

Predicting β -decay energy with machine learning

Jose M. Munoz,^{1,2,*} Serkan Akkoyun^{3,†}, Zayda P. Reyes^{2,4,‡} and Leonardo A. Pachon^{2,5,§}

¹Grupo de Física Teórica y Aplicada, EIA University, Envigado 055428, Colombia

²guane Enterprises, R+D+I Unit, Medellín 050010, Colombia

³Department of Physics, Faculty of Science, Sivas Cumhuriyet University, Sivas, 58140, Turkey

⁴Universidad Industrial de Santander, Grupo de Óptica y Tratamiento de Señales, Bucaramanga 680002, Colombia

⁵Grupo de Física Teórica y Matemática Aplicada, Instituto de Física, Facultad de Ciencias Exactas y Naturales, Universidad de Antioquia, Medellín 050010, Colombia



(Received 21 December 2022; accepted 13 February 2023; published xxxxxxxxx)

Q_β represents one of the most important factors characterizing unstable nuclei, as it can lead to a better understanding of nuclei behavior and the origin of heavy atoms. Recently, machine learning methods have been shown to be a powerful tool to increase accuracy in the prediction of diverse atomic properties such as energies, atomic charges, and volumes, among others. Nonetheless, these methods are often used as a black box not allowing unraveling insights into the phenomena under analysis. Here, the state-of-the-art precision of the β -decay energy on experimental data is outperformed by means of an ensemble of machine-learning models. The explainability tools implemented to eliminate the black box concern allowed to identify proton and neutron numbers as the most relevant characteristics to predict Q_β energies. Furthermore, a physics-informed feature addition improved models' robustness and raised vital characteristics of theoretical models of the nuclear structure.

DOI: [10.1103/PhysRevC.00.004300](https://doi.org/10.1103/PhysRevC.00.004300)

I. INTRODUCTION

There are about 250 stable isotopes and over 3000 unstable ones are known. One way that unstable atomic nuclei decay into stable ones is β decay. As a result, the β -decay energy Q_β that goes with it is a fundamental property of unstable atomic nuclei. Q_β values can be determined via several methods, such as β endpoint measurements, counting in coincidence with annihilation radiation, electron capture EC/β^+ ratio method, γ absorption with x-ray coincidence [1]. This process is complex to explore provided that its energy spectrum has a continuous structure. The decay energy seems to simply relate to the proton number and mass of the atomic nuclei. Experimental Q_β energies of nuclei can be verified in the $Z \in [4, 82]$, $N \in [4, 126]$ regions, whereas unknown energies can be calculated using α - β energy cycles by means of reliable α spectroscopic data. Moreover, some β -decay energies can be obtained with the help of energy cycles from α -decay energy systematic [2].

A. Explainable machine learning

Machine learning (ML) has recently emerged as an important tool for understanding physical phenomena, focusing on understanding how a model makes a particular decision,

rather than just accepting the output as a black box [3,4]. In the supervised approach, an algorithm is used as a functional approximation $\mathcal{F}(x)$ for an unknown function $F(x)$. Thus, it is possible to obtain the desired output for the collected physical data $x \in \mathbb{R}^n$. Classical statistical algorithms such as linear regression models or principal components regression (PCR) allow for characterizing the relevant predicting features of the model, thus providing an understanding of the physical phenomena underlying the data. However, they impose specific restrictions on the studied manifold, and thus, they are not well suited to most of the complex phenomena observed in nature [5]. On the other hand, modern approaches such as deep learning (DL) allow for a comprehensive representation of physical observables, as they work as universal function approximators [6,7]. The trade-off comes when it is necessary to understand what the algorithms are considering. To address this issue, explainable ML allows identifying of underlying characteristics such as feature importance [8–10], model the dynamics of complex systems [11,12], conservation law discovery [13,14], and symbolic expression extraction [15–17]. Besides, simple ML models have previously been used in the study of the β decay [18–22]; however, there is still the need to obtain physical intuition out of these techniques. Nuclear physics is not an exception, as determination of one and two proton separation energies [23], developing nuclear mass systematics [24], determination of ground state energies of the nuclei [19], identification of impact parameters in heavy-ion collisions [25], estimation of fusion reaction cross section [26], estimating nuclear rms charge radii [27], decoding β -decay systematic [28], and determination of

*jose.munoz25@eia.edu.co

†sakkoyun@cumhuriyet.edu.tr

‡zaydaprq@correo.uis.edu.co

§leonardo.pachon@udea.edu.co

TABLE I. Numerical distribution of the AME2020 data for each one of the measured parameters.

	$N - Z$	N	Z	m excess	δ_m	Binding E	Q_β	$\delta_{Q\beta}$	p	Atomic m	$\delta_a m$
mean	27.783	86.782	58.999	-24262.379	115.570	7985.498	1546.938	149.663	145.043	713121.060	120.919114
std	16.501	42.272	27.627	59122.015	239.781	666.922	7441.595	231.639	69.683	388483.889	199.856141
min	-3	1	0	-341875.725	0.000	0.000	-16673.000	0.000	1.000	99.560	0.000
25%	14	53	38	-67307.280	2.251	7731.891	-4103.108	5.004	91.000	171690.750	2.417
50%	26	85	59	-40851.305	10.614	8075.771	62.237	22.526	144.000	926617.954	11.394
75%	40	119	81	2679.998	179.083	8383.611	6087.989	206.267	199.000	954880.438	189.704
max	64	177	117	196397.000	8013.457	8794.555	32740.000	1082.000	294.000	999981.252	1078.000

fusion barrier heights [29] have already been studied via this tooling.

B. ML for β decay

To be precise, the β -decay energies [30] were estimated by considering the neutron and proton numbers of atomic nuclei as the only input parameters and the data values have been obtained based on the Hartree-Fock- Bardeen Cooper Schrieffer (BCS) method with the Skyrme force MSk7 [31]. By contrast, here, the recent AME2020 experimental data [32] are considered and the input parameters are augmented by considering a full range of physical properties of atoms (see Table I below). By using the experimental Q_β values available in the literature, the systematics of atomic nuclei related to this energy was obtained here by performing ML, considerably decreasing the root mean squared error (RMSE) of previous approaches by a factor of 2.5. In addition, by calculating the importance scores, the fundamental parameters of atomic nuclei that have an impact on the Q_β energies are highlighted.

C. The atomic mass table

The dataset consists of the classical nuclear variables, numerically described in Table I, where entries with any non-computable data entrances were removed, leaving a total of 2813 unstable isotopes. The relevant dataset variables are the number of neutrons in each isotope N , number of protons in each isotope Z , mass number A , difference between measured mass and the mass number M_e and binding energy B_E . Here, notice that a majority of the features are highly skewed. Further a description of the raw data is shown in the Appendix.

It is possible to identify high correlations between some of the variables provided that some of the features are computable from the others (such as the mass number from the atomic and nuclear numbers). This needs to be addressed since it is known that algorithms that are susceptible to multicollinear features generate unreliable predictions [33].

Therefore, an exploratory dimensionality reduction analysis is implemented using principal component analysis (PCA) with the normalized variables and evaluating the performance of a naive predictor model. Here, both the explained variance and a principal component regression (PCR) used a fivefold cross-validation scheme [34]. As expected, more than two principal components are necessary to give a complete sense of Q_β by evaluating the RMSE of a linear regression model. In addition, Fig. 1 depicts the asymptotic behavior that is reached

using more than three principal components. Nevertheless, neither the second nor the third principal components are negligible.

The PCA loadings, obtained as shown in Fig. 2, were used to understand the importance of each of the parameters. Interestingly enough, the second orthogonal component loads more heavily the uncertainties, rather than the components themselves, with a major role in the binding energy B_E . This suggests that there exists a significant contribution of the features related to the experimental uncertainties reported in the original dataset. However, the correlation analysis shown in Fig. 3 accounts for part of this effect. Nonetheless, it is expected that universal approximators can easily overcome this issue.

II. MODELING Q_β : A REGRESSION APPROACH

To start the modeling of the phenomena in a systematic and replicable way, training/testing/validation sets were generated with the dataset using a 70/20/10 split of the randomly shuffled AME2020 dataset. A variety of tabular models were fitted in an automatic manner, reporting the scores, and using only the training and validation splits. In this approach, in each one of the steps, the best-performing models are selected and their hyperparameters are tuned via Bayesian optimization techniques as designed by Ref. [35]. Based on a collection of regression metrics (such as explained variance, RSME, and their volatility among different splits of the training set), the models obtaining high scores while keeping explainability

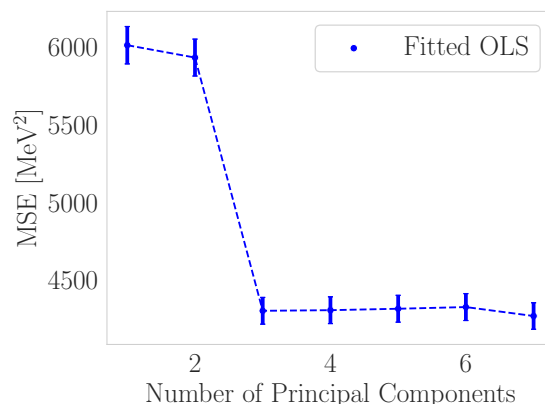


FIG. 1. RMSE of the fit for a different number of components in the PCA analysis.

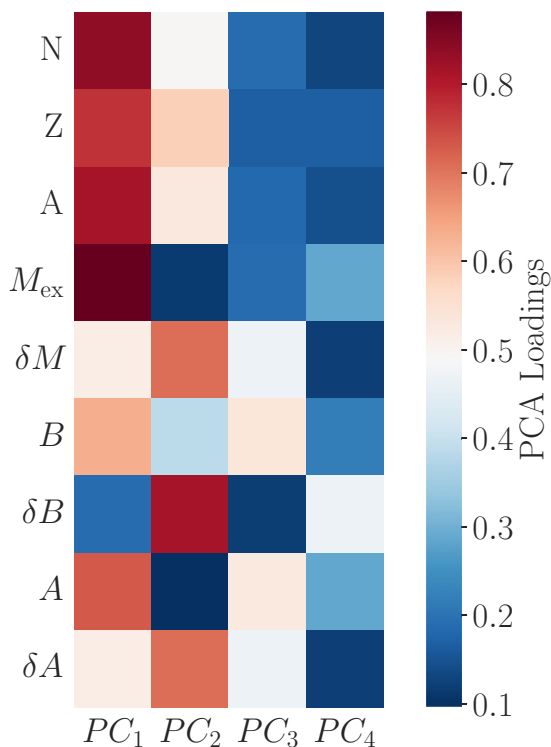


FIG. 2. Loadings of each one of the variables into the principal orthogonal components. See text for more details.

143 were selected and further explored via hyperparameter tuning
 144 and feature importance measurement techniques.

145 To have a baseline, the previous work [30] was replicated
 146 using a similar approach. Here, a vanilla fully connected
 147 multilayer perceptron (MLP) was implemented, but with the
 148 AME2020 dataset, which was trained only using the N and

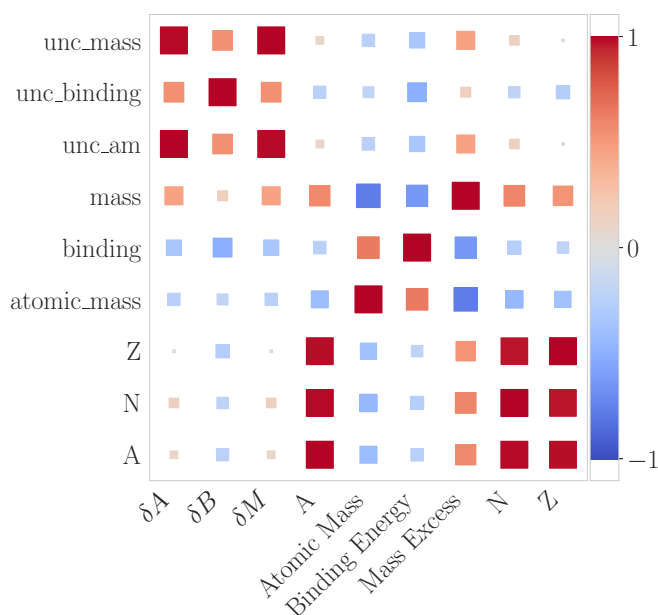


FIG. 3. Correlations between variables in the Atomic Mass Table.

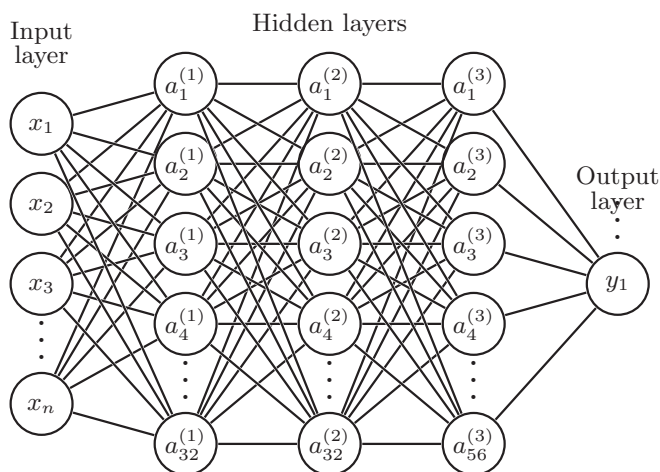


FIG. 4. Schematic representation of the NN used when fitting the Q_β .

Z of each one of the entries. The architecture consists of
 a single hidden layer with 100 neurons, hyperbolic-tangent
 nonlinearities, and classic stochastic gradient descent (SGD).

A. ML and AI models

1. Ordinary linear regression

Besides its simplicity, this approach is well suited to sci-
 entific tasks because of the transparency, a general regression
 allows. Thus, it serves both as a benchmark and brings insights
 into the process. The ordinary least squares (OLS) approach
 uses a projection model to fit the input vector \mathbf{X} containing the
 selected features to obtain an estimator Q_β via

$$Q_\beta = \mathbf{X}^T \mathbf{k} + \varepsilon. \tag{1}$$

The optimal parameters for the vector of coefficients \mathbf{k} and
 the scalar ε , which minimizes the difference between the
 observation Q_β and the outputs Q_β are obtained via the OLS
 procedure.

Moreover, this technique allows for obtaining a statistical
 measurement of the fitted coefficients and their uncertainties,
 thus, getting insight into the relevant features and characteris-
 tics of the model.

2. Modern MLP

Multilayer perceptrons (MLPs) are a type of artificial neu-
 ral network (ANN) in which neurons are connected via a
 directed acyclic graph. They have one or more inputs, more
 than one hidden layer, and one output layer. The hidden
 neurons simply propagate information forward to the next
 layer, each layer is a linear transformation of the previous
 one via nonlinearities, which are required to allow a general
 approximation of a black box function such as the studied
 $Q_\beta = Q_\beta(A, Z, N, \dots)$. Such functions are referred to as the
 layer activation function and the output layer corresponds to
 a prediction of Q_β . The neural network (NN) architecture
 schema is shown in Fig. 4.

The Gaussian error linear unit (GeLU) activation function
 was included as a nonlinearity, given that it has been shown

that helps reduce several problems associated with small datasets [36]. In addition, an interlayer dropout is included, which discards random connections between layers with a probability of 0.1. This has been shown also to help in the optimization procedure [37].

Several experiments modifying the hyperparameters via optimization methods were performed to obtain a suitable configuration of both the number of neurons per layer and the batch size [38]. A proper learning rate was also obtained using the method of cyclical learning rates as proposed in Ref. [39]. The training process used a decaying learning rate and was implemented using the PyTorch Lightning library [40]. This used minibatch gradient descent to optimize the mean squared error loss (MSELoss) with the Adam optimizer [41]. Before starting the actual training process, three warm-up rounds were used.

3. Gradient boosting machines

Besides being increasingly known and used in different areas of physics, ANN are far from being the only available option for an explainable and accessible model, a clear example is an extreme gradient boosting (with XGBOOST) algorithm [42], which uses the gradient boosted trees (BDT) method for fitting several tree regressors (or weak linear learners) using a regularized loss function. These are compared and later assembled with the approach of ML boosting. They perform especially well on tabular data and in most cases outperforming deep-learning approaches, requiring only fractions of the computational cost [43]. Here, the XGBoost model with the optimization of hyperparameters with Optuna [44] was implemented and several hyperparameters such as the depth, learning rate, and the number of learners were inspected. Moreover, callbacks to avoid overfitting were added to the loss minimization process.

4. Attention networks

The *attention mechanism* [45] is implemented here to properly learning weights and relations within the input via learning queries and keys. Besides, this approach was found to be successful for unstructured data in several tasks such as protein unfolding and natural language processing, it had not been especially useful for tabular data until the appearance of the TabNet model [46]. In this approach, the TabNet model was used with weighted Adam optimizer and early stopping callbacks to avoid overfitting.

5. Ensemble model

Even though the fitting generated with the methods above might fit accurately to the datasets, the models trained can generally either underfit, overfit, or just be poorly configured. The aim of the ensemble method is to facilitate the best of several base models to train a strong prediction model. This is achieved via a voting ensemble regression (VER), a linear combination of the predictions is done by weighting the individual models according to their performance on the validation split. Specifically, the weighting of the BDTs, the ANN, and the TabNet model is performed by evaluating both

TABLE II. Regression metrics for the different classifiers trained on the original AME2020 training set.

Model	R^2	$MSE_{[MeV^2]}$	$RMSE_{[MeV]}$	$MAE_{[MeV]}$
Baseline	0.926	4.134	2.033	1.518
OLS	0.868	7.943	2.818	2.043
ANN	0.946	2.724	1.650	1.289
BDT	0.958	2.271	1.715	1.275
Ensemble	0.957	2.305	1.518	1.191

the RMSE and the standard deviation over a fivefold in the validation set.

B. Data augmentation techniques

From the initial evaluation of the feature importance, experimental uncertainties arose as noteworthy in the regression of Q_β . Here, they come from the combination of both epistemic and systematic errors. Also, a characteristic often makes it difficult for the models to improve their performance is the limited experimental data they are training on.

Therefore, a method for obtaining more diverse and accurate data is useful. A Monte Carlo sampling on each one of the parameters with registered uncertainty is proposed. Assuming that each one of the variables has an error normally distributed and centered on the reported data. This allows obtaining a significant data set for further improving the ML models with increased robustness. This is only done in the training entries so that no isotopes are overlapped between the split, ensuring no data leakage. Notice that this approach can be followed by the same procedure as an augmentation at testing time, as it has been shown to considerably improve robustness [47].

Following this procedure, the original training data was overpopulated, from 1912 to a total of 573 600 samples, followed by the removal of the columns related to uncertainties, since this information has already been taken into account. The models that were trained on the original data (which will

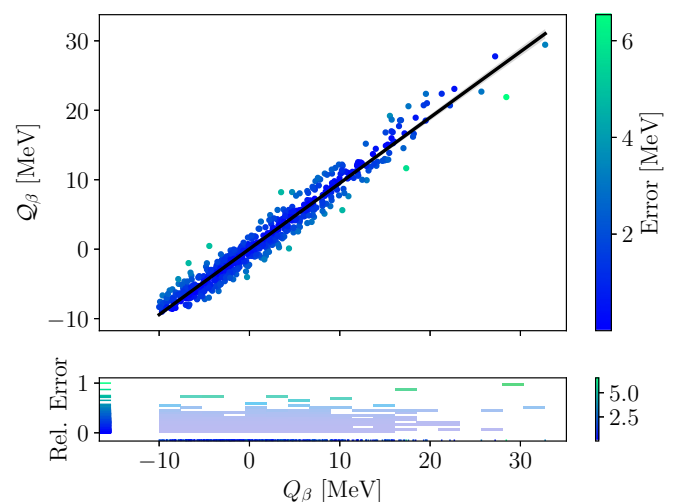


FIG. 5. Residuals of the ensemble of models trained on the original AME2020 dataset.

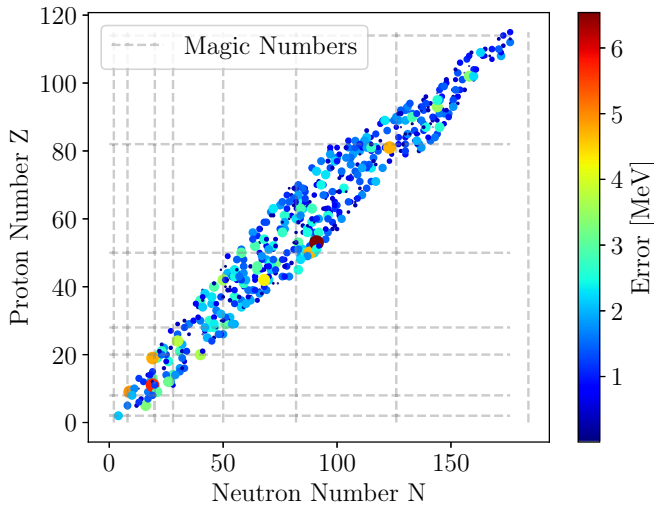


FIG. 6. Distribution of the errors in the predicted Q_β by the models trained in the original AME2020 dataset within the N and Z plane, the size of the points represented is also determined by the deviation in Q_β .

continue to be referred to as AME2020) were also trained and tuned on the augmented dataset.

C. Periodic feature augmentation

In the spirit of the augmentation technique presented in Sec. II B, physics-inspired features were added to both datasets (i.e., to the AME2020 and to the augmented dataset from Sec. II B). The periodic group was added to the data as a one-hot encoded feature. Moreover, the *magic numbers* such that they are arranged into complete shells in the nucleus were also taken for each one of the isotopes, both for N and Z . For this, the dataset includes the difference to the closes magic number (which we denote as m_N and m_Z , respectively). In addition, the ratio $m_Z/(m_N + 1)$ and Z/N were also included.

Given that this method heavily relies on the current atomic model, we denote the generated data as *feature augmented*.

TABLE III. Results on the fitted coefficients for the OLS model. c_i is the coefficient and corresponds to the rate of change of Q_β with respect to each variable, σ is the standard error, $P > \|t\|$ is the probability that the coefficient is measured, and C.I. @ 95% is representing the range of the coefficients within 95% confidence level.

Variable	c_i	σ	$P > \ t\ $	C.I. @ 95%
N	533.02	10.91	0.00	[511.60, 554.43]
Z	-925.62	16.08	0.00	[-957.19, -894.06]
ME	-0.00	0.01	0.97	[-0.01, 0.01]
δ_{ME}	-196.84	578.31	0.73	[-1331.34, 937.66]
E_b	1.21	0.04	0.00	[1.13, 1.30]
δ_{E_b}	263.86	20.87	0.00	[222.91, 304.80]
AM	-0.001	0.00	-2.01	[-0.01, -0.00]
δ_{AM}	186.21	538.77	0.73	[-870.73, 1243.14]

TABLE IV. Regression metrics for both the original AME2020 and the augmented data.

Model	R^2	MSE[MeV ²]	RMSE [MeV]	MAE[MeV]
Baseline	0.926	4.134	2.033	1.518
ANN	0.964	1.895	1.372	1.084
TabNet	0.958	0.995	1.149	0.987
BDT	0.986	0.749	0.865	0.371
Ensemble	0.987	0.686	0.828	0.460

III. RESULTS

For each one of the models described above, the coefficient of determination (R^2) of the fit, the mean squared error (MSE), the RMSE, and the mean absolute error (MAE) were calculated. We use this collection of metrics to get an estimation of how well distributed and big the deviations between Q_β and \hat{Q}_β is for the unseen testing dataset. Below, are the results obtained using the approaches of different ML models with several datasets.

I. AME2020

After fitting several regressors on the original split, modern architecture clearly outperforms the previous state-of-the-art RMSE. Notably, results obtained by the OLS are not considerably bigger compared to the baseline ANN, as shown in Table II. Moreover, experiments suggest that the number of parameters that have to be trained on the TabNet requires considerably more than one entry per isotope, thus the metrics are not taken into account.

Besides the distribution of errors seems to be uniformly distributed for different Q_β within the dataset as shown in Fig. 5, it is remarkable that the distribution of outliers is not with a clear higher deviation on low number of nucleons, as illustrated in Fig. 6 and more notably, on the intersection points for the atomic magic numbers.

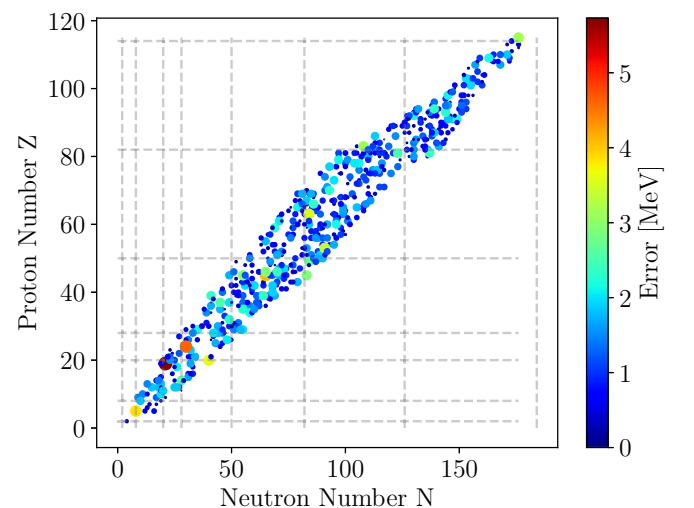


FIG. 7. Real Q_β versus predicted Q_β for the ensemble model trained on the augmented dataset.

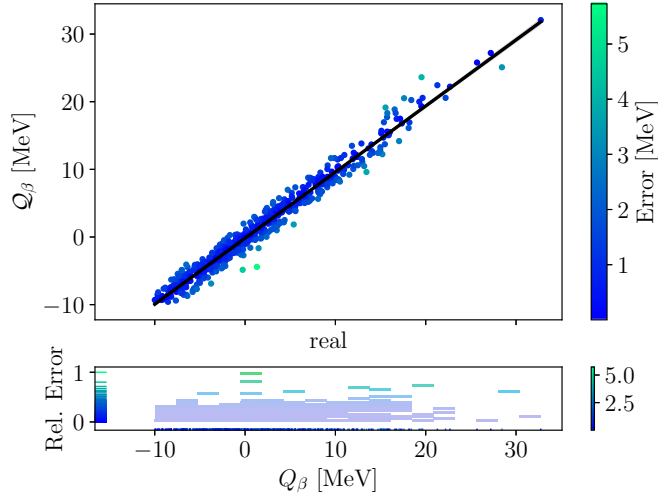


FIG. 8. Deviation in predicted Q_β for the test isotopes by the models trained on the augmented dataset.

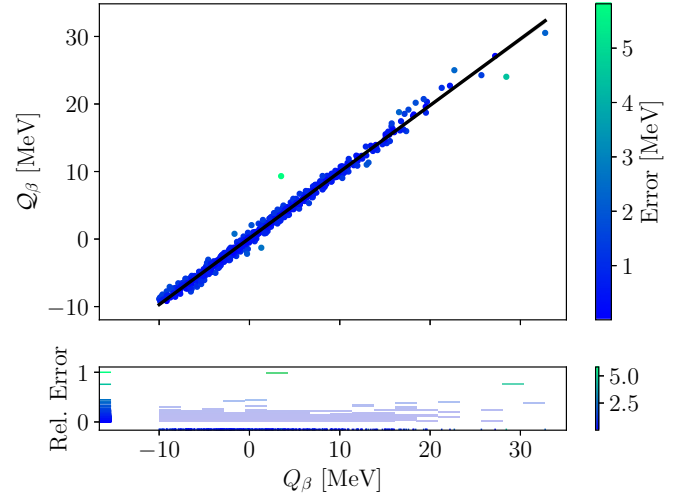


FIG. 9. Real Q_β versus predicted Q_β by the ensemble model after adding periodic features.

Furthermore, the coefficients extracted by the OLS regression reflect the fact of not only interference but also of irrelevance for some fitted parameters, evidenced in Table III.

2. Augmented dataset

Since the amount of data used in this approach is considerably big, the OLS was not implemented, for all other algorithms the results are presented in Table IV.

All the models were trained on the same test split that was established from the beginning of the process. And the good performance with no manifestations of overfitting shows that the models are capable of extrapolating to unseen atomic feature configurations. Remarkably, the ensemble model achieves remarkable accuracy with fewer error outliers than all of the other models, taking advantage of the uncertainties with the original AME2020 dataset and the augmentation process. Moreover, the voting based on splits approach allows to effectively target robustness, as the ensemble method achieved a 12% lower maximum deviation in Q_β over the test isotopes. This allows for state-of-the-art metrics and a remarkably low MSE, as shown in Fig. 7 for the case of the ensemble model.

TABLE V. Regression metrics for both the original AME2020 and the augmented data after performing feature augmentation.

Model	R^2	$MSE_{[MeV^2]}$	$RMSE_{[MeV]}$	$MAE_{[MeV]}$
AME2020				
Baseline	0.926	4.134	2.033	1.518
ANN	0.950	2.583	1.607	1.252
BDT	0.981	0.756	0.869	0.613
Augmented				
ANN	0.970	1.640	1.281	1.031
TabNet	0.982	0.993	0.997	0.774
BDT	0.989	0.576	0.759	0.534
Ensemble	0.991	0.511	0.714	0.510

Furthermore, besides the error corresponding to isotopes near magic numbers still having higher errors, the deviations were reduced uniformly for test isotopes, as illustrated in Fig. 8.

3. Feature augmented dataset

Within the same nature, the models trained on the dataset containing the periodic group, and information related to the magic numbers, as presented in Sec. II C. The first experiments showed that the encoded periodic group is detrimental to the performance, and thus those features were dropped. This approach was implemented both on the augmented and in the original AME2020 datasets, both of which are presented in Table V.

This approach demonstrates the importance of generating features based on theoretical models, as higher-level atomic

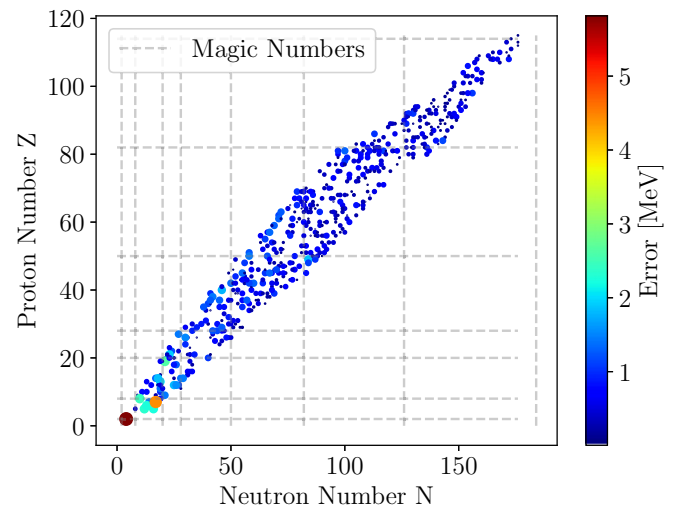


FIG. 10. Distribution of deviations in Q_β as predicted by the ensemble of all models on the testing isotopes with the augmented periodic features.

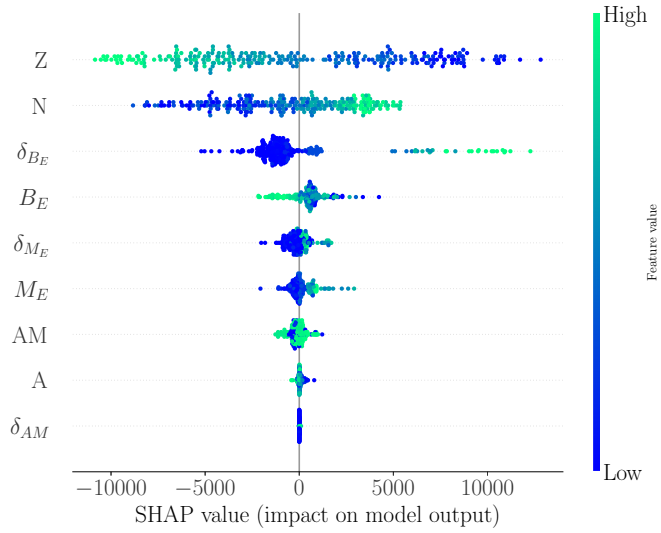


FIG. 11. Feature importance of the hypertuned BDT on the AME2020 dataset.

337 features, clearly allow models to fit the function better. This is
 338 noticeable in the distribution of deviations, as shown in Fig. 9.

339 As expected, this approach allows a considerable reduction
 340 in errors, especially for the critical isotopes in not feature
 341 augmented approaches, and with the highest error at a low
 342 number of isotopes, as illustrated in Fig. 10. Notably, the
 343 two isotopes with an error above the percentile 90 correspond
 344 to helium (isotope with $A = 6$) and nitrogen (isotope with
 345 $A = 24$).

IV. EXPLAINING THE BLACK BOX

346 Up to this point, ML techniques have been shown to pro-
 347 vide a highly accurate regression of the β -decay energies.
 348

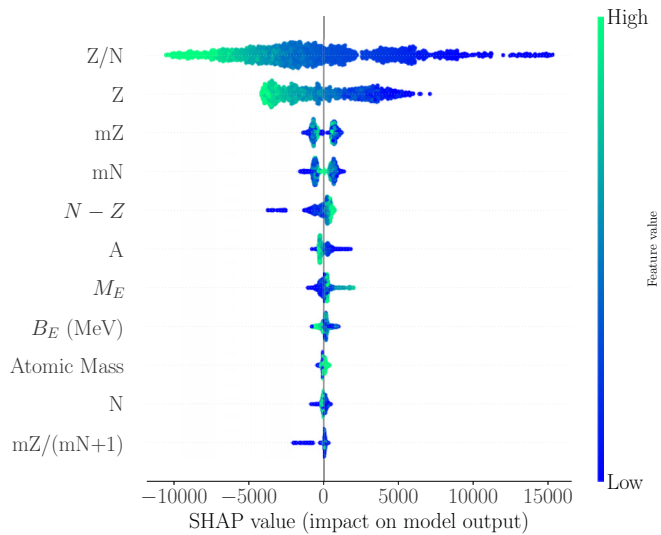


FIG. 12. Permutational feature importance for the model trained on the augmented set with magic-number features, only the eight most important features are displayed.

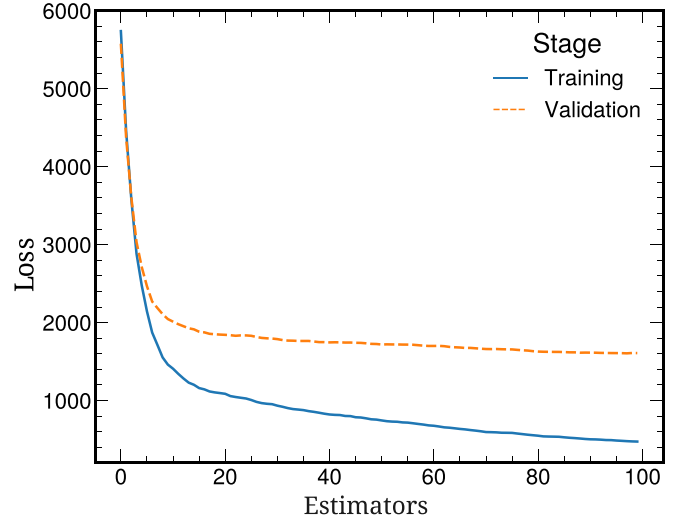


FIG. 13. MSE Loss of the BDT model after hyper-tuning trained on the augmented dataset with feature augmentation.

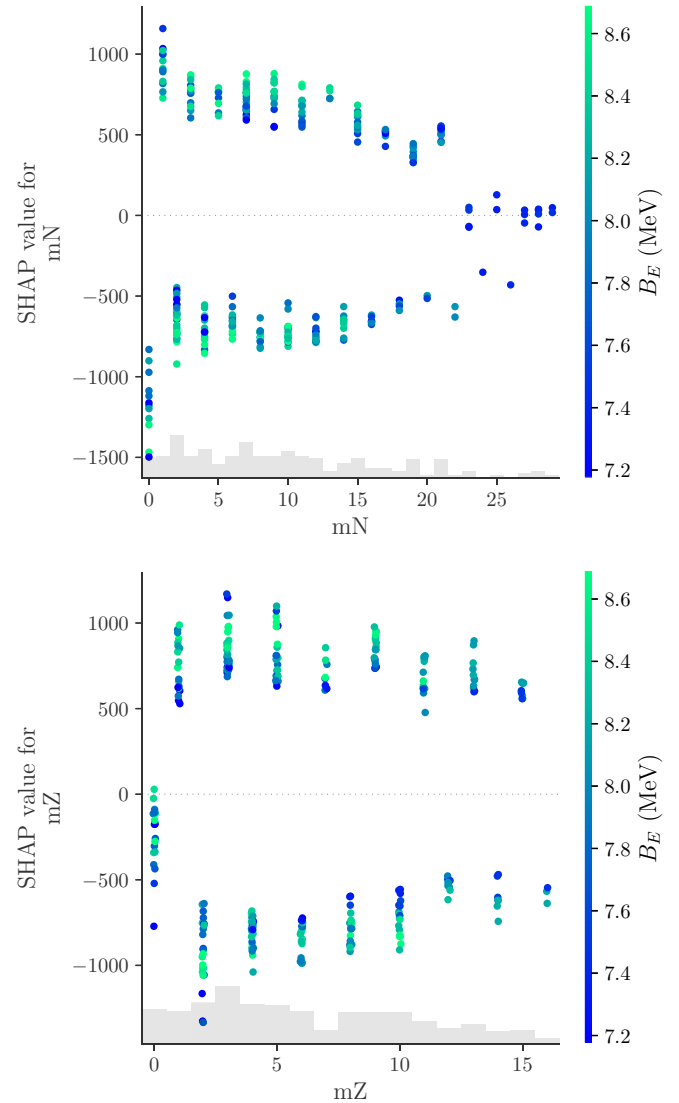


FIG. 14. Distribution of the feature importance of mZ and mN where the alternating pattern in parity is evident.

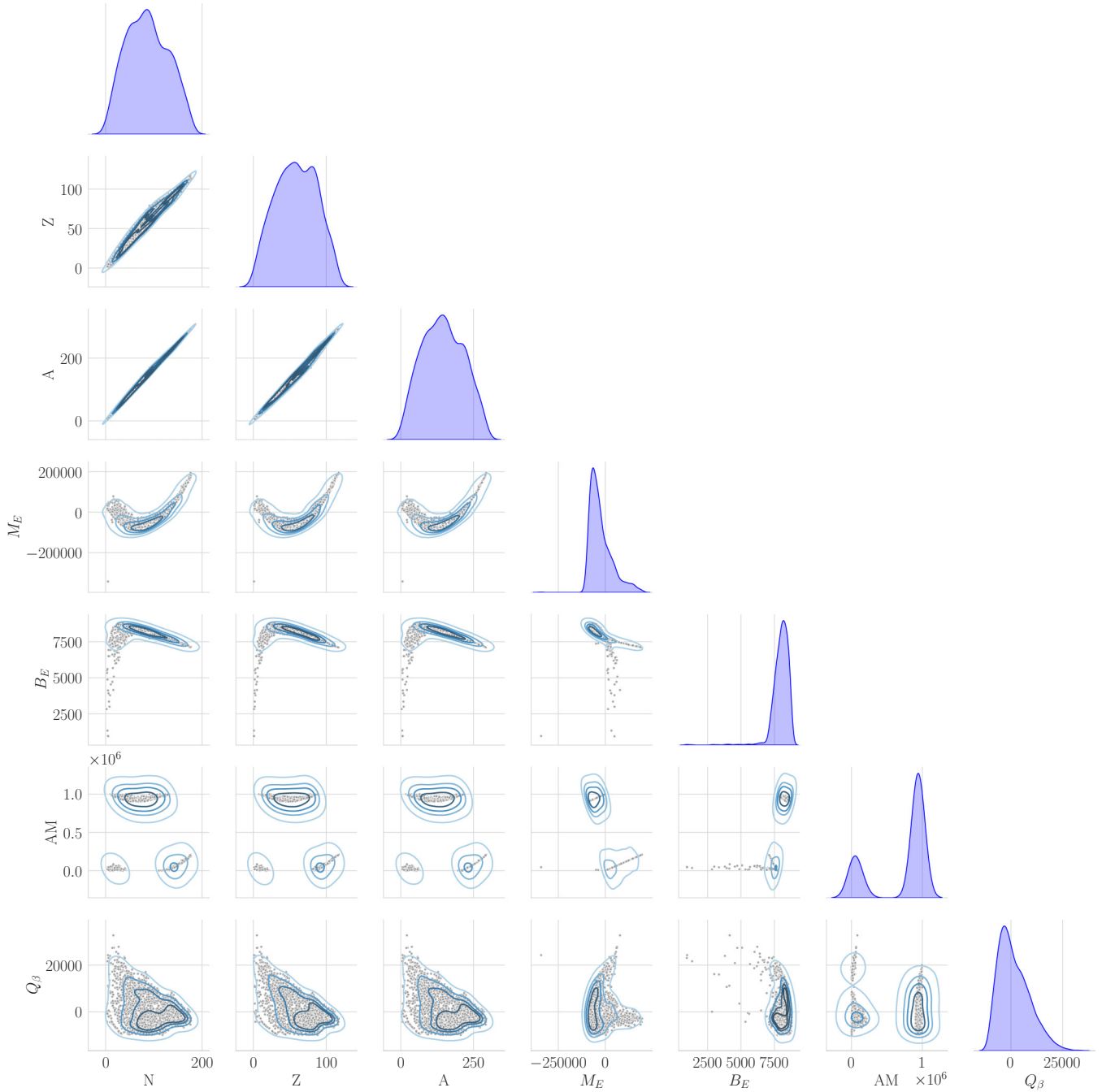


FIG. 15. Initial distribution of the atomic features in the AME2020 dataset [32].

349 However, it is possible further to inspect the impact of each
 350 one of the features on the output of the model. This can be
 351 done via the evaluation of the shapley additive explanations
 352 (SHAP), a model-agnostic explainability method inherited
 353 from game theory, where the feature importance of the feature
 354 is calculated via a global analysis of the marginals per each
 355 input feature [48]. For this end, the KernelSHAP technique
 356 implemented in Ref. [49] is applied. The results are presented
 357 in Fig. 11 for the case of BDT, which was fully trained and
 358 hyperoptimized on the AME2020 training data.

359 In accordance with our explorations with the PCA ap-
 360 proach, the uncertainties play a significant role in explaining

the model output and that in general, the mass excess is
 considered charged. The feature importance after training on
 the augmented the dataset was also studied. It was found that
 the importance scores for the physical features (N , Z , mass
 excess, B_E , atomic mass, and A) kept their assigned impor-
 tance as expected from an orthogonal permutational scoring.
 Moreover, the feature weights inside the trees for the BDT
 and the feature permutation importance for the ANN, which
 matches the overall order was also investigated.

The models thus fit the coefficients with a major impor-
 tance of Z and N , which could in fact explain the good
 results obtained in Ref. [35] with such a simple architecture.

Furthermore, the relation found by the black boxes, in general, is inversely proportional for Z and direct for N .

More importantly, the fact that the groups were detrimental to the performance of the regressors was validated by the permutational scoring. By considering the augmented periodic features, the high importance of the ratios N/Z , and the magic numbers give insights into why the critical isotopes near the magic numbers were predicted with lower errors [cf. Eq. (12)].

Remarkably, the feature importance averaged over five runs for the models are shown to have a relative low F score, experiments show that augmenting the features with periodic variables allows the BDT model to achieve the same metrics as the augmented dataset, and to increase metrics on the augmented dataset with virtually no overfitting, as illustrated in Fig. 13.

Further inspection of the augmented features also shows interesting behavior arising from physical phenomena. For instance, both for mZ (absolute difference between Z and the closest magic number) and mN (difference between N and the closest magic number) have alternating feature importance with parity as shown in Fig. 14. The oddness in the number of nucleons in relation to the magic numbers, in general, results in lower B_E , which is in excellent agreement with the nuclear shell model [50] as well as theoretical approximations of nuclear structure such as the Weizsäcker formula [51].

This alternation in the SHAP importance value is also larger near the magic numbers, which might explain the improvement in deviations near magic numbers with the feature augmentation, and is clear evidence that the model is already capable of getting insights into the nuclear structure.

V. CONCLUSIONS AND OUTLOOKS

In this work, an ML model was developed to predict the β -decay energies of isotopes using the experimental dataset AME2020 [32]. Using modern techniques and data augmentation strategies, it was shown that our models can

considerably decrease the error in estimating Q_β compared with previous ML approaches while maintaining interpretability. The importance of considering experimental uncertainties was illustrated as the models incorporating sampling techniques become considerably more robust to noise. Furthermore, it was demonstrated that by incorporating physical features such as magic numbers, models can be highly improved. Using explainability tools it is shown that the models have learned fundamental features of the atomic structure.

We look forward to generating clearer physical intuition from the explainability extracted from several ML techniques. Clearly, the intersection between ML and physics is only starting to showcase its potential.

ACKNOWLEDGMENTS

This work was supported by the Erasmus+ program and the R+D+I efforts from guane Enterprises.

APPENDIX: DATA DISTRIBUTION

The tabular data used in this work compiles the atomic features published in [32]. It contains the features:

- (i) Neutron number N : Number of neutrons in each isotope.
- (ii) Proton number Z : Number of protons in each isotope.
- (iii) Mass number A : Atomic mass ($N + Z$).
- (iv) Mass excess M_e : Difference between measured mass and the mass number (A).

Binding energy B_E : Measured per nucleon, refers to the ionization potential.

The distribution of the features is shown in Fig. 15, where no clear relationship between Q_β is evident, but well-distributed samples for the atomic features are evident, both in scattered and in the histograms.

-
- [1] G. Alkhalzov, L. Batist, A. Bykov, F. Moroz, S. Y. Orlov, V. Tarasov, and V. Wittmann, *Z. Phys. A: Hadrons Nucl.* **344**, 425 (1993).
 - [2] N. Kolesnikov and A. Krylova, *Zhur. Eksptl. Teoret. Fiz.* **37** (1959).
 - [3] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, *Rev. Mod. Phys.* **91**, 045002 (2019).
 - [4] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, *Nat. Rev. Phys.* **3**, 422 (2021).
 - [5] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Vol. 2 (Springer, Berlin, 2009).
 - [6] C. P. Chen, Z. Liu, and S. Feng, *IEEE Trans. Neural Netw. Learning Syst.* **30**, 1191 (2018).
 - [7] F. Scarselli and A. C. Tsoi, *Neural Networks* **11**, 15 (1998).
 - [8] G. Casalicchio, C. Molnar, and B. Bischl, *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (Springer, Berlin, 2018), pp. 655–670.
 - [9] M. Wojtas and K. Chen, *Advances in Neural Information Processing Systems* **33**, 5105 (2020).
 - [10] D. Rengasamy, J. M. Mase, M. T. Torres, B. Rothwell, D. A. Winkler, and G. P. Figueredo, [arXiv:2110.11713](https://arxiv.org/abs/2110.11713).
 - [11] M. Cranmer, S. Greydanus, S. Hoyer, P. Battaglia, D. Spergel, and S. Ho, [arXiv:2003.04630](https://arxiv.org/abs/2003.04630).
 - [12] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, *Adv. neural inf. process. syst.* **31**, 3 (2018).
 - [13] Z. Liu and M. Tegmark, *Phys. Rev. Lett.* **126**, 180604 (2021).
 - [14] K. Lee and K. Carlberg, [arXiv:1909.09754](https://arxiv.org/abs/1909.09754).
 - [15] M. Cranmer, A. Sanchez Gonzalez, P. Battaglia, R. Xu, K. Cranmer, D. Spergel, and S. Ho, *Advances in Neural Information Processing Systems* **33**, 17429 (2020).
 - [16] S.-M. Udrescu, A. Tan, J. Feng, O. Neto, T. Wu, and M. Tegmark, *Advances in Neural Information Processing Systems* **33**, 4860 (2020).
 - [17] J. Bakarji, K. Champion, J. N. Kutz, and S. L. Brunton, [arXiv:2201.05136](https://arxiv.org/abs/2201.05136).

- [18] Z. M. Niu, H. Z. Liang, B. H. Sun, W. H. Long, and Y. F. Niu, *Phys. Rev. C* **99**, 064307 (2019).
- [19] T. Bayram, S. Akkoyun, and S. O. Kara, *Ann. Nucl. Energy* **63**, 172 (2014).
- [20] M. Cruz, *Bull. Am. Phys. Soc.* **64** (2019).
- [21] N. Costiris, E. Mavrommatis, K. Gernoth, and J. Clark, [arXiv:nucl-th/0701096](https://arxiv.org/abs/nucl-th/0701096).
- [22] Z.-P. Gao, Y.-J. Wang, H.-L. Lü, Q.-F. Li, C.-W. Shen, and L. Liu, *Nucl. Sci. Tech.* **32**, 1 (2021).
- [23] S. Athanassopoulos, E. Mavrommatis, K. A. Gernoth, and J. W. Clark, [arXiv:nucl-th/0509075](https://arxiv.org/abs/nucl-th/0509075).
- [24] S. Athanassopoulos, E. Mavrommatis, K. Gernoth, and J. Clark, *Nucl. Phys. A* **743**, 222 (2004).
- [25] S. A. Bass, A. Bischoff, J. A. Maruhn, H. Stöcker, and W. Greiner, *Phys. Rev. C* **53**, 2358 (1996).
- [26] S. Akkoyun, *Nucl. Instrum. Methods Phys. Res. B* **462**, 51 (2020).
- [27] S. Akkoyun, T. Bayram, S. O. Kara, and A. Sinan, *J. Phys. G: Nucl. Part. Phys.* **40**, 055106 (2013).
- [28] N. J. Costiris, E. Mavrommatis, K. A. Gernoth, and J. W. Clark, *Phys. Rev. C* **80**, 044332 (2009).
- [29] S. Akkoyun and T. Bayram, *Int. J. Mod. Phys. E* **23**, 1450064 (2014).
- [30] S. Akkoyun, T. Bayram, and T. Turker, *Radiat. Phys. Chem.* **96**, 186 (2014).
- [31] F. Tondeur, S. Goriely, J. M. Pearson, and M. Onsi, *Phys. Rev. C* **62**, 024308 (2000).
- [32] M. Wang, W. J. Huang, F. G. Kondev, G. Audi, and S. Naimi, *Chin. Phys. C* **45**, 030003 (2021).
- [33] A. Garg and K. Tai, *2012 Proceedings of the International Conference on Modelling, Identification and Control* (IEEE, 2012), pp. 353–358.
- [34] I. T. Jolliffe, *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **31**, 300 (1982).
- [35] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, [arXiv:1907.10902](https://arxiv.org/abs/1907.10902).
- [36] A. Nguyen, K. Pham, D. Ngo, T. Ngo, and L. Pham, An analysis of state-of-the-art activation functions For supervised deep neural network, *2021 International Conference on System Science and Engineering (ICSSE)* (IEEE, 2021), pp. 215–220.
- [37] C. Garbin, X. Zhu, and O. Marques, *Multimed. Tools. Appl.* **79**, 12777 (2020).
- [38] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, 2019), pp. 2623–2631.
- [39] L. N. Smith, Cyclical learning rates for training neural network, [arXiv:1506.01186](https://arxiv.org/abs/1506.01186) (2015).
- [40] W. Falcon and The PyTorch Lightning team, PyTorch Lightning (2019).
- [41] P. Rajendra, H. Ravi. PVN, and G. Naidu T, *AIP Conference Proceedings* (AIP Publishing LLC, New York, 2021), Vol. 2375, p. 020034.
- [42] T. Chen and C. Guestrin, XGBoost, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, 2016).
- [43] R. Shwartz-Ziv and A. Armon, Tabular Data: Deep Learning is Not All You Need (2021).
- [44] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, Optuna: A next-generation hyperparameter optimization framework (2019).
- [45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, *Advances in Neural Information Processing Systems* **30** (2017).
- [46] S. O. Arik and T. Pfister, TabNet: Attentive interpretable tabular learning (2019).
- [47] I. Kim, Y. Kim, and S. Kim, Learning loss for test-time augmentation, [arXiv:2010.11422](https://arxiv.org/abs/2010.11422) (2020).
- [48] S. M. Lundberg and S.-I. Lee, *Advances in Neural Information Processing Systems* **30**, 4765 (2017).
- [49] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, *Nat. Mach. Intell* **2**, 56 (2020).
- [50] B. A. Brown and B. Wildenthal, *Annu. Rev. Nucl. Part. Sci.* **38**, 29 (1988).
- [51] J. Lilley, *Nuclear Physics: Principles and Applications* (John Wiley & Sons, New York, 2013).