Controlling extrapolations of nuclear properties with feature selection

Rodrigo Navarro Pérez1,∗ and Nicolas Schunck2,†

1Department of Physics, San Diego State University, 5500 Campanile Drive, San Diego, California 92182-1233, USA
2Nuclear and Chemical Sciences Division, Lawrence Livermore National Laboratory, Livermore, California 94551, USA

(Dated: January 24, 2022)

Predictions of nuclear properties far from measured data are inherently imprecise because of uncertainties in our knowledge of nuclear forces and in our treatment of quantum many-body effects in strongly-interacting systems. While the model bias can be directly calculated when experimental data is available, only an estimate can be made in the absence of such measurements. Current approaches to compute the estimated bias quickly lose predictive power when input variables such as proton or neutron number are extrapolated, resulting in uncontrolled uncertainties in applications such as nucleosynthesis simulations. In this letter, we present a novel technique to identify the input variables of machine learning algorithms that can provide robust estimates of model bias. Our process is based on selecting input variables, or features, based on their probability distribution functions across the entire nuclear chart. We illustrate our approach on the problem of quantifying the model bias in nuclear binding energies calculated with Density Functional Theory (DFT). We show that feature selection can systematically improve theoretical predictions without increasing uncertainties.

Introduction - Nuclear physics models are at the heart of some of the most important contemporary scientific problems. Prominent examples include probing the limits of the periodic table of elements [1], understanding the mechanisms of nuclear fission for basic science and carbon-free energy production sources [2] or simulating the formation of elements in the universe through a set of nuclear reactions in various stellar environments [3]. These problems require a complete description of nuclear properties for all nuclei across the nuclear chart, but only a very small subset of them have been measured experimentally. With the coming online of next-generation radioactive ion beam facilities, many of the gaps in our knowledge will be filled [4]. Yet, the vast majority of nuclear data will remain out of reach and will have to be determined through theoretical simulations.

Atomic nuclei are self-bound, strongly-interacting quantum many-body systems [5, 6]. The nuclear forces that bind them are not elementary interactions but are derived from non-perturbative quantum chromodynamics [7–9]. For these reasons, an exact description of the nucleus is impossible and approximations are needed [10–14]. As a result, all theories of the nucleus are in fact models characterized, among others, by a small number of parameters that must be calibrated to experimental data. Most importantly, the accuracy and precision of nuclear models tend to be far lower than what direct experimental measurements can deliver. For example, state-of-the-art mass models predict nuclear binding energies to within about 500 keV [15, 16] while Penning-trap mass measurements yield results within a few eV – about 500 000 times more precise [17, 18]. In other words, nuclear models are inherently imperfect. Extrapolations far from the calibration region with such models are bound to come with a potentially very large bias.

The problems of calibrating nuclear model parameters, of quantifying and propagating the related uncertainties, and of estimating prediction biases have spurred interest in machine learning (ML) algorithms [19–29]. These new applications have resulted in better estimates for the limits of the nuclear chart [30], improved mass tables [20–22], and extrapolations of no-core shell model ground-state energies and radii to very large model spaces [25]. A recent workshop on “AI for nuclear physics” presented an in-depth report of current and possible applications of the broader field of artificial intelligence (AI) to nuclear physics [31]. While AI/ML techniques have been used to correct imperfect nuclear models, the reliability of the resulting predictions far outside the range of measured data has been seldom addressed.

The usual approach in AI/ML starts by defining a specific relationship, represented as an algorithm, between a set of input variables (features) and a set of output variables (targets). This algorithm usually contains a set of adjustable parameters. Typical examples include Gaussian Processes, Support Vector Machines and Bayesian Neural Networks. The training of the algorithm is nothing but the determination of its parameters by minimizing the distance between the algorithm’s output for a set of features-targets pairs and the physical world observations. In its most fundamental sense, this is similar to using an interpolating polynomial to reproduce a set of given points.

Similar to polynomial interpolation, machine learning training can thus result in overfitting. This can be mitigated by randomly splitting the available physical world data into training and testing sets. The algorithm is trained with a subset of all available data and its performance is tested against the remaining data. The training process is fine-tuned until the algorithm’s performance is comparable with the training and testing data. This process ensures that the algorithm will make reliable pre-
dictions with new feature values, as long as those values are within the training/testing domain (i.e. interpolating). However there is no guarantee of reliable predictions when the features are taken outside of the training/testing domain (i.e. extrapolating). This would be akin to training an image recognition software with pictures of dogs and wolves, and then using it to identify the animals in a picture of a lake with ducks and geese. For a simple and pedagogical demonstration of the limits to extrapolations with an artificial neural network, see [32].

Nuclides throughout the nuclear chart are regularly identified by their number of protons Z and number of neutrons N. Practitioners of machine learning in nuclear physics have, for the most part, adopted Z and N as the only features. Making predictions for neutron-rich nuclei with AI/ML algorithms therefore implies that the value of these features is extrapolated outside the training/testing region. We will show below that such extrapolations are highly questionable.

In this work we propose a practical approach to avoid extrapolations. The core of our solution consists in selecting features that stay within the training/testing domain when making predictions. Our procedure relies on three steps: (i) once a target has been established, define the training/testing region, (ii) define the prediction region, which will impact our choice of features for the training of the AI/ML algorithm. The choice of the prediction region depends on the problem at hand. For example, neutron-rich nuclei are essential for r-process simulations while a single isotopic sequence may be sufficient for nuclear structure studies, (iii) identify features with similar distributions in the training and prediction regions. This step will ensure that the algorithm is not extrapolated when making predictions.

Theoretical Framework - Density Functional Theory (DFT) provides a particularly advantageous framework to implement our solution [14]. For every nuclide characterized by Z and N, DFT calculations generate a large amount of properties, which range from physical observables like the binding energy or the neutron skin, to purely theoretical quantities such as the deformation parameters or the contribution to the binding energy from a specific term in an Energy Density Functional (EDF). Regardless of their nature, any of these properties can be used as a feature. We refer to the choice of Z and N only as the direct approach (DA). Including all available properties as features is called the all-properties approach (APA). Finally, to avoid extrapolations we propose to select as features those available DFT properties with similar distributions in the training and prediction regions. We refer to this option as the selected-properties approach (SPA) and will show that it is the only approach that shows consistently reliable performance.

In this work we evaluate how these three approaches improve the prediction of binding energies from two different parametrizations of the Skyrme energy density: SLY4 was optimized with an emphasis on neutron-rich nuclei and neutron matter properties [33] while UNEDF0 focuses on properties of spherical and deformed nuclei and approximates particle number restoration with the Lipkin-Nogami (LN) prescription [34]. In the Supplemental material, we also include results obtained with the UNEDF1_HFB parametrization, which adds excitation energies of fission isomers to the UNEDF0 optimization protocol but drops the LN prescription and the center-of-mass correction [35].

Our target is the model bias in DFT binding energies, that is, the difference between the theoretical and world binding energy,

$$\Delta B(Z, N) = B_{\text{EDF}}(Z, N) - B_{\text{phys}}(Z, N).$$

The bias is also referred to in the literature as the model discrepancy or residual [20, 22, 36]. The goal is to obtain a reliable AI/ML estimate of the model bias $$\Delta B_{\text{ML}}(\hat{f}(Z, N))$$, where $$\hat{f} = (f_1, \ldots, f_p)$$ is a particular set of p features. Once the bias is estimated it can be subtracted from the bare DFT calculations to obtain an improved set of binding energies

$$B^*(Z, N) = B_{\text{EDF}}(Z, N) - \Delta B_{\text{ML}}(\hat{f}(Z, N)).$$

To select the features with similar distributions in the training and prediction regions we quantify the level of similarity between two distributions from their overlap

$$BC(p, q) = \int \sqrt{p(x)q(x)}dx.$$
algorithm. In contrast, using $\beta_2$ ($BC = 0.94$) as a feature does not require an extrapolation since the distributions in the training and prediction regions are fairly similar. For the SPA, we use as features only those properties with $BC_i(p_{\text{train}}, q_{\text{pred}}) \geq c$, where $c$ is an arbitrary cut-off.

We use random forest regression to evaluate the performance of each feature selection method [46]. A random forest is a collection of decision trees where each tree is trained with a random subset of all training data. The output of the random forest is the average output of all decision trees. This technique avoids the overfitting that a few outliers can cause on a single decision tree. Additionally, random forests have few hyperparameters to tune, which makes them ideal for validation studies like the one presented here; for a complete introduction to random forest regression and classification we refer the interested readers to [47] and [48]. Although the work presented here uses random forests, the concept of feature selection for extrapolation can be used with any other AI/ML algorithm.

**Results** - We test the performance of our three feature selection approaches with available data. To this end, we define three prediction regions, which contain measured nuclei only, for which we will compare the estimated bias against the actual bias. The first region, dubbed time-wise split (TWS), consists of all the 70 even-even nuclei in the AME2020 data set that are not in the AME2003 data set [49, 50]. This simulates a realistic scenario in which predictions are tested over time as new measurements become available but does not test predictions far away from the training data. The second region, dubbed large-N split (LNS), consists of the up to 6 most neutron-rich even-even nuclei in each isotopic chain present in the AME2020 data set. This scenario tests predictions towards neutron-rich nuclei but still remains relatively close to the training data. The third region, called large-Z split (LZS), consists of all even-even nuclei in AME2020 with $Z \geq 76$. This scenario will test predictions far away from the training data. In each case the training/testing region consists of all even-even nuclei in AME2020 that are not present in the corresponding prediction region. The split between training and testing data is performed by randomly selecting 20% nuclei for testing and leaving the remaining 80% for training. Figure 2 shows the training, testing and prediction regions as described above. Note
that changing the prediction region changes the distribution of each property and it is thus necessary to recalculate the Bhattacharyya coefficient in order to re-select a new set of features with \( BC_i \geq c \). As mentioned earlier the cut-off \( c \) is somewhat arbitrary and an optimal value depends on the \( BC \) value of all available properties. We set a cut-off of 0.95 and 0.90 in the TWS and LNS cases respectively since most properties have large \( BC_i \) values: a lower cut-off would include most properties as features and would not allow distinguishing between the APA and SPA scenarios. For the LZS case we allow for a more “lenient” cut-off of 0.75 since there is a wider spread in the \( BC_i \) values among all properties.

We quantify the performance of a theoretical model for binding energies by calculating the root mean square error (rmse) \( \sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} [B_{\text{theo}}(Z_i, N_i) - B_{\text{exp}}(Z_i, N_i)]^2} \), where \( n \) is the total number of nuclides in a particular region. To estimate the quality of the corrected model after removing the bias (1), we then define the improvement index \( \eta(\sigma) = \sigma / \sigma_0 \) where \( \sigma_0 \) is the rmse of the original EDF model and \( \sigma \) the one after removing the bias. Values \( \eta < 1 \) mean that removing the bias has reduced the rmse; values \( \eta > 1 \) mean that this modification has increased the rmse. Figure 3 shows the improvement index for the two EDFs considered in this work after removing the estimated bias with the three different feature selection methods. As expected, \( \eta \ll 1 \) in the training region, irrespective of the EDF, the type of data analyzed and the feature selection method. In the testing region, the improvement is less pronounced but in general \( \eta \) stays lower than 0.5. Results in the prediction region are somewhat comparable to, or slightly worse than, in the testing region when looking at data not too far from the training region (blue and orange) as already noticed in [22]. When making predictions far from the training region (green), however, the improvement index takes values greater than 1 for the UNEDF0 functional for the DA and APA feature selection methods. This means that removing the AI/ML-estimated bias has in fact degraded the predictive power of the model. Similar results are obtained for the UNEDF1 functional [51]. Only by carefully selecting features with large \( BC \) coefficients can we avoid this situation and keep \( \eta < 1 \). The case of SLy4 sheds additional light on this mechanism. The error on binding energies comes mostly from not removing the effect of the electronic correction in the fit of the EDF. As a result, it is rather large and has a very clear dependency on \( Z \); see Fig. 2 (b) in the Supplemental material [51]. This trend can be easily learned by the AI/ML model, and the resulting bias estimate is robust (as a function of \( Z \)), accounts for most of the error and, therefore, still gives significant improvements in the prediction region, even far from training data. Put another way, feature selection is especially relevant for models with good predictive power with no clear the dependency of the model discrepancy on available properties.

**Conclusions** - We have analyzed the performance of three feature selection approaches to estimate the bias in DFT binding energies with two different EDFs. The performance of each approach was evaluated with three different splits of available data between training, testing and prediction. We showed that the direct approach, which uses \( Z \) and \( N \) as the only features, produces reliable predictions only for nuclei that are close to the available training data. When making predictions at values of \( Z \) far away from the training data, removing the estimated bias can in fact reduce the accuracy of the DFT prediction. By selecting features with similar distributions in the training and prediction regions, our approach eliminates the need to extrapolate features when making predictions. It is the only one that consistently showed improvement among all three EDFs considered here and in all three splittings between training, testing and prediction data. While we demonstrated our feature-selection technique on nuclear binding energies computed from DFT, it could be generalized to any other observable and any other global theoretical model as long as enough training data and potential features are available.
In fact, it is also potentially applicable in every scientific discipline (i) that rely on approximate theoretical models with adjustable parameters and (ii) where extrapolations outside measured data are needed.

Discussions with M. Grosskopf, E. Lawrence, S. Wild and J. O’Neal are gratefully acknowledged. Support for this work was partly provided through Scientific Discovery through Advanced Computing (SciDAC) program funded by U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research and Nuclear Physics. It was partly performed under the auspices of the US Department of Energy by the Lawrence Livermore National Laboratory (LLNL) Institutional Computing Grand Challenge program.

* rnavarroperez@sdsu.edu
† schunck1@llnl.gov

[1] S. A. Giuliani, Z. Matheson, W. Nazarewicz, E. Olsen, P.-G. Reinhard, J. Sadhukhan, B. Schuetrumpf, N. Schunck, and P. Schwertdfeger, Rev. Mod. Phys. 91, 011001 (2019).
[2] N. Schunck and D. Regnier, “Theory of nuclear fission,” (2022), arXiv:2201.02719 [nucl-th].
[3] J. J. Cowan, C. Sneden, J. E. Lawler, A. Aprahamian, M. Wiescher, K. Langanke, G. Martínez-Pinedo, and F.-K. Thielemann, Rev. Mod. Phys. 93, 015002 (2021).
[4] A. B. Balantekin, J. Carlson, D. J. Dean, G. M. Fuller, R. J. Furnstahl, M. Hjorth-Jensen, R. V. F. Janssens, B.-A. Li, W. Nazarewicz, F. M. Nunes, W. E. Ormand, S. Reddy, and B. M. Sherrill, Mod. Phys. Lett. A 29, 1430010 (2014).
[5] A. Bohr and B. Mottelson, Nuclear Structure, Vol. I, Single-Particle Motion (World Scientific, 1998).
[6] P. Ring and P. Schuck, The Nuclear Many-Body Problem, Texts and Monographs in Physics (Springer, 2004).
[7] S. Weinberg, Phys. Lett. B 251, 288 (1990).
[8] R. Machleidt and F. Sammarruca, Phys. Scr. 91, 083007 (2016).
[9] H.-W. Hammer, S. König, and U. van Kolck, Rev. Mod. Phys. 92, 025004 (2020).
[10] G. Hagen, T. Papenbrock, M. Hjorth-Jensen, and D. J. Dean, Rep. Prog. Phys. 77, 096302 (2014).
[11] J. Carlson, S. Gandolfi, F. Pedervia, S. C. Pieper, R. Schiavilla, K. E. Schmidt, and R. B. Wiringa, Rev. Mod. Phys. 87, 1067 (2015).
[12] H. Hergert, S. K. Bogner, T. Duguet, J.-P. Ebran, J. Engel, M. F. Michael, M. Kortelainen, and T. Nakatsukasa, Energy Density Functional Methods for Atomic Nuclei (Institute of Physics Publishing, 2019).
[13] P. Møller, A. J. Sierk, T. Ichikawa, and H. Sagawa, Atom. Data Nucl. Data Tab. 109, 1 (2016).
[39] B. K. Patra, R. Launonen, V. Ollikainen, and S. Nandi, Knowledge-Based Systems 82, 163 (2015).
[40] P. K. Singh, M. Sinha, S. Das, and P. Choudhury, Appl Intell 50, 4708 (2020).
[41] P. Van Molle, T. Verbelen, B. Vankeirsbilck, J. De Vylder, B. Diricx, T. Kimpe, P. Simoens, and B. Dhoedt, Neural Comput & Applic 33, 10259 (2021).
[42] R. Navarro Pérez, N. Schunck, R. D. Lasseri, C. Zhang, and J. Sarich, Computer Physics Communications 220, 363 (2017).
[43] M. Rosenblatt, The Annals of Mathematical Statistics 27, 832 (1956).
[44] E. Parzen, The Annals of Mathematical Statistics 33, 1065 (1962).
[45] A. Gramacki, Nonparametric Kernel Density Estimation and Its Computational Aspects (Springer, 2017).
[46] L. Breiman, Machine Learning 45, 5 (2001).
[47] G. Biau and E. Scornet, TEST 25, 197 (2016).
[48] R. Genuer and J.-M. Poggi, Random Forests with R (Springer Nature, 2020).
[49] G. Audi, A. H. Wapstra, and C. Thibault, Nucl. Phys. A The 2003 NUBASE and Atomic Mass Evaluations, 729, 337 (2003).
[50] A. H. Wapstra, G. Audi, and C. Thibault, Nucl. Sci. Eng. The 2003 NUBASE and Atomic Mass Evaluations, 729, 129 (2003).
[51] See Supplemental Material.
Supplemental Material to “Controlling extrapolations of nuclear properties with feature selection”

Rodrigo Navarro Pérez1,* and Nicolas Schunck2,†

1Department of Physics, San Diego State University, 5500 Campanile Drive, San Diego, California 02182-1233, USA
2Nuclear and Chemical Sciences Division, Lawrence Livermore National Laboratory, Livermore, California 94551, USA

(Dated: January 24, 2022)

CALCULATIONS OF HFBTHO MASS TABLES

Calculations of nuclear binding energies for even-even nuclei were performed with the code HFBTHO [1] using an axially-deformed basis with \( N_0 = 20 \) shells. The oscillator length was \( b_0 = \sqrt{\hbar^2 c^2 / (mc^2 \omega_0)} \) fm, where \( mc^2 = 931.499413 \) MeV, \( \omega_0 = 1.2 \times 41.1^{-1/3} \) and \( A \) is the mass number of the nucleus. Pairing correlations were treated at the HFB approximation (SLy4, UNEDF1HFB) and HFB with Lipkin-Nogami (UNEDF0). In all cases, the pairing interaction was a mixed surface-volume, density-dependent pairing force. In the calculation of the densities, only quasiparticles with energies less than \( E_{\text{cut}} = 60 \) MeV were included. In the case of the SLy4 interaction, the pairing strengths were set to \( V_0^{(n)} = -325.25 \) MeV and \( V_0^{(p)} = -340.0625 \) MeV. In every nucleus, the ground state was identified by performing calculations constrained on the axial quadrupole moment \( Q_2 \) for a few iterations before the constraint was released to identify the nearest unconstrained minimum. The value of this temporary constraint was \( Q_2 = \beta_2 \sqrt{\frac{5}{\pi} A^{5/3}} \) where \( \beta_2 \) is the basis deformation, which varied in the interval \([-0.2, 0.2]\).

TABLE I. Root mean square error for the three different EDFs considered in this work. The first line of each EDF is for the direct DFT calculation. The following three lines are for the modified values after subtracting the bias trained on different sets of features (DA, APA, and SPA). The data is split into training, testing and prediction following the time-wise split as shown in Panel (a) of Fig. 2. of the main manuscript. All quantities are in units of MeV.

| Model            | Training | Testing | Prediction |
|------------------|----------|---------|------------|
| UNEDF0           | 1.62     | 1.68    | 1.80       |
| UNEDF0 + DA      | 0.27     | 0.99    | 0.82       |
| UNEDF0 + APA     | 0.26     | 0.75    | 0.77       |
| UNEDF0 + SPA     | 0.28     | 0.91    | 0.83       |
| SLy4             | 3.95     | 4.22    | 5.15       |
| SLy4 + DA        | 0.28     | 1.13    | 0.88       |
| SLy4 + APA       | 0.28     | 0.95    | 0.80       |
| SLy4 + SPA       | 0.31     | 1.07    | 0.88       |
| UNEDF1HFB        | 2.85     | 3.45    | 3.10       |
| UNEDF1HFB + DA   | 0.22     | 0.90    | 0.77       |
| UNEDF1HFB + APA  | 0.26     | 0.92    | 0.79       |
| UNEDF1HFB + SPA  | 0.29     | 0.93    | 1.12       |

ROOT MEAN SQUARE ERRORS OF DFT PREDICTIONS

We list in tables I-III the root mean square deviation

\[
\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (B_{\text{theo}}(Z_i, N_i) - B_{\text{exp}}(Z_i, N_i))^2}
\]

for each functional (SLy4, UNEDF0, UNEDF1HFB). For each functional, the tables list the rmse without any bias correction and with bias correction: direct-approach (DA), all-properties approach (APA) or selected-properties approach (SPA). The rmse are presented in the training, testing and prediction regions separately.

TABLE II. Same as Table I but for the large-N split between training, testing and prediction sets as described in the text and shown in Panel (b) of Fig. 2.

| Model            | Training | Testing | Prediction |
|------------------|----------|---------|------------|
| UNEDF0           | 1.34     | 1.32    | 1.88       |
| UNEDF0 + DA      | 0.25     | 0.69    | 1.41       |
| UNEDF0 + APA     | 0.24     | 0.51    | 1.27       |
| UNEDF0 + SPA     | 0.29     | 0.67    | 1.20       |
| SLy4             | 2.86     | 2.79    | 5.21       |
| SLy4 + DA        | 0.26     | 0.83    | 1.90       |
| SLy4 + APA       | 0.28     | 0.77    | 2.13       |
| SLy4 + SPA       | 0.29     | 0.77    | 2.11       |
| UNEDF1HFB        | 2.67     | 3.09    | 3.26       |
| UNEDF1HFB + DA   | 0.22     | 0.70    | 1.49       |
| UNEDF1HFB + APA  | 0.25     | 0.71    | 1.57       |
| UNEDF1HFB + SPA  | 0.27     | 0.78    | 2.14       |

TABLE III. Same as Table I but for the large-Z split between training, testing and prediction sets as described in the text and shown in Panel (c) of Fig. 2.

| Model            | Training | Testing | Prediction |
|------------------|----------|---------|------------|
| UNEDF0           | 1.73     | 1.78    | 1.27       |
| UNEDF0 + DA      | 0.32     | 0.92    | 2.43       |
| UNEDF0 + APA     | 0.26     | 0.68    | 1.64       |
| UNEDF0 + SPA     | 0.27     | 0.73    | 1.16       |
| SLy4             | 2.92     | 2.61    | 6.37       |
| SLy4 + DA        | 0.33     | 1.06    | 3.15       |
| SLy4 + APA       | 0.30     | 0.85    | 3.03       |
| SLy4 + SPA       | 0.40     | 0.98    | 3.40       |
| UNEDF1HFB        | 3.17     | 3.27    | 2.43       |
| UNEDF1HFB + DA   | 0.25     | 0.89    | 3.51       |
| UNEDF1HFB + APA  | 0.26     | 0.99    | 1.24       |
| UNEDF1HFB + SPA  | 0.52     | 1.61    | 1.68       |
FIG. 1. Value of the estimated AI/ML bias across the entire mass table for the SLy4, UNEDF0 and UNEDF1\textsubscript{HFB} parameterizations of the Skyrme functional and the three feature selection modes.

**ESTIMATED BIAS**

Figure 1 shows the estimated bias for the SLy4, UNEDF0 and UNEDF1\textsubscript{HFB} functionals across the entire mass table after using all even-even nuclei in the AME2020 data set for training and testing. In the direct approach (DA, left column), the bias shows very little structure as soon as we depart significantly from the training/testing region (marked by the contour in each plot. In the case of SLy4, where the actual discrepancy is very large and grows approximately like \( Z^2 \), this lack of structure does not have major consequences. In the case of the UNEDF0 functional, which reproduces binding energies reasonably well, it implies that the AI/ML algorithm has not learned any dependency of the bias as a function of \( Z \) and \( N \) outside of the training/testing region. The APA approach improves things significantly, since structures associated with magic numbers become more visible – although for UNEDF1\textsubscript{HFB} the differences between DA and APA are still small. In the selected approach (SPA), the bias has a clear, non-trivial dependency on \( Z \) and \( N \) with neutron shell closures, in particular, clearly visible: the AI/ML algorithm has learned physics trend from the available data.

Figure 2 provides additional insights on the differences in the improvement index \( \eta \) in the prediction region. The upper row of the figure shows the residuals \( E_{\text{th}} - E_{\exp} \) for isotonic chains as a function of proton numbers for UNEDF0, panel (a), and SLy4, panel (b). While both sets of results exhibit the “traditional” large biases around closed shells, the average error for the UNEDF0 is approximately centered around 0, as can be seen in panel (c). In contrast, the error for SLy4 is not only much larger, the overall error is clearly correlated with the value of the proton number \( Z \). Panels (c) and (d) show the distributions of the residuals: for the UNEDF0 functional, this distribution is approximately centered around 0, while for SLy4, it is centered at around 3 MeV. The dependency of the residual on proton number is simple and robust: it is in fact a consequence of the fact that in
FIG. 2. Residuals of nuclear binding energies as for isotonic sequences, as a function of proton number. Panel (a): results for UNEDF0; panel (b): results for SLy4. Panels (c) and (d) show the corresponding distributions of the residuals.

the fit of the SLy4 functional, nuclear masses were not corrected for the electronic binding energy, which grows approximately like $Z^2$. As a result, the total discrepancy for SLy4 can be written

$$\Delta E = \Delta E_{\text{elec}} + \Delta E_{\text{nucl}}. \quad (1)$$

It is the sum of a discrepancy originating from the electronic energy correction, $\Delta E_{\text{elec}}$, and an additional term accounting for every other missing nuclear physics, $\Delta E_{\text{nucl}}$. In heavier nuclei, $\Delta E_{\text{elec}} \gg \Delta E_{\text{nucl}}$. The AI/ML model can easily learn the trend of $\Delta E_{\text{elec}}$ from every data set we considered in this work. Most importantly, this trend remains valid at large $Z$ values, which means that the gain from removing the bias will remain significant in the prediction region, contrary to UNEDF0.

Discussions with M. Grosskopf, E. Lawrence, S. Wild and J. O’Neal are gratefully acknowledged. Support for this work was partly provided through Scientific Discovery through Advanced Computing (SciDAC) program funded by U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research and Nuclear Physics. It was partly performed under the auspices of the US Department of Energy by the Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Computing support for this work came from the Lawrence Livermore National Laboratory (LLNL) Institutional Computing Grand Challenge program.

* rnavarroperez@sdsu.edu
† schunck1@llnl.gov
[1] R. Navarro Pérez, N. Schunck, R. D. Lasserri, C. Zhang, and J. Sarich, Computer Physics Communications 220, 363 (2017).