Journal of Applied Physics

Toward machine learning for microscopic mechanisms: A formula search for crystal structure stability based on atomic properties

Cite as: J. Appl. Phys. **131**, 215703 (2022); https://doi.org/10.1063/5.0088177
Submitted: 14 February 2022 • Accepted: 16 May 2022 • Published Online: 07 June 2022

Udaykumar Gajera, Loriano Storchi, Danila Amoroso, et al.

Journal of Applied Physics ARTICLE scitation.org/journal/jap

Toward machine learning for microscopic mechanisms: A formula search for crystal structure stability based on atomic properties

Cite as: J. Appl. Phys. **131**, 215703 (2022); doi: 10.1063/5.0088177
Submitted: 14 February 2022 · Accepted: 16 May 2022 ·
Published Online: 7 June 2022

Udaykumar Gajera,[1,2,a)] Loriano Storchi,[3,b)] Danila Amoroso,[1,4,c)] Francesco Delodovici,[1,5] and Silvia Picozzi[1]

AFFILIATIONS

[1]Consiglio Nazionale delle Ricerche, CNR-SPIN c/o Università "G. D'Annunzio," 66100 Chieti, Italy
[2]Chemistry Department, University of Turin, via Pietro Giuria, 7, 10125 Torino, Italy
[3]Dipartimento di Farmacia, Università degli Studi G. D'Annunzio, 66100 Chieti, Italy
[4]NanoMat/Q-mat/CESAM, Université de Liège, B-4000 Liège, Belgium
[5]Université Paris-Saclay, CentraleSupélec, CNRS, Laboratoire SPMS, 91190 Gif-sur-Yvette, France

[a)]**Author to whom correspondence should be addressed:** uday.gajera@edu.unito.it
[b)]loriano@storchi.org. **URL:** https://www.storchi.org/
[c)]danila.amoroso@uliege.be

ABSTRACT

Machine-learning techniques are revolutionizing the way to perform efficient materials modeling. We here propose a combinatorial machine-learning approach to obtain physical formulas based on simple and easily accessible ingredients, such as atomic properties. The latter are used to build materials features that are finally employed, through linear regression, to predict the energetic stability of semiconducting binary compounds with respect to zinc blende and rocksalt crystal structures. The adopted models are trained using a dataset built from first-principles calculations. Our results show that already one-dimensional (1D) formulas well describe the energetics; a simple grid-search optimization of the automatically obtained 1D-formulas enhances the prediction performance at a very small computational cost. In addition, our approach allows one to highlight the role of the different atomic properties involved in the formulas. The computed formulas clearly indicate that "spatial" atomic properties (i.e., radii indicating maximum probability densities for *s*, *p*, *d* electronic shells) drive the stabilization of one crystal structure with respect to the other, suggesting the major relevance of the radius associated with the *p*-shell of the cation species.

I. INTRODUCTION

Modeling material properties with high accuracy and low computational cost is one of the grand-challenges in materials science and engineering. The development of *ab initio* methods has provided accurate tools for material properties prediction and their further optimization; nevertheless, one disadvantage of approaches relying only on first-principles simulations is the high cost required in terms of computational resources and simulation time. In recent years, the continuous growth of available computational power[1] has stimulated scientists to move in the direction of high-throughput simulations.[2–10] Along this line, open access databases, such as OQMD,[11,12] NOMAD,[13,14] Aflowlib,[15] C2DB,[16,17] QPOD,[18] Materials Project,[19] Materials Cloud,[20] and related AiiDA,[21,22] provide researchers with a huge collection of basic first-principles results. A large amount of *ab initio* data is thus available, which can be used for deeper analyses and studies, provided one can count on proper tools to extract relevant information out of them.

In the last few years, materials scientists have developed different machine-learning (ML) methods to rationalize the data analysis.[23–33] Each method has its own specific advantages and limitations. Methods, such as random forest[34] or neural network (NN),[35] which is mainly behind the Deep Learning (DL), are very efficient[36] but not always transparent, partially blurring the

comprehension of the role played by the input variables in the final results. Nonetheless, over the last few decades, improvements toward the interpretability of such "black-box" ML models have been made through additional methodologies,[37] such as model-agnostic methods, which in turn are divided into global and local interpretation techniques (see Ref. 38 and references therein). For instance, out of various global methods, we can cite: the *permutation feature importance*,[39,40] which associates each feature importance values depending on how much the model error increases when its values are shuffled; the *functional decomposition*,[41] which decomposes the complex prediction function into smaller parts; and the *global surrogate*,[42] which replace the original model with a simpler model that can be more easily interpreted. On the other hand, among the local methods, we can cite: the *local surrogate models* (LIME),[43] which replaces the complex model with a locally interpretable surrogate model, and the *SHapley Additive exPlanations* (SHAP),[44] which is based on Shapley values and computes the contribution of each feature to the prediction. Also, in the specific case of DL, which structures algorithms in multiple layers to create "artificial neural networks," thus enhancing the complexity in the prediction's interpretation, other specific interpretation methods have been proposed,[45–47] in addition to the already cited model-agnostic ones.

However, when targeted case studies allow, the easiest way to achieve a deeper understanding of machine-learning results is to rely on interpretable models, such as linear regression (LR),[48–51] logistic regression,[52] and decision trees.[53] This can apply here to our target case, i.e., the prediction of the difference in total energy (ΔE) between rocksalt (RS) and zinc blende (ZB) crystal structures in semiconductor binary AB compounds, a prototypical case in materials science. Accordingly, being our goal the creation of formulas linking the target label (ΔE) to a set of basic atomic features, we selected the LR algorithm both in its one-dimensional and multi-dimensional forms.

In closer detail, we here propose a ML-based approach to build sets of features (or descriptors) starting from a given set of basic variables (e.g., atomic properties), which are subsequently used to construct LR models (or formulas). The final outcome of our procedure is a transparent formula, not necessarily of easy mathematical formulation, but revealing which part of the input mostly affects the output,[54] i.e., allowing the identification of the main driving physical features.

To test our method, we target a prototypical case in materials science: indeed, inspired by the original work of Ghiringhelli *et al.*,[55] we optimized our models to predict the difference in energy between RS and ZB; from that optimization, a classification of the most stable crystal structure between RS and ZB for semiconductor AB binary compounds naturally derives.

To identify useful features, we generate combinations of basic atomic properties (i.e., the independent variables in our approach) of the material constituents through a combinatorial approach.[56] We then carry out an analysis of the emerging best-performing formulas, identifying the role of specific atomic features in determining the final stabilization of the crystal structure. Finally, we test the predictive capability of the obtained formulas by applying them to "new" compounds (i.e., outside the dataset used for training the model), finding an overall satisfactory agreement with first-principles results. As already mentioned, our approach is similar to what was originally proposed by Ghiringhelli *et al.*,[55] though with some differences and further extensions, which will be carefully discussed in what follows. Let us also remark that large packages are nowadays available to the scientific community, already providing advanced and well-tested features for use in ML applied to materials science.[57,58] However, we here strictly followed the spirit of Ghiringhelli *et al.*[55] and, therefore, chose their same atomic features, as detailed below.

II. METHODOLOGY

The approach we present here can be regarded as a combinatorial machine-learning: a set of basic atomic properties (APs, listed in Table S2 of the supplementary material) are randomly combined (though under certain initial constraints detailed below) to build a set of material features (MFs). The generated features are then used to train a LR model, where the energy difference between rocksalt and zinc blende structures is the dependent variable (i.e., the label). Then, we select the best-performing model according to standard performance metrics, such as the root mean squared error (RMSE). The final result of this procedure is a "formula," which is a concise and clear representation of the relationship between the used atomic properties and the energy difference between RS and ZB phases. In the following, we describe in detail the different steps of our approach.

A. Dataset preparation and materials

As mentioned, we aim at predicting the total energy difference ($\Delta E = E^{RS} - E^{ZB}$) between RS and ZB phases of cubic crystal structures for 82 semiconductor binary AB compounds (the dataset is reported in Table S2 of the supplementary material). We employed total energies reported in Ref. 55, which were calculated through density functional theory (DFT)[59,60] within the local density approximation (LDA[61]).

The construction of the material features is based on primary atomic properties of the constituents, also taken from Ref. 55. To facilitate the physical interpretation of each MF, the APs are subdivided into two different kinds: (i) "energy" properties, including the highest occupied Kohn–Sham level (HOMO), the lowest unoccupied Kohn–Sham level (LUMO), ionization potential (IP), electron affinity (EA) and (ii) "spatial" properties, including r_s, r_p, and r_d, i.e., the radii where the radial probability density of the valence *s*, *p*, and *d* orbitals, respectively, reaches its maximum.4

B. Formula construction

We rely on the LR[48,49] approach to obtain a direct interpretation of the dependent and independent variables. The construction of a useful LR model can become troublesome, requiring a linear dependence between features. In Ref. 55, the authors implemented an automated feature selection method employing the LASSO regression analysis method.[55,63] In our work, we use a combinatorial approach to generate the dependent variable (material features) to be used within the linear equations and thus to finally obtain the formulas.

In Fig. 1, we illustrate the workflow of the formula generation and selection using LR. The process starts with the selection of the APs to be combined. Afterward, we choose prototype functions that are simple analytical operations applied to the APs. In our case, we selected five prototype functions, $f(x)$, namely,

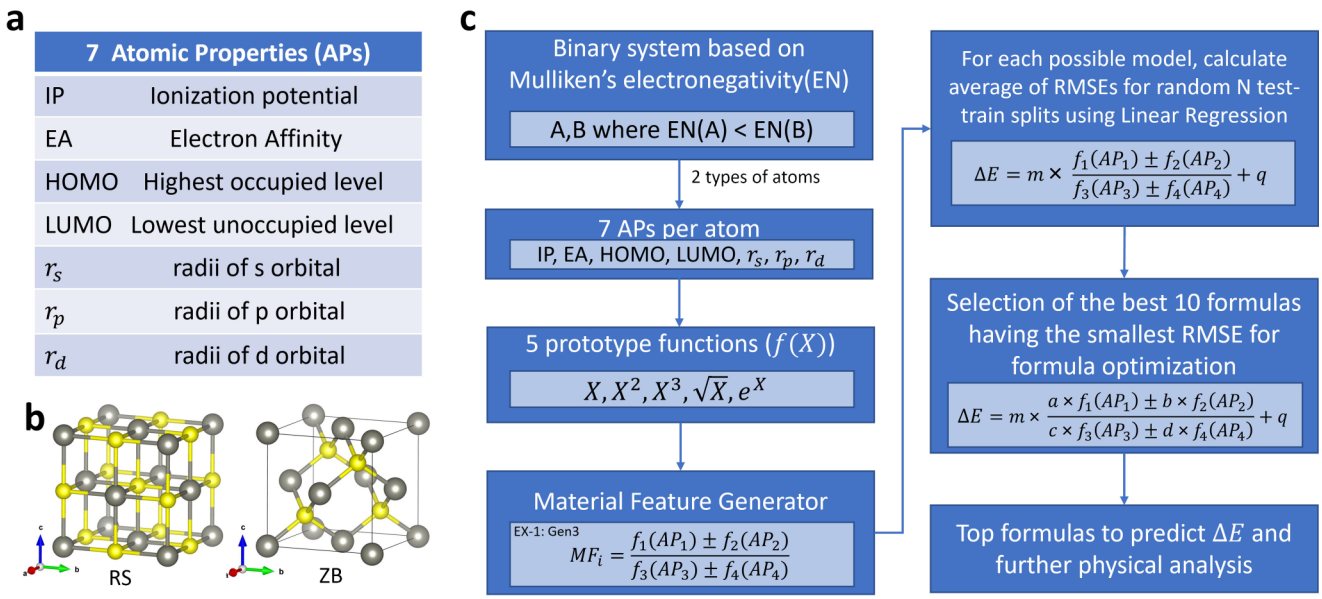

FIG. 1. (a) Basic atomic properties (APs) used to construct the material features. (b) Crystal structures of RS and ZB (plot made using the VESTA tool).[62] Gray (yellow) spheres represent A (B) atoms. (c) Workflow for formula construction, machine-learning methodology, validation, and MF selection procedures. In the AB compounds, A is the atom with the lowest electronegativity.

x, x^2, x^3, $\sqrt{x}$, e^x, where x is an AP. Then, we obtain the final set of MFs by combining different prototype functions via the combinatorial approach (see, for instance, Ref. 56) and applying the following additional set of rules:

- *GEN*1: combine two prototype functions in the numerator, forcing them to belong to the same kind of APs, which is both "spatial"-like or both "energy"-like; one prototype function is at the denominator with the only constraint to be non-zero, such as

$$MF = \frac{f_1(AP_1) \pm f_2(AP_2)}{f_3(AP_3)}. \quad (1)$$

- *GEN*2: combine two prototype functions with the same kind of APs at the numerator and a single prototype function at the denominator with an argument of a different kind with respect to the numerator ones. For instance, if AP_1 in $f_1(AP_1)$ and AP_2 in $f_2(AP_2)$ are "energy" terms (i.e., *EA* or *HOMO*), then AP_3 must be a "spatial" term (i.e., r_p),

$$MF = \frac{f_1(AP_1) \pm f_2(AP_2)}{f_3(AP_3)}. \quad (2)$$

- *GEN*3: combine two prototype functions at both the numerator and denominator without any constraints,

$$MF = \frac{f_1(AP_1) \pm f_2(AP_2)}{f_3(AP_3) \pm f_4(AP_4)}. \quad (3)$$

- *GEN*4: combine two prototype functions with the same physical dimensions at both the numerator and denominator,

$$MF = \frac{f_1(AP_1) \bigstar f_2(AP_2)}{f_3(AP_3) \bigstar f_4(AP_4)}, \quad (4)$$

where $\bigstar = + - \times \div$.

Each one of these set of rules corresponds to a different MF generator.

From the implementation point of view, each generator is a Python[64] function that produces a set of strings. Therefore, we can easily exploit the Python capability to parse a source code and run a Python expression (code) within a program[65] to compute all the MFs' values starting from the generated sets of strings. This allows for an easy implementation and plugin of other generators, as well as to easily adopt different sets of atomic properties, leaving the workflow unchanged: a new generator can be introduced implementing a Python function returning a list of strings, each one being a valid MF.

Finally, in order to choose the optimal formula, we build a LR model for each of the generated MF. To practically select the best model, i.e., the "best formula," we randomly split the full dataset into 90% as a training set to train/initialize the model and 10% as a test set to check model's performance. We perform this random splitting N times (with $N = 150$) for each model, and we calculate the RMSE from the test set for each run. Afterward, we again verify the top ten resulting best formulas with a higher value of training set and test set splitting, with $N = 1000$. We average it over all N splitting, and we obtain avg(*RMSE*), as reported in the tables of Sec. III.

We mention that different metrics for evaluating regression models can lead to different formula ranking. In this work, we rank the obtained models based on the lowest avg(*RMSE*) for direct comparison with a previous work.[55]

C. Formula optimization

In order to further improve the performance of our models, we introduce an additional step, which we refer to as "formula optimization." Specifically, we focus on the top ten formulas obtained using each generator and the subsequent LR, as described in Sec. II B. After that, we use a grid search to find the relative weights of each prototype function of the atomic properties [i.e., each $f_i(AP_i)$] within the formula. A first grid search ranging between −1 and 1 with the increasing step of 0.1 is used simultaneously for all the weight coefficients (i.e., an exhaustive search through the specified subset of values for a, b, c, d coefficients is simultaneously performed). We multiply each $f_i(AP_i)$ of the formula by the weight coefficient, and we optimize the final RMSE value. Once the procedure finds a set of optimal weight coefficients, two subsequent grid searches, with reduced incremental step values (0.01 and 0.001, respectively) and a range of search are performed to obtain the final set of refined weight coefficients. Of note, for each set of weight coefficients generated during the grid search, we also run the linear regression. Thus, we are performing a proper formula optimization, as at each step of the grid search, we are updating both the weight coefficients as well as the slope and intercept coming from the LR. In addition, it is important to underline that we made sure to find the global minimum solution when analyzing the N–D maps over the phase space of the parameters (in the specific case of *GEN*2, we also double-checked the optimization results through an analytical minimization approach via Mathematica[66,67]).

To further clarify the procedure, we show here an exemplary equation,

$$\Delta E = m \times \frac{a \times f_1(AP_1) \bigstar b \times f_2(AP_2)}{c \times f_3(AP_3) \bigstar d \times f_4(AP_4)} + q, \tag{5}$$

where ΔE is the targeted material feature (MF); a, b, c, d denote the weight coefficients scanned during the grid search; $f_1(AP_1), f_2(AP_2), f_3(AP_3), f_4(AP_4)$ are the prototype functions build on the primary atomic properties AP_i; and m and q are the slope or angular coefficient and intercept, respectively, recursively determined upon LR.

In Table II, we report the optimized, best-performing formula from the different generators; the top 10 formulas are reported in Table S1 of the supplementary material.

To benchmark our grid search, we also used automated coefficient-optimizing methods: Nelder–Mead,[68] conjugate gradient (CG),[69] Broyden–Fletcher–Goldfarbnno (BFGS),[70] and truncated Newton method (TNC).[71] Although the resulting sets of coefficients are different in terms of single values with respect to those obtained via the grid search, the ratios between them are almost preserved as well as the associated RMSE. In particular, for the case of *GEN*1 and *GEN*2, the ratio between the numerator coefficients a and b is preserved, and for *GEN*3 and *GEN*4, also, the denominator coefficient ratio, between c and d, is preserved. In Fig. S3 of the supplementary material, we show the evolution of the RMSE and different ratios for different methods using 1D feature generated by *GEN*3.

Finally, we would like to underline that the whole procedure, i.e., formula construction and its optimization, is not too expensive from a computational point of view. Indeed, as reported in Table S4 of the supplementary material, for almost all the generators, the whole computation can be performed in less than 4 h on a standard PC. Only GEN3, where 1 091 200 different formulas are generated and evaluated, is more time consuming (i.e., almost 15 h are needed). However, we underline that the 1D formula construction procedure can be easily parallelized in order to drastically reduce its computational burden.

D. Higher-dimensional features

Following the idea of Ghiringhelli *et al.* (see Ref. 55), we also build 2D and 3D formulas as follows: we combined in all possible ways two or three different 1D MFs extracted from the best 1000 ones and checked the avg(*RMSE*) using multiple LR for N test-train set splits. Thus, the final equations, which relate ΔE to the

TABLE I. 1D formulas, along with related statistics: avg(*RMSE*) denotes the root mean squared error for average over 1000 random train-test splits of dataset. Instead, the RMSE is the root mean squared error for the entire dataset as training and test. Similarly, the R^2 values are calculated considering the entire dataset, and they show the quality of fit between predicted and actual values. The success rate (in percent) shows how many RS or ZB phases out of 82 have been correctly identified by the descriptor. The "Generator type" column indicates the different generators used to produce the corresponding descriptor. RMSEs are in eV.

Formula	avg (RMSE)	RMSE	R^2	Success rate (%)	Generator type
$0.117 \times \frac{EA(B)-IP(B)}{r_p(A)^2} - 0.342$	0.1455	0.1423	0.89	89	1D descriptor[55]
$-0.751 \times \frac{r_p(B)^3-\exp[r_s(B)]}{r_p(A)^2} - 0.317$	0.1296	0.1193	0.92	90	*GEN*1
$0.285 \times \frac{\sqrt{\lvert IP(B)\rvert}+\sqrt{\lvert EA(A)\rvert}}{r_p(A)^2} - 0.387$	0.1367	0.1309	0.91	91	*GEN*2
$0.774 \times \frac{r_p(B)+\sqrt{\lvert r_d(A)\rvert}}{r_p(A)^3+r_p(B)^3} - 0.303$	0.0995	0.0963	0.95	94	*GEN*3
$1.155 \times \frac{r_s(B)+r_s(A)}{r_p(B)^3+r_p(A)^3} - 0.368$	0.1103	0.1058	0.94	96	*GEN*4

TABLE II. 1D formulas after the optimization step, along with related statistics. Notation as in Table I. RMSEs are in eV.

Formula	avg (RMSE)	RMSE	R^2	Success rate (%)	Generator type
$0.127 \times \frac{0.800 \times EA(B) - 1.000 \times IP(B)}{1.110 \times r_p(A)^2} - 0.352$	0.1457	0.1419	0.89	89	1D descriptor[55]
$-1.870 \times \frac{0.801 \times \sqrt{r_p(B)} - 0.606 \times \exp[r_p(A)]}{1.010 \times r_p(A)^3} - 0.968$	0.1191	0.1143	0.93	91	*GEN*1
$0.477 \times \frac{0.876 \times \sqrt{\lvert HOMO(B)\rvert} + 0.468 \times \sqrt{\lvert LUMO(B)\rvert}}{1.110 \times r_p(A)^2} - 0.372$	0.1340	0.1296	0.91	91	*GEN*2
$1.609 \times \frac{0.642 \times r_p(B) + 0.502 \times \sqrt{\lvert r_d(A)\rvert}}{1.170 \times r_p(A)^3 + 1.170 \times r_p(B)^3} - 0.309$	0.0991	0.0961	0.95	94	*GEN*3
$1.207 \times \frac{0.878 \times r_s(B) + 0.200 \times r_p(A)}{0.512 \times r_p(B)^3 + 0.610 \times r_p(A)^3} - 0.359$	0.1045	0.1016	0.94	99	*GEN*4

basic atomic features, are written as follows:

$$\Delta E = m_1 \times \frac{a_1 \times f_1(AP_1) \bigstar b_1 \times f_2(AP_2)}{c_1 \times f_3(AP_3) \bigstar d_1 \times f_4(AP_4)} + m_2 \times \frac{a_2 \times f_5(AP_5) \bigstar b_2 \times f_6(AP_6)}{c_2 \times f_7(AP_7) \bigstar d_2 \times f_8(AP_8)} + q \quad (6)$$

for the general 2D formulas and

$$\Delta E = m_1 \times \frac{a_1 \times f_1(AP_1) \bigstar b_1 \times f_2(AP_2)}{c_1 \times f_3(AP_3) \bigstar d_1 \times f_4(AP_4)} + m_2 \times \frac{a_2 \times f_5(AP_5) \bigstar b_2 \times f_6(AP_6)}{c_2 \times f_7(AP_7) \bigstar d_2 \times f_8(AP_8)} + m_3 \times \frac{a_3 \times f_9(AP_9) \bigstar b_3 \times f_{10}(AP_{10})}{c_3 \times f_{11}(AP_{11}) \bigstar d_3 \times f_{12}(AP_{12})} + q \quad (7)$$

for the 3D ones. The comparison between performance is discussed in Sec. III.

E. Test of predictive power of the ΔE formula for novel AB compounds

After obtaining the optimized 1D formulas for ΔE in the case of AB compounds, we aimed at further verifying their validity and predictive power by considering additional AB systems (i.e., which were not originally included in the ML training set) and by comparing values obtained from the ML-predicted ΔE formula with the corresponding *ab initio* calculated values. In closer detail, we focused on different alloys, obtained by changing, respectively, the concentration of A-site atoms, such as $[A_xA'_{1-x}]B$, and of B-site atoms, such as $A[B_xB'_{1-x}]$. Accordingly, one can test the efficiency of the formulas by checking the energy difference for intermediate concentrations as obtained from optimized 1D formulas and compare their trend with respect to first-principles results. To this end, *ab initio* electronic-structure simulations were carried out within DFT and LDA functional. Calculations were performed using the VASP[72–74] code, employing a $8 \times 8 \times 8$ k-mesh for the Brillouin zone sampling. We verified that the results obtained with the pseudopotential VASP for the parent binary compounds were consistent with those reported by Ghiringhelli *et al.*, calculated with the all-electron FHI-aims code.[75] For simulations at different concentrations, we adopted the so-called virtual crystal approximation (VCA) based on virtual atoms interpolating between the real constituent atoms.[76,77] However, as well known from the literature, the VCA approach neglects some effects, such as local distortions around atoms and, as such, should not be expected to reproduce fine details of disordered alloy properties.[78] Accordingly, in some cases (i.e., for $Mg_xCa_{1-x}Se$ alloys), in order to mimic disordered structures with an improved accuracy, we calculated total energies using supercell structures, rather than using the VCA method on primitive unit cells. Specifically, the considered supercell is the cubic unit cell composed of four AB formula units with planes of cations alternating along the **c** direction (see Fig. S4 in the

TABLE III. 2D formulas, along with related statistics. Notation as in Table I. RMSEs are in eV.

Formula	avg (RMSE)	RMSE	R^2	Success rate (%)	Generator type
$0.113 \times \frac{EA(B) - IP(B)}{r_p(A)^2} - 1.558 \times \frac{\lvert r_s(A) - r_p(B)\rvert}{\exp[r_s(A)]} - 0.133$	0.1041	0.0988	0.95	96	2D descriptor[55]
$-0.342 \times \frac{r_p(B)^3 - \exp[r_p(A)]}{r_p(A)^3} - 1.042 \times \frac{r_p(A)^2 - \sqrt{\lvert r_d(A)\rvert}}{\exp[r_p(A)]} - 0.062$	0.0989	0.0944	0.95	89	*GEN*1
$-0.081 \times \frac{IP(B) + \sqrt{\lvert IP(A)\rvert}}{r_p(A)^3} - 0.001 \times \frac{r_s(A)^3 - \sqrt{r_d(A)}}{\exp(HOMOKS(A))} - 0.062$	0.1163	0.1100	0.93	86	*GEN*2
$-1.175 \times \frac{r_p(A) - \sqrt{\lvert r_d(A)\rvert}}{r_s(B)^3 + r_p(A)^3} + 0.513 \times \frac{r_s(B) + \sqrt{\lvert r_p(B)\rvert}}{r_p(B)^3 + r_s(A)^3} - 0.250$	0.0911	0.0878	0.96	87	*GEN*3
$0.618 \times \frac{r_d(A)/r_p(B)}{r_p(A)^3 \times \sqrt{r_d(A)}} + 1.097 \times \frac{r_p(A) \times \sqrt{\lvert r_p(B)\rvert}}{r_p(B)^3 + r_p(A)^3} - 0.384$	0.0995	0.0955	0.95	92	*GEN*4

TABLE IV. 3D formulas, along with related statistics. Notation as in Table I. RMSEs are in eV.

Formula	avg (RMSE)	RMSE	R^2	Success rate (%)	Generator type
$0.108 \times \frac{EA(B)-IP(B)}{r_p(A)^2} - 1.806 \times \frac{\lvert r_s(A)-r_p(B)\rvert}{\exp[r_d(A)]} - 3.782 \times \frac{\lvert r_p(B)-r_s(B)\rvert}{\exp[r_d(A)]} - 0.023$	0.0818	0.0756	0.97	93	3D descriptor[55]
$0556 \times \frac{r_p(B)^3-\exp[r_p(A)]}{r_p(A)^3} + 0.364 \times \frac{r_p(A)^2-\sqrt{\lvert r_d(B)\rvert}}{\exp[r_p(A)]} - 0.124 \times \frac{r_p(B)^2-\sqrt{\lvert r_d(A)\rvert}}{r_p(A)^3} - 1.87$	0.1003	0.0933	0.95	90	*GEN1*
$-0.056 \times \frac{(LUMO(A)+HOMO(B)}{r_p(A)^3} + 0.266 \times \frac{\sqrt{\lvert EA(B)\rvert}+\exp(EA(B))}{r_s(A)^3} - 0.016 \times \frac{HOMO(A)-\exp(LUMO(B))}{(r_p(A)^3)} - 0.310$	0.1300	0.1205	0.92	91	*GEN2*
$-0.885 \times \frac{r_p(B)-\exp[r_p(A)]}{r_p(A)^3+r_p(A)^3} - 0.417 \times \frac{r_s(A)-\exp[r_s(B)]}{r_s(A)^3+r_p(B)^3} - 0.579 \times \frac{r_p(A)-\sqrt{\lvert r_d(A)\rvert}}{r_p(B)^2+r_s(A)^3} - 0.616$	0.0875	0.0834	0.96	98	*GEN3*
$0.635 \times \frac{\sqrt{IP(B)}/\sqrt{IP(A)}}{r_p(A)^3+r_p(B)^3} + 0.730 \times \frac{r_p(B)\times\sqrt{\lvert r_d(A)\rvert}}{r_p(A)^3+r_p(B)^3} + 0.038 \times \frac{IP(A)^2-EA(A)^2}{\exp(r_p(A))\times\exp(r_d(B))} - 0.358$	0.0989	0.0919	0.96	93	*GEN4*

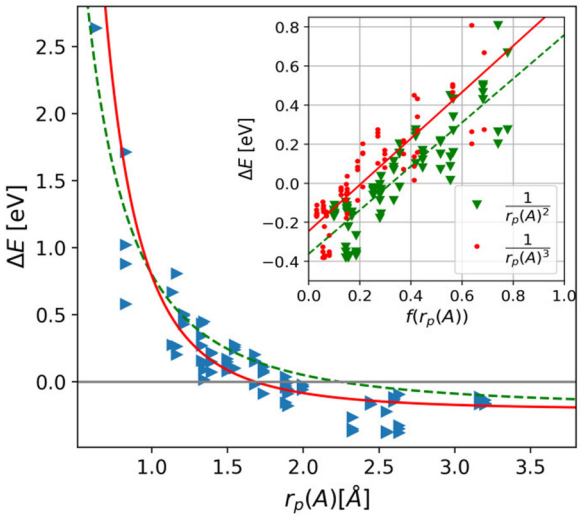

FIG. 2. Energy difference between rock salt and zinc blende, ΔE (in eV), as a function of $r_p(A)$ for different binary compounds (blue triangles). Data fit functions are also shown, using proportionality to $r_p(A)^{-2}$ and $r_p(A)^{-3}$ via a green dashed line and a red straight line, respectively. The inset shows the linear fit of ΔE for $f(r_p(A)) \in [0:1]$ as a function of $1/r_p(A)^2$ (green triangles) and $1/r_p(A)^3$ (red dots). Resulting slope (m) and intercept (q) are, respectively, 1.120 (eV Å^2), −0.360 (eV) for $1/r_p(A)^2$ and 1.184 (eV Å^3), −0.240 (eV) for $1/r_p(A)^3$.

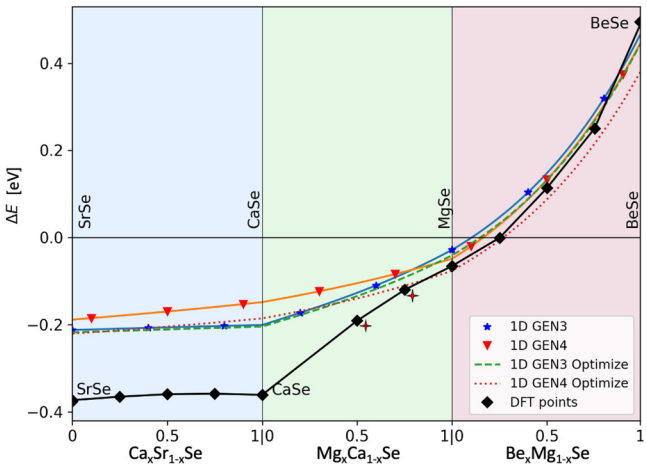

FIG. 3. Total energy difference ΔE as a function of concentration (x) for $[Ca_xSr_{1-x}]Se$, $[Mg_xCa_{1-x}]Se$, and $[Be_xMg_{1-x}]Se$ alloys, highlighted in blue, green, and pink regions, respectively. Energy differences are predicted using original and optimized 1D descriptors constructed using *GEN3* and *GEN4* and verified using DFT (black line with diamond points) within VCA. For an improved accuracy, the two asterisk-highlighted intermediate points in the $[Mg_xCa_{1-x}]Se$ region are calculated using the supercell approach rather than VCA.

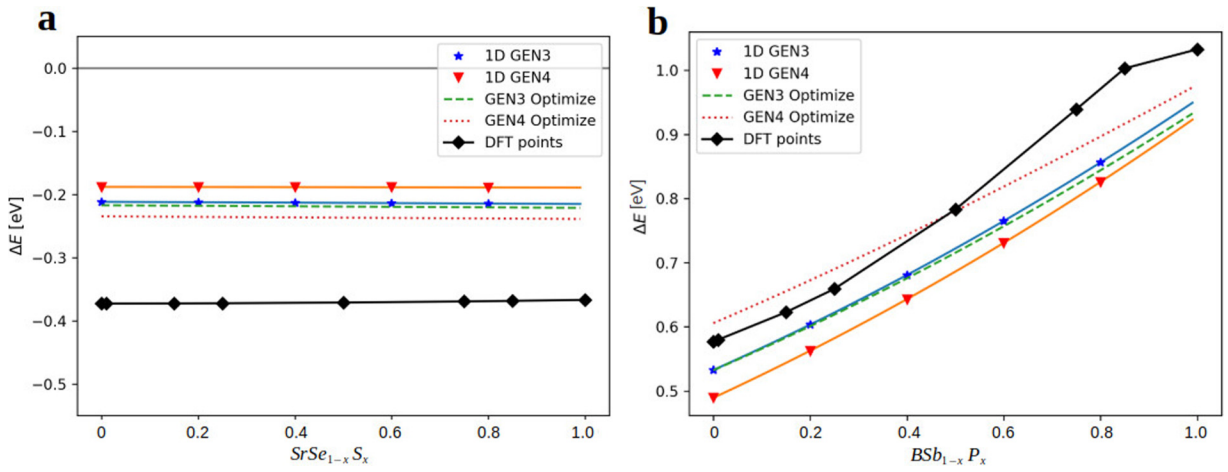

FIG. 4. Total energy difference ΔE as a function of concentration (x) for $Sr[S_xSe_{1-x}]$, see panel (a), and $B[P_xSb_{1-x}]$ alloys, see panel (b), predicted from original and optimized 1D descriptors constructed using *GEN*3 and *GEN*4. Model predictions are verified using energy differences calculated via DFT[59,60] (black line with diamond points).

supplementary material). The k-mesh was modified accordingly to maintain the same density of points employed in the simulations of primitive cells.

III. RESULTS AND DISCUSSION

In this section, we will analyze the final formulas as obtained from different generators. The results are shown in Tables I–IV; in the first row, we report the results obtained by Ghiringhelli *et al.*[55] for comparison.

First, by comparing the avg(*RMSE*) values, we note that all 1D formulas obtained from our different generators better perform with respect to the 1D ones reported in Ref. 55, where the authors used the automated feature selection method LASSO.[63] Remarkably, some atomic primary features appearing in 1D formulas of Ref. 55 also appear in our obtained list of 1D formulas using *GEN*1 and *GEN*2; nevertheless, those are characterized by a higher avg(*RMSE*) than other formulas we obtained via our combinatorial approaches. Additionally, formulas from *GEN*3 show the lowest avg(*RMSE*) among all the others. We also note, from Table I, that *GEN*1 and *GEN*3 provide lower avg(*RMSE*) compared to *GEN*2 and *GEN*4, respectively; however, *GEN*2 and *GEN*4 have a higher success rate in terms of classification prediction. (For instance, in Table S2 of the supplementary material, we report the formula optimized using the success rate as a target label.) It is noteworthy that the best formula from *GEN*4 shows similar terms to the corresponding case for the ΔE-based optimization. This testifies the fact that the choice of the performance metrics to rank the material features can be different according to the target problem to be studied; different models' performance metrics are, in fact, not always correlated.

In order to gather hints on the relative contribution of the individual primary atomic properties to the stabilization of either the rocksalt or the zinc blende structure, we extracted the best ten formulas with the lowest avg(*RMSE*) from each generator (so-called original formulas) and then apply the formula optimization, as detailed in Sec. II C. This procedure attributes relative weights to each $f(AP)$, allowing us to measure the importance of the individual atomic properties in driving the energy stabilization. In principle, the avg(*RMSE*) value depends on random test-train splits that we perform to our dataset. Therefore, to reduce the effect of randomization, as target model performance metrics, we rank our optimized formula based on the RMSE of the whole dataset, rather than based on avg(*RMSE*). By comparing Tables I and II, it is evident that the optimization procedure can further change the formula ranking, providing a different final "best formula" with respect to the non-optimized formulas. In particular, we notice an improvement in RMSE around 5%–10% after the formula optimization.

Interestingly, our results reveal the size of the A cation to play a leading role in the phase stabilization; in fact, the $r_p(A)$ radius appears in the best-performing formulas more frequently than the other basic atomic properties. Therefore, we further analyzed the dependence of ΔE on $r_p(A)$. In Fig. 2, we show ΔE as a function of $r_p(A)$, including fitting curves proportional to $r_p(A)^{-2}$ and $r_p(A)^{-3}$. What can be observed is a clear dependence of ΔE on $r_p(A)$: larger (smaller) $r_p(A)$ favors RS (ZB). Moreover, there is an overall good agreement with the fit, particularly using the $r_p(A)^{-3}$ function. The latter is, in fact, the most recurrent prototype function detected by the ML models. Such a strong dependence of the energy is not observed with respect to the other atomic properties; other comparative plots of ΔE as a function of other $f(p)$ are reported in Fig. S2 of the supplementary material. This behavior is in line with the further observation that the rocksalt structures systematically show larger interatomic distances with respect to the zinc blende counterparts (cf. lattice parameters reported in Ref. 55); therefore, larger

cations prefer to adopt octahedral coordination (i.e., RS) with longer bond-lengths—and bigger polyhedral volume—compared to ZB with tetrahedral coordination.

From the obtained results, we remark that formulas based on "spatial" atomic properties achieve higher ranking, thus better performance, with respect to those, including atomic energy terms, both in the original models and in the optimized ones. Accordingly, this behavior further confirms the primary role played by the atomic size (in terms of steric and/or bonding-related effects), in determining the energetics of the AB compounds, i.e., in selecting the preferred crystal structure.[55] In particular, we note from the results that the well-performing GEN3 and GEN4 contain all the four radii [$r_s(A)$, $r_s(B)$, $r_p(A)$ and $r_p(B)$], as expected from a basic understanding of bonding in octet binary semiconductors. Note that GEN3 and GEN4 generally show better performance as they are built to explore a wider space of search (see Sec. II and Table S4 in the supplementary material where the number of generated and evaluated formulas is also reported).

With the aim of further proving such trends and validate the implemented combinatorial ML method, we study the energetics in alloys of the type $[A_xA'_{1-x}]B$ and $A[B_xB'_{1-x}]$, where x is the relative concentration of the mixing ions, monotonically tuning thus

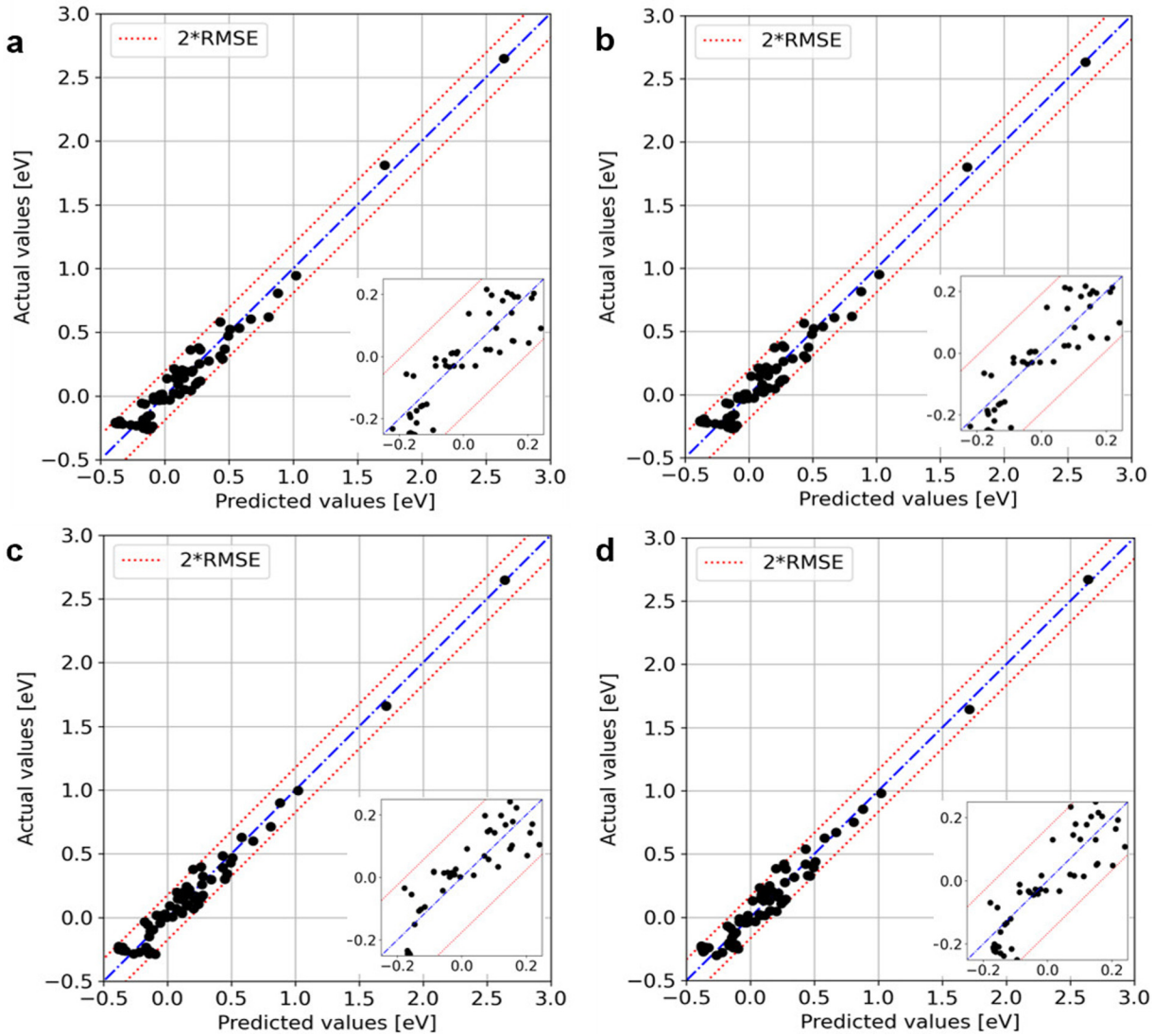

FIG. 5. Comparison of actual (i.e., DFT) vs predicted total energy difference ΔE for (a) 1D, (c) 2D, and (d) 3D formulas constructed using *GEN*3. Panel (b) shows the best 1D descriptors after formula optimization. Lower-right insets show a zoom in the relevant region where many compounds are concentrated. Red dotted lines correspond to a 2 × avg(*RMSE*) value. The respective descriptors can be inferred from Tables I–IV.

the average size of one ion with respect to the other. All the alloy input properties were linearly interpolated between corresponding values for end binaries (i.e., AB and $A'B$ in the $[A_xA'_{1-x}]B$ case) according to Vegard's law.[79] For the A-ion mixing case, we considered SrSe, CaSe, MgSe, and BeSe as parent AB compounds, already included in the original dataset. We then predicted the energy differences between RS and ZB phases for varying concentrations using the original and optimized 1D formulas constructed via *GEN*3 and *GEN*4 generators (Tables I and II, respectively). To confirm the obtained predictions, we thus calculated the energy difference via DFT simulations for a few intermediate concentrations. The results, shown in Fig. 3, demonstrate an overall agreement between first-principles calculated and machine-learning predicted energetics. In particular, we notice a change of sign in ΔE, reflecting the change in the stability of the RS with respect to the ZB phase, when moving from the larger strontium to the smaller beryllium at the A-site, in line with the previously discussed relation between atomic radii of the A-ions and phase stabilization. At variance, no such change of phase is observed when mixing ions at the B-site, keeping fixed the A-type one. This is confirmed by looking at the energetics in $B[Sb_{1-x}P_x]$ and $Sr[Se_{1-x}S_x]$ alloys, shown in Figs. 4(a) and 4(b), respectively. Despite the changing size of the average B-site, the two systems preserve the crystal structure adopted by the parent compounds, i.e., rock salt for the Sr-based compounds and zinc blende for the B-based compounds. Such behavior is still in line with the preferred atomic structure fixed by the ion at the A-site, consistently with strontium being larger than boron. Qualitative agreement between ML-predicted and DFT-calculated energetics is observed again.

After discussing the results related to 1D models, we now comment about the higher-dimensional formulas. Our best 2D and 3D formulas from different generators are reported in Tables III and IV, respectively.

To visualize the performance of the obtained formulas, we reproduce in Fig. 5 the scatterplots of DFT-calculated energies as a function of model-predicted energy differences for the best formulas obtained by *GEN*3—in terms of avg(*RMSE*)—for 1D, 1D after formula optimization, 2D, and 3D models. From these, one can infer the quality of the prediction for the different approaches: the narrower the area between red lines [representing $2 \times$ avg(*RMSE*)], the smaller the error or, equivalently, the more reliable the prediction. Notably, this is the case when building higher dimension formulas.

In addition, a careful comparison between our results and those reported in the reference paper, Ref. 55, is reported in Table S1 of the supplementary material. In particular, in Fig. S1 of the supplementary material, we compared the scatterplot of the 1D formula from *GEN*3 and Ref. 55, with bar graphs of errors for individual compounds. To check the improvement with respect to 1D formulas, we considered the avg(*RMSE*) value, as also chosen in Ref. 55. One can observe the improvement in avg(*RMSE*) if we examine 1D and 2D formulas in Tables I and III. We notice around 10%–20% improvement from the original 1D to 2D, but less than 10% of optimized 1D to original 2D formulas. Furthermore, we also notice that original and optimized 1D formulas from *GEN*3 and *GEN*4 better perform with respect to the corresponding 2D ones reported in Ref. 55.

We remark that the process of formula optimization is less computationally expensive than the construction of higher-dimensional formulas. In addition, from the formula optimization, one can gain better physical insights into the contribution of individual primary atomic properties. These comments overall suggest that lower-dimensional formulas constitute a better choice in terms of physical interpretation and computational efficiency.

IV. CONCLUSIONS

The knowledge of a material stable crystal structure constitutes the starting point for any *ab initio* modeling since material properties crucially depend on the periodic atomic arrangement in the crystal. Within this general framework, our aim here has been to exploit ML methods to correlate the energetic stability of different crystal structures (zinc blende vs rock salt) for popular binary semiconducting compounds with primary properties of their atomic constituents, the latter representing simple and easily accessible ingredients. Based on atomic properties, we, therefore, built the material features using a combinatorial approach, we trained the machine-learning model using the created features over a density-functional-theory dataset, and we obtained simple mathematical expressions to quantitatively predict the energetic stability of one crystal structure over the other (i.e., a formula). In addition, we have also introduced an extra step following the linear regression to explore the relative contributions of individual basic atomic properties.

To investigate the performance of the combinatorial approach, we compared our results with a reference paper,[55] where the authors predicted the stability of the crystal structure using an automated feature selection method. We found that our 1D formulas constructed using the combinatorial approach achieved a higher accuracy with respect to the reference ones. Furthermore, we also learned more about the underlying mechanism from the formula optimization, where we found that the stability of RS and ZB heavily depends on the r_p radius of A-sites. This kind of understanding is, in general, much more difficult to achieve in heavily automated artificial-intelligence methods, such as neural networks, where it is not possible to interpret directly the model results. In this respect, our approach based on linear regression allows the construction of physical models supported by machine-driven suggestions of relevant ingredients; as such, it should be regarded as a methodology offering a huge range of applications in addressing microscopic mechanisms underlying different phenomena, calling for extensive investigations in the near future.

SUPPLEMENTARY MATERIAL

See the supplementary material for technical details related to LR, DFT calculations of the alloy supercell, dataset, and for additional results related to 1D, 2D, and 3D formulas.

ACKNOWLEDGMENTS

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant (Agreement No. 861145-BeMAGIC). The authors acknowledge the Italian MIUR (Ministry of Education, University and Research) for supporting the PRIN project "TWEET:

ToWards fErroElectricity in Two dimensions" (Grant No. 2017YCTB59) and the "Nanoscience Foundries and Fine Analysis" (NFFA-MIUR Italy) project. Calculations were performed exploiting the computing resources at the Pharmacy Department, Univ. Chieti-Pescara. D.A. is grateful to M. Verstraete and B. Dupé (ULiège) for the time allowed to work on the writing of this paper. We are also thankful to L. Ghiringhelli for his fruitful support and insights.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

DATA AVAILABILITY

The data that support the findings of this study are available within the article and its supplementary material. The code for machine learning is available at https://github.com/lstorchi/matinformatics.

REFERENCES

[1]G. E. Moore, Electronics **38**, 114–117 (1965).
[2]R. Ramprasad, R. Batra, G. Pilania, A. Mannodi-Kanakkithodi, and C. Kim, npj Comput. Mater. **3**, 54 (2017).
[3]M. Fukuda, J. Zhang, Y.-T. Lee, and T. Ozaki, Mater. Adv. **2**, 4392 (2021).
[4]D. Schwalbe-Koda, S. Kwon, C. Paris, E. Bello-Jurado, Z. Jensen, E. Olivetti, T. Willhammar, A. Corma, Y. Román-Leshkov, M. Moliner, and R. Gómez-Bombarelli, Science **374**, 308 (2021).
[5]E. R. Homer, Comput. Mater. Sci. **161**, 244 (2019).
[6]S. Curtarolo, G. L. W. Hart, M. B. Nardelli, N. Mingo, S. Sanvito, and O. Levy, Nat. Mater. **12**, 191 (2013).
[7]M. L. Green, C. L. Choi, J. R. Hattrick-Simpers, A. M. Joshi, I. Takeuchi, S. C. Barron, E. Campo, T. Chiang, S. Empedocles, J. M. Gregoire, A. G. Kusne, J. Martin, A. Mehta, K. Persson, Z. Trautt, J. V. Duren, and A. Zakutayev, Appl. Phys. Rev. **4**, 011105 (2017).
[8]A. Walsh, Nat. Chem. **7**, 274 (2015).
[9]J. Shen, V. I. Hegde, J. He, Y. Xia, and C. Wolverton, Chem. Mater. **33**, 9486 (2021).
[10]S. D. Griesemer, L. Ward, and C. Wolverton, Phys. Rev. Mater. **5**, 105003 (2021).
[11]J. E. Saal, S. Kirklin, M. Aykol, B. Meredig, and C. Wolverton, JOM **65**, 1501 (2013).
[12]S. Kirklin, J. E. Saal, B. Meredig, A. Thompson, J. W. Doak, M. Aykol, S. Rühl, and C. Wolverton, npj Comput. Mater. **1**, 15010 (2015).
[13]C. Draxl and M. Scheffler, MRS Bull. **43**, 676 (2018).
[14]C. Draxl and M. Scheffler, J. Phys. Mater. **2**, 036001 (2019).
[15]S. Curtarolo, W. Setyawan, S. Wang, J. Xue, K. Yang, R. H. Taylor, L. J. Nelson, G. L. Hart, S. Sanvito, M. Buongiorno-Nardelli, N. Mingo, and O. Levy, Comput. Mater. Sci. **58**, 227 (2012).
[16]M. N. Gjerding, A. Taghizadeh, A. Rasmussen, S. Ali, F. Bertoldo, T. Deilmann, N. R. Knøsgaard, M. Kruse, A. H. Larsen, S. Manti, T. G. Pedersen, U. Petralanda, T. Skovhus, M. K. Svendsen, J. J. Mortensen, T. Olsen, and K. S. Thygesen, 2D Mater. **8**, 044002 (2021).
[17]S. Haastrup, M. Strange, M. Pandey, T. Deilmann, P. S. Schmidt, N. F. Hinsche, M. N. Gjerding, D. Torelli, P. M. Larsen, A. C. Riis-Jensen, J. Gath, K. W. Jacobsen, J. Jørgen Mortensen, T. Olsen, and K. S. Thygesen, 2D Mater. **5**, 042002 (2018).
[18]F. Bertoldo, S. Ali, S. Manti, and K. S. Thygesen, "Quantum point defects in 2D materials: The QPOD database," arXiv:2110.01961 (2021).
[19]A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, APL Mater. **1**, 011002 (2013).
[20]L. Talirz, S. Kumbhar, E. Passaro, A. V. Yakutovich, V. Granata, F. Gargiulo, M. Borelli, M. Uhrin, S. P. Huber, S. Zoupanos, C. S. Adorf, C. W. Andersen, O. Schütt, C. A. Pignedoli, D. Passerone, J. VandeVondele, T. C. Schulthess, B. Smit, G. Pizzi, and N. Marzari, Sci. Data **7**, 299 (2020).
[21]G. Pizzi, A. Cepellotti, R. Sabatini, N. Marzari, and B. Kozinsky, Comput. Mater. Sci. **111**, 218 (2016).
[22]S. P. Huber, E. Bosoni, M. Bercx, J. Bröder, A. Degomme, V. Dikan, K. Eimre, E. Flage-Larsen, A. Garcia, L. Genovese, D. Gresch, C. Johnston, G. Petretto, S. Poncé, G.-M. Rignanese, C. J. Sewell, B. Smit, V. Tseplyaev, M. Uhrin, D. Wortmann, A. V. Yakutovich, A. Zadoks, P. Zarabadi-Poor, B. Zhu, N. Marzari, and G. Pizzi, npj Comput. Mater. **7**, 136 (2021).
[23]H. Park, A. Ali, R. Mall, H. Bensmail, S. Sanvito, and F. El-Mellouhi, Mach. Learn.: Sci. Technol. **2**, 025030 (2021).
[24]C. Kim, G. Pilania, and R. Ramprasad, Chem. Mater. **28**, 1304 (2016).
[25]E. Tsymbalov, Z. Shi, M. Dao, S. Suresh, J. Li, and A. Shapeev, npj Comput. Mater. **7**, 76 (2021).
[26]C. J. Bartel, C. Sutton, B. R. Goldsmith, R. Ouyang, C. B. Musgrave, L. M. Ghiringhelli, and M. Scheffler, Sci. Adv. **5**, eaav0693 (2019).
[27]A. G. Kusne, T. Gao, A. Mehta, L. Ke, M. C. Nguyen, K.-M. Ho, V. Antropov, C.-Z. Wang, M. J. Kramer, C. Long, and I. Takeuchi, Sci. Rep. **4**, 6367 (2014).
[28]H. Koinuma and I. Takeuchi, Nat. Mater. **3**, 429 (2004).
[29]S. Manti, M. K. Svendsen, N. R. Knøsgaard, P. M. Lyngby, and K. S. Thygesen, "Predicting and machine learning structural instabilities in 2D materials," arXiv:2201.08091 (2022).
[30]K. Kim, L. Ward, J. He, A. Krishna, A. Agrawal, and C. Wolverton, Phys. Rev. Mater. **2**, 123801 (2018).
[31]K. Pal, C. W. Park, Y. Xia, J. Shen, and C. Wolverton, "Scale-invariant machine-learning model accelerates the discovery of quaternary chalcogenides with ultralow lattice thermal conductivity," arXiv:2109.03751 (2021).
[32]M. Kuban, S. Rigamonti, M. Scheidgen, and C. Draxl, "Density-of-states similarity descriptor for unsupervised learning from materials data," arXiv:2201.02187 (2022).
[33]K. Schütt, P.-J. Kindermans, H. E. Sauceda Felix, S. Chmiela, A. Tkatchenko, and K.-R. Müller, Adv. Neural Inf. Process. Syst. **30**, 992–1002 (2017).
[34]L. Breiman, Mach. Learn. **45**, 5 (2001).
[35]K. Gurney, *Introduction to Neural Networks* (UCL Press Limited, London, 1997).
[36]T. Xie and J. C. Grossman, Phys. Rev. Lett. **120**, 145301 (2018).
[37]R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, ACM Comput. Surv. **51**, 1 (2018).
[38]C. Molnar, *Interpretable Machine Learning*, 2nd ed. (Christoph Molnar, 2022).
[39]L. Breiman, Mach. Learn. **45**, 5 (2001).
[40]A. Fisher, C. Rudin, and F. Dominici, arXiv:1801.01489 (2019).
[41]S. Al-Askaar and M. Perkowski, in *2021 IEEE 51st International Symposium on Multiple-Valued Logic (ISMVL)* (IEEE, 2021), pp. 128–135.
[42]R. Elshawi, M. H. Al-Mallah, and S. Sakr, BMC Med. Inform. Decis. Mak. **19**, 1 (2019).
[43]M. T. Ribeiro, S. Singh, and C. Guestrin, "Model-agnostic interpretability of machine learning," arXiv:1606.05386 (2016).
[44]S. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," arXiv:1705.07874 (2017).
[45]Y. Zhang, P. Tiňo, A. Leonardis, and K. Tang, IEEE Trans. Emerg. Top. Comput. Intell. **5**, 726–745 (2021).
[46]Q. Zhang, Y. N. Wu, and S.-C. Zhu, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2018), pp. 8827–8836.
[47]D. Alvarez Melis and T. Jaakkola, arXiv:1806.07538 (2018).
[48]S. Chatterjee and J. S. Simonoff, *Handbook of Regression Analysis* (Wiley, Hoboken, NJ, 2013).

[49]J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets* (Stanford University, Stanford, CA, 2010).
[50]T. Miller, Artif. Intell. **267**, 1 (2019).
[51]B. Kim, R. Khanna, and O. O. Koyejo, Adv. Neural Inf. Process. Syst. **29**, 2280–2288 (2016).
[52]D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein, *Logistic Regression* (Springer, 2002).
[53]A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown, J. Chemom. **18**, 275 (2004).
[54]L. M. Ghiringhelli, "Interpretability of machine-learning models in physical sciences," arXiv:2104.10443 (2021).
[55]L. M. Ghiringhelli, J. Vybiral, S. V. Levchenko, C. Draxl, and M. Scheffler, Phys. Rev. Lett. **114**, 105503 (2015).
[56]B. Meredig, A. Agrawal, S. Kirklin, J. E. Saal, J. W. Doak, A. Thompson, K. Zhang, A. Choudhary, and C. Wolverton, Phys. Rev. B **89**, 094104 (2014).
[57]L. Ward, A. Dunn, A. Faghaninia, N. E. R. Zimmermann, S. Bajaj, Q. Wang, J. Montoya, J. Chen, K. Bystrom, M. Dylla, K. Chard, M. Asta, K. A. Persson, G. J. Snyder, I. Foster, and A. Jain, Comput. Mater. Sci. **152**, 60 (2018).
[58]L. Ward, A. Agrawal, A. Choudhary, and C. Wolverton, npj Comput. Mater. **2**, 16028 (2016).
[59]P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).
[60]W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).
[61]J. P. Perdew and Y. Wang, Phys. Rev. B **45**, 13244 (1992).
[62]K. Momma and F. Izumi, J. Appl. Crystallogr. **41**, 653 (2008).
[63]S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, New York, 2014).
[64]G. Van Rossum and F. L. Drake, Jr., *Python Tutorial* (Centrum voor Wiskunde en Informatica, Amsterdam, 1995).
[65]L. Storchi, "Open Source Code"; see https://github.com/lstorchi/matinformatics (2022).
[66]MATHEMATICA, version 13.0.0, W. R., Inc., Champaign, IL, 2021.
[67]L. Storchi, "Mathematica Notebook," startingtest.nb; see https://github.com/lstorchi/matinformatics (2022).
[68]F. Gao and L. Han, Comput. Optim. Appl. **51**, 259 (2012).
[69]G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., Johns Hopkins Studies in the Mathematical Sciences (The Johns Hopkins University Press, Baltimore, MD, 2013).
[70]C. G. Broyden, IMA J. Appl. Math. **6**, 76 (1970).
[71]R. dembo, S. Eisenstat, and T. Steihaug, SIAM J. Numer. Anal. **19**, 400 (1982).
[72]G. Kresse and J. Hafner, Phys. Rev. B **47**, 558 (1993).
[73]G. Kresse and J. Furthmüller, Comput. Mater. Sci. **6**, 15 (1996).
[74]G. Kresse and J. Furthmüller, Phys. Rev. B **54**, 11169 (1996).
[75]V. Blum, R. Gehrke, F. Hanke, P. Havu, V. Havu, X. Ren, K. Reuter, and M. Scheffler, Comput. Phys. Commun. **180**, 2175 (2009).
[76]C. Eckhardt, K. Hummer, and G. Kresse, Phys. Rev. B **89**, 165201 (2014).
[77]L. Bellaiche and D. Vanderbilt, Phys. Rev. B **61**, 7877 (2000).
[78]D. Amoroso, A. Cano, and P. Ghosez, Phys. Rev. B **97**, 174108 (2018).
[79]L. Vegard, Z. Phys. **5**, 17 (1920).