Supervised Machine Learning in the Search for the Higgs Boson.

Student Number 23220004
Machine Learning Department
Northeastern University London
London, England

Abstract—**This paper represents a two-part project which sets out to explore the application of supervised machine learning in the search for the Higgs boson. A review of five previous research pieces showcase the most popular and successful machine learning methods used when searching for Higgs bosons from 2014 to 2022, with the general consensus being that Deep Neural Networks provide the most accuracy at a high computational cost, and Gradient Boosted Trees offer a slightly less accurate but nevertheless effective compromise at a lower computational cost [1, 2, 3, 4, 5]. Then, XGBoost and AdaBoost models are implemented using Scikit-learn and evaluated on simulated collision data. XGBoost demonstrates higher precision and AMS scores, whereas AdaBoost scores higher in accuracy and recall.**

I. Introduction

The Higgs boson is a particle responsible for giving mass to matter [6]. To search for Higgs bosons, physicists use large particle colliders (such as the Large Hadron Collider) to smash particles together. However, the particles decay too quickly to directly observe a Boson, so information about the decaying particles is analysed to decide whether the event may be classified as signal (the decay of exotic particles) or background (the decay of other particles) [7]. Deciding whether an event is a signal or background is a binary classification problem, and so there have been many attempts to use machine learning to classify these events.

There does not yet exist sufficient real particle collision data on which to train or test machine learning algorithms, so all publicly documented attempts make use of simulated event data [8]. In the next section, five of the most relevant research papers using such data are reviewed.

II. Literature Review

The following pieces of research were chosen based on their relevance to supervised machine learning in high energy particle physics. They all focus on algorithms which can be implemented for supervised machine learning, though many were used for unsupervised learning problems. The research covered by this section begins in 2014 with the Higgs boson machine learning challenge which was the inspiration for this paper.

A. The Higgs boson machine learning challenge

The Higgs boson machine learning challenge ran in 2014, and was created by ATLAS scientists and hosted by Kaggle. Using data provided by the ATLAS experiment at CERN [9] (a labelled version of which was later made publicly available), participants were to use machine learning to classify each event in the dataset as signal or background. The purpose of the challenge was to try a "crowd-based" approach to find advanced and successful classifiers for the purpose of Higgs boson discovery, as well as foster collaboration between high-energy physicists and data scientists. Though the challenge did not appear to have any restrictions, it did lay out an unusual objective for participants: to optimise for the Approximate Median Significance (AMS), which is described as a "test statistic similar in spirit to the false discovery rate" [1].

The winning classifier was a computationally expensive but high-performing ensemble of moderately deep neural networks created by Gábor Melis. The second winner, Tim Salimans, used a solution based on the Regularised Greedy Forest algorithm and linear regression. Finally, a special award was given to Crowwork for participating with their creation XGBoost, an implementation of boosted decision trees which was also successfully employed by many other participants [1].

The competition was considered a success due to the high participation rate, and the successful implementation of machine learning algorithms to identify signal events. This success was partially attributed to the popularity of the subject at the time (the search for the Higgs boson) as well as the clever design of the challenge, which was open to participants with no previous physics knowledge and therefore provided a simplification of the problem setting [1].

B. Searching for Exotic Particles in High-Energy Physics with Deep Learning

The second piece of literature from 2014 comes from the publishers of the UCI dataset of simulated collision events [10] which, alongside the Kaggle dataset, is the only other publicly available source of Higgs boson event data [2].

The study aimed to demonstrate the efficacy of deep learning methods versus the methods being employed at the time, by testing them on two benchmark classification tasks. The first was to distinguish between signal and background decay processes where Higgs bosons are produced using the UCI Higgs dataset. The other benchmark test used a different dataset and was concerned with the creation of different particles, so for the sake of brevity only the Higgs boson benchmark will be covered by this review.

A five-layer neural network with 300 hidden units in each layer was utilised and compared against shallow neural networks (NN) that had been trained with the same number of units as well as the same hyper-parameters. A Boosted Decision Tree (BDT) was also trained for additional comparison. The main comparison metric was Area Under ROC (AUC) [2].

The study was a success in that it demonstrated that the deep neural network (DNN) performed much better than the shallow NN and BDT. Of note was the way that the performance of the shallow NN and BDT varied significantly when trained on only high-level or low-level features, whereas the performance of the DNN did not. The researchers claimed that this suggests the DNN was independently discovering the insight and discriminating power of the high-level features [2].

C. Stacking Machine Learning Classifiers to Identify Higgs Bosons at the LHC

In this paper, Alves set out to compare the performance of stacked generalisation against both a DNN and Boosted Decision Trees (BDTs) in two different Higgs boson discovery problems [3]. The second of these classification problems is the one laid out by the Kaggle ML challenge and as such will be the focus of this review.

Alves motivated that one ML algorithm may fail to learn all the correlations among the kinematic distributions (statistics about movement) in the data. However, Alves hypothesised that stacking multiple classifiers may stand a better chance at capturing these correlations. To test this hypothesis, the top results of the Kaggle ML challenge that use XGBoost were compared against Alves' own work of stacking ML outputs over the top of XGBoost. The exact method Alves used was to have three level-0 classifiers: XGBoost, a single layer NN implemented with Keras, and a naive Bayes classifier. The level-1 generaliser used was a Logistic Regression classifier implemented with Scikit-learn. In both multivariate analysis (MVA) and cut-and-count analysis, it was found that stacking outperformed XGBoost alone, even with less careful tuning [3].

In this sense, the main aim of the paper was a success. However, it was also shown that stacking performed slightly worse than DNNs- though Alves posits that the high cost of DNNs in both time and computational resources still makes stacking a viable option in comparison [3].

D. Higgs Boson Discovery using Machine Learning Methods with Pyspark

The aim of this short paper was to compare the accuracy and AUC of four ML methods in classifying signal and background events, specifically using the Pyspark environment. The four ML methods used were Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), and Gradient Boosted Tree (GBT). The research used both the UCI and Kaggle datasets in a 70% train and 30% test split [4].

The research found that GBT outperformed the other models on both the Kaggle and UCI datasets, achieving an accuracy of 83% and 70% respectively. They noted that the accuracies of each ML method they used reflected those ML method's rankings in the official Kaggle Higgs boson machine learning challenge, therefore the Spark implementation of these models was a success [4].

E. Application of Machine Learning Algorithms for Searching BSM Higgs Bosons Decaying to a Pair of Bottom Quarks

This research aimed to study the efficacy of ML algorithms which have been commonly used to search for the Standard Model (SM) Higgs boson, in order to improve the sensitivity of the search for Higgs bosons that are classified as "Beyond the Standard Model" (BSM). The four models chosen were three tree-based models: DTs, RF, Adaptive Boosting; and one 4-layer NN. The models were implemented using Scikit-learn and Keras [5].

No specific dataset was explicitly mentioned- instead, papers were cited which describe the process of generating simulated data. It seems reasonable to assume that this research therefore made use of privately generated data. When preprocessing the data, to help save computational resources, the researchers measured the linear correlation of the high-level features and the low-level features that they were derived from and disregarded the ones with high correlation [5].

The study was successful in showing that ML algorithms can be used effectively in the search for BSM Higgs bosons. NN and Adaptive Boosting models performed the best at classifying the signal and background events, with Adaptive Boosting performing only slightly worse than the NN [5].

III. Methodology

For this problem, a top-down quantitative approach is taken by selecting two models which feature prominently in the literature: XGBoost and AdaBoost. Both classifiers used the same splits of data, with 90% of the data being used in model selection, and 10% left out as unseen test data for evaluation. Care is taken in the tuning of both classifiers so they could be evaluated and compared against each other as fairly as possible. The entire pipeline from processing to evaluation is executed in Python, using the scikit-learn library.

Evaluation used the standard metrics available in scikit-learn's classification report, but also focused on maximising Approximate Median Significance (AMS), the scoring metric used for the original Higgs boson machine learning challenge, calculated as:

$$\text{AMS}_c = \sqrt{2\left((s+b+b_{\text{reg}})\ln\left(1+\frac{s}{b+b_{\text{reg}}}\right)-s\right)},$$

Where s, b are the true positive and false positive rates respectively, and breg is a constant regularisation term set to 10 [1].

A. Data and Preprocessing

The dataset used in this test is the Kaggle Higgs boson machine learning challenge dataset, which was labelled and made public after the conclusion of the Kaggle challenge in 2014 [9]. The data comprises 818,238 events and 30 features (13 high-level and 17 low-level), and events are labelled as signal or background. Sample weights are included to help counteract the dataset's negative bias.

The Higgs boson Machine Learning Challenge dataset is a reasonably high-dimensional dataset containing some high-level features which have been calculated using its low-level features. These features are all on significantly different scales. The dataset also contains a few features which have high nullity, as seen in figure 2, and most of the features are also heavily skewed. There are also a high amount of outliers present in the data. Based on these observations, the following pipeline was established:

- Discarding high-level features with a high correlation (r greater than 0.99) with any low-level features, as these would add to the complexity of the model without necessarily adding meaning.
- Mean imputation using scikit-learn's SimpleImputer.
- Detecting and removing outliers using a z-score method. This was done modestly to avoid potentially losing meaningful stand-out values which might denote a signal event.
- Log transformation on the features for which all values are greater than 0, to reduce the skew of as many features as possible.
- Feature scaling (standardisation) using scikit-learn's SimpleScaler.

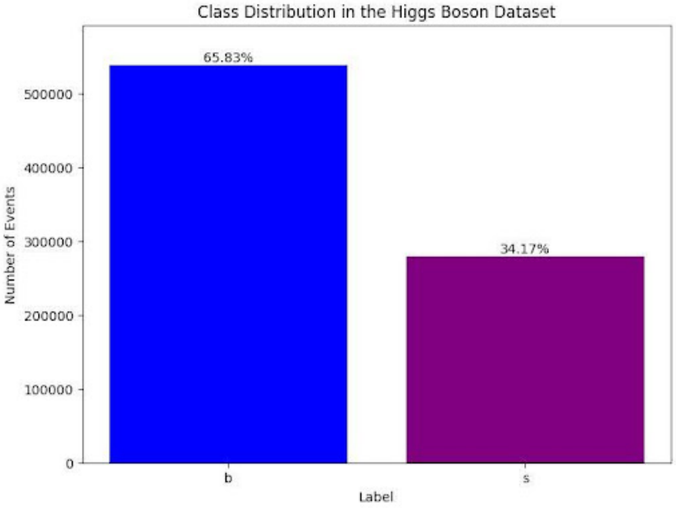


Fig. 1. Bias present in the Higgs dataset.

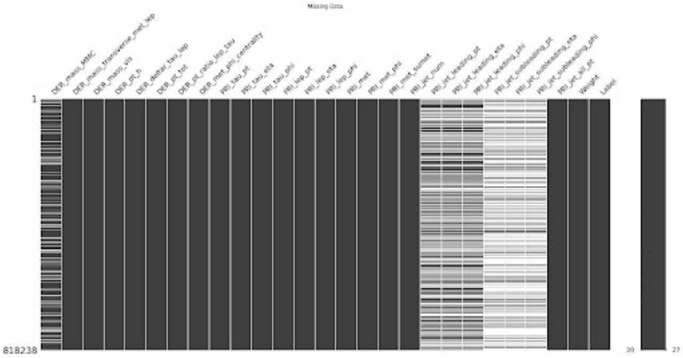


Fig. 2. Visualisation of missing data in the Higgs dataset.

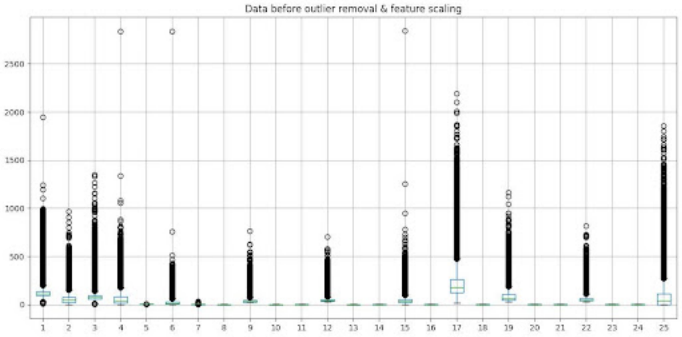


Fig. 3. Distribution and scale of each feature in the Higgs dataset before preprocessing.

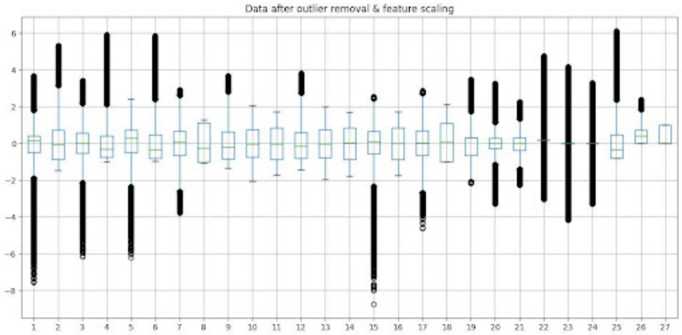


Fig. 4. Distribution and scale of each feature in the Higgs dataset after log transform, outlier removal, and standardisation.

B. The XGBoost Classifier

As the Gradient Boosted Tree (GBT) algorithm saw prolific success in the Kaggle ML challenge and in research thereafter, Scikit-learn's XGBoost was the first model to be selected. XGBoost is a GBT used for supervised learning problems- a tree ensemble algorithm which learns by using gradient descent to minimise a loss function, adding a new tree on each iteration [11]. GBTs have shown to be effective on imbalanced data such as the Kaggle dataset, and at making accurate predictions for rare events [12] such as Higgs boson production, and so is commonly used in high energy particle physics. Initial tuning was carried out using GridSearchCV on stratified 4-fold splits of the training data.

The main metric used during model selection was F1 score, as other metrics were found to cause exceptionally poor recall. At first, the parameters selected by GridSearchCV resulted in a badly underfitting model- however, it was noted that GridSearchCV was favouring parameters which increase the complexity of the model (such as maximum depth), though the limited parameter grid did not allow for much increase in these.

Based on this observation, further tuning was carried out by manually setting the maximum depth to be limitless and carrying out a 4-fold cross-validation using XGBoost's own cross-validation function. This added complexity allowed the model to perform better, and as seen in figure 5 the final model converges on the test set around 100 epochs with a test AUC approximately 0.07 behind the training AUC. The parameters of the final model were:

- learning_rate=0.5
- max_depth=0
- min_child_weight=1
- subsample=0.9
- colsampl_bytree=0.9

The final model was therefore trained and evaluated with 200 epochs and early stopping rounds set to 10, stopping the training if there was no increase in performance after 10 rounds.

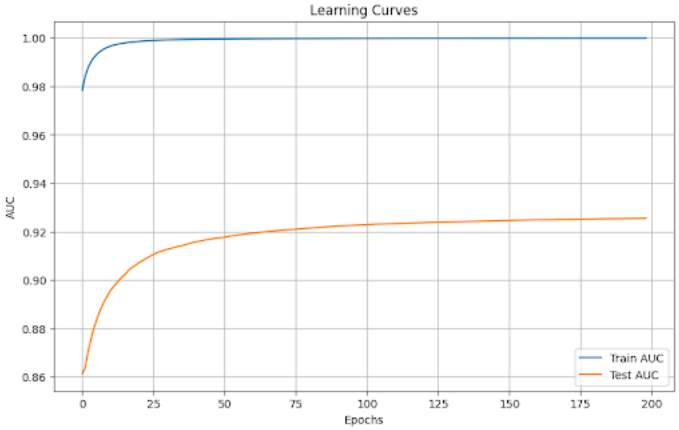


Fig. 5. XGBoost cross-validation learning curve.

C. The AdaBoost Classifier

The AdaBoost classifier is another form of Boosted Decision Tree (BDT) algorithm. It uses Adaptive Boosting, optimising the weights of each newly added weak learner on each iteration [13], and is the predecessor to XGBoost. AdaBoost was found to perform only slightly worse than NNs at the Higgs classification task [5], though its performance has not been directly compared to XGBoost yet, which is why it was selected as the second model for this study.

The AdaBoost classifier was tuned partially using GridSearchCV, though the computational cost of cross-validation was too high to make this an effective method of tuning. Instead, a trend was noticed by monitoring the GridSearchCV verbose output and the search was ended early. It was observed from this output that increasing the maximum depth of the base estimator trees increased performance, as did increasing the number of base estimator trees, however both came at a heavy cost to efficiency. A compromise was found by setting the number of trees to 100 and the maximum depth to 10, allowing for a model which trained in a matter of hours (compared to XGBoost's training time of a few minutes).

Since further cross-validation was impossible with the resources available, there is no learning curve data available for this model. The parameters of the final model were:

- n_estimators= 100
- learning_rate= 0.5
- (base estimator) max_depth= 10

IV. Results and Discussion

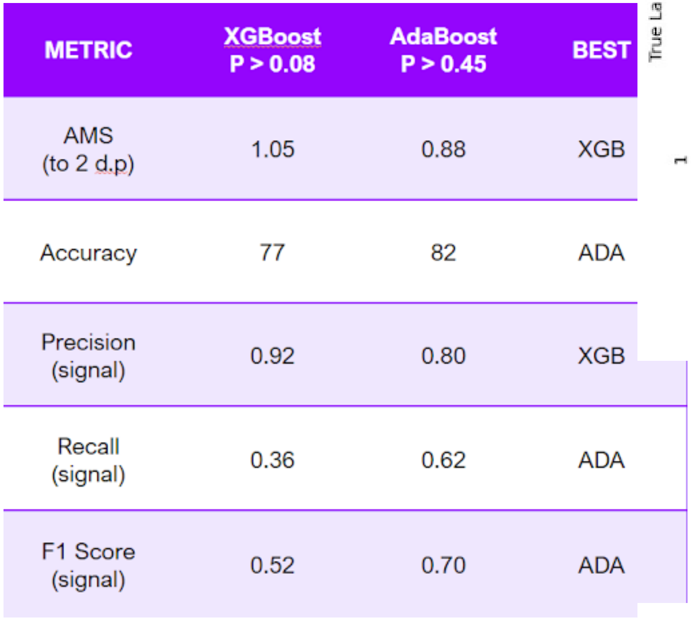
When evaluating both models' performance, a probability threshold was selected for each at which the AMS score was highest. The classification report was generated using predictions at these best thresholds for each model. Fig. 6 shows compares the performance metrics and AMS score for each model.

As shown by these scores and the confusion matrices shown by Fig. 7 and Fig. 8, XGBoost displayed higher precision which resulted in a higher AMS score, despite its low recall. However, AdaBoost outperformed XGBoost in every other metric. This means that the XGBoost model was good at minimising false positives, whilst the Adaboost model was better at minimising false negatives- both of these traits are considered important for Higgs discovery, though the AMS score prioritises the minimisation of false positives. For either of these to be developed into a more balanced and effective model for Higgs discovery, there are a few possible improvements to make.

In the case of AdaBoost, the model may benefit from more resources for hyperparameter tuning in order to complete a comprehensive GridSearchCV, as well as tuning for precision to improve the AMS score. More resources would also allow for a higher number of base estimators and a higher base estimator maximum depth, which both seemed to increase performance.

For XGBoost, again a more extensive GridSearchCV may be beneficial, and tuning for recall as this was the model's weakest area. The extensive GridSearchCV may help discover parameters which allow the model to improve recall without sacrificing AMS score.

Both models may improve with further data proces experimentation in terms of feature selection, feature scal and imputation methods. They may also improve with calculation of new high-level features, as demonstrated the “Cake” features created by team C.A.K.E as part of Higgs boson machine learning challenge [14], though would require specialist high-energy particle physics dor knowledge.



METRIC	XGBoost P > 0.08	AdaBoost P > 0.45	BEST
AMS (to 2 d.p)	1.05	0.88	XGB
Accuracy	77	82	ADA
Precision (signal)	0.92	0.80	XGB
Recall (signal)	0.36	0.62	ADA
F1 Score (signal)	0.52	0.70	ADA

Fig. 6. Performance metrics for both classifiers.

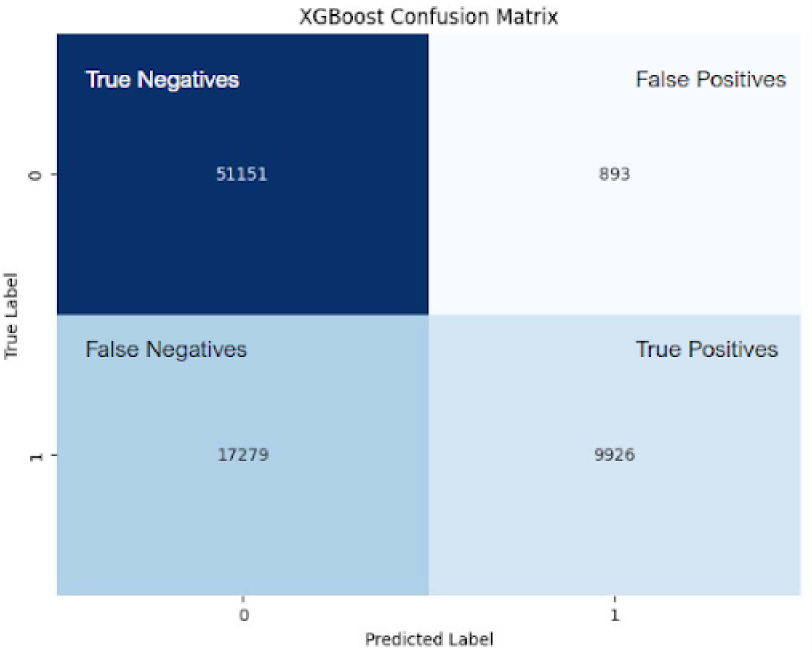


Fig. 7. Confusion matrix for the XGBoost classifier.

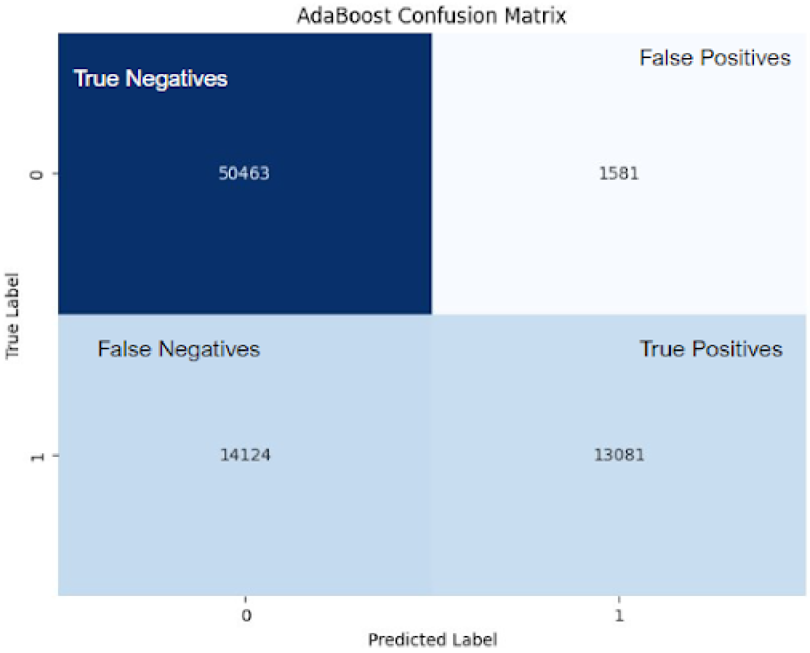


Fig. 8. Confusion matrix for the AdaBoost classifier.

V. Ethics

When using machine learning in bleeding-edge scientific pursuits such as Higgs boson discovery, it may speed up the search at the cost of interpretability. Using DNNs has so far proven to be the most accurate way of classifying signal events from background [1, 2, 3]; however, the complex nature of DNNs makes it impossible to know the exact process they are going through to classify the events. By having a completely transparent classification process key pieces of scientific knowledge about these events could be discovered, and by using DNNs it is possible that we are missing out on opportunities to advance our scientific understanding.

Secondly, if a major scientific breakthrough were to be made in the area of high-energy particle physics, it may be hard to pinpoint who is accountable for the discovery. It is important to identify a responsible party for scientific discoveries, not just so they can be rewarded for their ingenuity, but so they can be held accountable for the potential societal impacts their discovery has. How much of a discovery would we attribute to the person or team who designed and implemented the AI which made the discovery, versus the scientists who came up with the theory and designed the experiment?

Speeding up scientific advancement can also come at a societal cost, as policy struggles to keep up with innovation. At the speed that AI is advancing, it may begin making major scientific discoveries which have positive or potentially disastrous implications for humanity's future faster than we can roll out regulations or safety measures. Furthermore, when these major discoveries are made, public perception of them may be altered by the scientists using AI. With a general air of distrust towards AI technologies [15], it is possible that the public begin to distrust scientific establishments and perhaps science as a field in general.

However, the ethical implications for machine learning are not all bad. By combining high-energy particle physics with data and computer science, machine learning has the potential to improve interdisciplinary relations among scientists, as well as foster competition which can spark innovation and creativity.

VI. Conclusion and Future Work

This study explored the application of supervised machine learning in the search for the Higgs boson. A literature review showed that Deep Neural Networks (DNNs) and Gradient Boosted Trees (GBTs) have been the most successfully employed algorithms so far, with DNNs providing high accuracy albeit at a high computational cost, while GBTs offer a balance between accuracy and computational efficiency.

As part of this study, two GBT models were then tuned, trained, and evaluated on simulated particle collision data. Of the two selected GBT models, XGBoost demonstrated higher precision and AMS scores, whereas AdaBoost scored higher in accuracy and recall. The ethics of machine learning in high-energy particle physics were considered, touching on issues such as interpretability, trust, and accountability, as well as the possible societal impacts of making disruptive scientific discoveries using AI.

Future work may be needed to more effectively compare XGBoost and AdaBoost on the task of Higgs discovery, as this study lacked sufficient resources to carry out a more extensive hyperparameter tuning. Further experimentation with different data preprocessing techniques, such as feature selection, scaling, and imputation, could also improve model performance. High-energy particle physicists may also be able to calculate new features which generally improve the performance of machine learning in Higgs discovery.

References

Please number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first . . ."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

References

[1] C. Adam-Bourdarios, et al., "The Higgs boson machine learning challenge", JMLR: Workshop and Conference Proceedings 42:19-55, 2015.

[2] Baldi, P., Sadowski, P. and Whiteson, D, "Searching for exotic particles in high-energy physics with deep learning". Nat Commun 5, 4308. Available: https://doi.org/10.1038/ncomms5308, 2014.

[3] A. Alves. "Stacking machine learning classifiers to identify Higgs bosons at the LHC". JINST, 12. Available: https://iopscience.iop.org/article/10.1088/1748-0221/12/05/T05005, 2017.

[4] M. Azhari, A. Abarda, B. Ettaki, J. Zerouaoui and M. Dakkon, "Higgs Boson Discovery using Machine Learning Methods with Pyspark", Procedia Computer Science, 170, 1141-1146. Available: https://doi.org/10.1016/j.procs.2020.03.053, 2020.

[5] J. Waiwattana, C. Asawatangtrakuldee, P. Saksirimontri, V. Wachirapusitanand, and N. Pitakkultorn, "Application of Machine Learning Algorithms for Searching BSM Higgs Bosons Decaying to a Pair of Bottom Quarks", Trends Sci, 19, 5373, 2022.

[6] CERN, "The Higgs Boson" [Online]. Available: https://home.cern/science/physics/higgs-boson, 2024.

[7] CERN, "How did we discover the Higgs boson?" [Online]. Availabe: https://home.cern/science/physics/higgs-boson/how, 2024.

[8] M. Morrison, 'Without it there's Nothing: The Necessity of Simulation in the Higgs Search', Reconstructing Reality: Models, Mathematics, and Simulations, Oxford Studies in Philosophy of Science, Available: https://doi.org/10.1093/acprof:oso/9780199380275.003.0009, 2015.

[9] ATLAS collaboration, "Dataset from the ATLAS Higgs Boson Machine Learning Challenge 2014" CERN Open Data Portal. DOI:10.7483/OPENDATA.ATLAS.ZBP2.M5T8 Available: https://opendata.cern.ch/record/328, 2014.

[10] D. Whiteson, “HIGGS”. UCI Machine Learning Repository. Available: https://doi.org/10.24432/C5V312, 2014.

[11] https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/

[12] W. Feathers, ”Random Forest vs Gradient Boosted Trees: A Comparison” [Online]. Available: https://medium.com/@wilbossoftwarejourney/random-forest-vs-gradient-boosted-trees-pros-and-cons-8c1feec0ea0d, 2023.

[13] https://www.cs.toronto.edu/ mbrubake/teaching/C11/Handouts/AdaBoost.pdf

[14] https://www.kaggle.com/c/higgs-boson/discussion/10329.

[15] M. Bedford and V. Maclean, ”Public Trust in AI Technology Declines Amid Release of Consumer AI Tools” [Online]. Available: https://www.mitre.org/news-insights/news-release/public-trust-ai-technology-declines-amid-release-consumer-ai-tools, 2023.