



PAPER

Learning physical descriptors for materials science by compressed sensing

OPEN ACCESS

RECEIVED

3 December 2016

REVISED

4 January 2017

ACCEPTED FOR PUBLICATION

9 January 2017

PUBLISHED

7 February 2017

Original content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](#).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



Luca M Ghiringhelli^{1,5}, Jan Vybiral², Emre Ahmetcik¹, Runhai Ouyang¹, Sergey V Levchenko¹,
Claudia Draxl^{1,3} and Matthias Scheffler^{1,4}

¹ Fritz-Haber-Institut der Max-Planck-Gesellschaft, Berlin-Dahlem, Germany

² Charles University, Department of Mathematical Analysis, Prague, Czech Republic

³ Humboldt-Universität zu Berlin, Institut für Physik and IRIS Adlershof, Berlin, Germany

⁴ Department of Chemistry and Biochemistry and Materials Department, University of California-Santa Barbara, Santa Barbara, CA 93106-5050, United States of America

⁵ Author to whom any correspondence should be addressed.

E-mail: ghiringhelli@fhi-berlin.mpg.de

Keywords: crystal-structure prediction, big-data driven materials science, compressed sensing, feature selection

Abstract

The availability of big data in materials science offers new routes for analyzing materials properties and functions and achieving scientific understanding. Finding structure in these data that is not directly visible by standard tools and exploitation of the scientific information requires new and dedicated methodology based on approaches from statistical learning, compressed sensing, and other recent methods from applied mathematics, computer science, statistics, signal processing, and information science. In this paper, we explain and demonstrate a compressed-sensing based methodology for feature selection, specifically for discovering physical descriptors, i.e., physical parameters that describe the material and its properties of interest, and associated equations that explicitly and quantitatively describe those relevant properties. As showcase application and proof of concept, we describe how to build a physical model for the quantitative prediction of the crystal structure of binary compound semiconductors.

1. Introduction

Big-data-driven research offers new routes towards scientific insight. This big-data challenge is not only about storing and processing huge amounts of data, but also, and in particular, it is a chance for new methodology for modeling, describing, and understanding. Let us explain this by a simple, but most influential, historic example. About 150 years ago, D. Mendeleev and others organized the 56 atoms that were known at their time in terms of a table, according to their weight and chemical properties. Many atoms were not yet identified, but from the table, it was clear that they should exist, and even their chemical properties were anticipated. The scientific reason behind the table, i.e., the shell structure of the electrons, was unclear and was understood only 50 years later when quantum mechanics had been developed. Finding structure in the huge space of materials, e.g., one table (or a map) for each property, or function of interest, is one of the great dreams of (big-)data-driven materials science.

Obviously, the challenge to sort all materials is much bigger than that of the mentioned example of the periodic table of elements (PTE). Up to date, the PTE contains 118 elements that have been observed. About 200 000 materials are 'known' to exist [1], but only for very few of these 'known' materials, the basic properties (elastic constants, plasticity, piezoelectric tensors, thermal and electrical conductivity, etc) have been determined. When considering surfaces, interfaces, nanostructures, organic materials, and hybrids of these mentioned systems, the amount of possible materials is practically infinite. It is, therefore, highly likely that new materials with superior and up to now simply unknown characteristics exist that could help solving fundamental issues in the fields of energy, mobility, safety, information, and health.

For materials science it is already clear that big data are structured, and in terms of materials properties and functions, the space of all possible materials is sparsely populated. Finding this structure in the big data, e.g., asking for efficient catalysts for CO₂ activation or oxidative coupling of methane, good thermal barrier coatings, shape memory alloys as artery stents, or thermoelectric materials for energy harvesting from temperature gradients, just to name a few examples, may be possible, even if the actuating physical mechanisms of these properties and functions are not yet understood in detail. Novel big-data analytics tools, e.g., based on machine learning or compressed sensing (CS), promise to do so.

Machine learning of big data has been used extensively in a variety of fields ranging from biophysics and drug design to social media and text mining. It is typically considered a universal approach for learning (fitting) a complex relationship $y = f(x)$. Some, though few, machine-learning (ML) based works have been done in materials science, e.g., [2–17]. Most of them use kernel ridge regression (KRR) or Gaussian processes, where in both cases the fitted/learned property is expressed as a weighted sum over all or selected data points. A clear breakthrough, demonstrating a ‘game change’, is missing, so far, and a key problem, emphasized in [18], is that a good feature selection for descriptor identification is central to achieving a good-quality, predictive statistically learned equation.

One of the purposes of discovering structure in materials-science data, is to predict a property of interest for a given complete class of materials. In order to do this, a model needs to be built, that maps some accessible, descriptive input quantities, identifying a material, into the property of interest. In statistical learning, this set of input parameters is called descriptor [18]. For our materials science study, this descriptor (elsewhere also termed ‘fingerprint’ [5, 12]) is a set of physical parameters that uniquely describe the material and its function of interest. It is important that materials that are very different (similar) with respect to the property or properties of interest, should be characterized by very different (similar) values of the descriptors. Moreover, the determination of the descriptor must not involve calculations as intensive as those needed for the evaluation of the property to be predicted.

In most papers published so far, the descriptor has been introduced ad hoc, i.e., without performing an extensive, systematic analysis and without demonstrating that it is the best (in some sense) within a certain broad class. The risk of such ad hoc approach is that the learning step is inefficient (i.e., very many data points are needed), the governing physical mechanisms remain hidden, and the reliability of predictions is unclear. Indeed, machine learning is understood as an interpolation and not extrapolation approach, and it will do so well, if enough data are available. However, quantum mechanics and materials science are so multifaceted and intricate that we are convinced that we hardly ever will have enough data. Thus, it is crucial to develop these tools as domain-specific methods and to introduce some scientific understanding, keeping the bias as low as possible.

In this paper, we present a methodology for discovering physically meaningful descriptors and for predicting physically (materials-science) relevant properties. The paper is organized as follows: in section 2, we introduce compressed-sensing [19–22] based methods for feature selection; in section 3, we introduce efficient algorithms for the construction and screening of feature spaces in order to single out ‘the best’ descriptor. ‘Feature selection’ is a widespread set of techniques that are used in statistical analysis in different fields [23]. In the following sections, we analyze the significance of the descriptors found by the statistical-learning methods, by discussing the interpolation ability of the model based on the found descriptors (section 4), the stability of the models in terms of sensitivity analysis (section 5), and the extrapolation capability, i.e., the possibility of predicting new materials (section 6).

As showcase application and proof of concept, we use the quantitative prediction of the crystal structure of binary compound semiconductors, which are known to crystallize in zincblende (ZB), wurtzite (WZ), or rocksalt (RS) structures. The structures and energies of ZB and WZ are very close and for the sake of clarity we will not distinguish them here. For many materials, the energy difference between ZB and RS is larger, though still very small, namely just 0.001% or less of the total energy of a single atom. Thus, high accuracy is required to predict this difference. The property P that we aim to predict is the difference in the energies between RS and ZB for the given AB-compound material, ΔE_{AB} . The energies are calculated within the Kohn–Sham (KS) formulation [24] of density-functional theory [25], with the local-density approximation (LDA) exchange-correlation functional [26, 27]. Since in this paper we are concerned with the development of a data-analytics approach, it is not relevant that also approximations better than LDA exist, only the internal consistency and the realistic complexity of the data are important. The below described methodology applies to these better approximations as well.

Clearly, the sign of ΔE_{AB} gives the classification (negative for RS and positive for ZB), but the quantitative prediction of ΔE_{AB} values gives a better understanding. The separation of RS and ZB structures into distinct classes, on the basis of their chemical formula only, is difficult and has existed as a key problem in materials science for over 50 years [28–38]. In the rest of the paper, we will always write the symbol ΔE , without the subscript AB, to signify the energy of the RS compound minus the energy of the ZB compound.

2. Compressed-sensing based methods for feature selection

Let us assume that we are given data points $\{\mathbf{d}_1, P_1\}, \dots, \{\mathbf{d}_N, P_N\}$, with $\mathbf{d}_1, \dots, \mathbf{d}_N \in \mathbb{R}^M$ and $P_1, \dots, P_N \in \mathbb{R}$. We would like to analyze and understand the dependence of the output P (for ‘property’) on the input \mathbf{d} (for ‘descriptor’), that reflects the specific measurement or calculation. Specifically, we are seeking for an *interpretable* equation $P = f(\mathbf{d})$, e.g., an explicit analytical formula, where P depends on a small number of input parameters, i.e., the dimension of the input vector \mathbf{d} is small. Below, we will impose constraints in order to arrive at a method for finding a low-dimensional linear solution for $P = f(\mathbf{d})$. Later, we introduce nonlinear transformations of the vector \mathbf{d} ; dealing with such nonlinearities is the main focus of this paper.

When looking for a linear dependence $P = f(\mathbf{d}) = \mathbf{d} \cdot \mathbf{c}$, the most simple approach to the problem is the method of *least squares*. This is the solution of

$$\operatorname{argmin}_{\mathbf{c} \in \mathbb{R}^M} \sum_{j=1}^N \left(P_j - \sum_{k=1}^M d_{j,k} c_k \right)^2 = \operatorname{argmin}_{\mathbf{c} \in \mathbb{R}^M} \|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2, \quad (1)$$

where $\mathbf{P} = (P_1, \dots, P_N)^T$ is the column vector of the outputs and $\mathbf{D} = (d_{j,k})$ is the $(N \times M)$ -dimensional matrix of the inputs, and the column vector \mathbf{c} is to be determined. The equality defines the ℓ_2 or Euclidean norm ($\|\cdot\|_2$) of the vector $\mathbf{P} - \mathbf{D}\mathbf{c}$. The resulting function would then be just the linear relationship $f(\mathbf{d}) = \langle \mathbf{d}, \mathbf{c} \rangle = \sum_{k=1}^M d_k c_k$. It is the function that minimizes the error $\|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2$ among all linear functions of \mathbf{d} . Let us point out a few properties of equation (1). First, the solution of equation (1) is given by an explicit formula $\mathbf{c} = (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \mathbf{P}$ if $\mathbf{D}^T \mathbf{D}$ is a non-singular matrix. Second, it is a convex problem, therefore the solution could also be found efficiently even in large dimensions (i.e. if M and N are large) and there are many solvers available [39]. Finally, it does not put any more constraints on \mathbf{c} (or f) than that it is linear and minimizes equation (1).

A general linear function f of \mathbf{d} usually involves also an absolute term, i.e. it has a form $f(\mathbf{d}) = c_0 + \sum_{k=1}^M d_k c_k$. It is easy to add the absolute term (also called *bias*) to equation (1) in the form

$$\operatorname{argmin}_{c_0 \in \mathbb{R}, \mathbf{c} \in \mathbb{R}^M} \sum_{j=1}^N \left(P_j - c_0 - \sum_{k=1}^M d_{j,k} c_k \right)^2. \quad (2)$$

This actually coincides with equation (1) if we first enlarge the matrix \mathbf{D} by adding a column full of ones. We will tacitly assume in the following that this modification was included in the implementation and shall not repeat that any more.

If pre-knowledge is available, it can be (under some conditions) incorporated into equation (1). For example, we might want to prefer those vectors \mathbf{c} , which are small (in some sense). This gives rise to *regularized* problems. In particular, the *ridge regression*

$$\operatorname{argmin}_{\mathbf{c} \in \mathbb{R}^M} (\|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2 + \lambda \|\mathbf{c}\|_2^2) \quad (3)$$

with a penalty parameter $\lambda > 0$ weights the magnitude of the vector \mathbf{c} and of the error of the fit against each other. The larger λ , the smaller is the minimizing vector \mathbf{c} . The smaller λ , the better is the least square fit (the first addend in equation (3)). More specifically, if $\lambda \rightarrow 0$, the solutions of equation (3) converge to the solution of equation (1). And if $\lambda \rightarrow \infty$, the solutions of equation (3) tend to zero.

2.1. Sparsity, NP-hard problems, and convexity of the minimization problem

The kind of pre-knowledge, which we will discuss next, is that we would like $f(\mathbf{d})$ depend only on a small number of components of \mathbf{d} . In the notation of equation (1), we would like to achieve that most of the components of the solution \mathbf{c} are zero. Therefore, we denote the number of non-zero coordinates of a vector \mathbf{c} by

$$\|\mathbf{c}\|_0 = \#\{j : c_j \neq 0\}. \quad (4)$$

Here, $\{j : c_j \neq 0\}$ stands for the set of all the coordinates j , such that c_j is non-zero, and $\#\{\dots\}$ denotes the number of elements of the set $\{\dots\}$. Thus, $\|\mathbf{c}\|_0$ is the number of non-zero elements of the vector \mathbf{c} , and it is often called the ℓ_0 norm of \mathbf{c} . We say that $\mathbf{c} \in \mathbb{R}^M$ is k -sparse, if $\|\mathbf{c}\|_0 \leq k$, i.e. if at most k of its coordinates are not equal to zero.

When trying to regularize equation (1) in such a way that it prefers sparse solutions \mathbf{c} , one has to minimize:

$$\operatorname{argmin}_{\mathbf{c} \in \mathbb{R}^M} (\|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2 + \lambda \|\mathbf{c}\|_0). \quad (5)$$

This is a rigorous formulation of the problem that we want to solve, but it has a significant drawback: it is computationally infeasible when M is large. This can be easily understood as follows. The naive way of solving equation (5) is to look first on all index sets $T \subset \{1, \dots, M\}$ with one element and try to minimize over vectors

supported in such sets. Then one looks for all subsets with two, three, and more elements. Unfortunately, their number grows quickly with M and the level of sparsity k . It turns out, that this naive way can not be improved much. The problem is called ‘NP-hard’ [40]. In a ‘non-deterministic polynomial-time hard’ problem, a good candidate for the solution can be checked in a polynomial time, but the solution itself can not be found in polynomial time. The basic reason is that equation (5) is not convex.

2.2. Methods based on the ℓ_1 norm

Reaching a compromise between convexity and promoting sparsity is made possible by the least absolute shrinkage and selection operator (LASSO) [41] approach, in which the ℓ_0 regularization of equation (5) is replaced by the ℓ_1 norm:

$$\operatorname{argmin}_{\mathbf{c} \in \mathbb{R}^M} \|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2 + \lambda \|\mathbf{c}\|_1. \quad (6)$$

The use of the ℓ_1 norm ($\|\mathbf{c}\|_1 = \sum_k |c_k|$), also known as the ‘Manhattan’ or ‘Taxicab’ norm, is crucial here. On the one hand, the optimization problem is convex [42]. On the other hand, the geometry of the ℓ_1 -unit ball $\{\mathbf{x} \in \mathbb{R}^M : \|\mathbf{x}\|_1 \leq 1\}$ shows, that it promotes sparsity [21, 22].

Similarly to equation (3), the larger the $\lambda > 0$, the smaller the ℓ_1 -norm of the solution of equation (6) would be, and vice versa. Actually, there exists a smallest $\lambda_0 > 0$, such that the solution of equation (6) is zero. If λ then falls below this threshold, one or more components of \mathbf{c} become non-zero.

2.3. Compressed sensing

The minimization problem in equation (6) is an approximation of the problem in equation (5), and their respective results do not necessarily coincide. The relation between these two minimization problems was studied intensively [43–45] and we summarize the results, developed in the context of a recent theory of CS [19, 21, 22]. These will be useful for justifying the implementation of our method (see below). The literature on CS is extensive and we believe that the following notes are a useful compendium.

We are especially interested in conditions on \mathbf{P} and \mathbf{D} which guarantee that the solutions of equations (5) and (6) coincide, or at least do not differ much. We concentrate on a simplified setting, namely attempting to find $\mathbf{c} \in \mathbb{R}^M$ fulfilling $\mathbf{D}\mathbf{c} = \mathbf{P}$ with fewest non-zero entries. This is written as

$$\operatorname{argmin}_{\mathbf{c}: \mathbf{D}\mathbf{c}=\mathbf{P}} \|\mathbf{c}\|_0. \quad (7)$$

As mentioned above, this minimization is computationally infeasible, and we are interested in finding the same solution by its convex reformulation:

$$\operatorname{argmin}_{\mathbf{c}: \mathbf{D}\mathbf{c}=\mathbf{P}} \|\mathbf{c}\|_1. \quad (8)$$

The answer can be characterized by the notion of the null space property (NSP) [46]. Let $\mathbf{D} \in \mathbb{R}^{N \times M}$ and let $\Omega \in \{1, \dots, M\}$. Then \mathbf{D} is said to have the NSP of order Ω if

$$\sum_{j \in T} |v_j| < \sum_{j \notin T} |v_j| \quad \forall \mathbf{v} \neq 0 \text{ with } \mathbf{D}\mathbf{v} = 0 \quad \forall T \subset \{1, \dots, M\} \text{ with } \#T \leq \Omega. \quad (9)$$

It is shown [46] that every Ω -sparse vector \mathbf{x} is the unique solution of equation (8) with $\mathbf{P} = \mathbf{D}\mathbf{x}$ if, and only if, \mathbf{D} has the NSP of order Ω . As a consequence, the minimization of the ℓ_1 norm as given by equation (8) recovers the unknown Ω -sparse vectors $\mathbf{c} \in \mathbb{R}^M$ from \mathbf{D} and $\mathbf{P} = \mathbf{D}\mathbf{c}$ only if \mathbf{D} satisfies the NSP of order Ω . Unfortunately, when given a specific matrix \mathbf{D} , it is not easy to check if it indeed has the NSP.

However, the CS analysis gives us some guidance for the admissible dimension M of \mathbf{D} . It is known [22, 46] that there exists a constant $C > 0$ such that whenever

$$N \geq C \Omega \ln(M) \quad (10)$$

then there exist matrices $\mathbf{D} \in \mathbb{R}^{N \times M}$ with NSP of order Ω . Actually, a random matrix with these dimensions satisfies the NSP of order Ω with high probability. On the other hand, this bound is tight, i.e., if N falls below this bound no stable and robust recovery of \mathbf{c} from \mathbf{D} and \mathbf{P} is possible, and this is true for every matrix $\mathbf{D} \in \mathbb{R}^{N \times M}$. Later on, N will correspond to the number of AB compounds considered and will be fixed to $N = 82$. Furthermore, M will be the number of input features, and equation (10) tells us that we can expect reasonable performance of ℓ_1 -minimization only if their number is at most of the order $e^{\frac{N}{C\Omega}}$.

Furthermore, the CS analysis sets forth that the NSP is surely violated already for $\Omega = 2$ if any two columns of \mathbf{D} are a multiple of each other. In general, one can expect the solution of equation (8) to differ significantly from the solution of equation (7) if any two columns of \mathbf{D} are highly correlated or, in general, if any of the columns of \mathbf{D} lies close to the span of a small number of any other its columns.

Historically, LASSO appeared first in 1996 as a ML technique, able to provide stable solutions to underdetermined linear systems, thanks to the ℓ_1 regularization. Ten years later, the CS theory appeared,

sharing with LASSO the concept of ℓ_1 regularization, but the emphasis is on the reconstruction of a possibly noisy signal \mathbf{c} from a minimal number of observations. CS can be considered the theory of LASSO, in the sense that it gives conditions (on the matrix \mathbf{D}) on when it is reasonable to expect that the ℓ_1 and ℓ_0 regularizations coincide. If the data and a, typically overcomplete, basis set are known, then LASSO is applied as a ML technique, where the task is to identify the smallest number of basis vectors that yield a model of a given accuracy. If the setting is such that a minimal set of measurements should be performed to reconstruct a signal, as in quantum-state tomography [47], then CS tells how the data should be acquired. As it will become clear in the following, our situation is somewhat intermediate, in the sense that we have the data (here, the energy of the 82 materials), but we have a certain freedom to select the basis set for the model. Therefore, we use CS to justify the construction of the matrix \mathbf{D} , and we adopt LASSO as a mean to find the low-dimensional basis set.

2.4. A simple LASSO example: the energy differences of crystal structures

In this and following sections, we walk through the application of LASSO to a specific materials-science problem, in order to introduce step by step a systematic approach for finding simple analytical formulas, that are built from simple input parameters (the *primary features*), for approximating physical properties of interest. We aim at predicting ΔE_{AB} , the difference in DFT-LDA energy between ZB and RS structures in a set of 82 octet binary semiconductors. Calculations were performed using the all-electron full-potential code FHI-aims [48] with highly accurate basis sets, \mathbf{k} -meshes, integration grids, and scaled ZORA [49] for the relativistic treatment.

The order of the two atoms in the chemical formula AB is chosen such that element A is the cation, i.e., it has the smallest Mulliken electronegativity, $EN = -(\text{IP} + \text{EA})/2$. IP and EA are the ionization potential and electron affinity of the free, isolated, spinless, and spherical symmetric atom. As noted, the calculation of the descriptor must involve less intensive calculations than those needed for the evaluation of the property to be predicted. Therefore, we consider only properties of the free atoms A and B, that build the binary material, and properties of the gas-phase dimers. In practice, we identified the following *primary features*, exemplified for atom A: the ionization potential IP(A), the electron affinity EA(A), H(A) and L(A), i.e.,⁶ the energies of the highest-occupied and lowest-unoccupied KS levels, and $r_s(A)$, $r_p(A)$, and $r_d(A)$, i.e., the radii where the radial probability density of the valence *s*, *p*, and *d* orbitals are maximal. The same features were used for atom B. Clearly, these primary features were chosen because it is well known since long that the relative ionization potentials, the atomic radii, and *sp*-hybridization govern the bonding in these materials. Consequently, some authors [50–52] already recognized that certain combinations of r_s and r_p —called r_σ and r_π (see below)—may be crucial for constructing a descriptor that predicts the RS/ZB classification. Note that just the sign of ΔE was predicted, while the model we present here targets at a quantitative prediction of ΔE . In contrast to previous work, we analyze how LASSO will perform in this task and how much better the corresponding description is. We should also note that the selected set of atomic primary features contains redundant physical information, namely H(A) and L(A) contain similar information as IP(A) and EA(A). In particular H(A)–L(A) is correlated, on physical grounds, to IP(A)–EA(A). However, since the values of these two differences are not the same (see footnote 5), all four features were included. As it will be shown below and in particular in appendix A, the two pairs of features are not interchangeable.

We start with a simplified example, in order to show in an easily reproducible way the performance of LASSO in solving our problem. To this extent, we describe each compound AB by the vector of the following six quantities:

$$\mathbf{d}_{AB} = (r_s(A), r_p(A), r_d(A), r_s(B), r_p(B), r_d(B)). \quad (11)$$

All this data collected gives a 82×6 matrix \mathbf{D} , one row for each compound. We standardize \mathbf{D} to have zero mean and variance 1, i.e. subtract from each column its mean and divide by its standard deviation. Standardization of the data is common practice[42] and aimed at controlling the numerical stability of the solution. However, when non-linear combinations of the features and cross-validation is involved, the standardization has to be performed carefully (see below). The energy differences are stored in a vector $\mathbf{P} \in \mathbb{R}^{82}$.

We are aiming at two goals:

- On the one hand, we would like to minimize the mean-squared error (MSE), a measure of the quality of the approximation, given by

$$\frac{1}{N} \sum_{j=1}^N \left(P_j - \sum_{k=1}^M d_{j,k} c_k \right)^2. \quad (12)$$

In this tutorial example, the size N of the dataset is 82 and, with the above choice of equation (11), $M = 6$.

⁶We used for IP (EA) the energy of the half occupied KS orbital in the half positively (negatively) charged atom.

- On the other hand, we prefer sparse solutions \mathbf{c}^\dagger with small Ω , as we like to explain the dependence of the energy difference P based on a low-dimensional (small Ω) descriptor \mathbf{d}^\dagger .

These two tasks are obviously closely connected, and go against each other. The larger the coefficient λ in equation (6), the more sparse the solution \mathbf{c} will be; the smaller λ , the smaller the MSE will be. The choice of λ is at our disposal and let us weight sparsity of \mathbf{c} against the size of the MSE.

In the following, we will rather report the root mean square error (RMSE) as a quality measure of a model. The reason is that the RMSE has the same units as the predicted quantities, therefore easier to understand in absolute terms. Specifically, when a sparse solution \mathbf{c}^\dagger of equation (6) is mentioned, with Ω non-zero components, we report the RMSE:

$$\sqrt{\frac{1}{82} \sum_{j=1}^{82} \left(P_j - \sum_{k=1}^{\Omega} d_{j,k}^\dagger c_k^* \right)^2},$$

where $\mathbf{D}^\dagger = (d_{j,k}^\dagger)$ contains the columns corresponding to the non-zero components of \mathbf{c}^\dagger and $\mathbf{c}^* = (c_k^*)$ is the solution of the least square fit:

$$\operatorname{argmin}_{\mathbf{c}^* \in \mathbb{R}^\Omega} \|\mathbf{P} - \mathbf{D}^\dagger \mathbf{c}^*\|_2^2.$$

Let us first state that the MSE obtained when setting $\mathbf{c} = 0$ is 0.1979 eV², which corresponds to a RMSE of 0.4449 eV. This means that if one ‘predicts’ $\Delta E = 0$ for all the materials, the RMSE of such ‘model’ is 0.4449 eV. This number acts as a baseline for judging the quality of new models.

In order to run this and all the numerical tests in this paper, we have written PYTHON scripts and used the linear-algebra solvers, including the LASSO solver, from the scikit_learn (sklearn) library (see appendix D for the actual script used for this section). We like to calculate the coefficients for a decreasing sequence of a hundred λ values, starting from the smallest $\lambda = \lambda_1$, such that the solution is $\mathbf{c} = 0$. λ_1 is determined by $N\lambda_1 = \max_i |\langle \mathbf{d}_i, \mathbf{P} \rangle|$ [42]. The sequence of λ values is constructed on a log scale, and the smallest λ value is set to $\lambda_{100} = 0.001 \lambda_1$. In our case, $\lambda_1 = 0.3169$ and the first non-zero component of \mathbf{c} is the second one, corresponding to $r_p(A)$. At $\lambda_{19} = 0.0903$, the second non-zero component of \mathbf{c} appears, corresponding to $r_d(B)$ (see equation (11)). For decreasing λ , more and more components of \mathbf{c} are different from zero until, from the column corresponding to $\lambda_{67} = 0.0032$ down to $\lambda_{100} = 0.0003$, all entries are occupied, i.e. no sparsity is left. The method clearly suggests $r_p(A)$ to be the most useful feature for predicting the energy difference. Indeed, by enumerating all six linear (least-square) models constructed with only one of the components of vector \mathbf{d} at a time, the smallest RMSE is given by the model based on $r_p(A)$. We conclude that LASSO really finds the coordinate best describing (in a linear way) the dependence of P on \mathbf{d} .

Let us now proceed to the best pair. The second appearing non-zero component is $r_d(B)$. The RMSE for the pair ($r_p(A)$, $r_d(B)$) is 0.2927 eV. An exhaustive search over all the fifteen pairs of elements of \mathbf{d}_{AB} reveals that the pair ($r_p(A)$, $r_d(B)$), indeed, yields the lowest (least-square) RMSE among all possible pairs in \mathbf{d}_{AB} .

The outcome for the best triplet reveals the weak point of the method. The third non-zero components of \mathbf{d} is $r_p(B)$, and the RMSE of the triplet $r_p(A)$, $r_p(B)$, $r_d(B)$ is 0.2897 eV, while an exhaustive search over all the triplets found, suggests ($r_s(B)$, $r_p(A)$, $r_p(B)$) to be optimal with a RMSE error of 0.2834, i.e., some 2% better. The reason is that equation (6) is only a convex proxy of the actual problem in equation (5). Their solutions have similar performance in terms of RMSE, but they do not have to coincide. Now let us compare the norm $\|\mathbf{c}\|_1$ of both triplets, for \mathbf{c}^* obtained by the least square regression with standardized columns. For the optimal triplet, ($r_s(B)$, $r_p(A)$, $r_p(B)$), $\|\mathbf{c}\|_1 = 2.901$ and for ($r_p(A)$, $r_p(B)$, $r_d(B)$), $\|\mathbf{c}\|_1 = 0.478$. Here we observe, that the first one needs higher coefficients for a small $\|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2$, while the second one provides a better compromise between a small $\|\mathbf{P} - \mathbf{D}\mathbf{c}\|_2^2$ and a small $\lambda\|\mathbf{c}\|_1$ in equation (6). The reason for the large coefficients for the former is that considering the 82-dimensional vectors whose components are the values of $r_s(B)$ and $r_p(B)$ for all the materials (in the same order), these vectors are almost parallel (their Pearson correlation coefficient is 0.996).⁷ In order to understand this, let us have a look at both least square models:

$$\Delta E_{\ell_0} = +1.272r_s(B) - 0.296r_p(A) - 1.333r_p(B) + 0.106, \quad (13)$$

$$\Delta E_{\ell_1} = -0.337r_p(A) - 0.044r_p(B) - 0.097r_d(B) + 0.106. \quad (14)$$

In the optimal ℓ_0 model, the highly correlated vectors $r_s(B)$ and $r_p(B)$ appear approximatively as the difference $r_s(B) - r_p(B)$ (their coefficients in the linear expansion have almost the same magnitude, ~ 1.3 , and opposite

⁷ Differently from H – L and IP – EA, which are expected, and are, correlated on physical grounds for all atoms, r_s is not necessarily expected to be correlated to r_p . This is true for all the atoms considered in this work, but it could well be that such strong correlation is not valid for all atoms. This situation is somewhat reminiscent of the Berkson paradox, i.e., when a spurious correlation is found due to a biased selection of the data set.

Table 1. Definition of the feature space for the tutorial example described in section 2.5.

Columns of \mathbf{D}	Description	Typical formula
1–4	Primary features	$r_s(A), r_p(A), r_s(B), r_p(B)$
5–16	All ratios of all pairs of r 's	$r_s(A)/r_p(A)$
17–22	Differences of pairs	$r_s(A) - r_p(A)$
23–34	All differences divided by the remaining r 's	$(r_s(A) - r_p(A))/r_s(B)$
35–40	Absolute values of differences	$ r_s(A) - r_p(A) $
41–43	Sums of absolute values of differences with no r appearing twice	$ r_s(A) - r_p(A) + r_s(B) - r_p(B) $
44–46	Absolute values of sums of differences with no r appearing twice	$ r_s(A) - r_p(A) + r_s(B) - r_p(B) $

sign). The difference of these two almost collinear vectors with the same magnitude (due to the applied standardization) is a vector much shorter than both $r_s(B)$ and $r_p(A)$, but also much shorter than $r_p(B)$ and P , therefore $r_s(B) - r_p(B)$ needs to be multiplied by a relatively large coefficient in order to be comparable with the other vectors. Indeed, while the coefficient of $r_s(B) - r_p(B)$ is about 1.3, all the other coefficients, in particular in the ℓ_1 solution, are much smaller. Such large coefficient is penalized by the $\lambda\|c\|_1$ term and therefore a sub-optimal (according to ℓ_0 -regularized minimization) triplet is selected. This examples shows how with highly correlated features, $\lambda\|c\|_1$ may not be a good approximation for $\lambda\|c\|_0$.

Repeating the LASSO procedure for the matrix \mathbf{D} consisting once only of elements of the first triplet and once of the second triplet, shows that only for $\lambda \leq 0.00079$ the first triplet has a smaller LASSO-error, since then the contribution of the high coefficients to the LASSO-error are sufficiently small. When \mathbf{D} contains all features, at this λ already all coefficients are non-zero.

For the sake of completeness, let us just mention that the RMSE for the pair of descriptors defined by John and Bloch [31] as

$$r_\sigma = |(r_p(A) + r_s(A)) - (r_p(B) + r_s(B))|, \quad (15)$$

$$r_\pi = |r_p(A) - r_s(A)| + |r_p(A) - r_s(A)| \quad (16)$$

is 0.3055 eV. For predicting the energy difference in a linear way, the pair of descriptors ($r_p(A)$, $r_d(B)$) is already slightly better (by 4%) than this.

2.5. A more complex LASSO example: non-linear mapping

Until this point, we have treated only linear models, i.e., where the function $P = f(\mathbf{d})$ is a linear combination of the components of the input vector \mathbf{d} . This is a clear limitation. In this section, we describe how one can easily introduce non-linearities, without loosing the simplicity of the linear-algebra solution. To the purpose, the vector \mathbf{d} is mapped by a (in general nonlinear) function $\Phi: \mathbb{R}^{M_1} \rightarrow \mathbb{R}^{M_2}$ into a higher-dimensional space, and only then the linear methods are applied in \mathbb{R}^{M_2} . This idea is well known in ℓ_2 -regularized *kernel* methods [53], where the so-called kernel trick is exploited. In our case, we aim at explicitly defining and evaluating a higher dimensional-mapping of the initial features, where each new dimension is a non-linear function of one or more initial features.

We stay with the case of 82 compounds. To keep the presentation simple, we leave out the inputs $r_d(A)$ and $r_d(B)$. Hence, every compound is first described by the following four *primary features*,

$$\mathbf{d}_{AB} = (r_s(A), r_p(A), r_s(B), r_p(B)). \quad (17)$$

We will construct a simple, but non-trivial non-linear mapping $\Phi: \mathbb{R}^4 \rightarrow \mathbb{R}^{M^*}$ (with M^* to be defined) and apply LASSO afterwards. The construction of Φ involves some of our pre-knowledge, for example, on dimensional grounds, we expect that expressions like $r_s(A)^2 - r_p(A)$ have no physical meaning and should be avoided. Therefore, in practice, we allow for sums and differences only of quantities with the same units. We hence consider the 46 features listed in table 1: the 4 primary plus 42 *derived* features, building $\Phi(\mathbf{d}_{AB})$, represented by the matrix \mathbf{D} .

The descriptors of John and Bloch [31], r_π and r_σ , are both included in this set, with indexes 41 and 46, respectively. The data are standardized, so that each of the 46 columns of \mathbf{D} has mean zero and variance 1. Note that for columns 5–46, the standardization is applied only after the analytical function of the primary features is evaluated, i.e., for physical consistency, the primary features enter the formula unshifted and unscaled.

Applying LASSO gives the result shown in table 2, where we list the coordinates as they appear when λ decreases. Where we have truncated the list, at $\lambda = 0.028$, the features with non-zero coefficient are:

Table 2. Bookkeeping of (decreasing) λ values at which either a new feature gets non-zero coefficients (marked by a ‘+’ in column action) or a feature passes from non-zero to zero coefficient (marked by a ‘-’). The value of ℓ_0 counts the non-zero coefficients at each reported λ value. In a rather non-intuitive fashion, the number of non-zero coefficients fluctuates with decreasing λ , rather than monotonically increasing; this is an effect of linear correlations in the feature space.

λ	ℓ_0	Feature	Action
0.317	1	$r_p(A)$	+
0.257	2	$ r_p(A) - r_s(A) + r_p(B) - r_s(B) $	+
0.240	3	$r_s(B)/r_p(A)$	+
0.147	4	$r_s(A)/r_p(A)$	+
0.119	3	$ r_p(A) - r_s(A) + r_p(B) - r_s(B) $	-
0.111	4	$r_s(A)$	+
0.104	3	$r_p(A)$	-
0.084	4	$ r_s(B) - r_p(B) $	+
0.045	5	$(r_s(B) - r_p(B))/r_p(A)$	+
0.034	6	$ r_s(A) - r_p(B) $	+
0.028	5	$r_s(A)$	-

$$1 : \frac{r_s(A)}{r_p(A)}, 2 : \frac{r_s(B)}{r_p(A)}, 3 : \frac{r_s(B) - r_p(B)}{r_p(A)}, 4 : |r_s(A) - r_p(B)| \text{ and } 5 : |r_s(B) - r_p(B)|.$$

Let us remark, that the features 2 and 3 and the features 3 and 5 are strongly correlated, with covariance greater than 0.8 for both pairs⁸. This results in a difficulty for LASSO to identify the right features.

With these five descriptor candidates, we ran an exhaustive ℓ_0 test over all the $5 \cdot 4/2 = 10$ pairs. We discovered that the pair of the second and third selected features, i.e. of

$$\frac{r_s(B)}{r_p(A)}, \quad \frac{r_s(B) - r_p(B)}{r_p(A)} \quad (18)$$

achieves the smallest RMSE of 0.1520 eV, improving on John and Bloch’s descriptors [31] by a factor 2. We also ran an exhaustive ℓ_0 search for the optimal pair over all 8 features that were singled out in the LASSO sweep, i.e., including also $r_p(A)$, r_π , and $r_s(A)$. The best pair was still the one in equation (18). We then also performed a full search over the $46 \cdot 45/2 = 1035$ pairs, where the pair in equation (18) still turned out to be the one yielding the lowest RMSE. We conclude that even though LASSO is not able to find directly the optimal Ω -dimensional descriptor, it can efficiently be used for filtering the feature space and single out the ‘most important’ features, whereas the optimal descriptor is then identified by enumeration over the subset identified by LASSO.

The numerical test discussed in this section shows the following:

- A promising strategy is to build an even larger feature space, by combining the *primary features* via analytic formulas. In the next section, we walk through this strategy for systematically constructing a large feature space.
- LASSO cannot always find the best Ω -dimensional descriptor as the first Ω columns of \mathbf{D} with non-zero coefficient by decreasing λ . This is understood as caused by features that are (nearly) linearly correlated. However, an efficient strategy emerges: First using LASSO for extracting a number $\Theta > \Omega$ of ‘relevant’ components. These are the first Θ columns of \mathbf{D} with non-zero coefficients found when decreasing λ . Then, performing an exhaustive search over all the Ω -tuples that are subsets of the Θ extracted columns. The latter is in practice the problem formulated in equations (4) and (5). In the next section, we formalize this strategy, which we call henceforth LASSO + ℓ_0 , because it combines LASSO (ℓ_1) and ℓ_0 optimization.

The feature-space construction and LASSO + ℓ_0 strategies presented in the next section, are essentially those employed in our previous paper [18]. The purpose of this extended presentation is to describe a general approach for the solution of diverse problems, where the only requisite is that the set of basic ingredients (the *primary features*) is known. As a concluding remark of this section, we note that CS and LASSO were successfully demonstrated to help solving quantum-mechanics and materials-science problems in [54–58]. In all those papers, a ℓ_1 based optimization was adopted to select from a well defined set of functions, in some sense the ‘natural basis set’ for the specific problem, a minimal subset of ‘modes’ that maximally contribute to the accurate

⁸ ‘Having a large correlation’ is clearly not a transitive relationship, i.e., if the vector \mathbf{a} is highly correlated with \mathbf{b} (the absolute value of the covariance is close to 1), and \mathbf{b} is highly correlated to \mathbf{c} , \mathbf{c} is not necessarily highly correlated to \mathbf{a} .

Table 3. Comparison of descriptors selected by minimizing the ℓ_2 -norm of the fitting function (the usual LASSO problem) or the ℓ_∞ (maximum norm) over the feature space described in table 1. Reported is also the performance of the models in terms of RMSE and MaxAE.

Ω	Type	RMSE	MaxAE	Descriptor
1	ℓ_2	0.32	1.93	$r_p(A)$
1	ℓ_∞	0.36	1.70	$r_s(A)/r_s(B)$
2	ℓ_2	0.15	0.42	$r_s(B)/r_p(A), (r_s(B) - r_p(B))/r_p(A)$
2	ℓ_∞	0.15	0.42	$r_s(B)/r_p(A), (r_s(B) - r_p(B))/r_p(A)$

Table 4. Set of atomic primary features used for constructing the feature space, divided in groups. The ‘ID’ labels the group and ‘#’ indicates the number of features in the group.

ID	Description	Symbols	#
A1	Ionization potential (IP) and electron affinity (EA)	IP(A), EA(A), IP(B), EA(B)	4
A2	Highest occupied (H) and lowest unoccupied (L) Kohn–Sham levels	H(A), L(A), H(B), L(B)	4
A3	Radius at the max. value of $s, p,$ and d valence radial probability density	$r_s(A), r_p(A), r_d(A)$ $r_s(B), r_p(B), r_d(B)$	6

approximation of the property under consideration. In our case, the application of the ℓ_1 (and subsequent ℓ_0) optimization must be preceded by the construction of a basis set, or feature space, for which a construction strategy is not at all *a priori* evident.

In the discussed numerical tests until this point, we have always looked for the low-dimensional model that minimizes the square error (the square of the ℓ_2 norm of the fitting function). Another quantity of physical relevance that one may want to minimize is the maximum absolute error (maxAE) of the fit. This is called infinity norm and, for the vector \mathbf{x} , it is written as $\|\mathbf{x}\|_\infty = \max|x_k|$. The minimization problem of equation (6) then becomes

$$\operatorname{argmin}_{\mathbf{c} \in \mathbb{R}^M} \|\mathbf{P} - \mathbf{D}\mathbf{c}\|_\infty + \lambda \|\mathbf{c}\|_1. \quad (19)$$

This is still a convex problem as in equation (6). We have looked for the model that gives the lowest MaxAE, starting from the feature space of size 46 defined in table 1.

For the specific example presented here, we find (see table 3) that the 2D model is the same for both settings, i.e., the model that minimizes the RMSE also minimizes the MaxAE. This is, of course, not necessarily true in general. In fact, the 1D model that minimizes the RMSE differs from the 1D model that minimizes the MaxAE.

3. Generation of a feature space

For the systematic construction of the feature space, we first divide the *primary features* in groups, according to their physical meaning. In particular, necessary condition is that elements of each group are expressed with the same units. We start from atomic features, see table 4.

Next, we define the combination rules. We note here that building algebraic function over a set of input variables (in our case, the *primary features*) by using a defined dictionary of algebraic (unary and binary) operators and finding the optimal function with respect to a given cost functional is the strategy of *symbolic regression* [59]. In this field of statistical learning, the optimal algebraic function is searched via an evolutionary algorithm, where the analytic expression is evolved by replacing parts of the test functions with more complex functions. In other words, in symbolic regression, the evolutionary algorithm guides the construction of the algebraic functions of the primary features. In our case, we borrow from symbolic regression the idea of constructing functions by combining ‘building blocks’ in more and more complex way, but the selection of the optimal function (in our language, the descriptor) is determined by the LASSO + ℓ_0 algorithm over the whole set of generated functions, *after* the set of candidate functions is generated.

Our goal is to create ‘grammatically correct’ combinations of the *primary features*. This means, besides applying the usual syntactic rule of algebra, we add a physically motivated constraint, i.e., we exclude linear

Table 5. First set of operators applied to the primary features (table 4). Each group, labeled by a different ID, is formed by starting from a different group of primary features and/or by applying a different operator. The label A stays for both A and B of the binary material.

ID	Description	Prototype formula	#
B1	Absolute differences of A1	$ IP(A) - EA(A) $	6
B2	Absolute differences of A2	$ L(A) - H(A) $	6
B3	Absolute differences and sums of A3	$ r_p(A) \pm r_s(A) $	30
C3	Squares of A3 and B3 (only sums)	$r_s(A)^2, (r_p(A) + r_s(A))^2$	21
D3	Exponentials of A3 and B3 (only sums)	$\exp(r_s(A)), \exp(r_p(A) \pm r_s(A))$	21
E3	Exponentials of squared A3 and B3 (only sums)	$\exp[r_s(A)^2], \exp[(r_p(A) \pm r_s(A))^2]$	21

Table 6. Second set of operators applied to the groups defined in tables 4 and 5.

ID	Description	Prototype formula	#
{F1, F2, F3}	Abs. differences and sums of {B1, B2, B3}, without repetitions	$ r_p(A) \pm r_s(A) \mp r_p(B) \pm r_s(B) ^a$	72
G	ratios of any of {Ai, Bi}, $i = 1, 2, 3$ with any of {A3, C3, D3, E3}	$ r_p(B) - r_s(B) / (r_d(A) + r_s(B))^2$	~ 4300

^a With signs $(-, +, -)$ this is St. John & Bloch's r_m , with signs $(+, -, +)$ it is r_σ .

combinations of inhomogeneous quantities, such as $'IP + r_s'$ or $'r_s + r_p^2'$. In practice, quantities are inhomogeneous when they are expressed with different units. Except this exclusions of physically unreasonable combinations, we produce as many combinations as possible. However, compressed-sensing theory poses a limit on the maximum size M of the feature space from which the best (low-) Ω -dimensional descriptor can be extracted by sampling the feature space with the knowledge of N data points: $N = C \Omega \ln(M)$ [19, 60, 61], when the M candidate features are uncorrelated. C is not a universal constant, however it is typically estimated to be in the range between 4 and 8 (see [22]). For $\Omega = 2$ and $N = 82$, this implies a range of M between ~ 200 and ~ 30000 . Therefore, we regarded values of a few thousand as an upper limit for M . Since the number of thinkable features is certainly larger than few thousands, we proceeded iteratively in several steps, by learning from the previous step what to put and what not in the candidate-feature list of the next step. In the following, we describe how a set of ~ 4500 features was created. In appendixes A and B, we summarize how different feature space can be constructed, starting from different assumptions.

First of all, we form sums and absolute differences of homogeneous quantities and apply some unary operators (powers, exponential), see table 5.

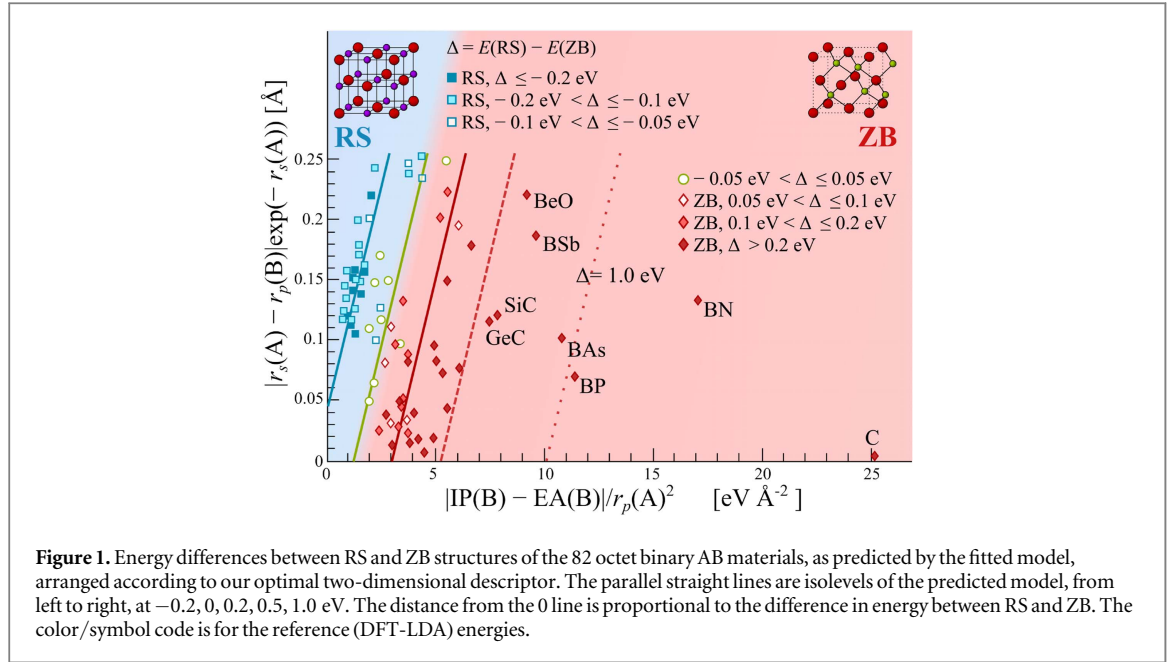
Next, the above combinations are further combined, see table 6.

LASSO was then applied to this set of ~ 4500 candidate features. If the features had low linear correlation, the first two features appearing upon decreasing λ would be the best 2D descriptor, i.e., the one that minimizes the RMSE⁹. Unfortunately, checking all pairs of features for linear correlation would scale with size M as unfavorably as just performing the brute force search for the best 2D descriptor by trying all pairs. Furthermore, such a screening would require the definition of a threshold for the absolute value of the covariance, for the decision whether or not any two features are correlated, and then possibly discarding one of the two. A similar problem would appear in case more refined techniques, like singular-value decomposition, were tried in order to discard eigenvectors with low eigenvalues. Still a threshold should be defined and thus tuned.

We adopted instead a simple yet effective solution: the best $\Omega = 30$ features with non-zero coefficients that emerge from the application of LASSO at decreasing λ are grouped, and among them an exhaustive ℓ_0 minimization is performed (equations (4) and (5) for $\|c\|_0 = 1, 2, 3, \dots$). The single features, pairs, triplets, etc that score the lowest RMSE are the outcome of the procedure as 1D, 2D, 3D, etc, descriptors. The validity of this approach was tested by checking that running it on smaller feature spaces ($M \sim$ few hundreds), where the direct search among all pairs and all triples could be carried out, gave the same result.

Our procedure, applied to the above defined set of features (tables 4–6), found both the best 1D, 2D, and 3D descriptor, as well as the coefficients of the equations for predicting the property ΔE as shown below (energies are in eV and radii are in Å):

⁹ Correlation between pairs of features means in this context that the absolute value of the covariance (i.e., the scalar product of the 82-dimensional vectors containing the values of features i and j , with each vector subtracted of its mean value and divided by its standard deviation) is close to 1. If their covariance is close to zero, the two features are uncorrelated.



$$\Delta E = 0.117 \frac{EA(B) - IP(B)}{r_p(A)^2} - 0.342, \quad (20)$$

$$\Delta E = 0.113 \frac{EA(B) - IP(B)}{r_p(A)^2} + 1.542 \frac{|r_s(A) - r_p(B)|}{\exp(r_s(A))} - 0.137, \quad (21)$$

$$\begin{aligned} \Delta E = & 0.108 \frac{EA(B) - IP(B)}{r_p(A)^2} + 1.737 \frac{|r_s(A) - r_p(B)|}{\exp(r_s(A))} \\ & + 9.025 \frac{|r_p(B) - r_s(B)|}{\exp(r_d(A) + r_s(B))} - 0.030. \end{aligned} \quad (22)$$

We removed the absolute value from ‘ $IP(B) - EA(B)$ ’ as this difference is always negative. In figure 1, we show a *structure map* obtained by plotting the 82 materials, where we used the two components of the 2D descriptor (equation (21)) as coordinates. We note that the descriptor we found contains physically meaningful quantities, like the HOMO-LUMO gap of B in the numerator of the first component and the size difference between valence s - and p -orbitals (numerators of the second and third component).

In closing this section, we note that the algorithm described above can be dynamically run in a web-based graphical application at: https://analytics-toolkit.nomad-coe.eu/tutorial-LASSO_L0.

4. Cross validation (CV), sensitivity analysis, and extrapolation

In this section, we discuss in detail a series of analyses performed on our algorithm. We start with the adopted CV scheme. Then, we investigate how the CV error varies with the size of the feature space (actually, its ‘complexity’, as will be defined below). We proceed by discussing the stability of the choice of the descriptor with respect to sensitivity analysis. Finally, we test the *extrapolation* capabilities of the model.

In the numerical test described above, we have always used all the data for the training of the models; the RMSE given as figure of merit were therefore *fitting errors*. However, in order to assess the predictive ability of a model, it is necessary to test it on data that have not been used for the training, otherwise one can incur the so-called *overfitting* [42]. Overfitting is in general signaled by a noticeable discrepancy between the fitting error (the RMSE over the training data) and the test error (the RMSE over control data, i.e., the data that were not used during the fitting procedure). If a large amount of data is available, one can just partition the data into a training and a test set, fully isolated one from another. In our case, that is not so unusual, the amount of data is too little for such partitioning, therefore, we adopted a CV strategy.

In general, the data set is still partitioned into training and test data, but this procedure repeated several times, choosing different test data, in order to achieve a good statistics. In our case, we adopted a leave-10%-out CV scheme, where the data set is divided randomly into a $\sim 90\%$ of training data (75 data points, in our case) and a $\sim 10\%$ of test data. The model is trained on the training data and the RMSE is evaluated on the test set. By ‘training’, we mean the whole LASSO + ℓ_0 procedure that selects the descriptor and determines the model (the

Table 7. Root mean square error (RMSE) and maximum absolute error (MaxAE) in eV for the fit of all data (first two lines) and of the test set in a leave-10%-out cross validation (L-10%-OCV), averaged over 150 random selections of the training set (last two lines), according to our LASSO + ℓ_0 algorithm.

Descriptor	1D	2D	3D	5D
RMSE	0.14	0.10	0.08	0.06
MaxAE	0.32	0.32	0.24	0.20
RMSE, CV	0.14	0.11	0.08	0.07
MaxAE, CV	0.27	0.18	0.16	0.12

coefficients of the linear equation) as in equations (20)–(22). Another figure of merit that was monitored is the maximum absolute error over the test set. The random selection, training, and error evaluation procedure was repeated until the average RMSE and maximum absolute errors did not change significantly. In practice, we typically performed 150 iterations, but the quantities were actually converged well before. We note that, at each iteration, the standardization is applied by calculating the average and standard deviation only of the data points in the training set. In this way, no information from the test set is used in the training, while if the standardization were computed once and for all over all the available data, some information on the test set would be used in the training. In fact, it can be shown [42] that such approach can lead to spurious performance.

The CV test can serve different purposes, depending on the adopted framework. For instance, in KRR, for a Gaussian kernel, the fitted property is expressed as a weighted sum of Gaussians (see also section 5): $P(\mathbf{d}) = \sum_{i=1}^N c_i \exp(-\|\mathbf{d}_i - \mathbf{d}\|_2^2 / 2\sigma^2)$, where N is the number of training data points, i.e., there are as many coefficients as (training) data points. The coefficients c_i are determined by minimizing $\sum_{i=1}^N (P(\mathbf{d}_i) - P_i)^2 + \lambda \sum_{i,j=1}^N c_i c_j \exp(-\|\mathbf{d}_i - \mathbf{d}_j\|_2^2 / 2\sigma^2)$, where $\|\mathbf{d}_i - \mathbf{d}_j\|_2^2 = \sum_{\alpha=1}^{\Omega} (d_{i,\alpha} - d_{j,\alpha})^2$ is the squared ℓ_2 norm of the difference of descriptors of different materials. A recommended strategy [42] is to use the cross validation to determine the optimal value of the so-called *hyper-parameters*, λ and σ , in the sense that it is selected the pair (λ, σ) that minimizes the average RMSE upon cross validation. In our scheme, we can regard the dimensionality Ω of the descriptor and the size M of the feature space as *hyper-parameters*. By increasing both parameters, we do not observe a minimum, but we rather reach a plateau, where no significant improvement on the CV average RMSE is achieved (see below).

Since in our procedure, the descriptor is found by the algorithm itself, a fundamental aspect of the CV scheme is that all the procedure, including the selection of the descriptor, is repeated from scratch with each training set. This means that, potentially, the descriptor changes at each training-set selection. We found a remarkable stability of the 1D and 2D descriptors. The 1D was the same as the all-data descriptor (equation (20) for 90% of the training sets, while the 2D descriptor was the same as in equation (21) in all cases. For 3D and higher dimensionality, as expected from the $N = C\Omega \ln(M)$ relationship, the selection of the descriptor becomes more unstable, i.e., for different training sets, the selected descriptor often differs in at least one of the three components. The RMSE, however, does not change much from one training set to the other, i.e., the instability of the descriptor selection just reflects the presence of many competing models. We show in table 7 the CV figures of merit, average RMSE and maximum absolute error, as a function of increased dimensionality. We also show in comparison the fitting error.

4.1. Complexity of the feature space

Our feature space is subdivided in 5 tiers.

- In tier zero, we have the 14 primary features as in table 4.
- In tier one, we group features obtained by applying only one unary (e.g., \mathcal{A}^2 , $\exp(\mathcal{A})$) or binary (e.g., $|\mathcal{A} - \mathcal{B}|$) operation on primary features, where \mathcal{A} and \mathcal{B} stand for any primary feature. Note that in this scheme the absolute value, applied to differences, is not counted as an extra operation, i.e., we consider the operator $|\mathcal{A} - \mathcal{B}|$ as a single operator.
- In tier two, two operations are applied, e.g., $\mathcal{A}/(\mathcal{B} + \mathcal{C})$, $(\mathcal{A} - \mathcal{B})^2$, $\mathcal{A}/\mathcal{B}^2$, $\mathcal{A} \exp(-\mathcal{B})$.
- In tier three, we apply three operations: $|\mathcal{A} \pm \mathcal{B}|/(\mathcal{C} + \mathcal{D})$, $|\mathcal{A} \pm \mathcal{B}|/\mathcal{C}^2$, $|\mathcal{A} \pm \mathcal{B}| \exp(-\mathcal{C})$, $\mathcal{A}/(\mathcal{B} + \mathcal{C})^2$, ...
- Tier four: $|\mathcal{A} \pm \mathcal{B}|/(\mathcal{C} + \mathcal{D})^2$, $|\mathcal{A} \pm \mathcal{B}| \exp(-(\mathcal{C} + \mathcal{D}))$, ...

Table 8. Errors after L-10%-OCV. ‘Tier x ’ means that *all tiers up to tier x* are included in the feature space.

	Tier 0	Tier 1	Tier 2	Tier 3	Tier 4	Tier 5
$\Omega = 1$						
RMSE, CV	0.31	0.19	0.14	0.14	0.14	0.14
MaxAE, CV	0.67	0.37	0.32	0.28	0.29	0.30
$\Omega = 2$						
RMSE, CV	0.27	0.16	0.12	0.10	0.10	0.10
MaxAE, CV	0.60	0.39	0.27	0.18	0.19	0.22
$\Omega = 3$						
RMSE, CV	0.27	0.12	0.10	0.08	0.08	0.08
MaxAE, CV	0.52	0.39	0.27	0.16	0.18	0.20

- Tier five: $|\mathcal{A} \pm \mathcal{B}| \exp(-(\mathcal{C} + \mathcal{D})^2)$, ...

Our procedure was executed with tier 0, then with tier 0 AND 1, then with tiers from 0 to 2, and so on. The results are shown in table 8. A clear result of this test is that little is gained, in terms of RMSE, when going beyond tier 3. The reason why MaxAE may increase at larger tiers is that the choice of the descriptor becomes more unstable (i.e., different descriptors may be selected) the larger the feature space is. This leads to less controlled maximal errors, and is a reflection of overfitting.

Incidentally, while for the 1D and 2D descriptor, the results presented in equations (20) and (21) contain only features up to tier 3, the third component of the 3D descriptor shown in equation (22) belongs to tier 4. The 3D descriptor and model limited to tier 3 is:

$$\begin{aligned} \Delta E = & 0.108 \frac{EA(B) - IP(B)}{r_p(A)^2} + 1.790 \frac{|r_s(A) - r_p(B)|}{\exp(r_s(A))} \\ & + 3.766 \frac{|r_p(B) - r_s(B)|}{\exp(r_d(A))} - 0.0267. \end{aligned} \quad (23)$$

This is the same as presented in [18]. The CV RMSE of the 3D model compared to the one in equation (22) is worse by less than 0.01 eV.

4.2. Sensitivity analysis

Cross validation tests if the found model is good only for the specific set of data used for the training or if it is stable enough to predict the value of the target property for unseen data. Sensitivity analysis is a complementary test on the stability of the model, where the data are perturbed, typically by random noise. The purpose of sensitivity analysis can be finding out which of the input parameters maximally affect the output of the model, but also how much the model depends on the specific values of the training data. In practice, the training data can be affected by *measurement errors* even if they are calculated by an *ab initio* model. This is because numerical approximations are used to calculate the actual values of both the primary features and the property. Since, through our LASSO + ℓ_0 methodology, we determine functional relationships between the primary features and the property, applying noise to the primary features and the property is a way of finding out how much the found functional relationship is affected by numerical inaccuracies; in other words, if it is an artifact of the level of accuracy or a deeper, physically meaningful, relationship.

4.2.1. Noise applied to the primary features

In this numerical test, each of the 14 primary features of table 4 was independently multiplied by Gaussian noise with mean 1 and standard deviation $\sigma = 0.001, 0.01, 0.03, 0.05, 0.1, 0.13, 0.3$, respectively. The derived features are then constructed by using these primary features including noise. We also considered multiplying by Gaussian noise (same level as for the independent features) all features at once. The test with independent features reflects the traditional sensitivity analysis test (see e.g., [62]), where the goal is to single out which input parameters maximally affect the results yielded by a model. The test with all features perturbed takes into account that all primary features (as well as the fitted property) are evaluated with the same physical model and computational parameters. Therefore, inaccuracies related to not fully converged computational settings are modeled as noise.

Table 9 summarizes the results. It reports the fractional number of times, in %, in which the 2D descriptor of equation (21) is found by LASSO + ℓ_0 as a function of the noise level. For each noise level, 50 random

Table 9. Number of times the 2D descriptor of the noiseless model (see equation 21) is found when noise is applied to the primary features. The noise is measured in terms of the standard deviation of the Gaussian-distributed set of random numbers that multiply the feature. Only for the 5 features contained in the 2D noiseless model, the noise affects the selection of the descriptor, therefore only 5 out of 14 primary features are listed. The last line shows the effect of noise applied to all primary features simultaneously. The results are displayed for both leave-one-out (LOOCV) and leave-10%-out CV schemes, as indicated by the ‘CV scheme’ column.

Feature	CV scheme	Measure of Gaussian noise, σ						
		0.001	0.010	0.030	0.050	0.100	0.130	0.300
IP(B)	LOOCV	99	99	98	70	4	0	0
IP(B)	L-10%-OCV	84	84	71	51	10	1	0
EA(B)	LOOCV	99	99	99	98	91	86	30
EA(B)	L-10%-OCV	86	84	84	84	80	72	28
r_s (A)	LOOCV	99	99	99	99	96	61	0
r_s (A)	L-10%-OCV	83	87	84	86	72	38	0
r_p (A)	LOOCV	99	98	86	64	2	0	0
r_p (A)	L-10%-OCV	85	85	67	42	0	0	0
r_p (B)	LOOCV	99	99	99	99	81	50	1
r_p (B)	L-10%-OCV	86	85	86	83	72	53	2
All 14	LOOCV	99	98	70	11	0	0	0
All 14	L-10%-OCV	85	82	52	15	0	0	0

extractions of the Gaussian-distributed random number were performed. For the leave-one-out CV (LOOCV) scheme, 82 iterations were performed for each random number, i.e. each materials was once the test material. For the leave-10%-out (L-10%-OCV) 50 iterations were performed, with 50 random selections of 74 materials as training and 8 materials as test set. As expected, for the 9 primary features that do not appear in the 2D descriptor of equation (21), the noise does not affect the final result at any level, i.e., the 2D descriptor of the noiseless model is always found, together with the fitting coefficients.

When one of the 5 features appearing in the 2D descriptor is perturbed, the result is affected by the noise level. For noise applied to some features, the percent of selection of the 2D descriptor of the noiseless model drops faster with the noise level than for others. Of course, even when the 2D descriptor of the noiseless model is found, the fitting coefficients differ from iteration to iteration (each iteration is characterized by a different value of the random noise). For the LOOCV, the RMSE goes from 0.09 eV ($\sigma = 0.001$) to 0.12 eV ($\sigma = 0.3$), while for the L-10%-OCV the RMSE goes from 0.11 to 0.15 eV. So, even when, at the largest noise level, the selected descriptor may vary each time, the RMSE is only mildly affected. This reflects the fact that many competitive models are present in the feature space, and therefore a model yielding a similar RMSE can always be found. It is interesting to note that upon applying noise to IP(B) in *all cases* the new 2D descriptor is

$$\left(\frac{\text{EA(B)}}{r_p(A)^2}, \frac{|r_s(A) - r_p(B)|}{\exp(r_s(A))} \right), \quad (24)$$

i.e, the very similar to the descriptor in equation (21), but here IP(B) is simply missing. It is also surprising that for quite large levels of noise (10%–13%) applied to EA(B), the descriptor containing this feature is selected. In general, up to noise levels of 5%, the descriptor of the noiseless model is recovered the majority of times or more. Therefore, we can conclude that the model selection is not very sensitive to the noise applied to isolated features. When the noise is applied to all features, however, the frequency of recovery of the 2D descriptor of equation (21) drops quickly. Still, for noise levels up to 1%, that could be related, e.g., to computational inaccuracies (non fully converged basis sets, or other numerical settings), the model is recovered almost always.

4.2.2. Adding noise to $P = \Delta E$

We have added *uniformly distributed* noise of size $\delta = \pm 0.01, \pm 0.03, \dots$ eV to the DFT data of ΔE . Here, we have selected two feature spaces of size 2924 and 1568, constructed by two different set of functions, but always including the descriptors of equations (20)–(22). The results are shown in table 10. For $\Omega = 2$, we report the fraction of trials for which the 2D descriptor of the unperturbed data was found in a L-10%-OCV test. (10 selection of random noise were performed and for each selection L-10%-OCV was run for 50 random selections of the training set, so the statistic is over 500 independent selections.) The selection of the descriptor is remarkably stable up to uniform noise of ± 0.1 eV (incidentally, at around the value of the RMSE), then it drops

Table 10. Performance of the model for increasing uniform noise added to the calculated ΔE . Besides the RMSE and the AveMaxAE, we report the number of times, the 2D descriptor of the unperturbed data is recovered.

Number of features	Quantity	Measure of uniform noise, δ (eV)				
		± 0.0	± 0.01	± 0.03	± 0.10	± 0.20
1536	% best 2D descriptor	100	99	99	93	63
	RMSE (eV)	0.10	0.11	0.11	0.12	0.17
	AveMaxAE (eV)	0.18	0.18	0.18	0.23	0.32
2924	% best 2D descriptor	96	94	93	86	68
	RMSE (eV)	0.11	0.11	0.12	0.13	0.18
	AveMaxAE (eV)	0.19	0.20	0.22	0.24	0.34

Table 11. Performance of the model found by LASSO + ℓ_0 when diamond and BN are excluded from the training. The rightmost column reports the LDA cohesive energy *per atom* of the ZB structure, referred to spinless atoms as used for determining the primary features in this work.

ΔE (eV)		E_{coh} (eV)
LDA	Predicted	LDA
-2.64	-1.44	-10.14
-1.71	-1.37	-9.72

rapidly. We note that the errors stay constant when the noise is in the ‘physically meaningful’ regime, i.e., the relative ordering of the materials along the ΔE scale is not much perturbed. Only when the noise starts mixing the relative order of the materials, then the prediction becomes also less and less accurate in terms of RMSE.

4.3. Extrapolation: (re)discovering diamond

Most of ML models, in particular kernel-based models, are known to yield unreliable performance for extrapolation, i.e., when predictions are made for a region of the input data where there are no training data. We note that, in condensed-matter physics, the distinction between what systems are similar and suitable for interpolation and what are not is difficult if not impossible. We test the extrapolative capabilities of our LASSO + ℓ_0 methodology, by setting up two exemplary numerical tests. In the first test, we remove from the training set the two most stable ZB materials, namely C-diamond and BN (the two rightmost points in figure 1), and then calculate for both of them ΔE , as predicted by the trained model. Although the prediction errors of 1.2 and 0.34 eV for C and BN, respectively, are very large, as can be seen in table 11 for the 2D descriptor, the model still predicts C and BN as the most stable ZB structures. Thus, in a setup where C and BN were unknown, the model would have predicted them as good candidates to be the most stable ZB materials.

In the other test, we remove from the training set all four carbon-containing materials, namely C-diamond, SiC, GeC, and SnC, and then calculate for all of them ΔE , as predicted by the trained model. The results are reported in table 12 for the model based on the 2D descriptor. The prediction error for C-diamond is comparable to the first numerical test, and also the other errors are relatively large. However, the remarkable thing, here, is that the trained model does not know anything about carbon as a chemical element, nevertheless, it is able to predict that it will form ZB materials, and the relative magnitude of ΔE is respected.

We conclude that a LASSO + ℓ_0 based model is likely to have good, at least qualitative, extrapolation capabilities. This is owing to the stability of the linear model and the physical meaningfulness of the descriptor, which contains elements of the chemistry of the chemical species building the material.

In closing this section, that we cannot draw general conclusions from the particularly robust performance of the descriptors that are found by our LASSO + ℓ_0 algorithm when applied to the features space constructed as explained above. The tests described in this section, however, form a useful basis for assessing the robustness of a found model. We regard such or similar strategy to be good practice. Two criteria give us confidence that the found models may have a physical meaning: these are the particular nature of the models found by our methodology, i.e., they are expressed as explicit analytic functions of the primary features, and the evidence of robustness with respect to perturbations on the training set and the primary features. We also note that the

Table 12. Performance of the model found by LASSO + ℓ_0 when the carbon atom is excluded from the training, i.e., C-diamond, SiC, GeC, SnC are excluded from the training. The rightmost column reports the LDA cohesion energy *per atom* of the ZB structure, referred to spinless atoms as used for determining the primary features in this work.

Material	ΔE (eV)		E_{coh} (eV) LDA
	LDA	Predicted	
C	-2.64	-1.37	-10.14
SiC	-0.67	-0.48	-8.32
GeC	-0.81	-0.46	-7.28
SnC	-0.45	-0.23	-6.52

Table 13. Performance of the KRR method with various descriptors, in terms of the minimum cross-validation RMSE over a grid of 30×30 (λ , σ) values. Descriptor 4 is built with the IUPAC group (G, from 1 to 18) and period (R from row, to avoid confusion with the property (P) of the elements in the PTE; performance of this set of four possible primary features with LASSO + ℓ_0 is discussed in appendix B.1. Descriptors 5 contains the 4 primary features used for building descriptor 2. Similarly, descriptor 6 contains the 5 primary features found in descriptor 3. Descriptor 7 contains all 14 atomic features listed in table 4. Descriptor 8 contains the 14 atomic features plus 9 dimer features (see table A1 in appendix A).

ID	Dim.	Descriptors	(λ , σ)	RMSE (eV)
1	2D	Z_A, Z_B	($1 \cdot 10^{-6}$, 0.008)	0.13
2	2D	John and Bloch's r_σ and r_π	($7 \cdot 10^{-6}$, 0.008)	0.09
3	2D	our 2D	($7 \cdot 10^{-6}$, 0.73)	0.10
4	4D	G(A), G(B), R(A), R(B)	($1 \cdot 10^{-6}$, 0.14)	0.09
5	4D	$r_s(A), r_p(A), r_s(B), r_p(B)$	($2 \cdot 10^{-6}$, 0.14)	0.08
6	5D	IP(B), EA(B), $r_s(A), r_p(A), r_p(B)$	($7 \cdot 10^{-6}$, 0.14)	0.07
7	14D	All atomic features	($1 \cdot 10^{-6}$, 0.42)	0.09
8	23D	All atomic and dimer features	($1 \cdot 10^{-6}$, 6.72)	0.24

functional relationships between a subset of the primary features and the property of interest that are found by our methodology cannot be automatically regarded as physical laws. In fact, both the primary features and ΔE are determined by the Kohn-Sham equations where the physically relevant input only consists of the atomic charges.

5. Comparison to Gaussian-KRR with various descriptors

In this section, we use Gaussian-KRR to predict the DFT-LDA ΔE for the 82 octet binaries, with various descriptors built from our primary features (see table 4) or functions of them. The purpose of this analysis is to point out pros and cons of using KRR, when compared to an approach such as our LASSO + ℓ_0 . In the growing field of data analytics applied to materials-science problems, KRR is perhaps the ML method that is most widely used to predict properties of a given set of molecules or materials [5, 8, 63, 64].

KRR solves the nonlinear regression problem:

$$\operatorname{argmin}_c \sum_{j=1}^N \left(P_j - \sum_{i=1}^N c_i k(\mathbf{d}_i, \mathbf{d}_j) \right)^2 + \lambda \sum_{i,j=1}^N c_i k(\mathbf{d}_i, \mathbf{d}_j) c_j, \quad (25)$$

where P_j are the data points, $k(\mathbf{d}_i, \mathbf{d}_j)$ is the kernel matrix built with the descriptor \mathbf{d} , and λ is the regularization parameter, with a similar role as λ in equations (3), (5), and (6). In KRR, λ is determined by minimizing the CV error. The fitting function determined by KRR is therefore $P(\mathbf{d}) = \sum_{i=1}^N c_i k(\mathbf{d}_i, \mathbf{d})$, i.e. a weighted sum over all the data points. The crucial steps for applying this method are the selection of the descriptor and of the kernel. The most commonly used kernel is the Gaussian kernel: $k(\mathbf{d}_i, \mathbf{d}_j) = \exp(-\|\mathbf{d}_i - \mathbf{d}_j\|_2^2 / 2\sigma^2)$. The parameter determining the width of the Gaussian, σ , is recommended [42, 65] to be determined together with λ , by minimizing the CV error, and this is the strategy used here. The results are summarized in table 13. In each case, the optimal (λ , σ) was determined by running LOOCV.

We make the following observations:

Table 14. KRR prediction of ΔE (all in eV) for C and BN, when these two materials are not included in the training set for various descriptors, compared to the LASSO + ℓ_0 result and the LDA reference. Descriptor CBN is the same 2D descriptor found by LASSO + ℓ_0 for this dataset (see table C1, row b) and descriptor CBN* is built with the 5 primary features found in descriptor CBN.

Method	Descriptor	Dim.	(λ, σ)	$\Delta E(\text{BN})$	$\Delta E(\text{C})$
LDA				-1.71	-2.64
LASSO + ℓ_0	CBN	2		-1.37	-1.44
KRR	CBN	2	$(1 \cdot 10^{-6}, 0.14)$	-2.85	-4.30
KRR	CBN*	5	$(7 \cdot 10^{-6}, 0.24)$	-0.96	-1.35
KRR	Z_A, Z_B	2	$(1 \cdot 10^{-6}, 0.0085)$	-0.68	-0.56
KRR	G(A), G(B), R(A), R(B)	4	$(1 \cdot 10^{-6}, 0.079)$	-0.50	-1.29
KRR	r_σ and r_π	2	$(0.013, 0.045)$	-1.00	-1.17
KRR	$r_s(A), r_p(A), r_s(B), r_p(B)$	4	$(1 \cdot 10^{-6}, 0.14)$	-1.27	-2.31
KRR	All atomic features	14	$(1 \cdot 10^{-6}, 0.42)$	-1.01	-1.91

Table 15. KRR prediction of ΔE (all in eV) for all four carbon compounds, when these materials are not included in the training set for various descriptors, compared to the LASSO + ℓ_0 result and the LDA reference. Descriptor xC* is the same 2D descriptor found by LASSO + ℓ_0 for this dataset (see table C1, row d) and descriptor xC* is built with the 5 primary features found in descriptor xC.

Method	Descriptor	Dim.	(λ, σ)	$\Delta E(\text{C})$	$\Delta E(\text{SiC})$	$\Delta E(\text{GeC})$	$\Delta E(\text{SnC})$
LDA				-2.64	-0.67	-0.81	-0.45
LASSO + ℓ_0	xC	2		-1.37	-0.48	-0.46	-0.23
KRR	xC	2	$(1.3 \cdot 10^{-5}, 0.079)$	-3.05	-0.66	-0.71	-0.24
KRR	xC*	5	$(5.7 \cdot 10^{-4}, 0.42)$	-2.28	-0.48	-0.44	-0.17
KRR	Z_A, Z_B	2	$(7.3 \cdot 10^{-3}, 0.015)$	-2.38	-0.22	-0.59	-0.29
KRR	G(A), G(B), R(A), R(B)	4	$(1 \cdot 10^{-6}, 0.13)$	-2.28	-0.48	-0.47	-0.28
KRR	r_σ and r_π	2	$(1.6 \cdot 10^{-4}, 0.079)$	-1.96	-0.67	-0.50	-0.31
KRR	$r_s(A), r_p(A), r_s(B), r_p(B)$	4	$(1.3 \cdot 10^{-5}, 0.079)$	-3.06	-0.66	-0.70	-0.24
KRR	All atomic features	14	$(1 \cdot 10^{-6}, 0.42)$	-1.55	-1.24	-0.31	-0.04

- With several atomic-based descriptors, KRR fits reach levels of RMSE comparable to or slightly better than our fit with the LASSO + ℓ_0 .
- However, the performance of KRR is not improving with the dimensionality of the descriptor: descriptor 7 contains the same features as descriptor 5 or 6, plus other, possibly relevant, features. One thus expects a better performance, which is not the case. The same happens when going to all 23 atomic and dimer features (descriptor 8).

5.1. Prediction test with KRR

We have repeated the tests as in section 4.3, i.e., we have trained a KRR model for all materials except C and BN and for all materials except all four carbon compounds. Then we have evaluated the predicted ΔE for the excluded materials. This test was done by using descriptors 1, 2, 4, 5, and 7 from table 13. Furthermore, the 2D LASSO + ℓ_0 descriptors were evaluated for the two datasets described above (details on these descriptors are given in appendix C). These two 2D descriptors are analytical functions of five primary features each and these were also used as a 5D descriptor. In all cases, we have determined the dimensionless hyperparameters λ and σ by minimizing the RMSE over a LOOCV run, and the descriptors are normalized component by component. Each component of the descriptor is *normalized* by the ℓ_2 norm of the vector of the values of that component over the whole *training* set. The results are shown in tables 14 and 15.

The performance of KRR in predicting the ΔE of selected subsets of materials is strongly dependent on the descriptor. In particular, when KRR is used for extrapolation (descriptor CBN, where C and BN data points are distant from the other data points in the metric defined by this 2D descriptor), the performance is rather poor in terms of quantitative error, even though still correctly predicting BN and C as very stable ZB materials. Some descriptors expected to carry relevant physical information, such as the set $(r_s(A), r_p(A), r_s(B), r_p(B))$, show also good predictive ability in these examples.

In summary, KRR is a viable alternative to the analytical models found by LASSO + ℓ_0 (as also noted in [14]), but only when a good descriptor is identified. The strength of our LASSO + ℓ_0 approach is that the descriptor is

determined by the method itself. Most importantly, LASSO + ℓ_0 is not fooled by features that are redundant or useless (i.e., carrying little or no information on the property). These features are simply discarded. In contrast, KRR cannot discard components of a descriptor, resulting in decreasing predictive quality when a mixture of relevant and non-relevant features is naively used as descriptor.

6. Conclusions

We have presented a compressed-sensing methodology for identifying descriptors (parameters that describe the material and its properties of interest) based on physically meaningful quantities and for quantitatively predicting properties relevant for materials-science. The methodology starts from introducing possibly relevant *primary features* that are suggested by physical intuition and pre-knowledge on the specific physical problem. Then, a large feature space is generated by listing nonlinear functions of the primary features. Finally, few features are selected with a compressed-sensing based method, that we call LASSO + ℓ_0 because it uses the LASSO for a prescreening of the features, and an ℓ_0 -norm minimization for the identification of the few most relevant features. This approach can deal well with linear correlations among different features. We analyzed the significance of the descriptors found by the LASSO + ℓ_0 methodology, by discussing the interpolation ability of the model based on the found descriptors, the robustness of the models in terms of stability analysis, and their extrapolation capability, i.e., the possibility of predicting new materials.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 676580, The NOMAD Laboratory, a European Center of Excellence, the BBDC (contract 01IS14013E), and the Einstein Foundation Berlin (project ETERNAL). J V was supported by the ERC CZ grant LL1203 of the Czech Ministry of Education and by the Neuron Fund for Support of Science. This research was initiated while C D, L M G, M S, and J V were visiting the Institute for Pure and Applied Mathematics (IPAM), which is supported by the National Science Foundation (NSF).

Appendix A. More details on the construction of the feature space

In order to determine the final feature space as described in section 3, we proceeded in this way:

- As scalar features describing the valence orbitals, we use the radii at which the radial probability densities of the valence s , p , and d orbitals have their maxima. This type of radii was, in fact, selected by our procedure, as opposed to the average radii (i.e., the quantum-mechanical expectation value of the radius). To the purpose, first two feature spaces starting from both sets of radii as primary features were constructed. In practice, in one case we started from the same primary features as in table 4, but without group A2 (in order to reduce the dimensionality of the final space), in the other case we substituted the average radii in group A3, again without group A2. We then constructed both spaces following the rules of tables 5 and 6. Finally, we joined the spaces and applied LASSO + ℓ_0 to this joint space. Only features containing the radii at maximum were selected among the best.
- Similarly, we also defined three other radii-derived features for the atoms: the radius of the highest occupied orbital of the neutral atom, r_0 , and analogously defined radii for the anions, r_- , and the cations, r_+ . r_0 is either r_s or r_p as defined above, depending on which valence orbital is the HOMO. As in the previous point, we constructed a feature space containing both $\{r_0, r_-, r_+\}$ and $\{r_s, r_p, r_d\}$ and their combinations, and found that only the latter radii were selected among the best.
- We have considered in addition features related to the AA, BB and AB dimers (see table A1). These new features were combined in the same way as the groups A1, A2, and A3, respectively (see tables 4–6). After running our procedure, we found that features containing dimer-related quantities are never selected among the most prominent ones.
- We constructed in sequence several alternative sets of features, in particular varying systematically the elements of group G (see table 6). Multiplication of the $\{A_i, B_i, E_i\}$, ($i = 1, 2, 3$), by the $\{A_3, B_3\}$ was included, as well as division of $\{A_i, B_i, E_i\}$, ($i = 1, 2, 3$), by the $\{A_3, B_3\}$ cubed (instead of squared as in table 6). Only division by C3 were selected by LASSO. At this stage, a descriptor in the form

Table A1. Set of primary features related to homo- and hetero-nuclear (spinless) dimers.

ID	Descriptor	Symbols	#
A1'	Binding energy	$E_b(\text{AA}), E_b(\text{BB}), E_b(\text{AB})$	3
A2'	HOMO-LUMO KS gap	$\text{HL}(\text{AA}), \text{HL}(\text{BB}), \text{HL}(\text{AB})$	3
A3'	Equilibrium distance	$d(\text{AA}), d(\text{BB}), d(\text{AB})$	3

$$\left(\frac{|\text{IP}(\text{B}) - \text{EA}(\text{B})|}{r_p(\text{A})^2}, \frac{|r_s(\text{A}) - r_p(\text{B})|}{r_s(\text{A})^2}, \frac{|r_p(\text{B}) - r_s(\text{B})|}{r_d(\text{A})^2} \right) \quad (\text{A.1})$$

was found. The persistence of the C3 group in the denominator suggested to try other decaying functions of r and $r + r'$; for instance, exponentials as defined in D3 and E3. Interestingly, when the set of features containing C3, D3, and E3 was searched, the second and third component of the above descriptor were substituted by corresponding forms where the denominator squared is replaced by exponentials of the same atomic features (see below). This descriptor was therefore found by LASSO, in the sense that the substitutions $1/r_s(\text{A})^2 \rightarrow \exp(-r_s(\text{A}))$ and $1/r_d(\text{A})^2 \rightarrow \exp(-r_d(\text{A}))$ are an outcome of the LASSO procedure, not of a directly attempted substitution.

- The fact that all selected features belong to group G (see table 6), which is the most populated, is *not* due to the fact that members of the other groups are ‘submerged’ by the large population of group G and ‘not seen’ by LASSO. We have run extensive tests on the groups excluding G and, indeed, the best-performing descriptors yield RMSE larger than those we have found.
- By noticing that the features in group A2 (see table 4) never appear in the selected descriptors, and that the information carried by the features in A1 is similar to those in A2, we investigated what happens if A1 is removed from the primary features (and therefore all derived features containing primary features from A1 are removed from the feature space). We find the following models

$$\Delta E = 0.518 \frac{|\text{H}(\text{A}) - \text{L}(\text{B})|}{\exp[r_p(\text{A})^2]} - 0.174, \quad (\text{A.2})$$

$$\Delta E = 4.335 \frac{r_d(\text{A}) - r_s(\text{B})}{(r_p(\text{A}) + r_d(\text{A}))^2} + 16.321 \frac{r_s(\text{B})}{\exp[(r_p(\text{A}) + r_p(\text{B}))^2]} - 0.406. \quad (\text{A.3})$$

In practice, $|\text{H}(\text{A}) - \text{L}(\text{B})|$ replaces $|\text{IP}(\text{B}) - \text{EA}(\text{B})|$ in the 1D model, with a similar denominator as in equation (21). However, the 2D model does not contain any feature from A2 and it is all built with features from A3. The RMSE of the 1D model of equation (A.2) is 0.145 eV, slightly worse than 0.142 eV for the model of equation (20). The RMSE of the 2D model in equation (A.3) is 0.109 eV, compared to 0.099 eV for equation (21). Equation (A.3) is also the best LASSO + ℓ_0 model if the features space is constructed by using only A3 as primary features. In the latter case, the 1D model would be (RMSE = 0.160 eV):

$$\Delta E = 19.384 \frac{r_p(\text{B})}{\exp[(r_s(\text{A}) + r_s(\text{B}))^2]} - 0.257. \quad (\text{A.4})$$

Appendix B. Other feature spaces

In this appendix, we show the performance of our algorithm with feature spaces constructed from completely different *primary features* than in the main text. The purpose is to underline the importance of the choice of the initial set of features. We note that when combination rules are established, performing new numerical tests takes just the (human) time to tabulate the new set of primary features for the data set.

B.1. Primary features including valence and row of the PTE

We included the ‘periodic-table coordinates’, period (labeled as R for row) and group (labeled as G, from 1 to 18 according to the IUPAC numbering) as features. The reason for this test is to see whether by introducing more information than just the atomic number Z_A and Z_B (see [18]), a predictive model for ΔE is obtained. The new information is the explicit similarity between elements of the same group (they share coordinate G), that is not contained in the atomic number Z . First we started with only R(A), R(B), G(A), and G(B) as primary features and constructed a features space using the usual combination rules. We note that G(B) is redundant for many but not

Table B1. Feature space with primary features R (row or period) and G (group) of the PTE. Performance over the whole set of 82 materials.

	$\Omega = 1$	$\Omega = 2$	$\Omega = 3$
RMSE (eV)	0.20	0.19	0.17
MaxAE (eV)	0.69	0.71	0.62

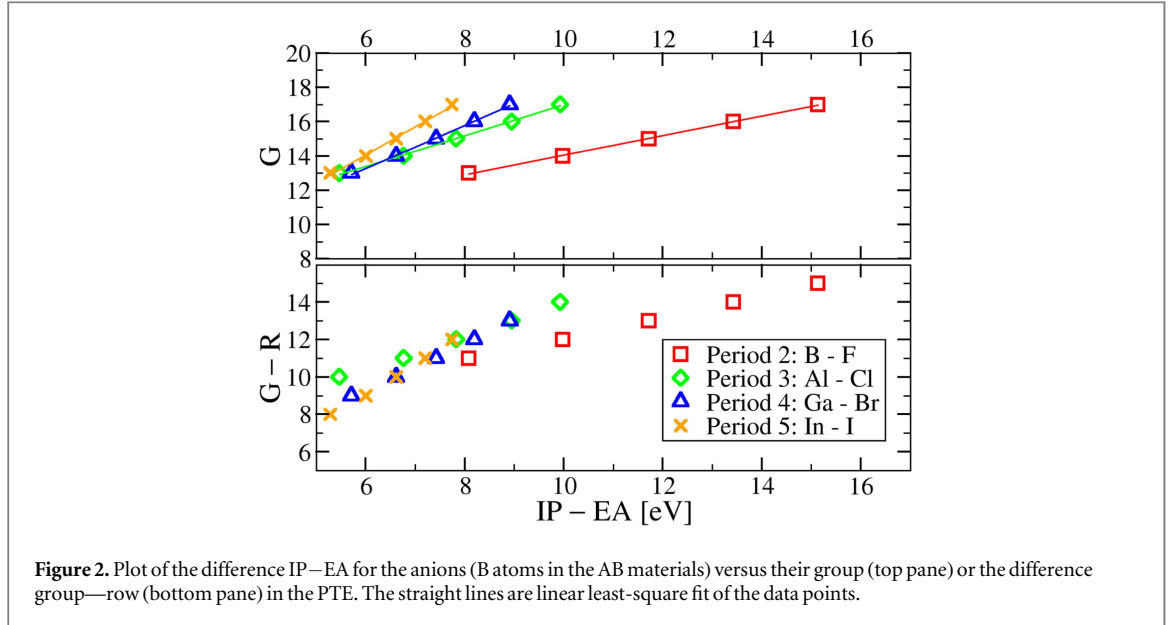


Figure 2. Plot of the difference $IP - EA$ for the anions (B atoms in the AB materials) versus their group (top pane) or the difference group—row (bottom pane) in the PTE. The straight lines are linear least-square fit of the data points.

all the cases, given $G(A)$. In fact, for sp octet binaries, the sum of the groups is 18, but for binaries containing Cu, Ag, Cd, and Zn (16 out of 82), the sum is different and therefore the coordinates are effectively 4. Next, we construct a feature space starting from the set of 14 primary features described in table 4—but where the group A_2 (the HOMO and LUMO Kohn-Sham levels) are substituted with $R(A)$, $R(B)$, $G(A)$, and $G(B)$ —and then following the rules summarized in tables 5 and 6.

For the case where the primary features are only $R(A)$, $R(B)$, $G(A)$, and $G(B)$, we generated a feature space of size 1143 and then ran $LASSO + \ell_0$. The outcome is shown in table B1. We conclude that $R(A)$, $R(B)$, $G(A)$, and $G(B)$ alone do not contain enough information for building a predictive model, following our algorithm.

For the case where the primary features are 14, including $R(A)$, $R(B)$, $G(A)$, and $G(B)$, we constructed a feature space of size 4605, and $LASSO + \ell_0$ found the following 1D optimal model:

$$\Delta E = -0.376 + 0.0944 \frac{|R(B) - G(B)|}{r_p(A)^2}. \quad (B.1)$$

In essence, $|R(B) - G(B)|$ replaced $|IP(B) - EA(B)|$ in equation (20), while the denominator remained unchanged. The RMSE of this new 1D model is only slightly better than the model in equation (20), namely 0.13 eV (versus 0.14 eV), but the MaxAE is much worse, i.e., 0.43 versus 0.32 eV. However, this finding is remarkable as it implies some correlation between $|R(B) - G(B)|$ and $|IP(B) - EA(B)|$, where the latter is a DFT-LDA property while the former comes simply from the number of electrons and the Aufbau principle. Indeed, there is a Pearson's correlation of 0.87 between the two quantities, while, when both quantities are divided by $r_p(A)^2$, the Pearson's correlation becomes as high as 0.98. The 2D and 3D optimal descriptor, though, are the same as in equations (21) and (22).

Even though it is expected that the difference $IP - EA$ grows when moving in the PTE to the right, a linear correlation between the difference $|R(B) - G(B)|$ and $IP - EA$ it is unexpected. Figure 2 shows in the bottom panel this correlation for the p elements (the anions B in the AB materials) of the PTE, while the top panel shows an even stronger linear correlation between $IP - EA$ and the group G in the PTE, for each period.

B.2. Adding John and Bloch's r_σ and r_π as primary features

Here, we added r_σ and r_π [31] to the primary features, in order to see whether combined with other features, according to our rules, yielded a more accurate and/or more robust modeling. The feature space was reduced, as plainly adding all the combinations with these 2 extra features, made the whole procedure unstable (remember that we have only 82 data points). By optimizing over a feature space of size 2924 and using all 82 materials for

Table B2. Performance of the model built from primary features based on force-constants.

Ω	RMSE all	RMSE CV	MAE CV	MaxAE CV
1	0.1910	0.1809	0.1473	0.3583
2	0.1532	0.1797	0.1303	0.3836
3	0.1346	0.2195	0.1553	0.4810

learning and testing, LASSO + ℓ_0 identified the same 2D descriptor as in equation (21). In other words, no function of r_σ or r_π won over the known descriptor. For the L-10%-OCV, in about 10% of the cases, a descriptor containing a function of r_σ or r_π was selected.

B.3. Use of force constants derived from the tetrahedral pentamers AB₄ and BA₄ as primary features

Here, we build a feature space exploring the idea that the difference in energy between RS and ZB may depend on the mechanical stability of the basic constituent of either crystal structure. For instance, we choose ZB and we look at the mechanical stability of the tetrahedral pentamers AB₄ and BA₄. In practice, we look at the elastic constants for the symmetric and antisymmetric stretching deformations.

We write the elastic energy of deformation of a tetrahedral pentamer XY₄, for a symmetric stretch:

$$\Delta E_{\text{harm}}^{\text{SYMM}} = 4\alpha_{\text{XY}} (\Delta r)^2, \quad (\text{B.2})$$

where α_{XY} is the bond-stretching constant (of one XY bond in the tetrahedral arrangement, which is not necessarily the same value as the one of the XY dimer). The factor 4 comes from the fact that there are 4 bonds equally stretched. The second derivative of $\Delta E_{\text{harm}}^{\text{SYMM}}$ with respect to Δr is $(\Delta E_{\text{harm}}^{\text{SYMM}})'' = 8\alpha_{\text{XY}}$. The left-hand side is evaluated from the (LDA) calculated $\Delta E_{\text{harm}}^{\text{SYMM}}(\Delta r)$ curve, fitted to a second order polynomial.

Considering an asymmetric stretch, which means that we moved the central atom X along one of the four XY bonds, we write:

$$\Delta E_{\text{harm}}^{\text{ASYMM}} = 2\alpha_{\text{XY}} (\Delta r)^2 + 3\beta_{\text{YXY}} (\theta_1)^2 + 3\beta_{\text{YXY}} (\theta_2)^2, \quad (\text{B.3})$$

where θ_1 and θ_2 are the two different distortion angles, that the 6 angles formed by the atoms X–Y–X, undergo upon the asymmetric stretch. After working out the geometrical relationship $\theta_{1,2}^2 = \theta_{1,2}^2((\Delta r)^2)$, we find:

$$\Delta E_{\text{harm}}^{\text{ASYMM}} = 2(\alpha_{\text{XY}} + 4\beta_{\text{YXY}})(\Delta r)^2. \quad (\text{B.4})$$

Since the asymmetric deformation is defined through Δr , for convenience, β_{YXY} is referred to the linear displacement, rather than the angular one. In practice, the relationship $\theta^2((\Delta r)^2)$ was derived. The second derivative of the above expression is equated to the second derivative of the calculated $\Delta E(\Delta r)$ curve. Since α_{XY} is known from equation (B.2), β_{YXY} is then inferred.

We have 4 primary features for each material, now. The first two, α_{AB} and α_{ABA} , come from the AB₄ tetrahedral molecule, while the third and fourth ones, α_{BA} and α_{BAB} come from the BA₄ molecule. In addition, by noting that α_{XY} and β_{YXY} enter the expression $\Delta E_{\text{harm}}^{\text{ASYMM}}$, with a ratio of 1/4, two primary features were added, i.e., $\alpha_{\text{AB}} + 4\beta_{\text{ABA}}$ and $\alpha_{\text{BA}} + 4\beta_{\text{BAB}}$. These 6 primary features were combined in (some of) the usual ways. Note that the above combination of α_{BA} and α_{BAB} opens a new level of complexity. At present, when constructing the feature space, we apply operations like A + B and |A – B|, but we do not allow for the freedom of arbitrary linear combinations of them. Note also that the information about the two atoms here are intermingled in all six primary features.

To give an idea, the so obtained 2D descriptor is:

$$\left([\beta_{\text{ABA}} - (\alpha_{\text{BA}} + 4\beta_{\text{BAB}})]\alpha_{\text{AB}}, \frac{\alpha_{\text{AB}} + 4\beta_{\text{ABA}} + \alpha_{\text{BA}} + 4\beta_{\text{BAB}}}{\alpha_{\text{AB}} + \alpha_{\text{BA}}} \right). \quad (\text{B.5})$$

The performance in terms of RMSE and CV is summarized in table B2. We conclude that the features based on the elastic energy do not yield a model with good predictive ability.

Appendix C. Numerical tests excluding selected elements

In this numerical test, we have removed several sets of materials from the data set. In practice, we have removed:

- (0) Nothing
- (a) C-diamond (1 material)
- (b) C-diamond and BN (2 materials)

Table C1. Summary of the training and CV errors for 1-, 2-, 3-dimensional descriptors for different datasets. The last column reports the descriptor found using all data in the dataset for the training and the column ‘Ratio’ reports the fraction of times the same descriptor was found over the L-10%-OCV iterations. CC means C-diamond.

Set	Dimension	RMSE (eV) all	RMSE (eV) CV	MaxAE (eV) CV	Ratio	Descriptor
0) all	1	0.14	0.15	0.28	0.85	Ⓐ
	2	0.10	0.11	0.18	0.99	Ⓐ, Ⓑ
	3	0.08	0.09	0.17	0.95	Ⓐ, Ⓑ, Ⓒ
a) no CC	1	0.12	0.14	0.29	0.91	Ⓓ
	2	0.10	0.13	0.26	0.39	Ⓑ, Ⓒ
	3	0.08	0.08	0.17	0.91	Ⓓ, Ⓑ, Ⓒ
b) no CC, BN	1	0.12	0.14	0.27	0.94	Ⓓ
	2	0.10	0.12	0.26	0.36	Ⓓ, Ⓒ
	3	0.08	0.08	0.16	0.91	Ⓓ, Ⓒ, Ⓑ
c) no BN	1	0.14	0.19	0.39	0.75	Ⓐ
	2	0.10	0.13	0.26	0.83	Ⓐ, Ⓑ
	3	0.08	0.11	0.24	0.87	Ⓐ, Ⓑ, Ⓒ
d) no C	1	0.12	0.13	0.27	0.76	Ⓔ
	2	0.09	0.12	0.23	0.74	Ⓑ, Ⓒ
	3	0.07	0.10	0.20	0.47	Ⓓ, Ⓑ, Ⓒ
e) no B	1	0.13	0.18	0.37	0.37	Ⓖ
	2	0.10	0.14	0.30	0.42	Ⓐ, Ⓑ
	3	0.08	0.12	0.24	0.38	Ⓐ, Ⓑ, Ⓒ
f) no CdO	1	0.14	0.18	0.38	0.77	Ⓐ
	2	0.10	0.12	0.23	0.86	Ⓐ, Ⓑ
	3	0.08	0.10	0.22	0.90	Ⓐ, Ⓑ, Ⓒ
g) no O	1	0.13	0.16	0.33	0.79	Ⓐ
	2	0.10	0.13	0.27	0.74	Ⓐ, Ⓑ
	2	0.08	0.12	0.25	0.33	Ⓐ, Ⓑ, Ⓜ
h) no Cd	1	0.14	0.18	0.38	0.80	Ⓐ
	2	0.10	0.13	0.26	0.84	Ⓐ, Ⓑ
	3	0.08	0.11	0.24	0.89	Ⓐ, Ⓑ, Ⓒ

- (c) BN (1 material)
 (d) All carbon compounds (4 materials)
 (e) All boron compounds (4 materials)
 (f) CdO (example of a material with $\Delta E \sim 0$ (1 material))
 (g) All oxygen-containing compounds (7 materials)
 (h) All cadmium-containing compounds (4 materials)

After the removal, the usual L-10%-OCV was performed on the remaining materials. The purpose was to analyze the stability of the model when some crucial (or less crucial) elements/materials are removed completely from the data set. Also, we aimed of identifying outliers, i.e. data points whose removal from the set improve the accuracy of the fit. The results for a tier-3 feature space of size 2420 are summarized in table C1.

The code for the descriptors is:

$$\begin{aligned}
 \mathfrak{A} &: \frac{\text{IP}(\text{B}) - \text{EA}(\text{B})}{r_p(\text{A})^2}, & \mathfrak{B} &: \frac{|r_s(\text{A}) - r_p(\text{B})|}{\exp(r_s(\text{A}))}, & \mathfrak{C} &: \frac{|r_s(\text{B}) - r_p(\text{B})|}{\exp(r_d(\text{A}))}, \\
 \mathfrak{D} &: \frac{\text{IP}(\text{B})}{\exp[(r_p(\text{A}))^2]}, & \mathfrak{E} &: \frac{|\text{H}(\text{B}) - \text{L}(\text{B})|}{r_p(\text{A})^2}, & \mathfrak{F} &: \frac{\text{H}(\text{B})}{r_p(\text{A})^2}, \\
 \mathfrak{G} &: \frac{\text{IP}(\text{A})}{[r_s(\text{A}) + r_p(\text{A})]^2}, & \mathfrak{H} &: \frac{\text{EA}(\text{A})}{\exp[(r_d(\text{A}))^2]}.
 \end{aligned} \tag{C.1}$$

We make the following observations:

- When C-diamond (and C-diamond together with BN) are excluded from the set, the errors are marginally smaller. However, the descriptor changes in both cases. In particular, the 3D descriptor is remarkably stable both for (b) and (a).
- Removal of all 4 carbon compound leads to a similar behavior as in (a) and (b). However, upon removal of BN or all boron compounds leads to a fit similar to (0).
- Carbon can be thus considered as an anomaly, but it also carries important information for the stability of the overall fit.

Appendix D. PYTHON script for running the LASSO algorithm with the scikit_learn library

```
from sklearn import linear_model
lasso = linear_model.Lasso(alpha=lambda)
lasso.fit(D, P)
coefficients = lasso.coef_
```

Here `sklearn` solves the problem in equation (6), with the ℓ_2 -norm scaled by a factor $\frac{1}{2N}$. The bias c_0 is included by default.

References

- [1] Springer Materials, Landolt-Börnstein, The most comprehensive collection of data in the fields of physics, physical and inorganic chemistry, materials science, and related fields (<http://materials.springer.com/>)
- [2] Lorenz S, Groß A and Scheffler M 2004 *Chem. Phys. Lett.* **395** 210
- [3] Hautier G, Fischer C C, Jain A, Mueller T and Ceder G 2010 *Chem. Mater.* **22** 3762
- [4] Behler J 2011 *Phys. Chem. Chem. Phys.* **13** 17930
- [5] Pilia G, Wang C, Jiang X, Rajasekaran S and Ramprasad R 2013 *Sci. Rep.* **3** 2810
- [6] Szlachta W, Bartók A and Csányi G 2014 *Phys. Rev. B* **90** 104108
- [7] Mueller T, Johlin E and Grossman J 2014 *Phys. Rev. B* **89** 115202
- [8] Schütt K, Glawe H, Brockherde F, Sanna A, Müller K and Gross E 2014 *Phys. Rev. B* **89** 205118
- [9] Atsuto S, Togo A, Hayashi H, Tsuda K, Chaput L and Tanaka I 2015 *Phys. Rev. Lett.* **115** 205901
- [10] Faber F, Lindmaa A, von Lilienfeld O and Armiento R 2015 *Int. J. Quantum Chem.* **115** 1094
- [11] Caccin M, Li Z, Kermode J and De Vita A 2015 *Int. J. Quantum Chem.* **115** 1129
- [12] Isayev O, Fourches D, Muratov E N, Oses C, Rasch K, Tropsha A and Curtarolo S 2015 *Chem. Mater.* **27** 753
- [13] Mannodi-Kanakithodi A, Pilia G, Doan Huan T, Lookman T and Ramprasad R 2016 *Sci. Rep.* **6** 20952
- [14] Kim C, Pilia G and Ramprasad R 2016 *Chem. Mater.* **28** 1304
- [15] Ueno T, Rhone T, Hou Z, Mizoguchi T and Tsuda K 2016 *Mater. Discovery* **4** 18
- [16] Solovyeva A and von Lilienfeld O 2016 *Phys. Chem. Chem. Phys.* **18** 31078–91
- [17] Bialon A F, Hammerschmidt T and Drautz R 2016 *Chem. Mater.* **28** 2550
- [18] Ghiringhelli L M, Vybiral J, Levchenko S V, Draxl C and Scheffler M 2015 *Phys. Rev. Lett.* **114** 105503
- [19] Donoho D 2006 *IEEE Trans. Inform. Theory* **52** 1289
- [20] Candès E J and Wakin M B 2008 *IEEE Signal Process. Mag.* **25** 21
- [21] Davenport M, Duarte M, Eldar Y and Kutyniok G 2012 *Introduction to Compressed Sensing: Theory and Applications* (Cambridge: Cambridge University Press)
- [22] Boche H, Calderbank R, Kutyniok G and Vybiral J 2015 *Compressed Sensing and its Applications* (Berlin: Springer) (<https://doi.org/10.1007/978-3-319-16042-9>)
- [23] Guyon I and Elisseeff A 2003 *J. Mach. Learn. Res.* **3** 1157 (www.jmlr.org/papers/v3/guyon03a.html)
- [24] Kohn W and Sham L 1965 *Phys. Rev.* **140** A1133
- [25] Hohenberg P and Kohn W 1964 *Phys. Rev.* **136** B864
- [26] Ceperley D M and Alder B J 1980 *Phys. Rev. Lett.* **45** 566
- [27] Perdew J P and Wang Y 1992 *Phys. Rev. B* **45** 13244
- [28] van Vechten J A 1969 *Phys. Rev.* **182** 891
- [29] Phillips J C 1970 *Rev. Mod. Phys.* **42** 317
- [30] Bloch A N and Simons G 1972 *J. Am. Chem. Soc.* **94** 8611
- [31] John J S and Bloch A 1974 *Phys. Rev. Lett.* **33** 1095
- [32] Zunger A 1980 *Phys. Rev. B* **22** 5839
- [33] Chelikowsky J 1982 *Phys. Rev. B* **26** 3433
- [34] Pettifor D G 1984 *Solid State Commun.* **51** 31
- [35] Villars P 1985 *Less-Common Met.* **109** 93
- [36] Andreoni W, Galli G and Tosi M 1985 *Phys. Rev. Lett.* **55** 1734
- [37] Andreoni W and Galli G 1987 *Phys. Rev. Lett.* **58** 2742
- [38] Saad Y, Gao D, Ngo T, Bobbitt S, Chelikowsky J R and Andreoni W 2012 *Phys. Rev. B* **85** 104104
- [39] Bertsekas D P 1999 *Nonlinear Programming* (Athena Scientific)
- [40] Arora S and Barak B 2009 *Computational Complexity: A Modern Approach* (Cambridge: University Press)

- [41] Tibshirani R 1996 *J. R. Stat. Soc. B* **58** 267 (www.jstor.org/stable/2346178)
- [42] Hastie T, Tibshirani R and Friedman J 2009 *The Elements of Statistical Learning* (New York: Springer) (<https://doi.org/10.1007/978-0-387-84858-7>)
- [43] Greenshtein E 2006 *Ann. Stat.* **34** 2367
- [44] van de Geer S 2008 *Ann. Stat.* **36** 614
- [45] Bühlmann P and van de Geer S 2011 *Statistics for High-Dimensional Data* (Berlin: Springer) (<https://doi.org/10.1007/978-3-642-20192-9>)
- [46] Cohen A, Dahmen W and DeVore R 2009 *J. Am. Math. Soc.* **22** 211
- [47] Gross D, Liu Y-K, Flammia S T, Becker S and Eisert J 2010 *Phys. Rev. Lett.* **105** 150401
- [48] Blum V, Gehrke R, Hanke F, Havu P, Havu V, Ren X, Reuter K and Scheffler M 2009 *Comp. Phys. Commun.* **180** 2175
- [49] van Lenthe E, Baerends E J and Snijders J G 1994 *J. Chem. Phys.* **101** 9783
- [50] Simons G and Bloch A N 1973 *Phys. Rev. B* **7** 2754
- [51] Chelikowsky J R and Phillips J C 1978 *Phys. Rev. B* **17** 2453
- [52] Andreoni W, Baldereschi A, Biémont E and Phillips J C 1979 *Phys. Rev. B* **20** 4814
- [53] Hofmann T, Schölkopf B and Smola A J 2008 *Ann. Stat.* **36** 1171
- [54] Nelson L J, Hart G L W, Zhou F and Ozoliņš V 2013 *Phys. Rev. B* **87** 035125
- [55] Nelson L, Ozoliņš V, Reese C, Zhou F and Hart G L W 2013 *Phys. Rev. B* **88** 155105
- [56] Ozoliņš V, Lai R, Caffisch R and Osher S 2013 *Proc. Natl Acad. Sci. USA* **110** 18368
- [57] Zhou F, Nielson W, Xia Y and Ozoliņš V 2014 *Phys. Rev. Lett.* **113** 185501
- [58] Budich J, Eisert J, Bergholtz E, Diehl S and Zoller P 2014 *Phys. Rev. B* **11** 115110
- [59] Koza J R 1992 *Genetic Programming* (Cambridge, MA: MIT Press)
- [60] Candès E J, Romberg J and Tao T 2006 *IEEE Trans. Inform. Theory* **52** 489
- [61] Foucart S and Rauhut H 2013 *A Mathematical Introduction to Compressive Sensing* (New York: Springer)
- [62] Meskine H, Matera S, Scheffler M, Reuter K and Metiu H 2009 *Surf. Sci.* **603** 1724
- [63] Rupp M, Tkatchenko A, Müller K-R and von Lilienfeld O A 2012 *Phys. Rev. Lett.* **108** 058301
- [64] Faber F A, Lindmaa A, von Lilienfeld O A and Armiento R 2016 *Phys. Rev. Lett.* **117** 135502
- [65] Hansen K, Montavon G, Biegler F, Fazli S, Rupp M, Scheffler M, von Lilienfeld O A, Tkatchenko A and Müller K-R 2013 *J. Chem. Theory Comput.* **9** 3404