PAPER • OPEN ACCESS

Advancing descriptor search in materials science: feature engineering and selection strategies

To cite this article: Benedikt Hoock *et al* 2022 *New J. Phys.* **24** 113049

View the article online for updates and enhancements.

New Journal of Physics

The open access journal at the forefront of physics

PAPER

CrossMark

Advancing descriptor search in materials science: feature engineering and selection strategies

Benedikt Hoock[1,2], Santiago Rigamonti[1,*] (iD) and Claudia Draxl[1,2] (iD)

[1] Institut für Physik and IRIS Adlershof, Humboldt-Universität zu Berlin, Berlin, Germany
[2] Fritz-Haber-Institut der Max-Planck-Gesellschaft, Berlin, Germany
* Author to whom any correspondence should be addressed.

E-mail: srigamonti@physik.hu-berlin.de

Keywords: compressed sensing, feature engineering, big-data driven materials science, ternary group-IV compounds

Abstract

A main goal of data-driven materials research is to find optimal low-dimensional descriptors, allowing us to predict a physical property, and to interpret them in a human-understandable way. In this work, we advance methods to identify descriptors out of a large pool of candidate features by means of compressed sensing. To this extent, we develop schemes for engineering appropriate candidate features that are based on simple basic properties of building blocks that constitute the materials and that are able to represent a multi-component system by scalar numbers. Cross-validation based feature-selection methods are developed for identifying the most relevant features, thereby focusing on high generalizability. We apply our approaches to an *ab initio* dataset of ternary group-IV compounds to obtain a set of descriptors for predicting lattice constants and energies of mixing. In particular, we introduce simple complexity measures in terms of involved algebraic operations as well as the amount of utilized basic properties.

1. Introduction

On the steady search for advanced materials with tailored properties and novel functions, high-throughput screening has become a popular branch of materials research. Focusing on the computational side, the amount of materials data produced on workstations, compute clusters, and supercomputers is growing exponentially. This situation can be considered as a *big-data problem* as well as a chance—the chance to learn from these data and obtain unprecedented insight, opening new routes for basic materials science and engineering. In this sense, *big data of materials science* is all about the question: How to exploit the wealth of information, inherently inside the materials data? For successfully exploring the chemical compound space, new concepts need to be developed that allow us for extracting knowledge from the materials data, identifying trends, anomalies, and mechanisms.

This is where machine learning (ML), and, more general, artificial intelligence (AI) come into play. Learning from materials data in an automatized manner, as ML/AI may suggest, is, however, far from being trivial, since learning the behavior of the entire quantum-mechanical many-body problem is a too difficult task. To make reliable predictions, the most critical step is to find suitable descriptors, i.e. physically meaningful parameters that describe a material's property best. This will enable us to develop and/or use AI tools to not only find optimal models but also to gain new insights into the underlying physical laws by relating properties and descriptors.

An important aspect of AI is to learn physical quantities that are computationally expensive and time-consuming from data that are easy to obtain. Predicting properties of bulk materials from atomic data or interface properties from bulk results are just two examples out of many possibilities [1]. For finite systems, atomization energies of relaxed molecules of similar kind have been successfully predicted within chemical accuracy [2]. Conversely, due to the long-range interactions, materials properties are much harder to predict. This has challenged both (a) the *feature engineering*, i.e. the design of convenient numerical representations most relevant to a property, and (b) the approaches for feature selection, i.e. the

identification of the relevant features out of a possibly (very) large pool. Important examples for (a) include the generation of features for modeling the potential-energy surface of solids [3–5] or learning the cohesive energies of compounds for arbitrary supercells [6]. More recently, crystal-graph representations applicable to arbitrary crystal structures, so called *n*-grams, were successfully employed, in combination with kernel ridge regression, to predict formation energies and band gaps of transparent conducting oxides [7]. An important example [8, 9], advancing both (a) and (b), concerns a *classical* classification problem, namely separating the binary octet semiconductors into whether they crystallize in rocksalt or zincblende structure. This problem has a long history in materials science (for an overview, see [10]). It has been solved by symbolic regressions via compressed sensing, in particular the LASSO + ℓ_0 [8, 9] and SISSO [11] approaches, and also by subgroup discovery techniques [12]. Other examples are crystal-structure classification problems using neural networks [13], the generation of structure-energy-property landscapes for molecular crystals [14], the prediction of vibrational free energies and entropies [15], the determination of the electrical breakdown field of insulators and the learning of the dielectric constant and band gap of polymers [16], the prediction of band gap energies or glass-forming ability of crystalline and amorphous materials [17], and unified frameworks for predicting atomic-scale properties of molecules and solids [18].

In this work, we advance some of the above methodology to identify suitable descriptors in two main aspects. First, we show how meaningful, yet simple, candidate features can be constructed from basic properties of the materials that constitute the building blocks of multi-component systems. Second, we devise and assess strategies for model selection. Thereby we also consider the predictive power of the model by making use of cross-validation (CV) for both model selection and validation. We demonstrate our approaches with the example of group-IV zincblende ternary alloys, for which we aim at predicting lattice constants and energies of mixing. We use an *ab initio* dataset obtained by density-functional theory on the level of the local-density approximation which yields reasonable accuracy for the quantities of interest in our work, i.e. ∼1% for lattice constants and ∼4% for energies of formation [19]. Since the properties of such alloys are sensitive to both the composition and the atomic arrangement, this poses a challenge to the construction of appropriate candidate features. Generating a feature space for disordered structures has, so far, not been tackled systematically in the context of symbolic regression. Focusing on the two learning tasks, we first determine a set of basic physical properties and mathematical operations for generating the feature space that provides a starting point for applying compressed-sensing methodology [8, 9]. In particular, we examine the quality of the models in terms of a measure of descriptor complexity, where the included primary features as well as the descriptors' algebraic form are considered. Further, we compare the different model-selection strategies based on cross validation as well as LASSO + ℓ_0 and SISSO, that we use for identifying the most meaningful descriptors. Finally, we present a set of *optimal* descriptors as a solution to the two learning tasks and discuss their interpretability.

2. Background

2.1. Building a linear model
As very often in a learning task, one wants to find a linear model that allows one to predict a material property P based on some input data $\mathbf{D}$. More specifically, the aim is to obtain, in the context of compressed sensing, a small subset of input vectors from the potentially huge matrix $\mathbf{D}$ which both gives an accurate prediction $\hat{P}$ of the property and desirably can be interpreted in a physically meaningful way. This subset is called Ω-dimensional descriptor $\mathcal{D}$, where Ω is the number of considered input vectors. Formally, the linear model can be written as

$$\hat{P}_s = \mathbf{D}_s \cdot \mathbf{c} = \sum_{i=1}^{\Omega} F_{sk_i} c_{k_i}. \tag{1}$$

The descriptor $\mathcal{D}$ can then be defined as a set of features $\mathcal{F}_k$ which map a material s to the corresponding set of real numbers, i.e.

$$\mathcal{F}_k : s \rightarrow F_{sk} \in \mathbb{R}, \tag{2}$$

$$\mathcal{D} : s \rightarrow \mathbf{D}_s = (F_{sk_1}, \ldots, F_{sk_\Omega}) \in \mathbb{R}^\Omega. \tag{3}$$

The components of the coefficient vector $\mathbf{c} = (c_{k_1}, c_{k_2}, \ldots, c_{k_\Omega})$ are found through fitting procedure to a training set $S = \{(\mathbf{D}_1, P_1), \ldots, (\mathbf{D}_N, P_N)\}$ where N is the number of training samples. The property P_s in each data-point $(\mathbf{D}_s, P_s)$ must be known beforehand, either from calculations or measurements. The task of finding the appropriate descriptor is achieved by first constructing a potentially huge pool of M candidate features $\{\mathcal{F}_1, \ldots, \mathcal{F}_M\}$ and then learning from this pool the Ω features yielding the *best* model.

2.2. Compressed sensing methods

In order to extract only a few highly relevant features from the plenty of candidates contained in **D** (see section 4), we make use of two different feature-selection methods $\mathcal{M}$, namely LASSO $+ \ell_0$ [8, 9] and SISSO [11] that are based on compressed-sensing theory and are briefly sketched in the following. Utilizing these two methods, we have developed various model-selection strategies, based on CV, which aim at generalizability of the resulting models. They will be described in section 5.

2.2.1. The LASSO $+ \ell_0$ approach

Mathematically, in compressed sensing, the search for a descriptor is expressed by the optimization problem

$$\underset{\mathbf{c}}{\operatorname{argmin}} \left\| \mathbf{P} - \mathbf{D}\mathbf{c} \right\|_2^2 + \lambda \left\| \mathbf{c} \right\|_0. \tag{4}$$

Here, the input matrix **D** is of size $N \times M$ containing the elements F_{sk}. $\left\| \mathbf{v} \right\|_p$ denotes the ℓ_p-norm of **v**, defined as $\left(\sum_{i=1}^{N} |v_i|^p \right)^{1/p}$. For the special case $p = 0$, $\left\| \mathbf{v} \right\|_0$ is defined as the number of non-zero elements of **v**, i.e. $\#\{i \text{ for which } v_i \neq 0\}$. The solution of equation (4) yields the coefficients **c** that minimize the mean square error (MSE) between the target **P** and the prediction of the model **Dc**, on the condition that the vector **c** is *sparse*, i.e. $c_i \neq 0$ for a few $i \in \{1, \ldots, M\}$ only. The parameter λ is a positive real number. Higher values of λ yield sparser models. If the solution has exactly Ω non-zero coefficients c_i, the associated subset of features $\{\mathcal{F}_i\}$ is an Ω-dimensional descriptor.

A solution to this problem consists of extracting all possible descriptors $\mathcal{D}$ of dimension Ω from the pool of features and selecting the one yielding the lowest MSE. This is, however, practically impossible due to the combinatorial explosion of the number of possible descriptors at increasing Ω (*NP*-hard problem [20]). To overcome this computational difficulty, in [8, 9] an approximation to the ℓ_0-problem has been proposed, consisting of two steps. First, one solves the convex optimization problem

$$\underset{\mathbf{c}}{\operatorname{argmin}} \left\| \mathbf{P} - \mathbf{D}\mathbf{c} \right\|_2^2 + \lambda \left\| \mathbf{c} \right\|_1. \tag{5}$$

Compared to equation (4), this equation contains the ℓ_1-norm (also called *Manhattan* norm) of the coefficient vector **c**. This approach is known as the least absolute shrinkage and selection operator (LASSO) [21] and is the best convex proxy to equation (4). It yields a subset of moderate size $\tilde{M}$, containing the most relevant features. Second, equation (4) is then solved exactly by going through all combinations of features, in the subset obtained in the first step. This guarantees that the best Ω-dimensional descriptors out of $\tilde{M}$ are obtained [9].
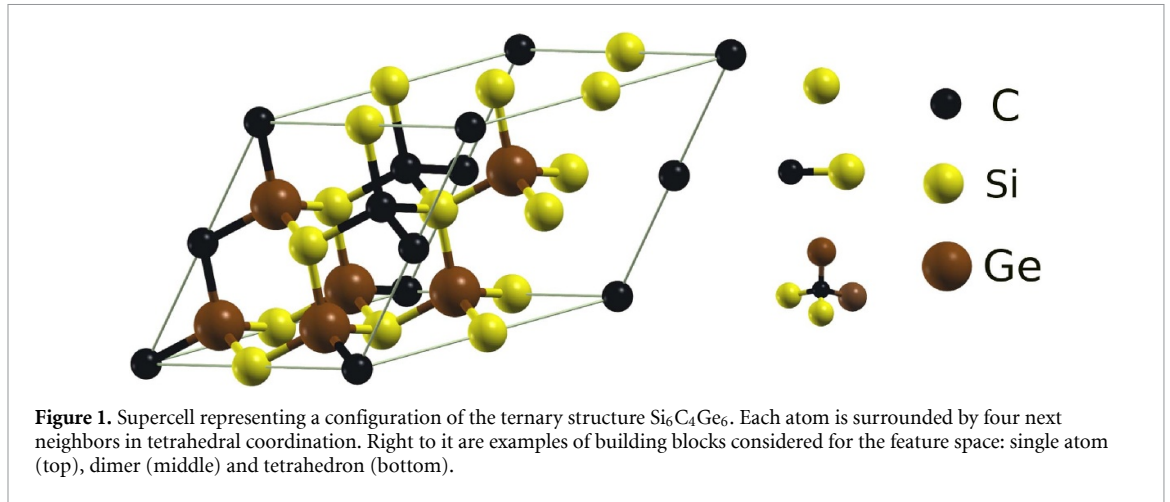
2.2.2. The SISSO approach

Although the LASSO $+ \ell_0$ approach has been successfully applied, e.g. to classification problems [8], it turned out to experience difficulties. This happens on the one hand, at large feature matrix **D** where the procedure becomes computationally very costly, and, on the other hand, if **D** contains highly correlated features. As a solution to this undesirable situation, the *SISSO* method was proposed [11] as an alternative approach to tackle the ℓ_0-problem, also enabling to work with much larger feature spaces.

Similar to LASSO $+ \ell_0$, SISSO combines two feature-selection methods. First, the *sure independence screening* method (SIS) [22] is applied to reduce the number of candidate features to a drastically smaller subset. Second, the *sparsifying operator* (SO) further reduces the dimensionality, eventually determining the Ω-dimensional descriptor. In more detail, SIS determines a first lower-dimensional subspace S_1 of features by choosing those being mostly correlated with the target **P**. Here, the *Pearson correlation coefficient* is used as a measure. SO will then find an optimal 1D descriptor $\mathcal{D}_1$ from S_1 (for 1D, this coincides with the first ranked by SIS). After this, SIS again is used to find a second subspace S_2 that now contains the features of highest correlation with the residual $\mathbf{P} - \mathbf{D}^{(1)}\mathbf{c}_1$ ($\mathbf{D}^{(1)}$ being the feature matrix with rows given by equation (3) with $\mathcal{D} \to \mathcal{D}_1$). Like above, SO determines the best 2D descriptor—but now from the union of subspaces $S_1 \cup S_2$. Thus, this procedure yields a sequence of almost orthogonal subsets and is iterated until a threshold in accuracy or in descriptor dimension Ω is reached. For the selection operator SO, several options are discussed in [11] that may put restrictions on the feasible size of the subspaces. In this work, we choose SO(ℓ_0) to solve the ℓ_0 problem exactly on the subset. Thus, the essential difference between LASSO $+ \ell_0$ and SISSO is the set onto which ℓ_0 is operated, being LASSO or SIS, respectively.

3. Dataset of group-IV ternary alloys

We consider group-IV ternary compounds that are composed of three out of the four elements carbon, silicon, germanium, and tin. This material class is of great interest because of its potential electronic,

Figure 1. Supercell representing a configuration of the ternary structure $Si_6C_4Ge_6$. Each atom is surrounded by four next neighbors in tetrahedral coordination. Right to it are examples of building blocks considered for the feature space: single atom (top), dimer (middle) and tetrahedron (bottom).

photoelectronic, and photovoltaic applications [23–25]. Here, we focus on ternary structures that can be described by a 16-atom supercell (see figure 1) as constructed by doubling the two-atomic zincblende parent lattice in each direction. The concentrations $x, y, 1 - x - y$ of types *A*, *B* and *C* are restricted to $0 < x, y \leqslant 0.5$ and $0 < 1 - x - y \leqslant 0.25$. The code *enumlib* [26] was used to determine all symmetrically distinct structures that satisfy these restrictions. These leads to a total of 504 different configurations, out of which 445 were used [27].

In general, a certain composition (x, y) can be realized by several symmetrically inequivalent atomic arrangements, i.e. (x, y) does not define a structure *s* uniquely. Typically, not only the composition but also the configuration has an impact on the physical properties. This needs to be reflected in the construction of the feature space. Specifically, for modeling the alloys' properties $P(x, y)$, this requires to go beyond pure linear interpolation schemes like Vegard's law [28]
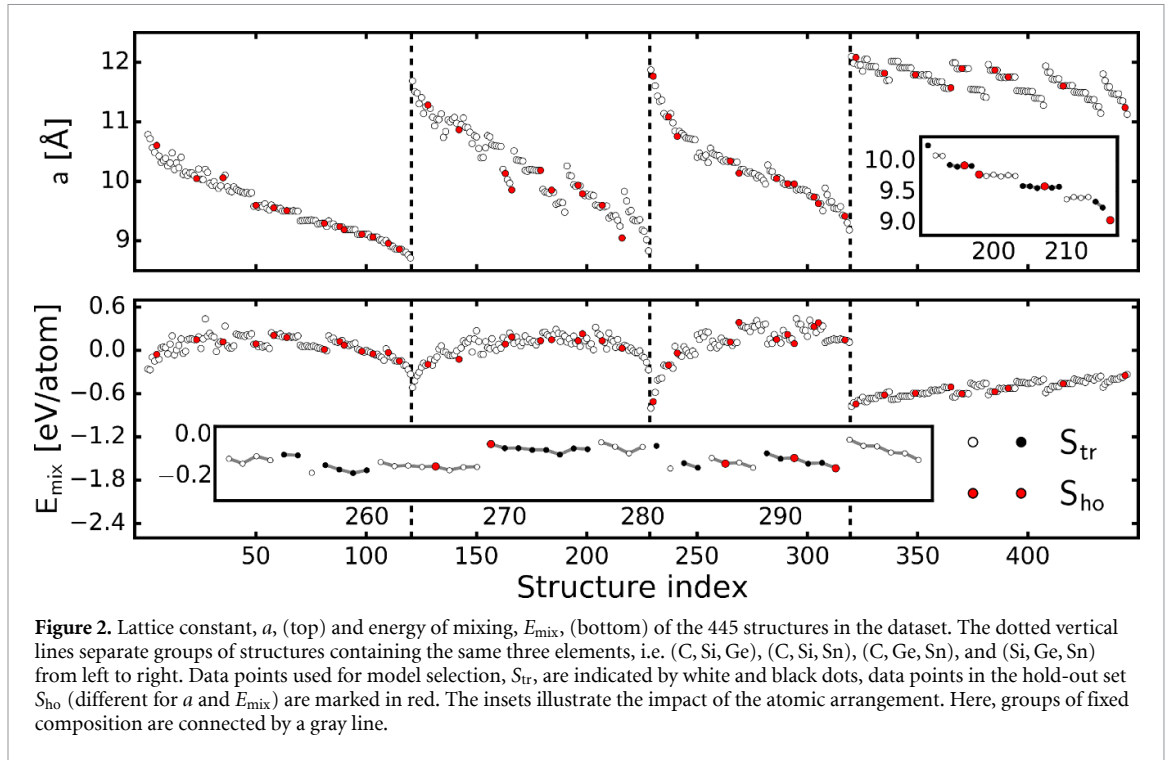
$$P_{\text{lin}}(x, y) = xP_A + yP_B + (1 - x - y)P_C. \tag{6}$$

or interpolation of the properties of the binary compounds P_{AB}, P_{AC}, and P_{BC} [29]. Instead, configuration-aware features must be introduced.

The dataset used in this work was calculated with the all-electron full-potential code `exciting` [30] using the Perdew–Wang exchange-correlation potential [31]. Muffin-tin radii of 1.45, 1.65, 1.8, and 1.9 a_0 were employed for C, Si, Ge, and Sn, respectively, and the basis-set size was determined by the dimensionless parameter $R_{MT,\text{min}} |\mathbf{G} + \mathbf{k}|_{\text{max}} = 7.0$ for C, i.e. the species with the smallest sphere. For the density and the potential, a plane-wave cut-off $|G|$ of 12.0 a_0^{-1} was applied. Brillouin-zone integrations were carried out on a $4 \times 4 \times 4$ **k**-point mesh. The atomic positions were relaxed until all forces were converged better than 3.0×10^{-4} Ha/a_0. The lattice constants *a* were determined by fitting the Murnaghan equation of state [32] to five values of the volume varying by up to $\pm 4\%$ around an initial guess from Vegard's law. A final atomic relaxation of the equilibrium supercell yielded the groundstate energy $E(s)$ for each structure. The energy of mixing per atom, is obtained as $E_{\text{mix}} = [E(s) - E_{\text{lin}}(x, y)]/16$, where $E_{\text{lin}}(x, y) = x \cdot E_A + y \cdot E_B + (1 - x - y) \cdot E_C$ is the linearly interpolated groundstate energy according to equation (6), and E_A (E_B, E_C) is the energy of the pristine crystal in the diamond structure for species *A* (*B*, *C*). Negative (positive) values of E_{mix} indicate that the compound *s* is stable (unstable) against phase separation into the pristine crystals A, B, and C.

For these data, figure 2 shows the two target properties *a* (top) and E_{mix} (bottom). The structures are sorted by ascending concentration of atom type *A*, then *B*, and then *C* where *A*, *B*, and *C* are sorted by ascending nuclear number. The insets, showing several groups of fixed composition, visualize the dependence on the arrangement as mentioned above. This dependence can be as large as 0.22 Å for the lattice constant and 0.41 eV for E_{mix}, considering the entire data set. Further analyzing the dataset, we find that the relaxed atomic positions frequently deviate quite markedly from those of the ideal zincblende structure. For some structures, the bond lengths can differ by up to 10%. We note that even bond lengths within the same bonding motif, for example bonds between silicon and germanium, and within the same structure may take different values.

The data used in this work are available for download at the NOMAD Repository [33, 34], DOI https://doi.org/10.17172/NOMAD/2022.05.20-1.

Figure 2. Lattice constant, *a*, (top) and energy of mixing, E_{mix}, (bottom) of the 445 structures in the dataset. The dotted vertical lines separate groups of structures containing the same three elements, i.e. (C, Si, Ge), (C, Si, Sn), (C, Ge, Sn), and (Si, Ge, Sn) from left to right. Data points used for model selection, S_{tr}, are indicated by white and black dots, data points in the hold-out set S_{ho} (different for *a* and E_{mix}) are marked in red. The insets illustrate the impact of the atomic arrangement. Here, groups of fixed composition are connected by a gray line.

4. Construction of feature spaces

How to build a feature space for materials-science problems, has been extensively described in [8, 9]. In short, one first defines a set of *primary features*, $\{f_k\}$, that are basic physical quantities of lower complexity than the target properties, i.e. requiring (much) lower computational cost than what is needed for computing the desired property. From this, one builds the full feature space by applying mathematical operators. In [8, 9], binary semiconductors with up to only two atoms in the unit cell were considered, hence, a representation with site-dependent f_k's was suitable. For the larger unit cells with disorder considered in this work, such representation is not convenient, since it (a) is not invariant under symmetry operations of the underlying crystal structure, and (b) relies on a fixed unit (super) cell. Therefore, we extend the above described approach [8, 9] to make it applicable to supercells of arbitrary size with disorder. Essentially, we use the site-dependent f_k's to define averaged primary features F_k. These are invariant under symmetry operations of the underlying lattice and also independent of the supercell size. From these one then builds the full feature space similar to [8, 9].

Starting from the primary features, as displayed in tables 1 and 2, these quantities are transformed into scalar primary features that characterize a sample material with a potentially complex structure as a whole. To obtain this, we perform an average of the $\{f_k\}$ over the crystal supercell as

$$F_{sk} = \frac{1}{m} \sum_{\gamma \in s} f_k(\gamma) \qquad (7)$$

where the sum runs over all building blocks γ of a certain type, and *m* is the number of these building blocks γ in the structure *s*. In the applications discussed in section 6, for instance, we employ averages over all single atoms, all pairs (or, equivalently, all bonds) and all tetrahedral clusters in the 16-atom supercell (figure 1, right). As an example, the equilibrium dimer distance $f_k = d_{2di}$ from the pool of primary features is summed over all pairs of nearest neighbors $\langle i, j \rangle$ in the supercell and divided by the total number of pairs in these structures ($m = 32$). This averaging procedure can be viewed as a generalization of Vegard's law.

Additional features are obtained by generalizations of equation (7), taking the *q*th *raw moments* of f_k,

$$\bar{F}_{sk}^q = \frac{1}{m} \sum_{\gamma \in s} f_k(\gamma)^q \qquad (8)$$

Table 1. Primary features f_k and different feature spaces $X_{a,i}$ ($i = 1$–3) used for learning the lattice constant a. The crosses indicate that a quantity is included in $X_{a,i}$. *at, es, bi, 1di, 2di, tet* used for atomic, elemental solid, binary solid, 1-atomic dimer, 2-atomic dimer, and tetra, respectively.

Feature (f_k)	Description	$X_{a,1}$	$X_{a,2}$	$X_{a,3}$
Properties of mono-atomic systems				
Z	Atomic number	×	×	×
P	Period (in periodic table)	×	×	×
r_s	s-orbital radius	×	×	×
r_p	p-orbital radius	×	×	×
r_d	d-orbital radius	×	×	×
a_{es}	Lattice constant of elemental solid		×	×
d_{1di}	Bond length of mono-atomic dimer		×	×
Properties of two-atomic systems				
a_{bi}	Lattice constant of binary material		×	×
d_{2di}	Bond length of bi-atomic dimer		×	×
Properties of tetrahedra				
d_{tet}	Bond length			×

Table 2. Primary features f_k and different feature spaces $X_{E,i}$ ($i = 1$–3) used for learning the energy of mixing, E_{mix}. The quantities Z, P, r_s, d_{2di} and d_{tet} are the same as in table 1. In addition, the Kohn–Sham band-gap and the energy of formation are considered here.

Feature	Description	$X_{E,1}$	$X_{E,2}$	$X_{E,3}$
Properties of mono-atomic systems				
Z	Atomic number	×	×	×
P	Period	×	×	×
r_s	s-orbital radius	×	×	×
$E_{g,at}$	HOMO-LUMO gap of atom	×	×	×
$E_{g,es}$	Gap of elemental solid at Γ-point		×	×
$E_{f,es}$	Energy of formation of elemental solid		×	×
$E_{f,1di}$	Energy of formation of mono-atomic dimer		×	×
$E_{g,1di}$	HOMO-LUMO gap of mono-atomic dimer		×	×
Properties of two-atomic systems				
$E_{f,bi}$	Energy of formation of binary material		×	×
$E_{g,bi}$	Gap of binary material at the Γ-point		×	×
d_{2di}	Bond length of two-atomic dimer		×	×
$E_{f,2di}$	Energy of formation of two-atomic dimer		×	×
$E_{g,2di}$	HOMO-LUMO gap of two-atomic dimer		×	×
Properties of tetrahedra				
d_{tet}	Bond length in tetrahedron			×
$E_{f,tet}$	Energy of formation of tetrahedron			×
$E_{g,tet}$	HOMO-LUMO gap of tetrahedron			×

and the qth root of the qth *central moments*,

$$\widetilde{F}_{sk}^q = \left[\frac{1}{m} \sum_{\gamma \in s} \left(f_k(\gamma) - F_{sk} \right)^q \right]^{1/q} \tag{9}$$

where $q \in \mathbb{N}$, $q \geqslant 2$ defines the order of the moment. Equation (8) defines an average of higher powers of the primary features. The central moments (equation (9)) carry information on the distribution of f_k relative to the mean. The case $q = 2$ can be understood as the standard deviation of f_k, and $q = 3$ is related to its skewness.

In many cases, materials' properties depend on differences in atomic properties, both with respect to geometry or composition. For instance, differences in atomic size can give rise to deviations from Vegard's law [35, 36]. We address this fact by augmenting the pool of candidate features with averages over nearest neighbor differences of the atomic primary features:

$$\Delta F_{sk} \equiv \frac{1}{m} \sum_{\gamma = \langle i,j \rangle \in s} \left| f_k(i) - f_k(j) \right|. \tag{10}$$

Table 3. Feature-space sizes, $N_{\mathcal{F}}$, used for the comparison of LASSO and SISSO as well as the model selection strategies for learning the lattice parameters (left columns) and the energy of mixing (right columns).

	a		E_{mix}	
$\mathcal{T}_i$	$N_{\mathcal{F}}$		$\mathcal{T}_i$	$N_{\mathcal{F}}$
$\mathcal{T}_1$	5257		$\mathcal{T}_1$	8736
$\mathcal{T}_2^{\mathrm{r}}$	239 797		$\mathcal{T}_2^{\mathrm{r}}$	572 351
$\mathcal{T}_2$	13 820 653		$\mathcal{T}_2$	38 163 216

Likewise, we also generalize this average by qth raw ($\bar{\Delta}F^q_{sk}$) and central ($\widetilde{\Delta}F^q_{sk}$) moments as in equations (8) and (9). Primary features, F_k, that are derived from building blocks consisting of single crystal sites (e.g. $f_k = Z$) will be independent to the atomic arrangement. On the contrary, if the averaging is carried out over building blocks containing more atoms, there will be an impact of the local atomic environment. An example of this are the tetrahedra, capturing the short-range order of five atomic sites (the center and the corners). Thus, the bond length (feature $f_k = d_{tet}$ in table 1) may have different values dependent on the configuration of atoms in the tetrahedron.

To obtain the full feature space, we combine the primary features F_k by mathematical operations in analogy to [8, 9]:

(a) first, the primary features F_k obtained from equations (7)–(10), with f_k's as listed in tables 1 and 2, are added to the pool. These features are called tier 0, $\mathcal{T}_0$.
(b) Tier 1, $\mathcal{T}_1$, comprises the features obtained by the following procedure: first, the binary operations $(\cdot + \cdot)$ and $|\cdot - \cdot|$ (here $\cdot$ represents a feature) are applied to all combinations of features F_k in $\mathcal{T}_0$, provided they have the same physical dimension. Then, the resulting expressions are acted on by the unary operators $(\cdot)^n$ ($n \in \{\pm 1/3, \pm 1/2, \pm 2, \pm 3\}$), $\exp(\cdot)$, $1/\exp(\cdot)$, $\log(\cdot)$, and $1/\log(\cdot)$. The power $n = 1$ is not included since it leads to linear combinations already represented by two-dimensional descriptors.
(c) Furthermore, the previously generated features are combined by the product $(\cdot * \cdot)$ operation, yielding tier 2, $\mathcal{T}_2$. This can quickly lead to a combinatorial explosion of the number of features and thus become computationally untractable. As a bypass, we also define a restricted subspace $\mathcal{T}_2^{\mathrm{r}}$ formed by all the products between one feature from $\mathcal{T}_0$ and one from $\mathcal{T}_1$. Note that all the features in $\mathcal{T}_1$ and $\mathcal{T}_2$ introduce non-linearities in the feature space.

The final feature space is then obtained from the union of all considered tiers. The corresponding sizes for the two learning tasks are displayed in table 3. We generally will use the symbols $\mathcal{T}_i$ as a shorthand notation for feature spaces made up from tiers *up to* $\mathcal{T}_i$.

5. Model selection strategies

Applying a model-selection method to the total available data set, guarantees optimal interpolation within these data, it may, however, poorly perform with respect to predictions on new data. To evaluate the predictive power of the model, typically, CV [37] is used, partitioning the data into training set and test set, and building the model upon the training data only. Usually, this procedure is repeated several times on different training and test sets until convergence is reached. CV can, however, not only be used for model assessment but also for *model selection*, i.e. to choose between several competing candidate models. In this work, we devise strategies that transfer the concept of using CV to identifying descriptors as described in section 2.

In the following, we describe three model selection strategies that we probe in this work. In the first one, which corresponds to a typical usage of CV, termed $\mathcal{S}_{\mathrm{all}}$, some kind of learning method $\mathcal{L}$ is applied to *all* data to determine an optimal model $\mathcal{D}$ by minimizing a suitable cost function, here the root mean square error (RMSE). In $\mathcal{S}_{\mathrm{all}}$, N_{CV}-fold CV runs are performed in order to quantify the model's performance. Importantly, at each CV split $i = 1\text{-}N_{\mathrm{CV}}$, $\mathcal{L}$ has to be applied to the training set $S_{\mathrm{tr}}^{(i)}$ in exactly the same way as before to the total data (i.e. with the same hyperparameters defining the subset size $\tilde{M}$, the descriptor dimension Ω, etc). This procedure gives N_{CV} models $\mathcal{D}^{(i)}$ together with their individual training and test errors, $\mathrm{Err}_{\mathrm{tr},i}$ and $\mathrm{Err}_{\mathrm{te},i}$, respectively. The frequency of CV splits i in which the same model is found, $\mathcal{D}^{(i)} = \mathcal{D}$, serves to measure the stability of $\mathcal{D}$ in [8, 9].

By contrast, our new strategies $\mathcal{S}_{\mathrm{tr}}^{\mathrm{CV}}$ and $\mathcal{S}_{\mathrm{te}}^{\mathrm{CV}}$ use the CV error, i.e. the average errors $\overline{\mathrm{Err}}_{\mathrm{tr}}$ and $\overline{\mathrm{Err}}_{\mathrm{te}}$ for model selection. Both strategies consist of the same three steps, (1) a pre-selection of candidate models

$\{\mathcal{D}^{(j)}\}$, through a CV procedure; (2) the calculation of the average training and test errors, $\overline{\text{Err}}_{\text{tr}}$ and $\overline{\text{Err}}_{\text{te}}$, for these candidates via a second CV procedure, and (3) the selection of the best candidate from the $\{\mathcal{D}^{(j)}\}$ based on the average errors. The difference between the strategies is that the final selection of the best candidate model in step (3) is performed using average *training* errors in $\mathcal{S}_{\text{tr}}^{\text{CV}}$ and average *test* errors in $\mathcal{S}_{\text{te}}^{\text{CV}}$. Note that step (2) is carried out by a CV run where in each split individual training and test errors are calculated for *all* candidates. In contrast to the CV procedure of step (1) or to the CV procedure in $\mathcal{S}_{\text{all}}$, this step does not include any model selection but solely fits the free parameters **c** (see equation (1)) for all candidates to the training set of the corresponding split. This serves to generate unbiased values for $\overline{\text{Err}}_{\text{tr}}$ and $\overline{\text{Err}}_{\text{te}}$ since, even if a model was selected in many CV-splits in step 1, it may still perform poorly for the splits in which it was not selected.

Formally, step (3), yielding the candidate with the lowest average error, can be expressed as

$$\underset{\mathcal{D}^{(j)}}{\text{argmin}} \, \overline{\text{Err}}_{\text{tr}}(\mathcal{D}^{(j)}) \quad \text{or} \quad \underset{\mathcal{D}^{(j)}}{\text{argmin}} \, \overline{\text{Err}}_{\text{test}}(\mathcal{D}^{(j)}) \tag{11}$$

in $\mathcal{S}_{\text{tr}}^{\text{CV}}$ and $\mathcal{S}_{\text{te}}^{\text{CV}}$, respectively.

6. Application to group-IV ternary alloys

In the following, we demonstrate and compare the approaches, introduced in sections 4 and 5, with the example of ternary group-IV compounds in the zincblende structure as described in section 3. Since an important aim is to keep the dimension Ω of the descriptors low, we investigate $\Omega = 1, \ldots, 5$. The number of relevant features in LASSO $+ \ell_0$ and SISSO is set to $\tilde{M} = 30$ (see section 2.2). For this value, an exact solution of the ℓ_0 problem is still computationally feasible for $\Omega = 5$. For SISSO, we use the selection operator $\text{SO}(\ell_0)$, i.e. the Ω-dimensional descriptor is obtained by explicitly solving the ℓ_0-problem on the actual union of subspaces. The latter is obtained by iterating SISSO five times, yielding in each iteration subspaces S_i of dimension 6, and then taking the union $\cup_{i=1}^5 S_i$, of dimension $\tilde{M} = 30$. Unless indicated otherwise, we use the RMSE as an error metric.

As explained in section 3, the target properties a and E_{mix} are influenced by both the composition and the actual atomic arrangements at fixed composition. In order to quantify how well the different models capture them, the RMSE is employed. This measure, though, is not useful to assess a model's ability to capture the mere effect of atomic arrangements, as a model may capture them well but still poorly score on the RMSE. To this end, we first calculate the mean absolute error (MAE) between the centered property $(P - \langle P \rangle)$ and the centered prediction $(\hat{P} - \langle \hat{P} \rangle)$ at fixed composition, and then average this for all considered compositions:
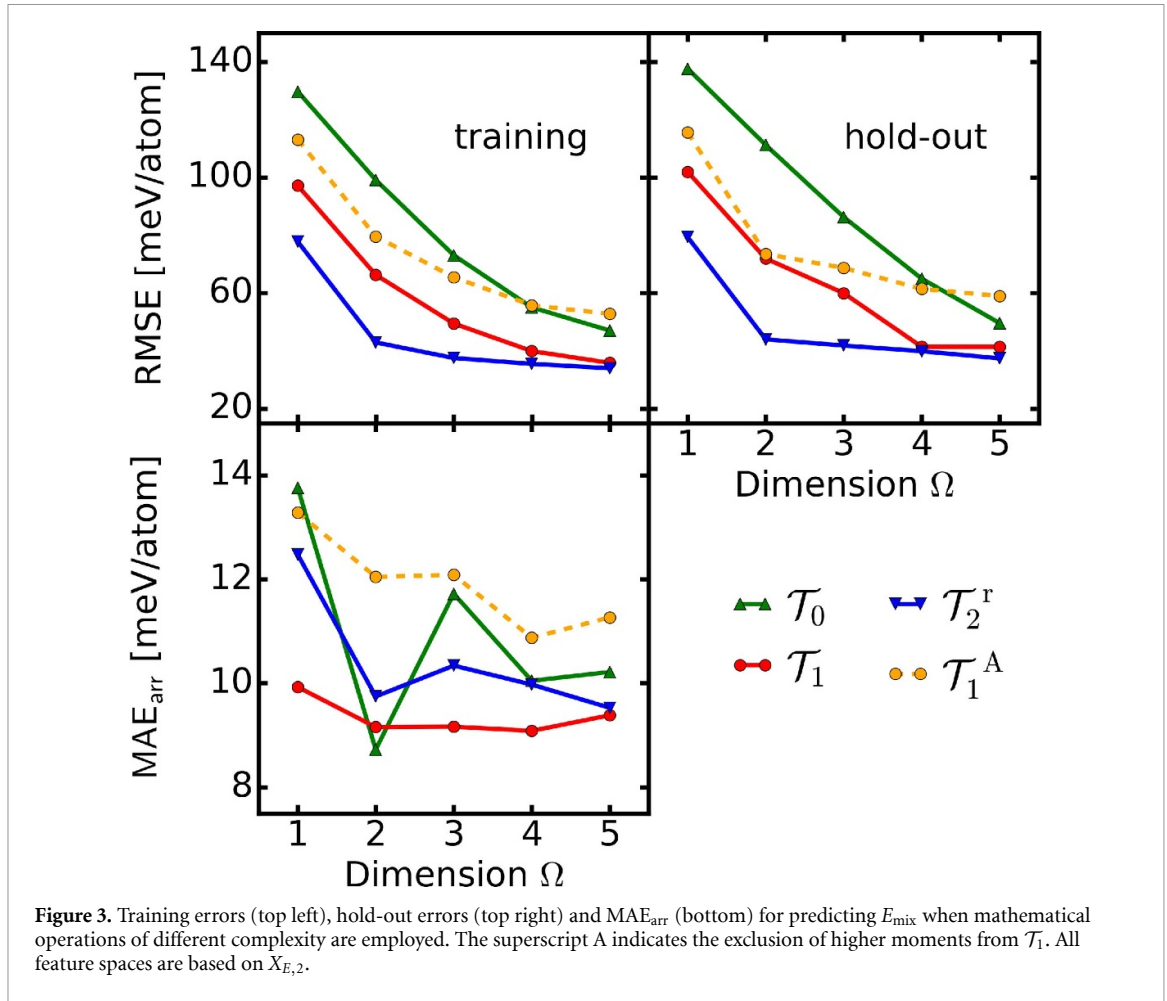
$$\text{MAE}_{\text{arr}} \equiv \overline{\langle |(P - \langle P \rangle) - (\hat{P} - \langle \hat{P} \rangle)| \rangle}. \tag{12}$$

Here, the overline indicates the average over all compositions (x, y), and $\langle \cdot \rangle$ is the average over atomic arrangements within composition (x, y). The subscript 'arr' stands for 'arrangement'. This quantity is invariant under the addition of a constant to the predictions, either concerning the whole data set or the data with a certain composition.

All the numerical computations using LASSO $+ \ell_0$, SISSO, and the various feature selection strategies, are performed with an in-house Python code.

6.1. Feature spaces
In this work, we generate three different feature spaces. The corresponding primary features are listed in tables 1 and 2, respectively, for the two different learning tasks. They contain physical properties of single atoms, molecular dimers, tetrahedral clusters, as well as elemental and binary solids. The radii r_s, r_p and r_d are defined as the maximum radial probability density for the s-, p- and d-orbitals of the isolated atoms, respectively, taken from [9]. Dimer bond-lengths (d_{1di}, d_{2di}) and bulk lattice-constants (a_{es}, a_{bi}) are obtained by relaxation. The lattice constants of the elemental solids (a_{es}) are computed in the diamond structure, those of the binary materials (a_{bi}) in the zincblende structure. We also consider all tetrahedral clusters present in the ternary materials. These are relaxed under the constraint of being fully symmetric, i.e. determined by the distance d_{tet} between the central atom and the corner atoms. Lattice constants, dimer bond lengths, and tetrahedral bond lengths are scaled to radius equivalents (e.g. dimer lengths are divided by 2). This is done to allow for a better interpretation of features that algebraically combine these properties. Also energies of formation and Kohn–Sham band-gaps are included. For bulk materials, we consider the values at the Γ-point, while for finite systems, the differences between the highest occupied and the lowest unoccupied

Figure 3. Training errors (top left), hold-out errors (top right) and MAE$_{arr}$ (bottom) for predicting E_{mix} when mathematical operations of different complexity are employed. The superscript A indicates the exclusion of higher moments from $\mathcal{T}_1$. All feature spaces are based on $X_{E,2}$.

molecular orbitals (HOMO-LUMO gap). For the energies of formation, the ground-state energies of the isolated atoms are used as a reference.

The feature spaces ($X_{a,i}$ and $X_{E,i}$, $i = 1, \ldots, 3$) are constructed for the two learning tasks using the primary features listed in tables 1 and 2, respectively. The first one, $X_{a/E,1}$, contains single-atom features only, i.e. any average in equation (7) will be over atomic building blocks only ($\gamma = i$). $X_{a/E,2}$ additionally contains pair features which are averaged over all $\gamma = \langle i, j \rangle$, and $X_{a/E,3}$ adds tetrahedral features.

6.2. Complexity of feature space

In the following analysis, we choose the model-selection strategy $\mathcal{S}_{all}$ in combination with LASSO $+ \ell_0$. We verified that the use of SISSO leads to similar results.

6.2.1. Impact of mathematical operations

We start by studying how the mathematical operations used to construct the feature space impact the performance of the resulting descriptors, in terms of the tiers $\mathcal{T}_i$, $i = 0, 1, 2$ (section 4) as well as the specific operations. Thereby, we restrict the analysis to the learning of E_{mix} and note that the learning of the lattice constant leads to very similar results.

Figure 3 shows the RMSE of the training and the hold-out set as well as the MAE$_{arr}$ on the training data of the resulting descriptors for this learning task. We use feature spaces based on single-atom and pair features ($X_{E,2}$) on the three complexity levels $\mathcal{T}_0$, $\mathcal{T}_1$ and $\mathcal{T}_2^r$ [38]. Furthermore, to evaluate the effect of the higher moments, we consider the feature space where the respective elements are eliminated from $\mathcal{T}_1$ (indicated by $\mathcal{T}_1^A$) As expected, an increase of complexity overall reduces the fitting error. The decrease of the hold-out error indicates that the models do not suffer from overfitting. For $\mathcal{T}_1$ and $\mathcal{T}_2^r$, the training error converges to about $\sim$30 meV/atom. This means that *optimal* models are obtained for $\mathcal{T}_1$ at $\Omega = 5$ and for $\mathcal{T}_2^r$ already at $\Omega = 3$. For $\mathcal{T}_0$, a higher-dimensional descriptor could reach even lower training errors. Arrangement effects are surprisingly well captured by medium complexity, i.e. tier $\mathcal{T}_1$. Notably, for the 2D case, even $\mathcal{T}_0$ does best. However, the differences in MAE$_{arr}$ are generally rather small. Finally, the worse
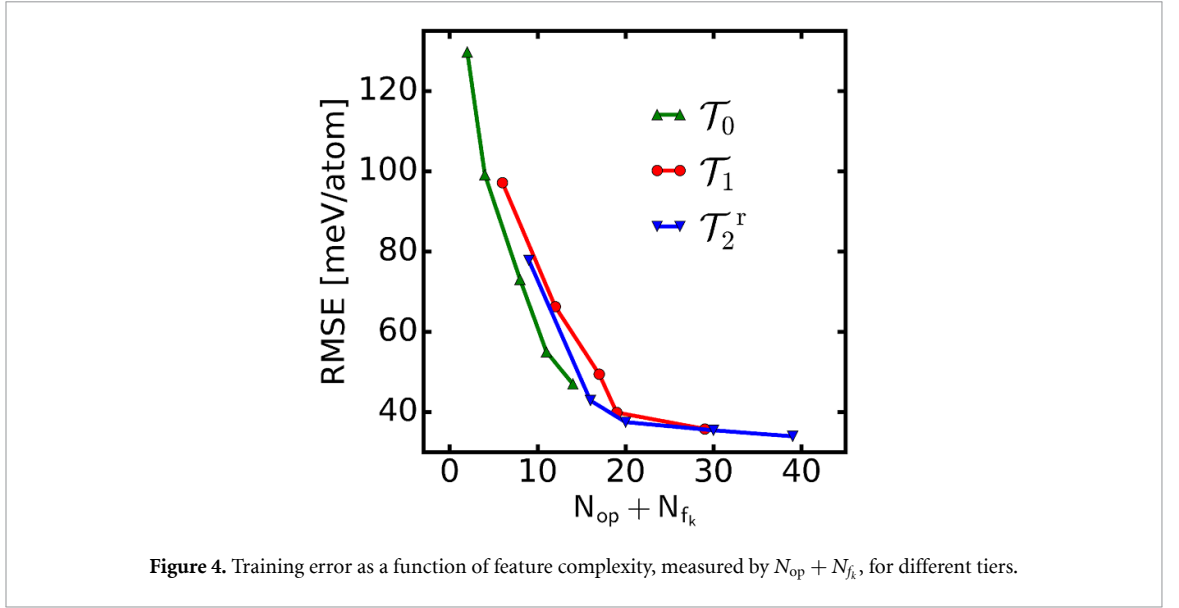
Figure 4. Training error as a function of feature complexity, measured by $N_{op} + N_{f_k}$, for different tiers.

Table 4. 1D and 2D descriptors resulting from feature spaces with complexity $\mathcal{T}_0$, $\mathcal{T}_1$, and $\mathcal{T}_2^r$ together with the number of operations, N_{op}, and the number of basic features, N_{f_k}, considered in the descriptors.

$\mathcal{T}_i$	1D		N_{op}	N_{f_k}
$\mathcal{T}_0$	$\tilde{E}_{g,es}^2$		1	1
$\mathcal{T}_1$	$(\tilde{E}_{g,es}^2 + \tilde{E}_{g,1di}^2)^3$		4	2
$\mathcal{T}_2^r$	$(\bar{E}_{f,1di}^2) \cdot \sqrt{\left\lvert \bar{r}_s^2 - \bar{d}_{2di}^2 \right\rvert}$		6	3
$\mathcal{T}_i$	2D		N_{op}	N_{f_k}
$\mathcal{T}_0$	$\tilde{d}_{2di}^2$	$E_{g,bi}$	2	2
$\mathcal{T}_1$	$(\tilde{E}_{g,es}^2 + \tilde{E}_{g,2di}^2)^3$	$\exp(\Delta E_{g,es} + \Delta E_{g,1di})$	9	3
$\mathcal{T}_2^r$	$\tilde{d}_{2di}^2 \cdot (\bar{\Delta}E_{g,at}^2 + \bar{E}_{g,bi}^2)^2$	$E_{m,es} \cdot \sqrt[3]{\lvert r_s - d_{2di} \rvert}$	11	5

performance of $\mathcal{T}_1^A$ as compared to $\mathcal{T}_1$, indicates that the higher moments accounting for non-linearities turn out to be an essential ingredient in feature construction.

From the previous analysis, it becomes apparent that, rather independently of the tier used (or the feature space, see below), the increase in descriptor complexity improves the fit. To quantify this qualitative observation, we define a practical measure for the complexity as the sum of two terms, namely, the total number of operations in a descriptor, N_{op}, and the number of primary features, N_{f_k}. While the former captures the algebraic complexity, the latter accounts for the amount of physical information in the descriptor. In figure 4, the training errors are plotted versus this complexity measure. Interestingly, the curves of the three tiers approximately follow a common line. This uncovers a general dependency of the fitting error on the total complexity where the different tiers cover different regions of complexity.

Table 4 summarizes the 1D and 2D descriptors found for the tiers $\mathcal{T}_0$, $\mathcal{T}_1$, and $\mathcal{T}_2^r$, along with the complexity measures for each of them.

6.2.2. Influence of primary features

We now study how the adoption of different feature spaces influences the obtained descriptors. To better emphasize the mere effect of the feature space on the results, we limit this analysis to the low-complexity tier $\mathcal{T}_1^A$ defined in the previous section. The left panels of figure 5 show the RMSE and the MAE_{arr} for the training data versus descriptor dimension Ω for predicting lattice constants for the three feature spaces $X_{a,1-3}$. The dotted and dashed horizontal lines indicate the errors of Vegard's law and the averaged binary lattice constant (equation (7)), respectively. Depending on the composition only, the former is totally blind to effects of atomic arrangement. The latter captures them to some extent, thus exhibiting a somewhat smaller error. For all three feature spaces, the increase in dimension Ω, naturally, lowers the training error, which is most distinctive for $X_{a,1}$. Except for the 1D descriptor of $X_{a,1}$ ($\mathbf{D}_{1D} = 1/\sqrt[3]{r_p}$), all models outperform
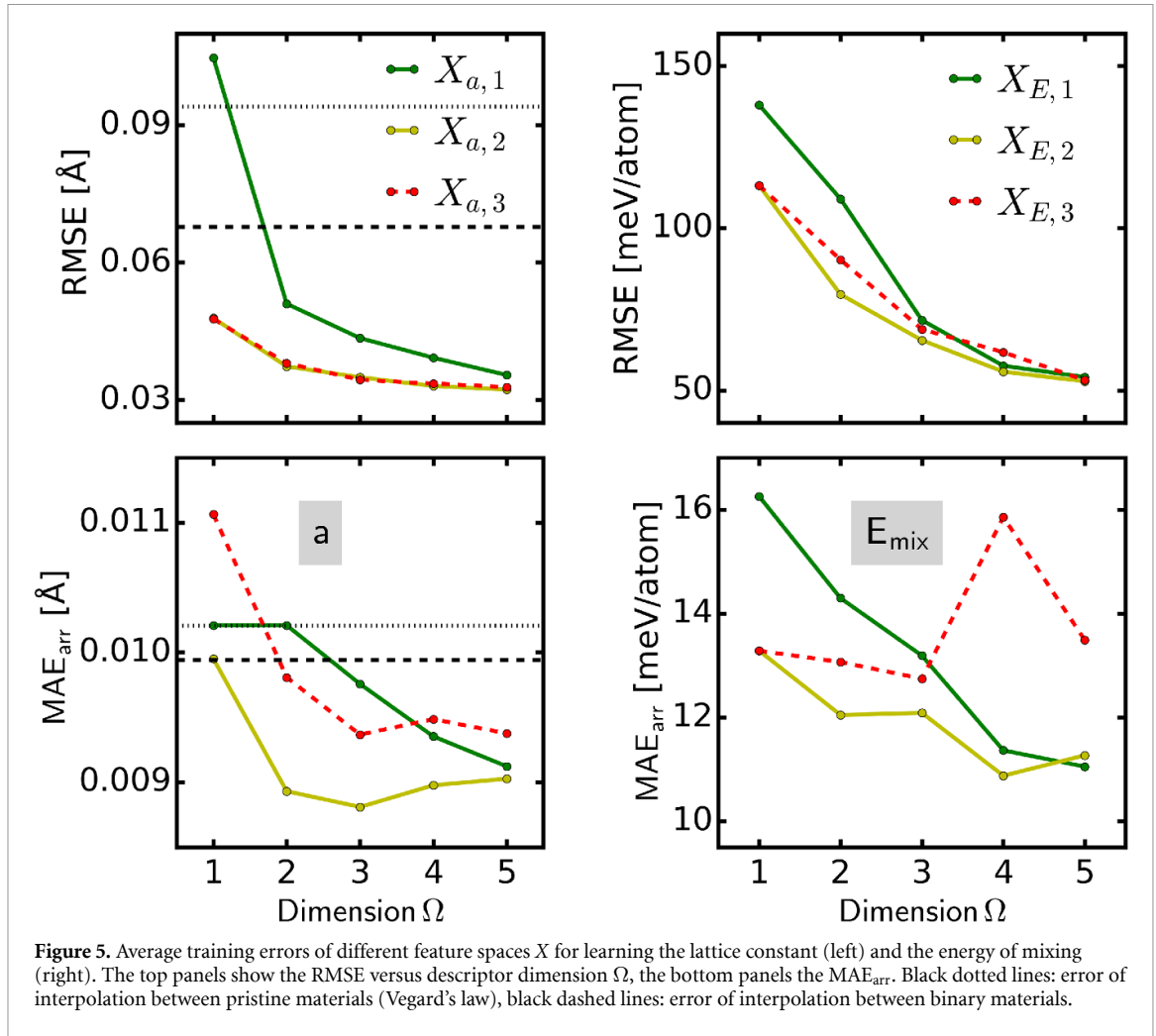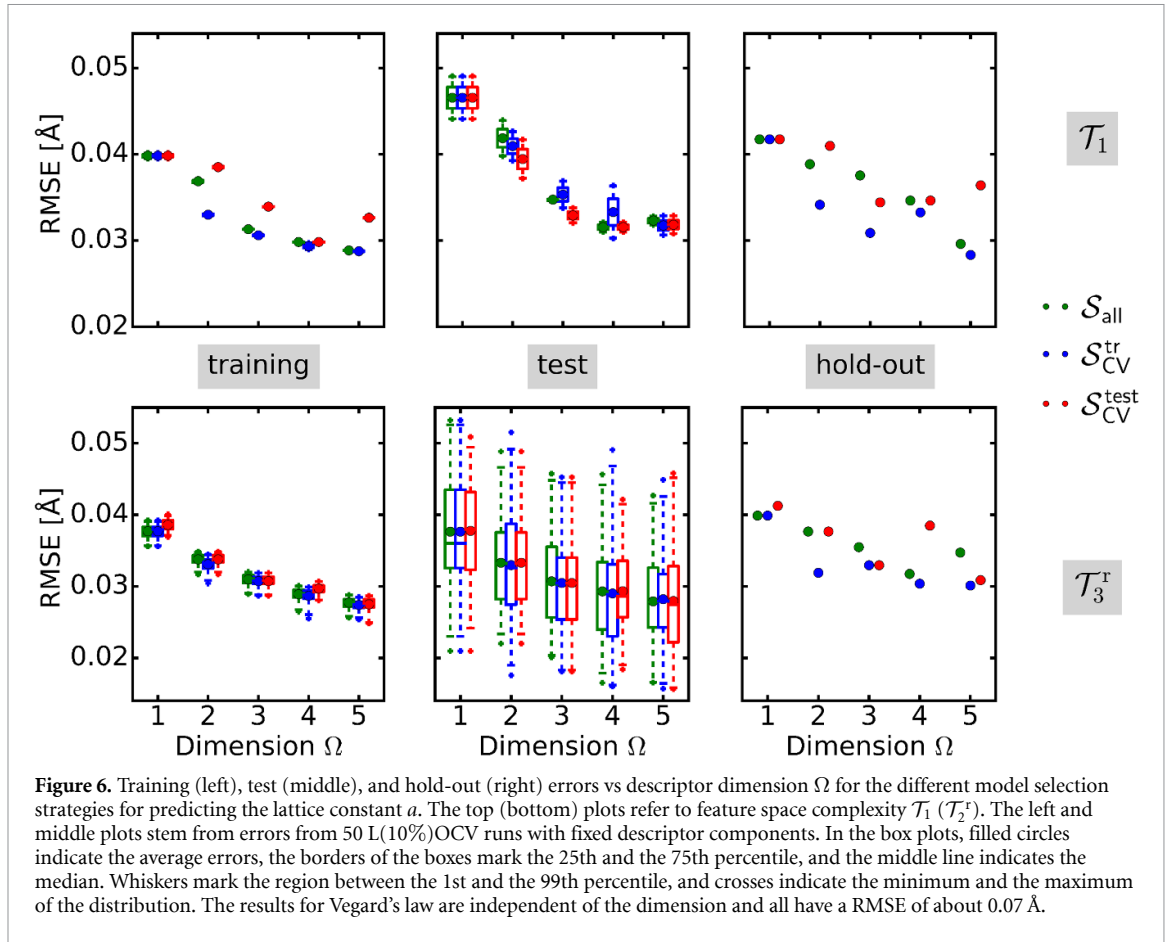
Figure 5. Average training errors of different feature spaces X for learning the lattice constant (left) and the energy of mixing (right). The top panels show the RMSE versus descriptor dimension Ω, the bottom panels the MAE_{arr}. Black dotted lines: error of interpolation between pristine materials (Vegard's law), black dashed lines: error of interpolation between binary materials.

Vegard's law as well as the simple averaging procedure. Including pair quantities in the feature spaces, reduces the RMSE drastically. Adding also the tetrahedral features, by contrast, has a negligible effect on the training error, though the descriptors emerging from $X_{a,3}$ are not identical to those from $X_{a,2}$ but contain also the tetrahedral feature d_{tet}.

The MAE_{arr}, in the lower panel, also shows an overall decrease with higher Ω, but exhibits a shallow minimum at $\Omega = 3$ for $X_{a,2}$ and $X_{a,3}$. Even descriptors from the pure atomic feature space ($X_{a,1}$) can outperform Vegard's law. This is the case for $\Omega \geqslant 3$ where the descriptors include nearest-neighbor differences in the fashion of equation (10). The best result is achieved by combined atomic and pair features ($X_{a,2}$) for all descriptor dimensions considered. Surprisingly, $X_{a,3}$, though doing well in the training error, is performing worse in the MAE_{arr}. This is presumably due to the rigidity of the tetrahedra, departing considerably from the actual physical situation in the solid. To sum up, optimal feature spaces for predicting lattice constants of the considered data, should be based on a combination of atomic and pair features, and the tetrahedra-based features can be discarded here safely.

The right part of figure 5 shows RMSE and MAE_{arr} for learning E_{mix}. As expected, also in this case the fitting error decreases with increasing Ω for all $X_{E,i}$. Also expected, the 1D and 2D descriptors based on $X_{E,1}$ exhibit a larger RMSE than the ones from $X_{E,2}$ and $X_{E,3}$, while considering more complex descriptors ($\Omega \geqslant 3$) does not improve the RMSE anymore, i.e. all three features spaces have similar error. Adding tetrahedral features, overall slightly increases the RMSE as compared to $X_{E,2}$, except for $\Omega = 1$ where $X_{E,2}$ and $X_{E,3}$ give the same descriptor. One would expect that offering more features cannot increase the error, and just the solution from the smaller feature space would be found. While this is true for solving the ℓ_0 problem in an exact manner, for the approximation by LASSO $+ \ell_0$ this does not need to be the case [39].

Finally, we have a look at the related MAE_{arr} (lower panel). In short, also for the energy of mixing, next-neighbor differences play an important role, where descriptors from $X_{E,2}$ generally perform best. Like

Figure 6. Training (left), test (middle), and hold-out (right) errors vs descriptor dimension Ω for the different model selection strategies for predicting the lattice constant a. The top (bottom) plots refer to feature space complexity $\mathcal{T}_1$ ($\mathcal{T}_2^{\mathrm{r}}$). The left and middle plots stem from errors from 50 L(10%)OCV runs with fixed descriptor components. In the box plots, filled circles indicate the average errors, the borders of the boxes mark the 25th and the 75th percentile, and the middle line indicates the median. Whiskers mark the region between the 1st and the 99th percentile, and crosses indicate the minimum and the maximum of the distribution. The results for Vegard's law are independent of the dimension and all have a RMSE of about 0.07 Å.

for the RMSE, addition of tetrahedral features never leads to an improvement, most likely because the local environment in the real data often differs markedly from the regular tetrahedral shape. In contrast, the most basic atomic features are performing quite well.

6.3. Model selection strategies

Now we apply the different model selection strategies described in section 5, to the prediction of lattice constants, employing LASSO + ℓ_0 and the feature spaces $\mathcal{T}_1$ and $\mathcal{T}_2^{\mathrm{r}}$. The results depicted in figure 6, comprise the distribution of training (left), test (middle) and hold-out errors (right). The error distributions for training and test data result from 50 CV runs where the descriptors obtained by the three strategies were kept fixed.

For all three strategies, the training errors do not show significant spread. $\mathcal{S}_{\mathrm{CV}}^{\mathrm{tr}}$ exhibits slightly smaller values compared to the other two. For $\mathcal{T}_2^{\mathrm{r}}$, $\mathcal{S}_{\mathrm{CV}}^{\mathrm{test}}$ is very similar as well, for $\mathcal{T}_1$ and some descriptor dimensions Ω it is markedly worse, however. The test errors have a narrow distribution at smaller matrix size ($\mathcal{T}_1$, top panel) and a wide distribution at larger matrix size ($\mathcal{T}_2^{\mathrm{r}}$, bottom panel). This is to be expected for increasing model complexity [37]. $\mathcal{S}_{\mathrm{CV}}^{\mathrm{test}}$ here performs better or similar to the other strategies, as expected, though no large difference between the strategies is apparent. Generally, $\mathcal{S}_{\mathrm{CV}}^{\mathrm{tr}}$ yields the smallest hold-out errors and $\mathcal{S}_{\mathrm{CV}}^{\mathrm{test}}$ tends to yield the largest. The results for learning E_{mix} exhibit the analogous trends.

6.4. LASSO versus SISSO

As described in section 2.2, SISSO has been proven to overcome the issue of strongly correlated features in LASSO + ℓ_0 [11], besides being able to deal with much larger feature matrices **D**. Hence, it is interesting to compare these two feature selection methods for identifying optimal descriptors. To do so, we first compare the pre-selected subspaces and then the resulting 5D descriptors, applying the strategy $\mathcal{S}_{\mathrm{all}}$. As a measure of (linear) correlations between features, we use $\left|\rho_{i,j}\right|$, the absolute value of the Pearson correlation coefficient. The symmetric matrix $\left|\rho_{i,j}\right|$ is visualized in figure 7 in terms of a correlogram. Here, pairwise correlations of the pre-selected subspaces are shown for predicting the lattice constant, with LASSO in the lower-left triangle and SISSO in the upper-right one. (Note that the subset obtained with SISSO (LASSO) contains 29 (30)
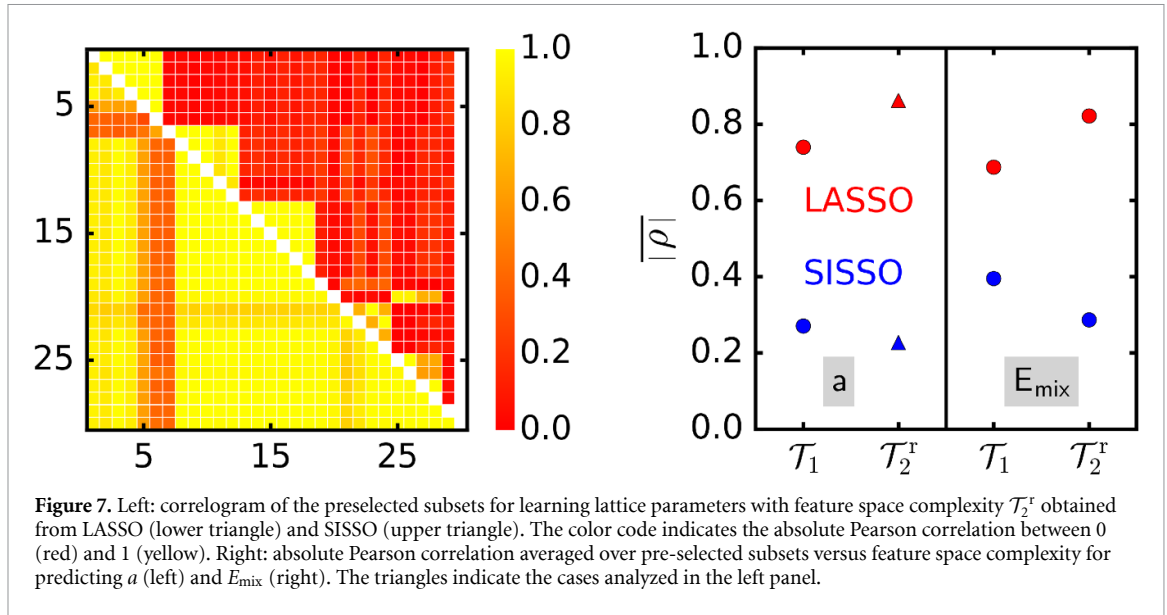
Figure 7. Left: correlogram of the preselected subsets for learning lattice parameters with feature space complexity $\mathcal{T}_2^r$ obtained from LASSO (lower triangle) and SISSO (upper triangle). The color code indicates the absolute Pearson correlation between 0 (red) and 1 (yellow). Right: absolute Pearson correlation averaged over pre-selected subsets versus feature space complexity for predicting a (left) and E_{mix} (right). The triangles indicate the cases analyzed in the left panel.

Table 5. Components of the 5D descriptors for predicting a with feature space complexity $\mathcal{T}_2^r$, obtained by applying an exhaustive ℓ_0 search on the subsets from LASSO or SISSO, respectively. The components are sorted by their absolute Pearson correlation with a in decreasing order.
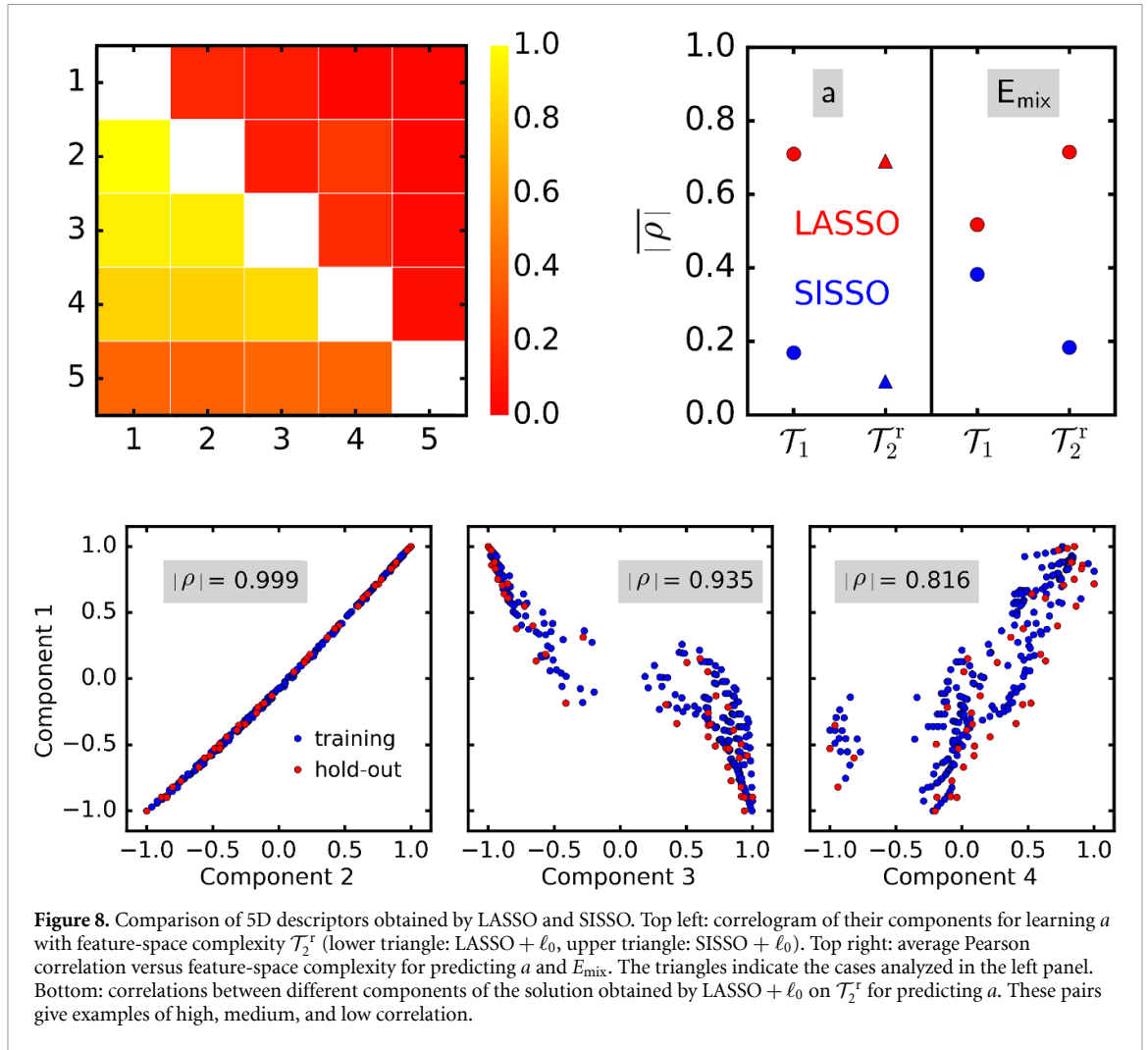
Component	1	2	3	4	5
LASSO	$\dfrac{\bar{a}_{bi}^2}{\sqrt{\bar{d}_{1di}}}$	$\dfrac{\bar{a}_{bi}^2}{a_{es}}$	$\sqrt[3]{\left|\bar{r}_p^2 - \bar{d}_{2di}^2\right|}$	$\bar{d}_{2di}^3 \cdot \sqrt[3]{\left|\bar{r}_d^2 - \bar{d}_{2di}^3\right|}$	$\dfrac{\tilde{d}_{2di}^3}{\sqrt[3]{\tilde{r}_s^2 + \overline{r}_s^3}}$
SISSO	$\dfrac{\bar{a}_{bi}^2}{\sqrt{\bar{r}_p^2 + \bar{a}_{bi}^2}}$	$\dfrac{\tilde{Z}^3}{\bar{r}_p^2 + \bar{r}_p^3}$	$\bar{r}_d^2 \cdot \sqrt{\tilde{a}_{bi}^3}$	$\tilde{a}_{bi}^2 \cdot (\tilde{d}_{1di}^2 - \tilde{a}_{bi}^3)^3$	$\dfrac{\bar{P}^3}{\bar{r}_s^2 + \bar{r}_d^3}$

features. Such a difference may occur when certain features are repeatedly selected to different subsets in SISSO.) The feature space complexity here is $\mathcal{T}_2^r$. It is a challenging case for LASSO where indeed many highly correlated candidate features dominate its subspace. By contrast, most of the features selected by SISSO are less correlated (predominance of the red color). The small yellow triangles inside the upper-right triangle correspond to the bunch-like subspaces which, by construction of SISSO, contain highly correlated features.

In the right panel of the figure, the mean absolute correlations $\overline{|\rho_{i,j}|}$ are compared for feature spaces of different complexity, for both predicting a and E_{mix}. Clearly, in all cases the overall correlation is significantly higher for subsets obtained with LASSO compared to SISSO. Moreover, at increasing matrix size, it increases for LASSO and decreases for SISSO. For the more difficult learning task of predicting E_{mix}, the overall correlation is lower compared to the simpler task of learning a when using LASSO, while for SISSO it is the opposite. As a general—and expected—conclusion, SISSO selects subsets with less redundancy, in particular for large and highly correlated feature spaces. This is a prerequisite for the following ℓ_0 step to select low-correlated descriptors.

For learning the lattice parameter at complexity level $\mathcal{T}_2^r$, table 5 lists the components of the 5D descriptors obtained when applying the ℓ_0 step to either the LASSO or the SISSO subspace. Figure 8 shows their correlations in the upper left panel. Since the preselected subspace offers only four low correlated pairs (see lower triangle in the left panel of figure 7), all possible 5D descriptors from the LASSO subspace inevitably contain at least one highly correlated feature pair. In fact, components 1 and 2 are particularly highly correlated ($|\rho_{1,2}| = 0.999$) and also appear to have a similar algebraic form (see table 5). Also the correlations between components 1 and 3 as well as 2 and 3 are very high ($|\rho_{1,3}| = 0.935$ and $|\rho_{2,3}| = 0.929$, respectively), however, they seem to carry different information. This can be seen in the bottom panel of the figure where the correlations are displayed for the training data and the hold-out set, exhibiting a larger spread (see middle panel). The right panel shows the correlations between components 1 and 4 ($|\rho_{2,3}| = 0.816$). By contrast to the LASSO descriptor, the 5D descriptor from SISSO does not contain highly correlated components.

The upper right panel of the figure compares the mean correlation of the 5D descriptors, obtained by the two methods. For LASSO, increasing feature-space complexity rises $\overline{|\rho_{i,j}|}$ in case of E_{mix} and slightly drops in

Figure 8. Comparison of 5D descriptors obtained by LASSO and SISSO. Top left: correlogram of their components for learning *a* with feature-space complexity $\mathcal{T}_2^r$ (lower triangle: LASSO + ℓ_0, upper triangle: SISSO + ℓ_0). Top right: average Pearson correlation versus feature-space complexity for predicting *a* and E_{mix}. The triangles indicate the cases analyzed in the left panel. Bottom: correlations between different components of the solution obtained by LASSO + ℓ_0 on $\mathcal{T}_2^r$ for predicting *a*. These pairs give examples of high, medium, and low correlation.

case of *a*. For SISSO, we observe a decrease in both cases, and the values are always lower than the LASSO counterparts.

It is also interesting to compare the predictive power of the descriptors obtained from LASSO and SISSO. This is shown in figure 9, displaying the fitting and hold-out errors for the two learning tasks. Thereby, feature spaces with complexity $\mathcal{T}_1$ and $\mathcal{T}_2^r$ are considered. While the fitting errors do not differ much between LASSO and SISSO for both learning tasks, the hold-out errors for learning the lattice parameter (left panel) indicate that for larger Ω, the SISSO descriptors may be overfitted. Additionally, results for $\mathcal{T}_2$ are shown obtained from SISSO only (LASSO being limited due to its high demands in terms of memory). This further reduces the fitting errors but enhances overfitting for $\Omega \geqslant 3$. In contrast, for learning E_{mix}, there is no evidence of overfitting, even for $\Omega = 5$. Also in this case, the higher complexity of feature space ($\mathcal{T}_2$) reduces the fitting error without leading to overfitting.

Overall, in the studied cases, the descriptors from LASSO and SISSO are very similar with respect to predictive power. As expected, descriptors from SISSO benefit from less redundant components and can handle big feature matrices.

6.5. Optimal descriptors

There is always some degree of arbitrariness as there are plenty of ways to construct candidate features. This also holds for model parameters like the descriptor dimension Ω. One would, for example, select a small value of Ω if simplicity is the main goal, at the expense of accuracy. Moreover, when feature complexity increases, the selection methods yield more and more competing models of virtually the same predictive power—which itself can be defined in different ways. Thus, in table 6, we present some exemplary descriptors for both *a* and E_{mix}, based on the previous analyses. For *a*, we have selected the 3D descriptor on the very simple complexity level $\mathcal{T}_0$ and the 2D descriptor on the rather complex $\mathcal{T}_2$; for E_{mix}, the 4D descriptor on $\mathcal{T}_0$ and the 2D descriptor on $\mathcal{T}_2$. All of these were identified by SISSO using the strategy $\mathcal{S}_{\mathrm{all}}$.
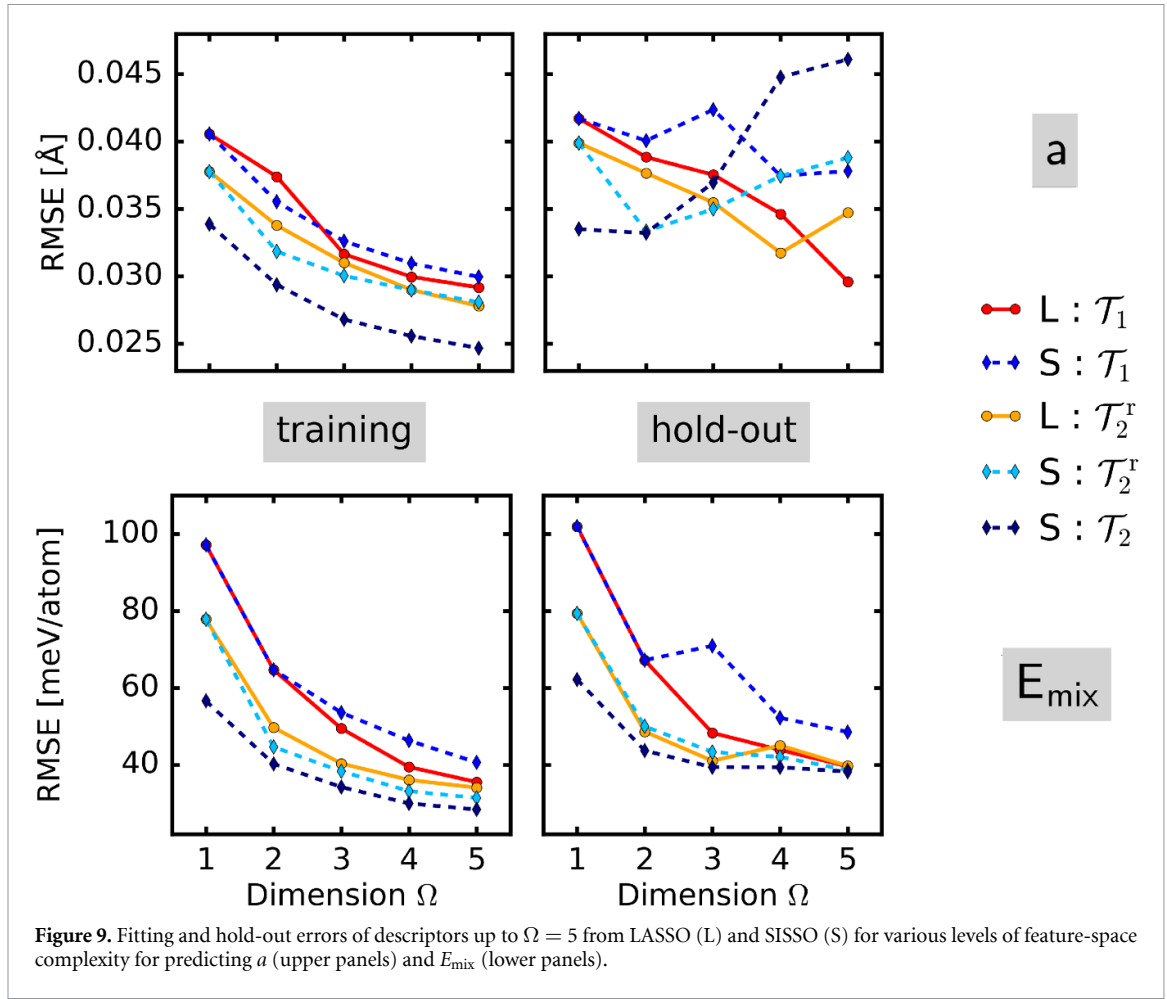
Figure 9. Fitting and hold-out errors of descriptors up to $\Omega = 5$ from LASSO (L) and SISSO (S) for various levels of feature-space complexity for predicting a (upper panels) and E_{mix} (lower panels).

Table 6. Components of the descriptors for predicting a and E_{mix} on complexity levels $\mathcal{T}_0$ and $\mathcal{T}_2$. The parenthesis indicate components that, when exchanged by each other, are virtually indistinguishable in performance.
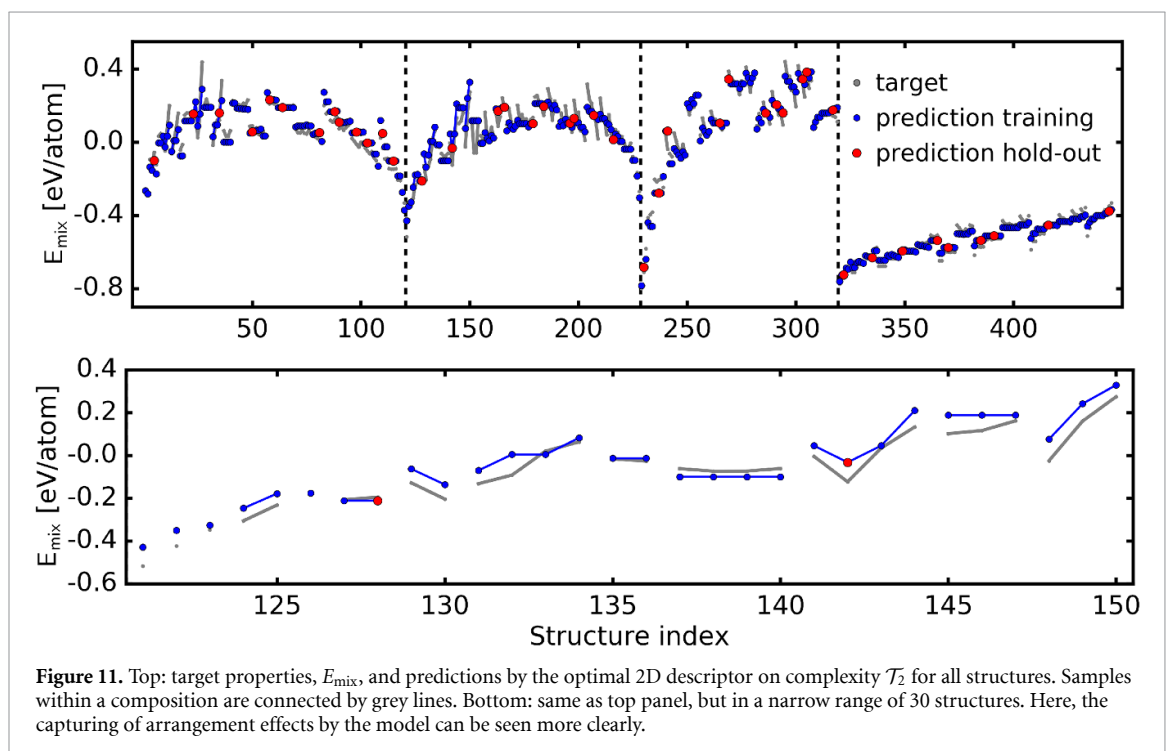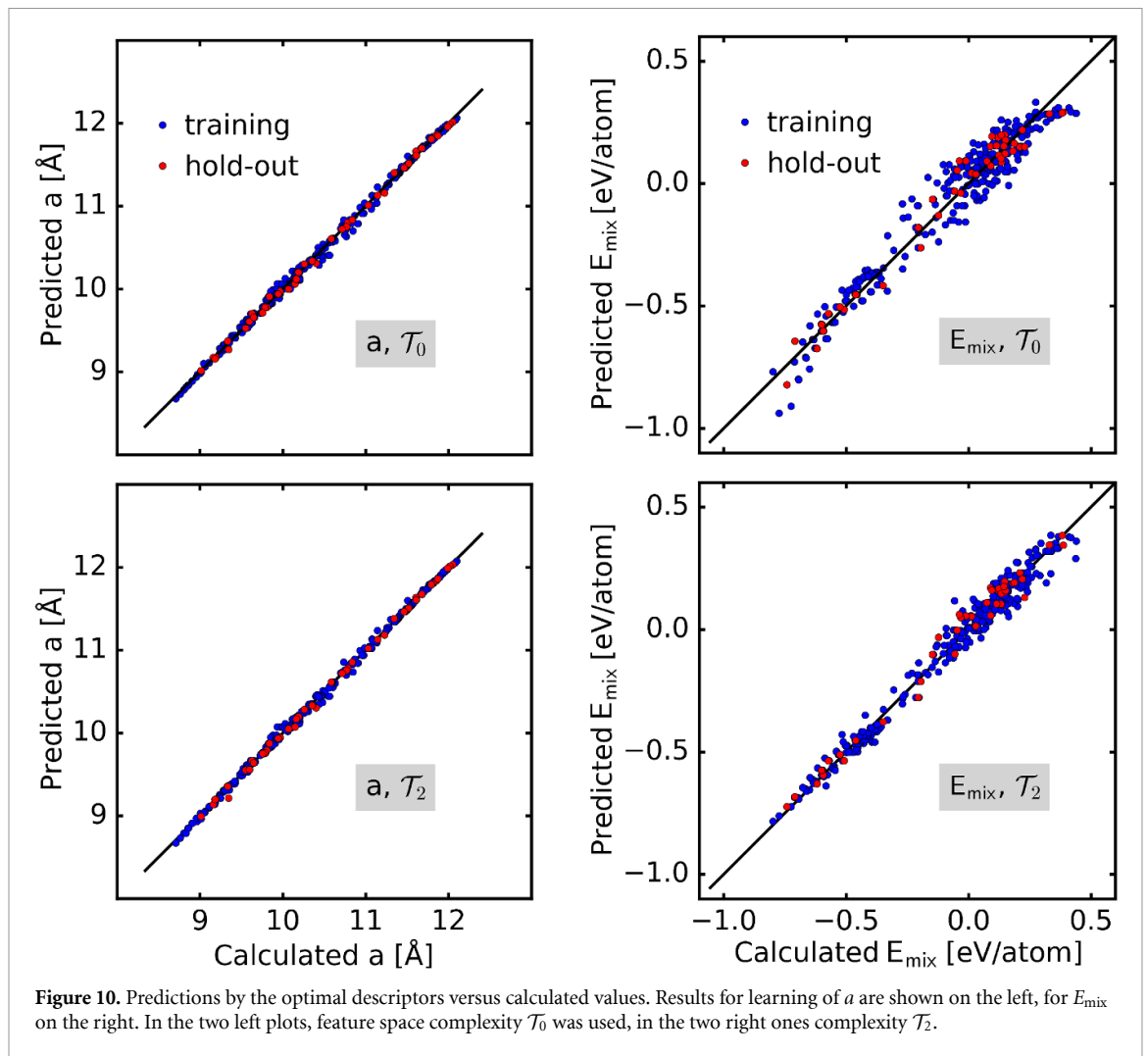
Target	Complexity	Descriptor components			
a	$\mathcal{T}_0$	a_{bi}	$\tilde{a}_{bi}^2(\tilde{d}_{2di}^2)$	$\tilde{a}_{bi}^3$	
	$\mathcal{T}_2$	$\dfrac{\sqrt{\overline{a_{es}^2}+\overline{a_{bi}^2}}}{\sqrt[3]{\overline{r_p^2}+\overline{d_{1di}^2}}}$	$\dfrac{\sqrt[3]{\left|\overline{r_d^2}-\tilde{d}_{2di}^3\right|}}{(r_d+\tilde{r}_d^3)^3}$		
E_{mix}	$\mathcal{T}_0$	$\bar{Z}^3$	$\tilde{P}^2$	$\tilde{E}_{g,bi}^2$	$\tilde{a}_{bi}^3(\tilde{d}_{2di}^3)$
	$\mathcal{T}_2$	$\dfrac{1}{E_{f,1di}+\tilde{E}_{g,at}^2}\cdot\dfrac{1}{\sqrt{E_{f,1di}+E_{g,at}}}$	$(E_{g,1di}+E_{g,bi})^2\cdot\sqrt{\left|\overline{r_s^3}-\overline{d}_{2di}^3\right|}$		

Table 7. Performance of the *optimal* descriptors for predicting a and E_{mix}. RMSE and MAE averaged over training and test sets of CV; MAE and maximum absolute error (maxAE) on the hold-out set; and MAE$_{arr}$. The mean absolute percentage error (MAPE) is shown in training and hold-out sets for learning a. Lenghts are in units of mÅ and energies in meV/atom.

Target	Complexity	Ω	N_{f_k}	N_{op}	Training			Test		Hold-out			MAE$_{arr}$
					RMSE	MAE	MAPE	RMSE	MAE	MAE	MAPE	maxAE	
a	$\mathcal{T}_0$	3	1	4	38	26	0.26	41	28	29	0.28	93	9
	$\mathcal{T}_2$	2	6	18	29	19	0.19	30	20	21	0.21	133	9
E_{mix}	$\mathcal{T}_0$	4	4	7	64	51	—	64	51	46	—	130	11
	$\mathcal{T}_2$	2	6	15	40	30	—	40	31	35	—	99	10

The corresponding error analysis is presented in table 7. Figure 10 visualizes the performance of the obtained models, and figure 11 shows in more detail the performance of the 2D descriptor on $\mathcal{T}_2$ for E_{mix}.

We start with the 3D descriptor for a on complexity level $\mathcal{T}_0$. It is favorable because it only contains the binary bulk lattice constant a_{bi} and takes a simple form in the style of a polynomial expansion and thus

Figure 10. Predictions by the optimal descriptors versus calculated values. Results for learning of *a* are shown on the left, for E_{mix} on the right. In the two left plots, feature space complexity $\mathcal{T}_0$ was used, in the two right ones complexity $\mathcal{T}_2$.

Figure 11. Top: target properties, E_{mix}, and predictions by the optimal 2D descriptor on complexity $\mathcal{T}_2$ for all structures. Samples within a composition are connected by grey lines. Bottom: same as top panel, but in a narrow range of 30 structures. Here, the capturing of arrangement effects by the model can be seen more clearly.

effectively augments a pure interpolation between the binary materials. We note that the three components are not redundant as they do not exhibit a high pairwise correlation (maximum $|\rho_{i,j}| = 0.49$).

The alternative descriptor presented for this task, the 2D descriptor on $\mathcal{T}_2$, has much higher informational ($N_{\mathcal{F}} = 6$) and algebraic ($N_{op} = 18$) complexity as the former ($N_{\mathcal{F}} = 1$ and $N_{op} = 4$). It uses all basic properties of length dimension from our pool of basic properties $\{f_k\}$, except r_s. The increased complexity results in an improved predictive performance on the expense of a simple, intuitive interpretation. Note also here that the two components are hardly correlated ($|\rho_{1,2}| = 0.14$). As can be seen from figure 10, both presented descriptors for a have good predictive power without any outliers. So, apart from the fact that the descriptor obtained from complexity $\mathcal{T}_2$ exhibits slightly smaller errors, it appears sufficient to stay with that from $\mathcal{T}_0$.

$\tilde{E}^2_{g,bi}$ and $\tilde{a}^3_{bi}$ appear as the third and fourth component for learning E_{mix}. As the lattice constant and the band gap of the binary materials are sensitive to the local atomic environments, the descriptor can, up to some extent, capture the effects of atomic arrangements. It additionally includes the very simple features Z and P. Also in this case, the components are not highly correlated (maximum $|\rho_{i,j}| = 0.38$).

Also like above, the higher algebraic and informational complexity of $\mathcal{T}_2$ improves the predictive performance. The descriptor depends on band-gap and formation energies, the atomic radius r_s, and the dimer distance d_{2di}, and these components are almost uncorrelated ($|\rho_{1,2}| = 0.13$). The larger dispersion in the right panels of figure 10 demonstrates that the learning of E_{mix} is more difficult than that of a, requiring higher descriptor complexity.

7. Conclusions

We have advanced and applied methods based on compressed sensing that allow one to identify low-dimensional models for predicting materials properties from a potentially huge pool of candidate features. The feature spaces were constructed to contain averages over building blocks of the supercell for being applicable to disordered materials. Moreover, we have derived model selection strategies based on CV to improve the generalizability of the models as compared to approaches that solely use training errors for model selection.

We have evaluated this methodology for predicting the *ab initio* lattice constants and energies of mixing of a dataset of group-IV ternary zincblende compounds. For extracting a low-dimensional subset of the most relevant features, we have applied LASSO $+ \ell_0$ [8, 9] and SISSO [11]. Our analysis yields that atomic, bulk, and dimer properties are essential features for this dataset while tetrahedral properties can be discarded. We further found that next-neighbor differences of atomic features and, alternatively, pair features are able to capture arrangement-specific effects of the materials. We have also found a trade-off between larger descriptor dimension and the complexity of the considered features with respect to the algebraic operations. Comparing the feature-selection methods, we confirmed that SISSO reduces correlations between the descriptor components and allows for going beyond the limitations on feature-space size that LASSO $+ \ell_0$ suffers from. Although the differences to a simple one-shot strategy [8] are small for the dataset under consideration, our strategy that minimizes the average training error on CV (strategy $\mathcal{S}^{tr}_{CV}$) was able to improve descriptor generalizability in terms of hold-out error. We emphasize that the differences between the strategies are small here presumably due to the homogeneity of the dataset. It would thus be desirable to test them in future research on more diverse datasets, for example comprising materials differing in space group or with a larger variety of elements.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://doi.org/10.17172/NOMAD/2022.05.20-1.

Acknowledgments

ORCID iDs

Santiago Rigamonti ⦿ https://orcid.org/0000-0002-8939-6173
Claudia Draxl ⦿ https://orcid.org/0000-0003-3523-6657

References

[1] Mueller T, Kusne A G and Ramprasad R 2016 *Machine Learning in Materials Science* (New York: Wiley)
[2] Rupp M, Tkatchenko A, Müller K R and Von Lilienfeld O A 2012 *Phys. Rev. Lett.* **108** 058301
[3] Behler J 2011 *J. Chem. Phys.* **134** 074106
[4] Bartók A P, Kondor R and Csányi G 2013 *Phys. Rev. B* **87** 184115
[5] Seko A, Takahashi A and Tanaka I 2014 *Phys. Rev. B* **90** 024101
[6] Seko A, Hayashi H, Nakayama K, Takahashi A and Tanaka I 2017 *Phys. Rev. B* **95** 144110
[7] Sutton C, Ghiringhelli L M, Yamamoto T, Lysogorskiy Y, Blumenthal L, Hammerschmidt T, Golebiowski J R, Liu X, Ziletti A and Scheffler M 2019 *npj Comput. Mater.* **5** 111
[8] Ghiringhelli L M, Vybiral J, Levchenko S V, Draxl C and Scheffler M 2015 *Phys. Rev. Lett.* **114** 105503
[9] Ghiringhelli L M, Vybiral J, Ahmetcik E, Ouyang R, Levchenko S V, Draxl C and Scheffler M 2017 *New J. Phys.* **19** 023017
[10] Butcher P N, March N H and Tosi M P 2013 *Crystalline Semiconducting Materials and Devices* (New York: Springer Science & Business Media)
[11] Ouyang R, Curtarolo S, Ahmetcik E, Scheffler M and Ghiringhelli L M 2018 *Phys. Rev. Mater.* **2** 083802
[12] Goldsmith B R, Boley M, Vreeken J, Scheffler M and Ghiringhelli L M 2017 *New J. Phys.* **19** 013031
[13] Ziletti A, Kumar D, Scheffler M and Ghiringhelli L M 2018 *Nat. Commun.* **9** 2775
[14] Musil F, De S, Yang J, Campbell J E, Day G M and Ceriotti M 2018 *Chem. Sci.* **9** 1289–300
[15] Legrain F, Carrete J, van Roekeghem A, Curtarolo S and Mingo N 2017 *Chem. Mater.* **29** 6220–7
[16] Ramprasad R, Batra R, Pilania G, Mannodi-Kanakkithodi A and Kim C 2017 *npj Comput. Mater.* **3** 54
[17] Ward L, Agrawal A, Choudhary A and Wolverton C 2016 *npj Comput. Mater.* **2** 16028
[18] Bartók A P, De S, Poelking C, Bernstein N, Kermode J R, Csányi G and Ceriotti M 2017 *Sci. Adv.* **3** e1701816
[19] Yin M T and Cohen M L 1981 *Phys. Rev. B* **24** 6121–4
[20] Arora S and Barak B 2009 *Computational Complexity: A Modern Approach* (Cambridge: Cambridge University Press)
[21] Tibshirani R 1996 *J. R. Stat. Soc.* B **58** 267–88 (available at: www.jstor.org/stable/2346178)
[22] Fan J and Lv J 2008 *J. R. Stat. Soc.* B **70** 849–911
[23] Ventura C I, Querales Flores J D, Fuhr J D and Barrio R A 2015 *Prog. Photovolt., Res. Appl.* **23** 112–8
[24] Fischer I A *et al* 2015 *Opt. Express* **23** 25048–57
[25] Wendav T, Fischer I A, Montanari M, Zoellner M H, Klesse W, Capellini G, Von Den Driesch N, Oehme M, Buca D and Busch K 2016 *Appl. Phys. Lett.* **108** 242104
[26] Hart G L and Forcade R W 2008 *Phys. Rev. B* **77** 224115
[27] For 59 configurations, no stable geometry could be found within the given supercell
[28] Vegard L 1921 *Z. Phys.* **5** 17–26
[29] Adachi S 2009 *Properties of Semiconductor Alloys: Group-IV, III–V and II–VI Semiconductors* vol 28 (New York: Wiley)
[30] Gulans A, Kontur S, Meisenbichler C, Nabok D, Pavone P, Rigamonti S, Sagmeister S, Werner U and Draxl C 2014 *J. Phys.: Condens. Matter* **26** 363202
[31] Perdew J P and Wang Y 1992 *Phys. Rev. B* **45** 13244
[32] Murnaghan F 1944 *Proc. Natl Acad. Sci.* **30** 244–7
[33] Draxl C and Scheffler M 2019 *J. Phys. Mater.* **2** 036001
[34] Draxl C and Scheffler M 2018 *MRS Bull.* **43** 676
[35] Denton A R and Ashcroft N W 1991 *Phys. Rev. A* **43** 3161–4
[36] Murphy S, Chroneos A, Jiang C, Schwingenschlögl U and Grimes R 2010 *Phys. Rev. B* **82** 073201
[37] Friedman J, Hastie T and Tibshirani R 2001 *The Elements of Statistical Learning* Springer Series in Statistics vol 1 (New York: Springer)
[38] We restrict ourselves to $\mathcal{T}_2^r$ due to computational limitations in combination with LASSO
[39] The LASSO preselection step is sub-optimal as compared to solving the exact ℓ_0 problem. This means that, even if the initial feature space is enlarged, there is no guarantees that the subspace $\bar{M}$ left for the ℓ_0 step will lead to a better descriptor