Uncertainty Aware Learning for High Energy Physics With A Cautionary Tale

Aishik Ghosh
Department of Physics and Astronomy
University of California, Irvine.
Physics Division
Lawrence Berkeley National Laboratory, Berkeley.
aishikghosh@lbl.gov

Benjamin Nachman
Physics Division
Lawrence Berkeley National Laboratory, Berkeley.
Berkeley Institute for Data Science
University of California, Berkeley.
bpnachman@lbl.gov

Daniel Whiteson
Department of Physics and Astronomy
University of California, Irvine.
daniel@uci.edu

Abstract

Machine learning tools provide a significant improvement in sensitivity over traditional analyses by exploiting subtle patterns in high-dimensional feature spaces. These subtle patterns may not be well-modeled by the simulations used for training machine learning methods, resulting in an enhanced sensitivity to systematic uncertainties. Contrary to the traditional wisdom of constructing an analysis strategy that is invariant to systematic uncertainties, we study the use of a classifier that is fully aware of uncertainties and their corresponding nuisance parameters. We show on two datasets that this dependence can actually enhance the sensitivity to parameters of interest compared to baseline approaches. Finally, we provide a cautionary example for situations where uncertainty mitigating techniques may serve only to hide the true uncertainties.

1 Introduction

The usefulness of physical measurements is tied to the magnitude and reliability of their estimated uncertainties. The most troublesome, systematic uncertainties, are often modeled as the dependence of a parameter of interest on other degrees of freedom, *nuisance parameters*.

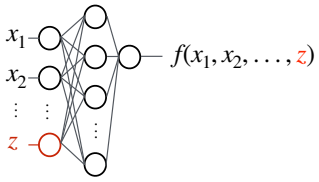


Figure 1: Uncertainty-aware architecture.

In high energy physics, machine learning models are typically trained on synthetic datasets generated with assumed values of the nuisance parameters. We will refer to this as the *baseline* approach. Several approaches have been considered to incorporate uncertainties into the training. Data augmentation trains a model on a concoction of synthetic data with different values of the nuisance parameters. Another possibility is to train a model to explicitly be insensitive to nuisance parameters [1–15], such as with adversarial training [1–4]. Maximizing overall sensitivity requires a compromise between the level of independence to nuisance parameters and the classification power. These three approaches will serve as important baselines in this paper.

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

We advocate for the opposite of decorrelation. Classifiers are constructed to be explicitly dependent on nuisance parameters. As nuisance parameters are profiled, the classifier will change and the best classifier will be used for each value of the nuisance parameter. Parameterized classifiers have been studied in the context of parameters or features of interest [16, 17], and full dependence on nuisance parameters for inference has been advocated in Ref. [18–22].

In this paper, we provide specific examples of profiled classifiers and show explicitly that parameterized classifiers can enhance analysis sensitivity over strategies that render networks insensitive to nuisance parameters. We focus on only the construction of classifiers as useful statistics for downstream analysis and not on full likelihood (ratio) estimation. In this way, our uncertainty-aware classifier approach [23] is a straightforward extension of existing analyses performed at the Large Hadron Collider (LHC) and elsewhere, and therefore may result in immediate improvements in sensitivity. In addition, this prescription allows for easy post-hoc histogram-based diagnostics. These may include quantification of the impact of additional sources of systematic uncertainties that are not used for training, and checks for whether the measurement over-constrains the nuisance parameter.

While we focus on the profiling aspect of uncertainty awareness, there is a complementary line of research on the use of inference-aware loss functions [24–30] and Bayesian neural networks for estimating uncertainties [31–34]. We leave the combination of these methods with our uncertainty-aware approach to future work. Additional information about the interplay between uncertainties and machine learning can be found in recent reviews [22, 35].

All the neural networks discussed in this paper were trained using KERAS [36] with a TENSORFLOW [37] backend on a single NVIDIA GEFORCE GTX GPU. Further implementational details are available with the code at https://github.com/hep-lbdl/systaware.

2 Uncertainty-Aware Classifier

The uncertainty-aware network is trained with the true value of the nuisance parameter z as an input to the network in additional to the observables x, see Fig. 1. Trained with a Binary Cross-Entropy loss, the network approximates the score,

$$s(x,z) = \frac{p(x|Z=z,S)}{p(x|Z=z,S)+p(x|Z=z,B)}. \tag{1}$$

where $p(\cdot)$ denotes a probability density, S represents the signal class and B represents the background class. Note that Eq. 1 depends on z, in contrast to the standard search paradigm in which the analysis observables are fixed and the sensitivity to z is evaluated post-hoc.

3 Evaluation Methodology

To evaluate the power of various approaches, we apply them to a common use case, fitting a signal hypothesis in the presence of background, where both signal and background depend on nuisance parameters. For ease of calculations we perform a binned likelihood fit.

For each strategy, template histograms of the classifier score are constructed from simulated signal and background events for several values of the nuisance parameter z. These templates are the basis of the binned likelihood calculation $\mathcal{L}(\mu, z|\{x_i\})$ over the parameters μ, z, where $\{x_i\}$ is the full observed dataset. The likelihood is a product of a Poisson term for each histogram bin and a Gaussian constraint on the nuisance parameter. The Gaussian constraint can readily be replaced with any other prior or a Poisson term from an auxiliary measurement if z is directly constrained with control region data. The Negative Log-Likelihood (NLL) is (up to an irrelevant constant),

$$\begin{aligned} &-\log \mathcal{L}(\mu, z|\{x_i\}) \\ &\quad = -\sum_{j=1}^{n_{\text{bins}}} \left[N_j \cdot \log\left(\mu s_j + b_j\right) - \mu s_j - b_j - \log(\Gamma(N_i)) \right] \\ &\qquad + \left(\frac{z-z_0}{\sqrt{2}\sigma_z}\right)^2, \end{aligned} \tag{2}$$

where s_j, b_j are the expected number of signal and background events in bin j, respectively, and N_j is the number of events observed in data for that bin. The Γ function is the generalized factorial function which can handle decimal values in the simulated test dataset. Although the $\log(\Gamma(N_i))$ term is usually irrelevant, it not a constant while using an uncertainty-aware network and cannot be ignored.

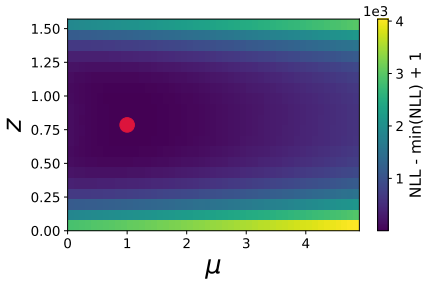


Figure 2: Example NLL as a function of μ and z for baseline classifier.

The fitted value of μ is obtained by minimizing Eq. 2. Since the measurement of the nuisance parameter is not the final objective, it is in fact the profile likelihood, $\mathcal{L}_p(\mu) = \max_z \mathcal{L}(\mu, z)$, that is the most relevant metric for determining the relative power of the various approaches. As a diagnostic, the parameter of interest may be profiled over instead to check if the measurement over-constrains the nuisance parameter.

4 Gaussian Example

We begin with a Gaussian example with a two-dimensional feature space and a single nuisance parameter. Signal events are drawn from Gaussian distributions in the two features, with means at $\cos(z)$ and $\sin(z)$, respectively; the width of each is set to 0.7. Background events are generated in same fashion, but with means for the two features at $-\cos(z)$ and $-\sin(z)$ respectively.

A set of 4.2×10^7 events are generated at 21 values of z equally spaced between 0 and $\pi/2$ for the signal and background. $z = \frac{\pi}{2}$ is treated as the nominal value. Ten bins are used to construct the template and observed histograms. The parameter of interest is the signal strength μ with a true value of 1.

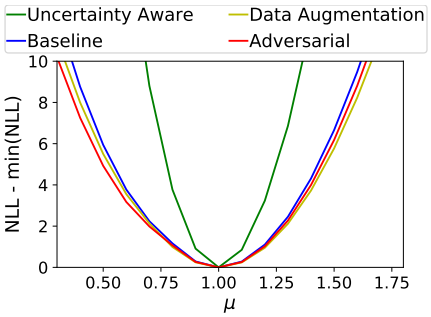


Figure 3: The profile likelihood $\max_z \mathcal{L}(\mu, z)$ as a function of the parameter of interest, μ for various classifiers. Narrower curves indicate more precise measurements having accounted for systematic and statistical uncertainties.

Results: For some observed data, the NLL (Eq. 2) is calculated as a function of the parameter of interest μ and the nuisance parameter z for each approach. An example of this two dimensional NLL distribution is shown in Fig. 2, which was computed by comparing templates from the baseline classifier to the "observed data" generated at $z = \frac{\pi}{4}$.

The profile likelihood for each method is shown in Fig. 3 for data generated with $z = \frac{\pi}{2}$ .We see that the uncertainty-aware classifier provides the best performance.

5 Realistic Example

The study is also performed on datasets [38] produced [39] for the HiggsML Kaggle challenge [40] and later enhanced [41] as benchmark datasets for uncertainty quantification [42, 43]. The nuisance parameter is related to the uncertainty of the measured τ lepton transverse energy.

Results: The performance of the four approaches are compared on data generated at the nominal value of $z = 1$ as well as shifted values of $z = 0.8$ and $z = 1.1$. In addition to these approaches, classifiers trained on data from the shifted values of z are added to the comparisons. The true value of μ was set to 1 throughout. Thirty bins are used to construct the template and observed histograms.

Figure. 4 shows that the uncertainty-aware classifier maintains ideal performance for all values of z while all other approaches are at best able to match the performance only for a single value of z.

6 Theory Uncertainties

While incorporating uncertainties in the training is desirable, caution must be taken to include only nuisance parameters with a statistical origin. For example, uncertainties due to fragmentation modelling are often estimated using the difference of two models (PYTHIA and HERWIG), and a full theoretical uncertainty decomposition is unknown. An example [44] of two classifiers trained to identify W boson jets (signal) from quark and gluon jets (background) is shown in Fig. 5, where adversarial training is used to reduce the difference in performance between PYTHIA and HERWIG. By sacrificing separation power, this difference is successfully reduced when compared to the large gap in performances for the nominal classifier. However, the difference in performance to data generated with a third model (SHERPA) remains large in both classifiers, indicating that the the true uncertainty will be underestimated in the case of the adversarial classifier if a third independent sample is unavailable.

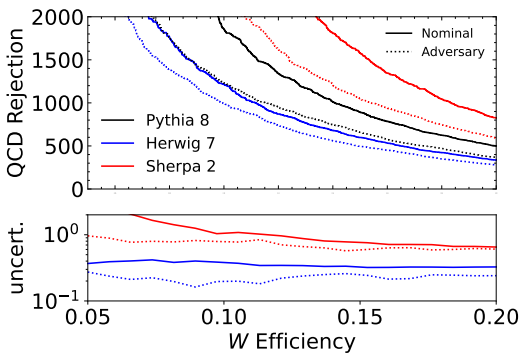
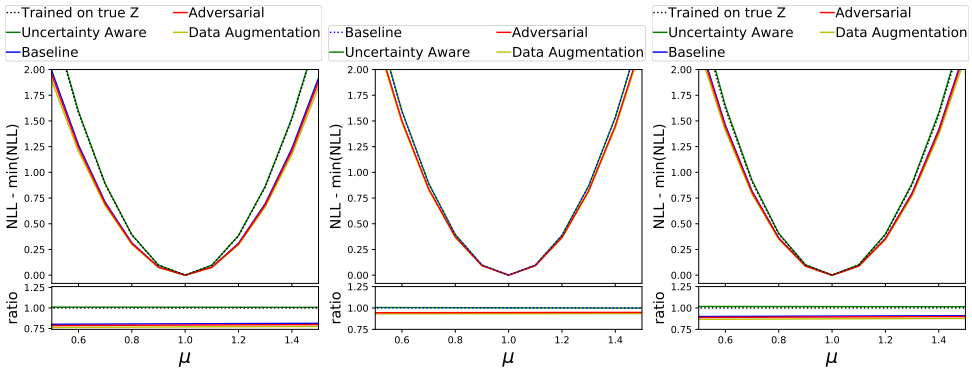


Figure 5: Performance of classifiers on data generated from PYTHIA, HERWIG, and SHERPA. Solid lines correspond to the nominal classifier trained with PYTHIA while dotted lines correspond to the adversarial setup using PYTHIA and HERWIG. The bottom panel shows the relative absolute difference with respect to PYTHIA (nominal or adversarial, as appropriate). Note that the lower panel has a logarithmic vertical axis.

7 Conclusions

In this paper, we have advocated for uncertainty-aware classifiers where the dependence on nuisance parameter is maximized during training by exploiting parameterized classifiers [16, 17]. Using a Gaussian example and a realistic $H \to \tau\tau$ example, we have shown that the uncertainty-aware approach outperforms alternative methods that either are unaware of uncertainties or try to reduce the dependence on them during training[1]. Our approach is successful because it provides the most effective classifier for all values of the nuisance parameter. This is useful when uncertainties are evaluated and when the nuisance parameter is profiled. It should be straightforward to apply this approach to multiple nuisance parameters although it was demonstrated on a single nuisance parameter in this paper.

[1] Further details can be found in Ref. [23]



(a) Syst Down Data ($z_T = 0.8$) (b) Nominal Data ($z_T = 1.0$) (c) Syst Up Data ($z_T = 1.1$)

Figure 4: Physics Dataset: Profiled NLL curves for all four classifiers evaluated three values of z where the true value of μ is 1. Narrower curves indicate more precise measurements having accounted for systematic and statistical uncertainties.

We also recommend that caution must be taken in applying uncertainty mitigating solutions along with an explicit example of the possible danger. We show a case where decorrelating the dependence of a classifier to a theoretical uncertainty only serves to hide the size of the true uncertainty[2]. While demonstrated for decorrelation, this cautionary tale remains relevant for other uncertainty or inference aware machine learning approaches [18–22, 24–30]. Ultimately, the decision to use the additional feature or not depends on how the test statistic will be used in the analysis.

The uncertainty-aware technique proposed here is a straightforward extension of existing LHC analyses and will require minimal changes or computational overhead. The biggest improvements are expected in analyses limited by experimental systematics. A large number of analyses will fall into this category at the High-Luminosity LHC and beyond.

Acknowledgements

AG thanks David Rousseau and Victor Estrade for fruitful discussions over several years, and for sharing code to generate the physics dataset, and also thanks Glen Cowan for fruitful discussions.

We are grateful to Yi-Lun Chung for producing the fragmentation variation samples. We thank Tommaso Dorigo, Victor Estrade, David Rousseau, Kingman Cheung, Shih-Chieh Hsu, Tilman Plehn, Michael Spannowsky and Kyle Cranmer for providing helpful comments on the manuscript. BN is supported by the U.S. Department of Energy (DOE), Office of Science under contract DE-AC02-05CH11231. AG and DW are supported by the U.S. Department of Energy (DOE), Office of Science under Grant No. DE-SC0009920.

References

[1] A. Blance, M. Spannowsky, and P. Waite, "Adversarially-trained autoencoders for robust unsupervised new physics searches," *JHEP*, vol. 10, p. 047, 2019. DOI: 10.1007/JHEP10(2019)047. arXiv: 1905.10384 [hep-ph].

[2] C. Englert, P. Galler, P. Harris, and M. Spannowsky, "Machine Learning Uncertainties with Adversarial Neural Networks," *Eur. Phys. J.*, vol. C79, no. 1, p. 4, 2019. DOI: 10.1140/epjc/s10052-018-6511-8. arXiv: 1807.08763 [hep-ph].

[3] G. Louppe, M. Kagan, and K. Cranmer, "Learning to Pivot with Adversarial Networks," 2016. arXiv: 1611.01046 [stat.ME].

[4] C. Shimmin, P. Sadowski, P. Baldi, E. Weik, D. Whiteson, E. Goul, and A. Søgaard, "Decorrelated Jet Substructure Tagging using Adversarial Neural Networks," 2017. DOI: 10.1103/PhysRevD.96.074034. arXiv: 1703.03507 [hep-ex].

[5] J. Stevens and M. Williams, "uBoost: A boosting method for producing uniform selection efficiencies from multivariate classifiers," *JINST*, vol. 8, P12013, 2013. DOI: 10.1088/1748-0221/8/12/P12013. arXiv: 1305.7248 [nucl-ex].

[6] L. Bradshaw, R. K. Mishra, A. Mitridate, and B. Ostdiek, "Mass Agnostic Jet Taggers," 2019. DOI: 10.21468/SciPostPhys.8.1.011. arXiv: 1908.08959 [hep-ph].

[7] "Performance of mass-decorrelated jet substructure observables for hadronic two-body decay tagging in ATLAS," *ATL-PHYS-PUB-2018-014*, 2018. [Online]. Available: http://cds.cern.ch/record/2630973.

[8] G. Kasieczka and D. Shih, "DisCo Fever: Robust Networks Through Distance Correlation," 2020. DOI: 10.1103/PhysRevLett.125.122001. arXiv: 2001.05310 [hep-ph].

[9] S. Wunsch, S. Jörger, R. Wolf, and G. Quast, "Reducing the dependence of the neural network function to systematic uncertainties in the input space," *Comput. Softw. Big Sci.*, vol. 4, no. 1, p. 5, 2020. DOI: 10.1007/s41781-020-00037-9. arXiv: 1907.11674 [physics.data-an].

[10] A. Rogozhnikov, A. Bukva, V. V. Gligorov, A. Ustyuzhanin, and M. Williams, "New approaches for boosting to uniformity," *JINST*, vol. 10, no. 03, T03002, 2015. DOI: 10.1088/1748-0221/10/03/T03002. arXiv: 1410.4140 [hep-ex].

[2] Further details can be found in Ref. [44]

[11] C. Collaboration, "A deep neural network to search for new long-lived particles decaying to jets," *Machine Learning: Science and Technology*, 2020. DOI: 10.1088/2632-2153/ab9023. eprint: 1912.12238.

[12] J. M. Clavijo, P. Glaysher, and J. M. Katzy, "Adversarial domain adaptation to reduce sample bias of a high energy physics classifier," 2020. arXiv: 2005.00568 [stat.ML].

[13] G. Kasieczka, B. Nachman, M. D. Schwartz, and D. Shih, "ABCDisCo: Automating the ABCD Method with Machine Learning," Jul. 2020. DOI: 10.1103/PhysRevD.103.035021. arXiv: 2007.14400 [hep-ph].

[14] O. Kitouni, B. Nachman, C. Weisser, and M. Williams, "Enhancing searches for resonances with machine learning and moment decomposition," Oct. 2020. arXiv: 2010.09745 [hep-ph].

[15] V. Estrade, C. Germain, I. Guyon, and D. Rousseau, "Systematic aware learning - A case study in High Energy Physics," *EPJ Web Conf.*, vol. 214, A. Forti, L. Betev, M. Litmaath, O. Smirnova, and P. Hristov, Eds., p. 06 024, 2019. DOI: 10.1051/epjconf/201921406024.

[16] K. Cranmer, J. Pavez, and G. Louppe, "Approximating Likelihood Ratios with Calibrated Discriminative Classifiers," Jun. 2015. arXiv: 1506.02169 [stat.AP].

[17] P. Baldi, K. Cranmer, T. Faucett, P. Sadowski, and D. Whiteson, "Parameterized neural networks for high-energy physics," *Eur. Phys. J.*, vol. C76, no. 5, p. 235, 2016. DOI: 10.1140/epjc/s10052-016-4099-4. arXiv: 1601.07913 [hep-ex].

[18] J. Brehmer, F. Kling, I. Espejo, and K. Cranmer, "MadMiner: Machine learning-based inference for particle physics," *Comput. Softw. Big Sci.*, vol. 4, no. 1, p. 3, 2020. DOI: 10.1007/s41781-020-0035-2. arXiv: 1907.10621 [hep-ph].

[19] J. Brehmer, G. Louppe, J. Pavez, and K. Cranmer, "Mining gold from implicit models to improve likelihood-free inference," *Proc. Nat. Acad. Sci.*, p. 201 915 980, 2020. DOI: 10.1073/pnas.1915980117. arXiv: 1805.12244 [stat.ML].

[20] J. Brehmer, K. Cranmer, G. Louppe, and J. Pavez, "Constraining Effective Field Theories with Machine Learning," 2018. DOI: 10.1103/PhysRevLett.121.111801. arXiv: 1805.00013 [hep-ph].

[21] ——, "A Guide to Constraining Effective Field Theories with Machine Learning," 2018. DOI: 10.1103/PhysRevD.98.052004. arXiv: 1805.00020 [hep-ph].

[22] B. Nachman, "A guide for deploying Deep Learning in LHC searches: How to achieve optimality and account for uncertainty," Sep. 2019. DOI: 10.21468/SciPostPhys.8.6.090. arXiv: 1909.03081 [hep-ph].

[23] A. Ghosh, B. Nachman, and D. Whiteson, "Uncertainty-aware machine learning for high energy physics," *Phys. Rev. D*, vol. 104, p. 056 026, 5 Sep. 2021. DOI: 10.1103/PhysRevD.104.056026. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevD.104.056026.

[24] S. Wunsch, S. Jörger, R. Wolf, and G. Quast, "Optimal statistical inference in the presence of systematic uncertainties using neural network optimization based on binned Poisson likelihoods with nuisance parameters," *Comput. Softw. Big Sci.*, vol. 5, no. 1, p. 4, 2021. DOI: 10.1007/s41781-020-00049-5. arXiv: 2003.07186 [physics.data-an].

[25] A. Elwood, D. Krücker, and M. Shchedrolosiev, "Direct optimization of the discovery significance in machine learning for new physics searches in particle colliders," *J. Phys. Conf. Ser.*, vol. 1525, p. 012 110, 2020. DOI: 10.1088/1742-6596/1525/1/012110.

[26] L.-G. Xia, "QBDT, a new boosting decision tree method with systematical uncertainties into training for High Energy Physics," *Nucl. Instrum. Meth.*, vol. A930, pp. 15–26, 2019. DOI: 10.1016/j.nima.2019.03.088. arXiv: 1810.08387 [physics.data-an].

[27] P. De Castro and T. Dorigo, "INFERNO: Inference-Aware Neural Optimisation," *Comput. Phys. Commun.*, vol. 244, pp. 170–179, 2019. DOI: 10.1016/j.cpc.2019.06.007. arXiv: 1806.04743 [stat.ML].

[28] T. Charnock, G. Lavaux, and B. D. Wandelt, "Automatic physical inference with information maximizing neural networks," *Physical Review D*, vol. 97, no. 8, Apr. 2018, ISSN: 2470-0029. DOI: 10.1103/physrevd.97.083004. [Online]. Available: http://dx.doi.org/10.1103/PhysRevD.97.083004.

[29] J. Alsing and B. Wandelt, "Nuisance hardened data compression for fast likelihood-free inference," *Mon. Not. Roy. Astron. Soc.*, vol. 488, no. 4, pp. 5093–5103, 2019. DOI: 10.1093/mnras/stz1900. arXiv: 1903.01473 [astro-ph.CO].

[30] L. Heinrich and N. Simpson, *Pyhf/neos: Initial zenodo release*, version 0.0.2, Mar. 2020. DOI: 10.5281/zenodo.3697981. [Online]. Available: https://doi.org/10.5281/zenodo.3697981.

[31] G. Kasieczka, M. Luchmann, F. Otterpohl, and T. Plehn, "Per-Object Systematics using Deep-Learned Calibration," Mar. 2020. DOI: 10.21468/SciPostPhys.9.6.089. arXiv: 2003.11099 [hep-ph].

[32] S. Bollweg, M. Haußmann, G. Kasieczka, M. Luchmann, T. Plehn, and J. Thompson, "Deep-Learning Jets with Uncertainties and More," *SciPost Phys.*, vol. 8, no. 1, p. 006, 2020. DOI: 10.21468/SciPostPhys.8.1.006. arXiv: 1904.10004 [hep-ph].

[33] J. Y. Araz and M. Spannowsky, "Combine and Conquer: Event Reconstruction with Bayesian Ensemble Neural Networks," *JHEP*, vol. 04, p. 296, 2021. DOI: 10.1007/JHEP04(2021)296. arXiv: 2102.01078 [hep-ph].

[34] M. Bellagente, M. Haußmann, M. Luchmann, and T. Plehn, "Understanding Event-Generation Networks via Uncertainties," Apr. 2021. arXiv: 2104.04543 [hep-ph].

[35] T. Dorigo and P. de Castro, "Dealing with Nuisance Parameters using Machine Learning in High Energy Physics: a Review," Jul. 2020. arXiv: 2007.09121 [stat.ML].

[36] F. Chollet *et al.* (2015). "Keras," [Online]. Available: https://github.com/fchollet/keras.

[37] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Y. Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015. [Online]. Available: http://tensorflow.org/.

[38] ATLAS Collaboration, *Dataset from the ATLAS Higgs Boson Machine Learning Challenge 2014. CERN Open Data Portal.* DOI: 10.7483/OPENDATA.ATLAS.ZBP2.M5T8. [Online]. Available: http://opendata.cern.ch/record/328.

[39] G. Aad *et al.*, "Evidence for the Higgs-boson Yukawa coupling to tau leptons with the ATLAS detector," *JHEP*, vol. 04, p. 117, 2015. DOI: 10.1007/JHEP04(2015)117. arXiv: 1501.04943 [hep-ex].

[40] C. Adam-Bourdarios, G. Cowan, C. Germain, I. Guyon, B. Kégl, and D. Rousseau, "The Higgs boson machine learning challenge," in *Proceedings of the NIPS 2014 Workshop on High-energy Physics and Machine Learning*, G. Cowan, C. Germain, I. Guyon, B. Kégl, and D. Rousseau, Eds., ser. Proceedings of Machine Learning Research, vol. 42, Montreal, Canada: PMLR, 13 Dec 2015, pp. 19–55. [Online]. Available: http://proceedings.mlr.press/v42/cowa14.html.

[41] V. Estrade, *Victor-estrade/datawarehouse: First release*, version v1.0.0, Dec. 2018. DOI: 10.5281/zenodo.1887847. [Online]. Available: https://doi.org/10.5281/zenodo.1887847.

[42] V. Estrade, C. Germain, I. Guyon, and D. Rousseau, "Adversarial learning to eliminate systematic errors: a case study in High Energy Physics," 2017. [Online]. Available: %7Bhttps://dl4physicalsciences.github.io/files/nips_dlps_2017_1.pdf%7D.

[43] Estrade, Victor, Germain, Cécile, Guyon, Isabelle, and Rousseau, David, "Systematic aware learning - a case study in high energy physics," *EPJ Web Conf.*, vol. 214, p. 06 024, 2019. DOI: 10.1051/epjconf/201921406024. [Online]. Available: https://doi.org/10.1051/epjconf/201921406024.

[44] A. Ghosh and B. Nachman, "A Cautionary Tale of Decorrelating Theory Uncertainties," Sep. 2021. arXiv: 2109.08159 [hep-ph].

For all authors...

1. Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
2. Did you describe the limitations of your work? [Yes] Section 6 as well as conclusion

3. Did you discuss any potential negative societal impacts of your work? [No]
4. Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

If you are including theoretical results...

1. Did you state the full set of assumptions of all theoretical results? [N/A]
2. Did you include complete proofs of all theoretical results? [N/A]

If you ran experiments...

1. Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
2. Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [No] The linked code has all the details.
3. Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
4. Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]

If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

1. If your work uses existing assets, did you cite the creators? [Yes]
2. Did you mention the license of the assets? [No] Information available at cited link
3. Did you include any new assets either in the supplemental material or as a URL? [Yes]
4. Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] It was all made publicly available
5. Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

If you used crowdsourcing or conducted research with human subjects...

1. Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
2. Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
3. Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]