centered in the bulk of the exponential background

$$n(x|S_2) = N(S_2) \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu_2)^2/2\sigma^2}, \quad (40)$$

$$\mu_2 = 0.2, \quad \sigma = 0.02, \quad N(S_2) = 90,$$

- A non-resonant signal given by

$$n(x|S_3) = N(S_3) 256 x^2 e^{-8x}, \quad N(S_3) = 90. \quad (41)$$

We select a large reference sample of size $N_0 = 2 \times 10^5$ and an expected number of background events $N(0) = 2 \times 10^3$. The size of the data sample is then $N_1 \sim \text{Pois}(N(y))$,

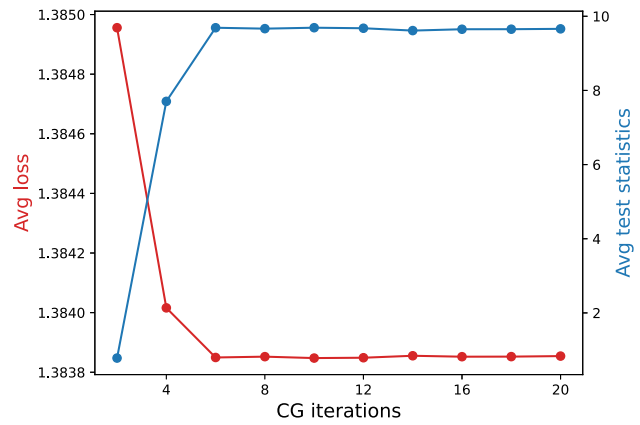
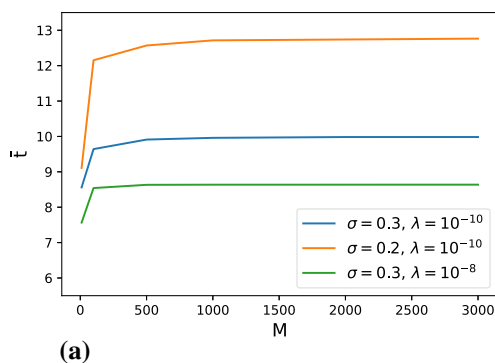


Fig. 11 Average loss and test statistics as functions of the number of conjugate gradient iterations (reference toys)

with $N(1) = N(0) + N(S)$ when the data sample is generated according to the true data distribution.

Model tuning In Fig. 9 we show the distribution of the pairwise distances from which we select the bandwidth as approximately the 90th percentile. In this case, it corresponds to $\sigma \approx 0.3$.

In Fig. 10a we show that the test statistics averaged over twenty independent runs (with reference data only) reaches a plateau at $M \approx 500 \approx \sqrt{N_0}$. This suggests that the estimated distribution of the test statistics under the null hypothesis does not change if more centers are selected. On the other hand, larger values of M might increase the sensitivity of the model to new physics, at the expense of efficiency in training times and memory. By looking at the average training time, as reported for instance in Fig. 10b, we fix $M = 3000$. Figure 10a also shows that the value of the test statistics increases for more complex models (smaller σ and/or λ). Finally, we take $\lambda = 10^{-10}$ because the training would show occasional instabilities at smaller values. In Fig. 11, we show the average loss and test statistics as functions of the conjugate gradient iterations of the algorithm.

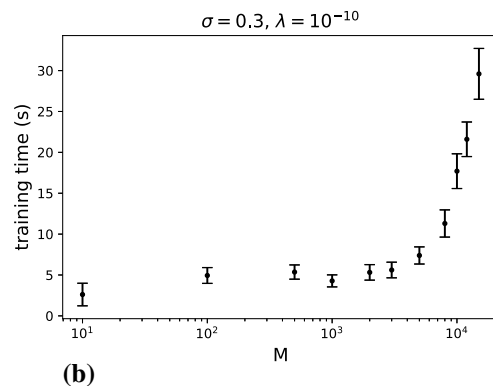
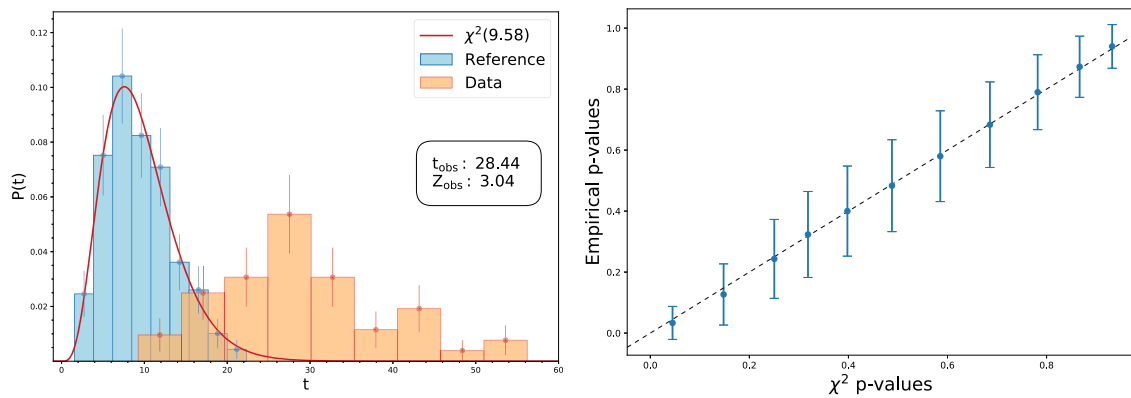


Fig. 10 **a** Average test statistics as a function of the number of Nyström centers. **b** Average training time as a function of the number of Nyström centers



(a) Distribution of the test statistics under the null and alternative hypotheses (non-resonant signal). (b) Empirical and χ^2 null distributions, integrated to p-values.

Fig. 12 a Distribution of the test statistics under the null and alternative hypotheses (non-resonant signal). b Empirical and χ^2 null distributions, integrated to p-values

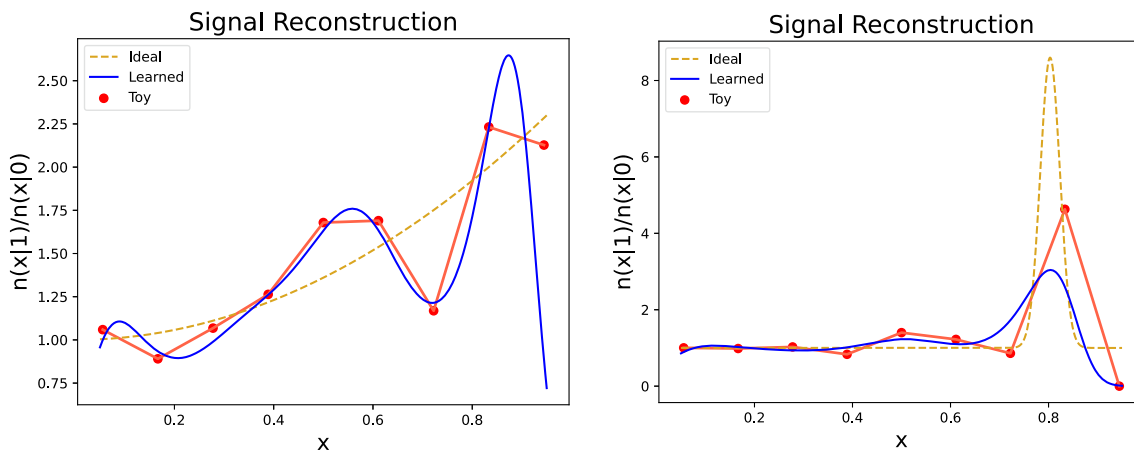


Fig. 13 Reconstructed density ratios. Non-resonant signal (left) and Gaussian in the tail (right)

Model training and results We first reconstruct the distribution of the test statistics under the null hypothesis $p(t|0)$. We train the algorithm on $N_{toys} = 300$ toy reference samples ($N(S) = 0$). In this simple scenario, we use our complete knowledge of the problem to reconstruct the distribution of the test statistics under the alternative hypothesis $p(t|1)$ by performing multiple experiments with $N_{toys} = 100$ toy data samples with injection of new physics events. The reconstructed distribution for the non-resonant case is shown in Fig. 12a. We can see from Fig. 12b that the test statistics with reference data follows a χ^2 distribution with 9.58 degrees of freedom (determined with a Kolmogorov–Smirnov test). The median observed significance for the three cases is $Z_{obs} = (2.43, 2.82, 3.04)$. The average training time for a single reference toy is $t_{train} \approx 2.11$ s. In this case we can compute the ideal test statistics exactly using the true distributions as follows

$$t_{id}(S) = -2 \left[-N(S) + \sum_{x \in S} \log \left(1 + \frac{n(x|S)}{n(x|0)} \right) \right] \quad (42)$$

This quantity is then evaluated on a large number (10M) of reference examples to accurately reconstruct $p(t|0)$ and on 300 data samples for each type of signal. This was done in Ref. [1] and the resulting values are $\hat{Z}_{id} = (4.7, 4.1, 4.4)$. In this examples, we lose approximately 1.6σ of sensitivity on average.

We can also inspect the learned density ratio to characterize the potential new physics clues. In this 1D case, this amounts to simply look at where $\exp(f_{\hat{w}}(x))$ deviates significantly from one. We show some examples in Fig. 13. They are obtained by showing the ideal (exact) likelihood ratio, the ratio between the (binned) toy and reference samples and the learned functions.

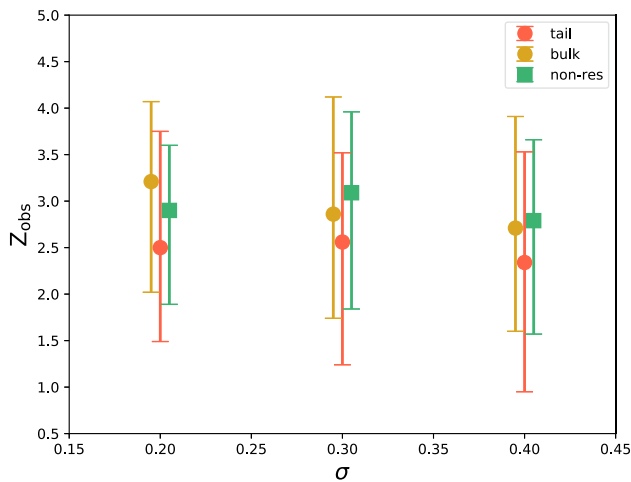


Fig. 14 Observed significance at varying kernel bandwidth

Finally, Fig. 14 shows that the results are stable around the selected bandwidth σ across the different types of new physics signals.

References

- R.T. D'Agnolo, A. Wulzer, Learning new physics from a machine. Phys. Rev. D **99**(1), 015014 (2019). [arXiv:1806.02350](#) [hep-ph]
- G. Choudalakis, On hypothesis testing, trials factor, hypertests and the BumpHunter. In *PHYSTAT 2011*, 1 (2011). [arXiv:1101.0390](#) [physics.data-an]
- B. Abbott et al., Search for new physics in $e\mu X$ data at DØ using SLEUTH: a quasi-model-independent search strategy for new physics. Phys. Rev. D **62**, 092004 (2000). [arXiv:hep-ex/0006011](#)
- V.M. Abazov et al., A quasi model independent search for new physics at large transverse momentum. Phys. Rev. D **64**, 012004 (2001). [arXiv:hep-ex/0011067](#)
- A. Aktas et al., A general search for new phenomena in ep scattering at HERA. Phys. Lett. B **602**, 14–30 (2004). [arXiv:hep-ex/0408044](#)
- F.D. Aaron et al., A general search for new phenomena at HERA. Phys. Lett. B **674**, 257–268 (2009). [arXiv:0901.0507](#) [hep-ex]
- P. Asadi, M.R. Buckley, A. DiFranzo, A. Monteux, D. Shih, Digging deeper for new physics in the LHC data. JHEP **11**, 194 (2017). [arXiv:1707.05783](#) [hep-ph]
- T. Aaltonen et al., Model-independent and quasi-model-independent search for new physics at CDF. Phys. Rev. D **78**, 012002 (2008). [arXiv:0712.1311](#) [hep-ex]
- T. Aaltonen et al., Global search for new physics with 2.0 fb^{-1} at CDF. Phys. Rev. D **79**, 011101 (2009). [arXiv:0809.3781](#) [hep-ex]
- A. Meyer, CMS Collaboration, Music-an automated scan for deviations between data and monte carlo simulation. In *AIP Conference Proceedings*, vol. 1200, pp. 293–296 (American Institute of Physics, 2010)
- CMS Collaboration, Music: a model-unspecific search for new physics in proton–proton collisions at. Eur. Phys. J. C **81**, 629 (2021)
- A general search for new phenomena with the ATLAS detector in pp collisions at $\sqrt{s} = 8 \text{ TeV}$. Technical report, CERN, Geneva (Mar 2014). All figures including auxiliary figures are available at <https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2014-006>
- M. Aaboud et al., A strategy for a general search for new phenomena using data-derived signal regions and its application within the ATLAS experiment. Eur. Phys. J. C **79**(2), 120 (2019). [arXiv:1807.07447](#) [hep-ex]
- C. Weisser, M. Williams, Machine learning and multivariate goodness of fit. 12 (2016). [arXiv:1612.07186](#) [physics.data-an]
- O. Cerri, T.Q. Nguyen, M. Pierini, M. Spiropulu, J.-R. Vlimant, Variational autoencoders for new physics mining at the large hadron collider. JHEP **05**, 036 (2019). [arXiv:1811.10276](#) [hep-ex]
- R.T. D'Agnolo, G. Grosso, M. Pierini, A. Wulzer, M. Zanetti, Learning multivariate new physics. Eur. Phys. J. C **81**(9), 89 (2021). [arXiv:1912.12155](#) [hep-ph]
- A. De Simone, T. Jacques, Guiding new physics searches with unsupervised learning. Eur. Phys. J. C **79**(4), 289 (2019). [arXiv:1807.06038](#) [hep-ph]
- M. Farina, Y. Nakai, D. Shih, Searching for new physics with deep autoencoders. Phys. Rev. D **101**(7), 075021 (2020). [arXiv:1808.08992](#) [hep-ph]
- J.H. Collins, K. Howe, B. Nachman, Anomaly detection for resonant new physics with machine learning. Phys. Rev. Lett. **121**(24), 241803 (2018). [arXiv:1805.02664](#) [hep-ph]
- A. Blance, M. Spannowsky, P. Waite, Adversarially-trained autoencoders for robust unsupervised new physics searches. JHEP **10**, 047 (2019). [arXiv:1905.10384](#) [hep-ph]
- J. Hajer, Y.-Y. Li, T. Liu, H. Wang, Novelty detection meets collider physics. Phys. Rev. D **101**(7), 076015 (2020). [arXiv:1807.10261](#) [hep-ph]
- T. Heimele, G. Kasieczka, T. Plehn, J.M. Thompson, QCD or what? SciPost Phys. **6**(3), 030 (2019). [arXiv:1808.08979](#) [hep-ph]
- J.H. Collins, K. Howe, B. Nachman, Extending the search for new resonances with machine learning. Phys. Rev. D **99**(1), 014038 (2019). [arXiv:1902.02634](#) [hep-ph]
- B. Nachman, D. Shih, Anomaly detection with density estimation. Phys. Rev. D **101**, 075042 (2020). [arXiv:2001.04990](#) [hep-ph]
- A. Andreassen, B. Nachman, D. Shih, Simulation assisted likelihood-free anomaly detection. Phys. Rev. D **101**(9), 095004 (2020). [arXiv:2001.05001](#) [hep-ph]
- O. Amram, C.M. Suarez, Tag N' Train: a technique to train improved classifiers on unlabeled data. JHEP **01**, 153 (2021). [arXiv:2002.12376](#) [hep-ph]
- B.M. Dillon, D.A. Faroughy, J.F. Kamenik, M. Szwec, Learning the latent structure of collider events. JHEP **10**, 206 (2020). [arXiv:2005.12319](#) [hep-ph]
- T. Cheng, J.-F. Arguin, J. Leissner-Martin, J. Pilette, T. Golling, Variational autoencoders for anomalous jet tagging 7 (2020). [arXiv:2007.01850](#) [hep-ph]
- C.K. Khosa, V. Sanz, Anomaly Awareness, 7 (2020). [arXiv:2007.14462](#) [cs.LG]
- B. Nachman, Anomaly detection for physics analysis and less than supervised learning, 10 (2020). [arXiv:2010.14554](#) [hep-ph]
- S.E. Park, D. Rankin, S.-M. Udrescu, M. Yunus, P. Harris, Quasi anomalous knowledge: searching for new physics with embedded knowledge. JHEP **21**, 030 (2020). [arXiv:2011.03550](#) [hep-ph]
- B. Bortolato, B.M. Dillon, J.F. Kamenik, A. Smolkovič, Bump hunting in latent space, 3 (2021). [arXiv:2103.06595](#) [hep-ph]
- T. Finke, M. Krämer, A. Morandini, A. Mück, I. Oleksiyuk, Autoencoders for unsupervised anomaly detection in high energy physics. JHEP **06**, 161 (2021). [arXiv:2104.09051](#) [hep-ph]
- J. Gonski, J. Lai, B. Nachman, I. Ochoa, High-dimensional anomaly detection with radiative return in e^+e^- collisions, 8 (2021). [arXiv:2108.13451](#) [hep-ph]
- A. Hallin, J. Isaacson, G. Kasieczka, C. Krause, B. Nachman, T. Quadfasel, M. Schlaffer, D. Shih, M. Sommerhalder, Classifying Anomalies THrough Outer Density Estimation (CATHODE), 9 (2021). [arXiv:2109.00546](#) [hep-ph]

36. B. Ostdiek, Deep set auto encoders for anomaly detection in particle physics, 9 (2021). [arXiv:2109.01695](#) [hep-ph]
37. P. Chakravarti, M. Kuusela, J. Lei, L. Wasserman, Model-independent detection of new physics signals using interpretable semi-supervised classifier tests, 2 (2021). [arXiv:2102.07679](#) [stat.AP]
38. G. Kasieczka et al., The LHC Olympics 2020 a community challenge for anomaly detection in high energy physics. Rep. Prog. Phys. **84**(12), 124201 (2021). [arXiv:2101.08320](#) [hep-ph]
39. T. Aarrestad et al., The dark machines anomaly score challenge: benchmark data and model independent event classification for the large hadron collider. SciPost Phys. **12**(1), 043 (2022). [arXiv:2105.14027](#) [hep-ph]
40. G. Meanti, L. Carratino, L. Rosasco, A. Rudi, Kernel methods through the roof: handling billions of points efficiently. Adv. Neural Inf. Process. Syst. **33**, 14410–14422 (2020). [arXiv:2006.10350](#) [cs.LG]
41. R.T. d’Agnolo, G. Grosso, M. Pierini, A. Wulzer, M. Zanetti, Learning new physics from an imperfect machine. Eur. Phys. J. C **82**(3), 275 (2022). [arXiv:2111.13633](#) [hep-ph]
42. C. Elkan, The foundations of cost-sensitive learning. In *International Joint Conference on Artificial Intelligence*, vol. 17, pp. 973–978 (Lawrence Erlbaum Associates Ltd, 2001)
43. S.S. Wilks, The large-sample distribution of the likelihood ratio for testing composite hypotheses. Ann. Math. Stat. **9**(1), 60–62 (1938)
44. A. Wald, Tests of statistical hypotheses concerning several parameters when the number of observations is large. Trans. Am. Math. Soc. **54**(3), 426–482 (1943)
45. G. Cowan, K. Cranmer, E. Gross, O. Vitells, Asymptotic formulae for likelihood-based tests of new physics. Eur. Phys. J. C **71**, 1554 (2011). [arXiv:1007.1727](#) [physics.data-an] [Erratum: Eur. Phys. J. C **73**, 2501 (2013)]
46. T. Hastie, R. Tibshirani, J.H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics (Springer, Berlin, 2009)
47. C.A. Micchelli, X. Yuesheng, H. Zhang, Universal kernels. J. Mach. Learn. Res. **7**(95), 2651–2667 (2006)
48. A. Christmann, I. Steinwart, *Support Vector Machines* (Springer, Berlin, 2008)
49. A. Rudi, L. Carratino, L. Rosasco, Falcon: an optimal large scale kernel method. Adv. Neural Inf. Process. Syst., 30 (2017). [arXiv:1705.10958](#) [stat.ML]
50. Y. Sun, A. Gilbert, A. Tewari, But how does it work in theory? Linear SVM with random features (2018). [arXiv:1809.04481](#)
51. A. Rudi, L. Rosasco, Generalization properties of learning with random features. Adv. Neural Inf. Process. Syst., 30 (2017). [arXiv:1602.04474](#) [stat.ML]
52. F. Bach, Sharp analysis of low-rank kernel matrix approximations. In *Conference on Learning Theory*, pp. 185–209 (PMLR, 2013). [arXiv:1208.2015](#) [cs.LG]
53. A. Rudi, R. Camoriano, L. Rosasco, Less is more: Nyström computational regularization. Adv. Neural Inf. Process. Syst., 28 (2015). [arXiv:1507.04717](#) [stat.ML]
54. D. Calandriello, L. Rosasco, Statistical and computational trade-offs in kernel k-means. Adv. Neural Inf. Process. Syst., 31 (2018). [arXiv:1908.10284](#) [stat.ML]
55. Z. Li, J.-F. Ton, D. Ogljic, D. Sejdinovic, Towards a unified analysis of random Fourier features. In: *International Conference on Machine Learning*, pp. 3905–3914 (PMLR, 2019). [arXiv:1806.09178](#) [stat.ML]
56. S. Shalev-Shwartz, S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, Cambridge, 2014)
57. U. Marteau-Ferey, F. Bach, A. Rudi, Globally convergent newton methods for ill-conditioned generalized self-concordant losses. Adv. Neural Inf. Process. Syst., 32 (2019). [arXiv:1907.01771](#) [math.OC]
58. P. Baldi, P. Sadowski, D. Whiteson, Searching for exotic particles in high-energy physics with deep learning. Nat. Commun. **5**, 4308 (2014). [arXiv:1402.4735](#) [hep-ph]
59. U. Marteau-Ferey, D. Ostrovskii, F. Bach, A. Rudi, Beyond least-squares: fast rates for regularized empirical risk minimization through self-concordance. In *Conference on Learning Theory*, pp. 2294–2340 (PMLR, 2019)