Journal of MATERIALS RESEARCH

Article
DOI:10.1557/s43578-023-01164-w

Interpretable machine learning for materials design

James Dean[1], Matthias Scheffler[2,3], Thomas A. R. Purcell[3], Sergey V. Barabash[4], Rahul Bhowmik[5], Timur Bazhirov[1,a)]

[1] Exabyte Inc., San Francisco, CA, USA
[2] University of California Santa Barbara, Isla Vista, CA, USA
[3] The NOMAD Laboratory at the Fritz Haber Institute, Berlin, Germany
[4] Intermolecular Inc., San Jose, CA, USA
[5] Polaron Analytics, Beavercreek, OH, USA
[a)] Address all correspondence to this author. e-mail: timur@exabyte.io

Received: 31 December 2022; accepted: 5 September 2023; published online: 12 October 2023

Fueled by the widespread adoption of machine learning and the high-throughput screening of materials, the data-centric approach to materials design has asserted itself as a robust and powerful tool for the in silico prediction of materials properties. When training models to predict material properties, researchers often face a difficult choice between a model's interpretability and performance. We study this trade-off by leveraging four different state-of-the-art machine learning techniques: XGBoost, SISSO, Roost, and TPOT for the prediction of structural and electronic properties of perovskites and 2D materials. We then assess the future outlook of the continued integration of machine learning into materials discovery and identify the key problems that will continue to challenge researchers as the size of the literature's datasets and complexity of models increases. Finally, we offer several possible solutions to these challenges with a focus on retaining interpretability and share our thoughts on magnifying the impact of machine learning on materials design.

Introduction

Today, big data and artificial intelligence revolutionize many areas of our daily life, and materials science is no exception [1–3]. More scientific data are available now than ever before and the size of the literature is growing at an exponential rate [4–7]. This has led to multiple efforts in building the digital ecosystem for material discovery, most notably the Materials Genome Initiative (MGI) [8, 9]. The MGI is a multinational effort focused on improving the tools and techniques surrounding materials research, which recently has included suggestions to adopt the set of Findable, Accessible, Interoperable, and Reusable (FAIR) principles when reporting data [10]. In the years since the creation of the MGI, a number of large materials and chemical datasets have emerged, including the 2D Materials Encyclopedia (2DMatPedia) [11], Automatic Flow (AFLOW) database [12, 13], Computational 2D Materials Database (C2DB) [14, 15], Computational Materials Repository (CMR) [16], Joint Automated Repository for Various Integrated Simulations (JARVIS) [17], Materials Project [18], Novel Materials Discovery (NOMAD) repository [19], and the Open Quantum Materials Database (OQMD) [20]. We note that all of these are primarily computational in nature, and that there is still a scarcity of large databases containing comprehensively characterized experimental data. Despite this, at least in computational materials discovery, the current availability of data has been a boon for exploration of the materials space, as it allows for highly flexible, data-hungry [21] models to be trained.

One such approach that has seen widespread popularity in recent years is gradient boosting. Gradient boosting [22] is an ensemble technique in which a collection of weak learners (typically decision trees) are incrementally trained with respect to the gradient of the loss function [23]. A well-known variant is eXtreme Gradient Boosting (XGBoost) [24], which reformulates the algorithm to provide stronger regularization and improved protection against overfitting. In chemistry, its applications have been diverse: XGBoost has been used to predict the adsorption energy of noble gases to Metal-Organic Frameworks (MOFs) [25], biological activity of pharmaceuticals [26], atmospheric transport [27], and has even been combined with the representations found in Graph Neural Networks (GNNs) to generate accurate models of various molecular properties, as Deng et al demonstrated for several well-known datasets including

TOX-21 [28] (a toxicology dataset), FreeSolv [29] (a dataset of small molecule hydration free energies), SIDER [30] (a dataset of adverse drug reactions), and others. [31].

Neural networks have also seen a lot of interest, owing to their ability to learn new features from input data. This has included the influential Behler-Parinello [32] and Crystal Graph Convolutional Neural Network (CGCNN) [33] architectures based on chemical structure, the Representation Learning from Stoichiometry (Roost) [34] architecture based on chemical formula, and many other approaches [1, 2, 35–41]. Historically, interpretability of neural networks has been a major challenge, although there has also been substantial recent work in addressing this problem [42].

The modern machine learning (ML) toolbox is large, although it is still far from complete. As a result, model selection techniques are becoming increasingly necessary: this has led to the field of automated machine learning (AutoML). This area of work has seen much progress in recent years [43, 44], and has even been extended to Neural Architecture Search (NAS) [45], for the automated optimization of neural network architectures. In this work, we leverage the Tree-based Pipeline Optimization Tool (TPOT) approach to AutoML [46–48], which uses a Genetic Algorithm (GA) to create effective ML pipelines. Although it generally draws from the models of SciKit-Learn [49], it can also be configured to explore gradient boosting models via XGBoost [22], and neural network models via PyTorch [50]. Moreover, TPOT also performs its own hyperparameter optimization, thus providing a more hands-off solution to identifying ML pipelines. The success of GA-based approaches in ML is not isolated to AutoML. Indeed, they are a fundamental part of genetic programming, where they are used to optimize functions for a particular task [51, 52]. Eureqa [53] is a particularly successful example of this [54], leveraging a GA to generate equations fitting arbitrary functions, and has been used in several areas of chemistry, including the generation of adsorption models to nanoparticles [55] and metal atoms to oxide surfaces [56]. This approach of fitting arbitrary functions to a task is also known as "symbolic regression." Recent work surrounding compressed sensing has yielded the Sure Independence Screening and Sparsifying Operator (SISSO) approach [57]. SISSO also generates equations mapping descriptors to a target property, proceeding by combining descriptors using various building blocks, including trigonometric functions, logarithms, addition, multiplication, exponentiation, and many others. This methodology has been highly successful in a variety of areas including crystal structure classification [58], as well as the prediction of perovskite properties [59–61] and 2D topological insulators [62].

While the recent raise in the availability of scientific data has led to the increased integration of machine learning and artificial intelligence techniques into materials science, one of the ongoing challenges associated with using these methods is getting physically interpretable results. By physical interpretation, we mean an understanding of the relationship between the chosen descriptors and the target property. Although a black-box model which has a high level of accuracy but little physical interpretation may lend itself well to the Edisonian screening of a wide range of materials, it may be difficult to understand exactly what feature (or combination of features) actually matters to the design of the material. Once the screening is done and the target values are calculated, little may be done to improve performance aside from including new features, adjusting the model's hyperparameters, or increasing the size of the training set. Alternatively, consider a model which has less accuracy, but which has an intuitive explanation, such as an equation describing an approximate relationship between features and target. Although such a model may at first glance seem less useful than a highly accurate black-box, such a model can help deliver insight into the underlying process that results in the target property. Moreover, by understanding which features are important, the model can give clues into what may be done to further improve it — driving the rational discovery of materials. In addition, interpretability versus accuracy is not a strict trade-off, and it is possible for interpretable and black-box models to deliver similar accuracy [63]. Therefore, in this work we take steps to compare the performance of TPOT, XGBoost, SISSO, and Roost for each problem with respect to i) performance and ii) interpretability.

We leverage a diverse selection of techniques in order to draw comparisons of model accuracy and interpretability. Taking advantage of the current abundance of chemical data, we can re-use the Density Functional Theory (DFT) calculations of others stored on several FAIR chemical datasets. A set of three different problems are investigated: (1) the prediction of perovskite volumes, (2) the prediction of 2D material bandgaps, and (3) the prediction of 2D material exfoliation energies. These problems allow for coverage over a range of relevant areas within materials science. Perovskites are well-studied systems with relevance to catalysis and solar cells [64] [65], and the unit cell is a fundamental property of crystalline materials. 2D materials are an exciting new field within nanotechnology with applications in electronics [66] [67]. The bandgap in particular is a crucial property for electronics [68] and the exfoliation energy is often a key parameter in the production of 2D materials [69].

For the perovskite volume problem, we leverage the ABX_3 perovskite dataset (containing 144 examples) published by Körbel, Marques, and Botti [70]; this dataset is hosted by NOMAD [19], whose repository has strong focus on enabling researchers to report their data such that it satisfies the FAIR data principles [71]. For the 2D material problems, we apply the 2DMatPedia published by Zhou et al [11], a set of 6,351 hypothetical 2D materials identified via a high-throughput

screening of systems on the Materials Project [18]. Although at first glance this may seem like a relatively small dataset, we note the general rarity of known 2D materials in the literature. Haastrup et. al. released the C2DB [14], which contains around 4,000 systems algorithmically generated from a set of prototypes. The JARVIS [17] database maintained by NIST contains, among many other systems, around 1,000 low-D materials. Thus, we choose the 2DMatPedia because it offers us access to a large number of structures out of the box, without needing to combine together multiple data sources. In addition, we note that data can sometimes be hard to come by in the materials science space: datasets with large numbers of entries may not always exist for properties of interest. Thus, evaluating the performance of ML approaches on smaller datasets is an important (and realistic) benchmark to measure.

The manuscript is organized as follows: we begin by training a diverse set of four models, which are XGBoost, TPOT, Roost, and SISSO to investigate each of the three problems, resulting in a total of 12 trained models. Performance metrics (and comparative plots) are presented for each trained model to facilitate comparison, and we discuss the interpretation we can achieve from each of these models. Overall, between the four categories of model we trained, we leverage the XGBoost model as a baseline, as it is the simplest among them. Additionally, it is a common workhorse oftentimes achieving good results on tabular data. Framing our analysis as a comparison to the interpretability and accuracy relative to the XGBoost model, we can then draw conclusions about the interpretation and applicability of the other three model types. Finally, we provide a discussion of the future outlook of ML in the digital materials science ecosystem and what can be done to further accelerate materials discovery.

We find that TPOT delivers high-quality models, generally outperforming the other methods in terms of fitness metrics. Despite this, interpretability is not guaranteed, as it can create highly complex pipelines. XGBoost lends itself to interpretation more consistently, as it allows for an importance metric, although it may be harder to understand exactly what the relationship is between the different features (or combinations of different features) and the target variable. We found that Roost performed well on problems that could be approached via compositional descriptors (i.e., without structural descriptors); as a result, it can help us understand when a target property requires more than just the composition. Finally, we achieve the easiest interpretability from SISSO, as it provides access to descriptors which directly capture the relationship between the features and target variable. Using these results, we discuss the advantages and disadvantages of each method, and discuss areas where the digital ecosystem surrounding materials discovery could be improved to improve adherence to FAIR principles. Our work provides a comparison of several common ML techniques on challenging (but relevant) materials property prediction problems.

Results

Perovskite volume prediction

XGBoost, TPOT, and SISSO were applied to investigate the volume of perovskites as a function of the compositional features described in Sect. "Compositional Descriptors. Additionally, we trained a Roost model on the chemical formula of the perovskites to predict the volume. The train/test split resulted in a total of 129 entries in the training set, and 15 in the test set. We find generally good performance on the perovskite volume problem across all 4 models, although the TPOT and SISSO model display the best performance by all metrics investigated (see Table 1), including respective test-set R^2 of 0.996 and 0.990. We note again here that we only used the compositional descriptors for this problem, and not the structural descriptors. The Roost model also performs well with a test-set R^2 of 0.935, but it also has a non-normal error, as can be seen in Figure 1. Finally, we find that while XGBoost is the worst performing method, it still has a relatively good test-set R^2 of 0.866.

The performance of all 4 models is summarized in Figure 1. Visually, we find a very tight fit by the TPOT model in both the training and test sets, with good correlation from the XGBoost and SISSO models. We also find a systematic under-prediction of perovskite volumes in the Roost model in both the training and test set, with the under-prediction beginning at approximately 75 $Å^3$/formula unit, achieving a maximum deviation at approximately 130 $Å^3$/formula unit, and returning to parity at approximately 200 $Å^3$/formula unit.

The good performance of the TPOT model results from a generated pipeline with seven stages. The first three stages are based on the **Familywise Error**, Feature Variance, and Familywise Error (FWE) again. This down-selects features according to the FWE error, and removes features with a variance under 0.20.

TABLE 1: Performance metrics for the XGBoost, TPOT, Roost, and SISSO models on the perovskite volume prediction problem.

Error Metric	Partition	XGBoost	TPOT	Roost	SISSO
MAE	Train	8.15	1.194	9.11	7.28
RMSE	Train	11.88	1.6227	11.21	9.08
Max Error	Train	43.29	5.7475	22.94	26.74
R^2	Train	0.949	0.999	0.955	0.971
MAE	Test	12.89	3.6362	8.83	4.05
RMSE	Test	17.17	4.9172	12.00	4.71
Max Error	Test	37.10	13.611	31.69	10.27
R^2	Test	0.866	0.996	0.935	0.990

The parity plots for these models are depicted in 1.

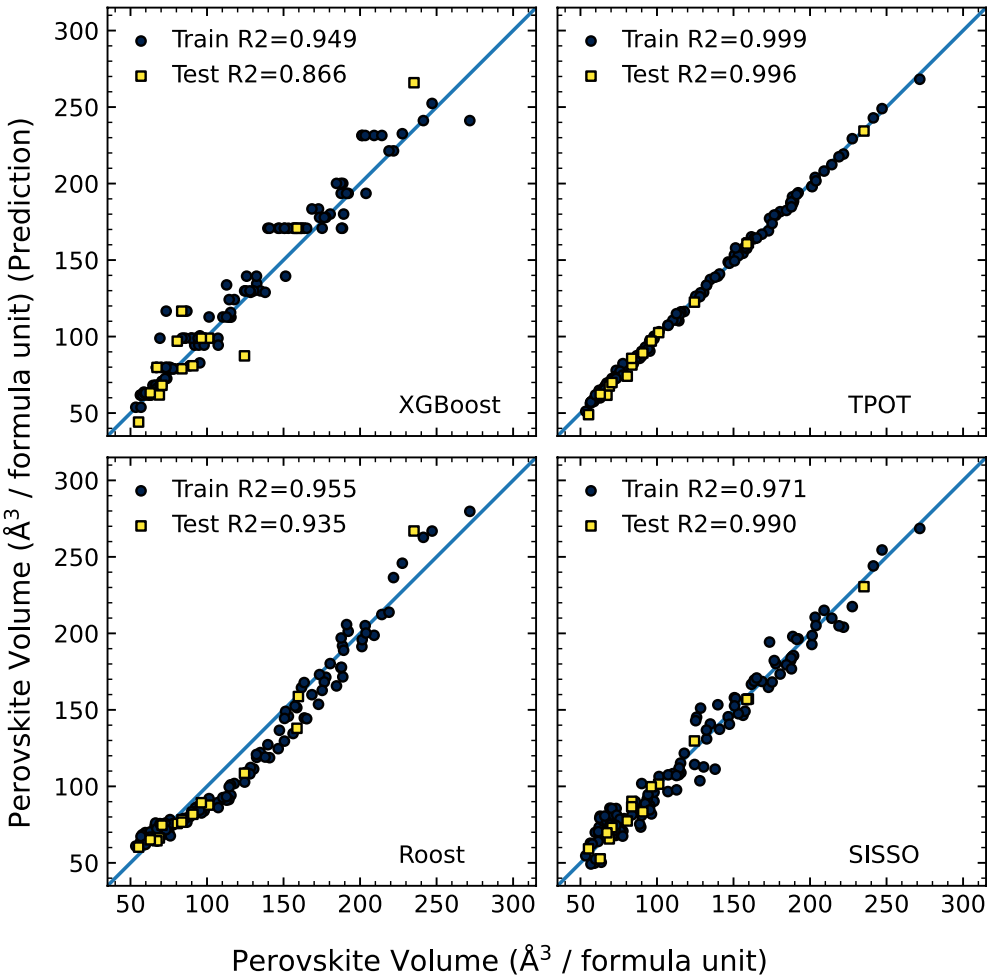

Figure 1: Parity plots for the XGBoost, TPOT, Roost, and SISSO models on the perovskite unit cell volume problem. Included are the training and testing sets. A diagonal line indicating parity is drawn as a guide to the eye.

The alpha values for the two FWE thresholds are 0.047 and 0.046, respectively, which means the highest allowed uncorrelated p-value for a feature is 0.046. From here, the remaining features are passed to a series of stacked Random Forest, Extremely Randomized Trees [72], and XGBoost. The Random Forest uses 100 trees with bootstrapping, can use at most 60% of the features, and each leaf must contain at least 16 samples. The Extremely Randomized Trees model averages over 100 trees without bootstrapping and at most 20% of the features, with each leaf containing at least 16 samples. The XGBoost stage has 100 estimators, and is rather shallow with any individual tree having a maximum depth of 1. The "stacked" component of this series of 3 regressors means that each regressor adds its own predictions to the dataset as a new column, which informs further models down the pipeline. Finally, a LASSO model is fit with a Least Angle Regressor.

Moving onward from TPOT, in the case of our XGBoost model, we can extract feature importances. Although various different feature importance metrics can be derived from XGBoost, in this case we use the "gain" metric, which describes how the model's loss function improves when a feature is chosen for a split while constructing the trees. A large number of features were input into this model, so we display only the 10 most important features identified by XGBoost in Supporting Information Figure 5. Here, we find that the average Rahm atomic radii [73, 74] (importance score 0.48) have the highest importance score, followed by the average Van der Waals radius used by the Universal Force Field (UFF) [75] (importance score 0.27). The remaining 288 features fall off as a long tail of low importance scores, indicating that they did little to improve the model's performance in predicting the perovskite volume.

For SISSO, we used the feature space as outlined in Sect. "Symbolic Regression with SISSO," with the pre-screened features listed in the Supporting Information along with the assumption we made about the units of the descriptor when fed into SISSO.

Generally, we find that the main descriptors selected by the procedure are related to volume and atomic radius. Some other descriptors with less interpretability are found, such as

the C6 dispersion coefficients, polarizability, melting points, and Herfindahl-Hirschman Index (HHI) [76] production and reserve values. Although typically used to help indicate the size of a company within a particular sector of the economy, the XenonPy definition of HHI appears to come from the work of Gaultois et al [76]. In the referenced work, the HHI production value refers to the geographic distribution of elemental production (in other words, it assesses how concentrated or dispersed the global industrial effort is which produces those elements), and HHI reserve value describes the geographic distribution of known deposits of these materials (e.g., whether they are spread out over a wide area, or concentrated in a small area).

We report the best descriptor found in Eq. 1. In this equation, the variables c_0, a_0, a_1 are the regression coefficients determined by SISSO.

$$V_{Perovskite} \approx c_0 + a_0 \cdot \frac{Z^{ave}}{C^{ave} \cdot \left(r^{ave}_{Slater} - r^{ave}_{pyykko,triple}\right)} + a_1 \cdot \left(V^{ave}_{gs} - V^{\min}_{gs}\right) \cdot \frac{r^{ave}_{pyykko,triple}}{r^{ave}_{pyykko}} \quad (1)$$

where $c_0 = -10.547, a_0 = 4.556, c_1 = 3.050$, Z^{ave} is the average atomic number, C^{ave} is the average mass-specific heat capacity of the elemental solid, r^{ave}_{Slater} is the average atomic covalent radius predicted by Slater, $r^{ave}_{pyykko,triple}$ is the average triple bond covalent radius predicted by Pyyko, r^{ave}_{pyykko} is the average single bond covalent radius predicted by Pyyko, and V^{ave}_{gs} and V^{min}_{gs} are the average and minimum ground state volume per atom as calculated by DFT. Unsurprisingly the ground state atomic volumes and covalent radii play an important role in determining the final volume of the perovskite structures. Interestingly, both the atomic number and specific heat capacity of the material appear in the final descriptor. This is interesting because they do not intuitively have a connection to the unit cell volume. It's possible that these just act as another source of variance for the model to pick up on, but we also note here that it could just be a correlation with the size of the individual atoms (e.g., another source of information about the volume). For the atomic number, it is well known that it has a periodic trend with the atomic radius (e.g., He is a small atom and Cs is a very large atom).

2D material bandgaps

The bandgap predictions leveraged a data filtering strategy (described in Sect. "Data Filtering." As a result of our data filtering approach, the 6351 entries in the dataset were reduced to 1412 entries. The train/test split divided the data into a training set of 1270 rows, and a test set containing 142 entries. The performance metrics of the XGBoost, TPOT, Roost, and SISSO models of 2D Material Bandgap can be found in Table 2.

TABLE 2: Performance metrics for the XGBoost, TPOT, Roost, and SISSO models on the 2D material bandgap problem.

Error metric	Partition	XGBoost	TPOT	Roost	SISSO
MAE	Train	0.16	0.109	0.11	0.28
RMSE	Train	0.29	0.208	0.27	0.46
Max Error	Train	2.81	2.056	2.83	4.34
R^2	Train	0.965	0.982	0.968	0.912
MAE	Test	0.89	0.273	0.65	0.31
RMSE	Test	0.47	0.460	1.07	0.53
Max Error	Test	2.28	2.220	4.80	3.41
R^2	Test	0.903	0.908	0.507	0.880

The rung-2, 4-term SISSO model is reported. The parity plots for these models are depicted in 2.

Performance is generally worse on this problem when compared to the perovskite volume predictions. As a result, in addition to the compositional features of XenonPy (Sect. "Compositional Descriptors) we also used several structural features (Sect. "Structural descriptors"). We also leveraged the bulk bandgap of the parent-3D material for each of the 2D materials, as we observed the performance of the TPOT, SISSO, and XGBoost models increased when this value was included.

Although test-set model performance was worse compared to the perovskite problem, XGBoost, TPOT, and SISSO models all perform well with nearly equivalent metrics for the test-set R^2, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). We find the Roost model overfit the data to some extent on the data, as the test-set error metrics are considerably worse than their training set counterparts. A parity plot summarizing these results can be found in Figure 2. In all cases, we can see a spike of misprediction for systems with a DFT bandgap of 0. We note here that a large portion of these entries had DFT bandgaps of 0: of the 382 of the 1412 entries in the dataset, a total of 27% of all training data.

The pipeline generated by TPOT is less complex than that of the perovskite volume problem. The first stage of the pipeline is a `MaxAbsScaler` unit, scaling each feature by the maximum absoulte value of the feature. The second stage is then an `ElasticNetCV` unit, which uses 5-fold cross-validation to optimize the alpha and L1/L2 ratio of the Elastic Net model. The converged alpha value was 0.0001, and the converged L1/L2 ratio was 0.85, which strongly leans toward the L1 (Least Absolute Shrinkage and Selection Operator (LASSO)) regularization penalty. Finally, it uses an `ExtraTreesRegressor` and averages over 100 decision trees to estimate the target property. Each tree in this final step can use 80% of the features with each leaf having at least two samples and each internal node splitting at least 14 samples.

We can also extract feature importances from the XGBoost model, and we report the 10 highest-ranked features in

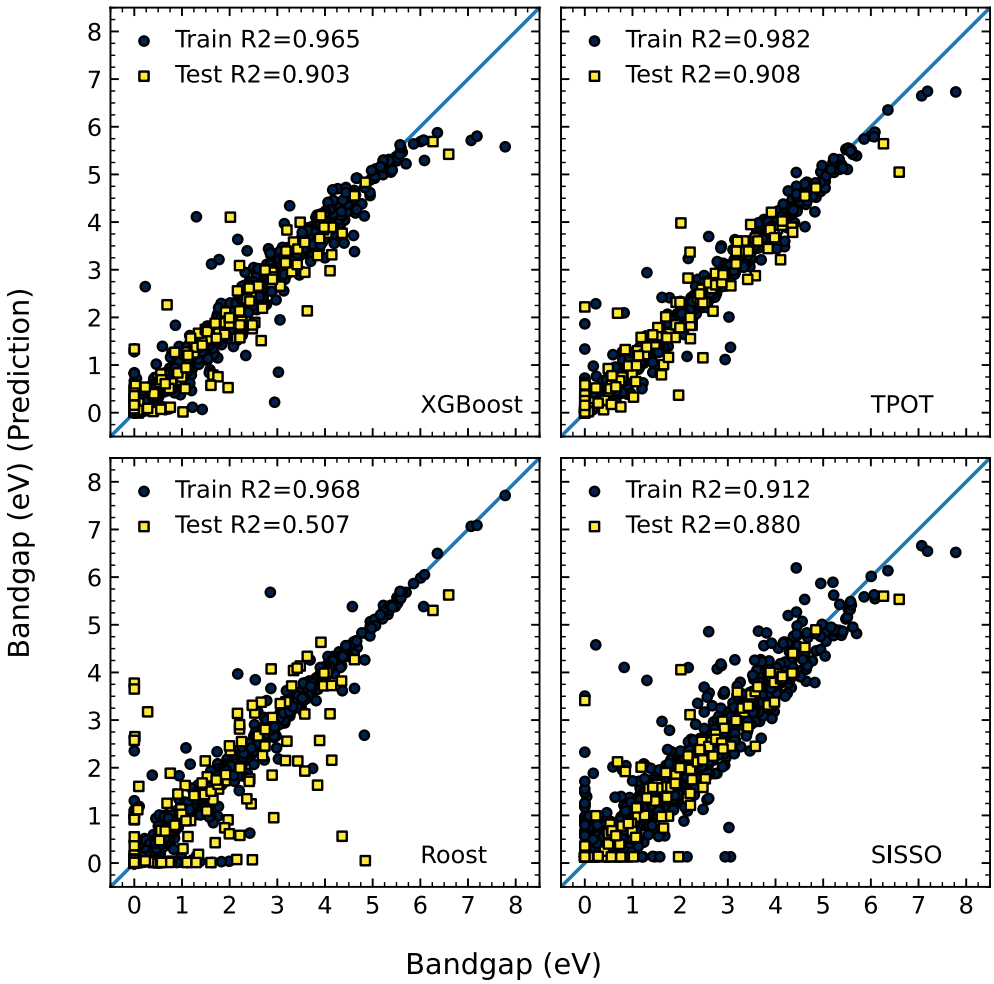

Figure 2: Parity plots for the XGBoost, TPOT, Roost, and SISSO models on the 2D material bandgap problem. Included are the training and testing sets. A diagonal line indicating parity is drawn as a guide to the eye. Regression statistics for the models shown on this plot can be found in Table 4.

Supporting Information Figure 6. Similar to the perovskite results the XGBoost model is dominated by a single feature, namely the bandgap of the parent-3D material (importance score 0.44). This feature also very important for the SISSO models, as shown in Supporting Information Table 7, signaoling that the similarity of the performance of these three models could be attributed to this feature. In fact the selected SISSO model is

$$E^{2D}_{Bandgap} \approx c_0 + a_0 \cdot \frac{Period^{ave} \cdot r^{ave}_{cov,slater}}{\left(r^{min}_{cov,cordero}\right)^3} + a_1 \cdot \frac{E^{3D,parent}_{Bandgap}}{r^{min}_{rahm}} \left(r^{min}_{vdw} + r^{min}_{cov,cordero}\right) \quad (2)$$

where $c_0 = -0.3296, a_0 = 1.69 \times 10^3, a_1 = 6.59 \times 10^{-1}$, r^{min}_{vdw} is the minimum Van der Waals radius of the atoms in the material, $r^{ave}_{cov,slater}$ is the average Slater covalent radius of an atom in the material, $r_{cov,cordero}$ is the Cordero covalent radius of an atom in the material, $E^{3D,parent}_{Bandgap}$ is the bandgap of the 3D-parent material, r^{min}_{rahm} is the minimum Rahm radius of an atom in the material, and $Period^{ave}$ is the average period of the elements in the material. This descriptor primarily represents a simple rescaling and shifting of the bandgap of the 3D-parent material, further implying the dominant role this feature plays in describing the bandgap of the 2D material.

2D material exfoliation energy

In the case of the 2D material exfoliation energy problem, the training and test-set statistics for the XGBoost, TPOT, Roost, and SISSO models can be found in Table 3. In this case, our feature selection methodology down-selected the 6351 rows of our dataset into 3388 rows. The train/test split further divided this into a training-set of 3049 entries, and a test set of 339 entries. Generally, we see the worst performance of the models in this problem, compared to the perovskite volume and 2D material bandgap problems.

TABLE 3: Performance metrics for the XGBoost, TPOT, Roost, and SISSO models on the 2D material exfoliation energy problem.

Error Metric	Partition	XGBoost	TPOT	Roost	SISSO
MAE	Train	0.20	0.04	0.06	0.27
RMSE	Train	0.35	0.13	0.24	0.48
Max Error	Train	7.11	5.12	9.63	8.89
R^2	Train	0.624	0.941	0.827	0.274
MAE	Test	0.23	0.18	0.19	0.30
RMSE	Test	0.35	0.31	0.34	0.78
Max Error	Test	1.64	1.86	1.96	12.30
R^2	Test	0.476	0.603	0.498	-1.558

Please see Methodology Sect. Data Filtering for details on how this data was filtered. The parity plots for these models are depicted in 3.

A set of parity plots for all four models is presented in Figure 3. To facilitate easier comparison at experimentally relevant energy ranges, we have zoomed the plot in such that the highest exfoliation energy is 2 eV. Plots showing the entire energy range explored can be found in Supporting Information Figure 8. Here, we find that all models perform generally poorly, with the largest errors occurring at higher exfoliation energies in the case of XGBoost, TPOT, and SISSO (see Figure 3). The best test-set R^2 and RMSE this time is only TPOT, although they are still relatively poor, with a test-set R^2 of only 0.603. Roost displays the best test-set MAE, although the model seems to have overfit, as it displays drastically better performance on the training set than it does on the test set. The XGBoost model performs slightly worse than either TPOT or Roost, and the SISSO approach did not perform well for this problem.

The TPOT algorithm results in a relatively complicated model pipeline, with multiple scaling and estimation steps. The pipeline firsts standardizes the features and then creates a linear model using linear ridge regression. From here it counts the number of zero and non-zero feature values for each samples

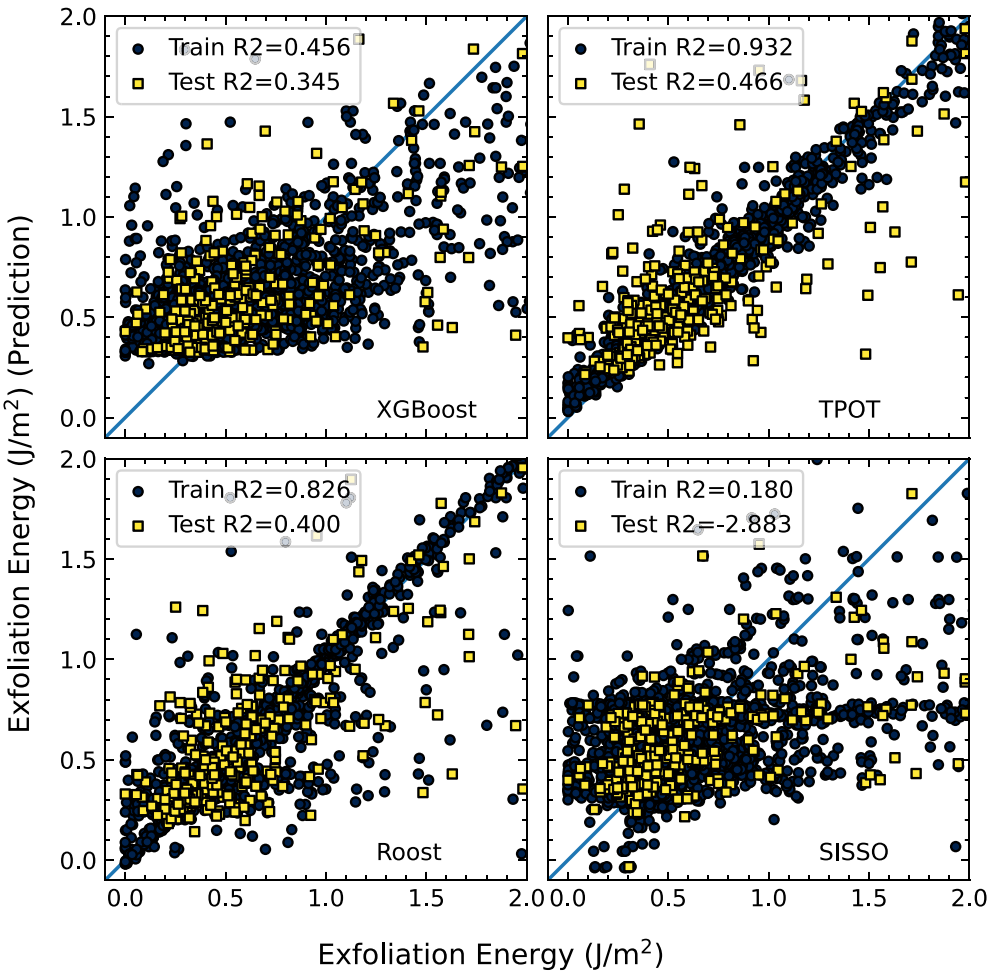

Figure 3: Parity plots for the XGBoost, TPOT, Roost, and SISSO models on the 2D material exfoliation energy problem. Included are the training and testing sets. A diagonal line indicating parity is drawn as a guide to the eye. Regression statistics for the models shown on this plot can be found in Table 3, the values presented here are for only the demonstrated data.. To facilitate comparison at energy ranges that are more experimentally relevant, we have zoomed in the plot to study energies no higher than 2 J/m^2. The full data range is plot in Supporting Information Figure 8.

and adds it to the feature set, and then rescales the features by the maximum absolute value of each feature. It then adds an `ExtraTreesRegressor` using 100 trees, 70% of the features, a minimum samples per leaf of 6, and a minimum number of samples per split of 15. The next stage is a `SelectFwe` unit, which down-selects the features according to the FWE [23]. An alpha value of 0.011 is selected for this purpose. This is then fed into a linear support vector regressor with a C value of 0.5 and rescaled such that each feature is between 0 and 1. Finally, a last `ExtraTreesRegressor` unit is used with 100 trees, a maximum of 10% of all features used and a minimum of 5 samples at each split.

We again extract features from the XGBoost model (Supporting Information Figure 7), and find the Mendeleev Number again appears as an important feature (importance score 0.08), albeit as the maximum instead of the minimum. Additionally, we see descriptors related to bond strengths in the corresponding elemental systems: average melting points (importance score 0.05), and average heats of evaporation (importance score 0.05).

The list of preselected features can be found in the Supporting Information Table 8. The best SISSO model found for this problem is

$$E_{Exf} \approx c_0 + a_0 BP^{\max} n_p^{ave} + a_1 \left(EA^{\text{var}} + EA^{ave}\right) + a_2 \frac{q_{ev}^{\min}}{V_{ICSD}^{\max}} \tag{3}$$

where $c_0 = 8.29 \times 10^{-1}$, $a_0 = -1.48 \times 10^{-4}$, $a_1 = -8.56\times$, 10^{-1}, BP^{max} is the maximum boiling point of an elemental solid of an atom in the material, $q_{evaporation}^{min}$ is the minimum atomic evaporation heat of each element in the material, EA is atomic electron affinity of an atom in the material, V_{ICSD}^{ave} is the average atomic volume in the ICSD database, and n_p^{ave} is the average number of p valance electrons. Examining this equation gives insights into why the SISSO model's performance is as poor as it is. The V_{icsd}^{ave} of graphene is 5.67, lower than any other data point in the dataset. Removing this single data point increases the Test R^2 to 0.329 and reduces the MAE, RMSE, and max error to 0.26, 0.40, and 1.88, respectively. All of which are inlined with the training results.

Discussion

We have developed a series of models which are capable of generating predictions for (1) the volume per formula unit of a series of ABX_3 perovskites, (2) the DFT-calculated bandgap of several 2D materials, and (3) several 2D material exfoliation energies. These problems encompass a variety of outcomes that one may find when training models of predictive properties.

Perovskite volume per formula unit

In the case of the volume per formula unit of ABX_3 perovskites, we observe all four model types perform well. Overall, we find that the volume per formula unit for ABX_3 perovskites can be predicted using only compositional descriptors (i.e., with no structural descriptors). The likely reason all four models perform well despite having no structural information is the general similarity in crystal structure between these systems—they are all perovskites, and therefore all possess very similar crystal structures. Supporting this is that the Roost model, which only leverages the chemical formula as an input, and which we did not optimize the hyperparameters or architecture for, performed just as well on this problem—albeit with some systematic deviation from parity at intermediate volumes (see Figure 1). Although interpretability is reduced by virtue of being a neural network, we can still achieve an important insight from this model—just by knowing the chemical formula of the system, we can achieve accurate predictions of perovskite volumes, which further justifies our use of compositional descriptors (see Sects. "Compositional Descriptors) on this problem as we move to the SISSO, XGBoost, and TPOT models. Additionally, we note this performance was achieved with a dataset containing only 129 entries—compared to the original Roost paper [34] that leveraged approximately 275,000 entries from the OQMD datset [77].

Like the Roost model, we have difficulty in interpreting the pipeline generated by TPOT. The TPOT model delivers the best performance—which is clearly visible from the parity plot in Figure 1. This performance came at a price, however, and the rather complex pipeline containing multiple feature selection steps, three estimators stacked together (the predictions from the previous are added as a new feature to the next), and a LASSO model fit using Least Angles regression.

Entering into the realm of interpretability, although the XGBoost model does not produce a direct formula for perovskite volumes, we can still gain some insight using it. It is still, however, relatively accurate—and allows us access to a feature importance metric (see Supporting Information Figure 6). In this case, we see the five most important features are the average Rahm [73, 74] atomic radius, average UFF [75] atomic radius, sum of elemental velocities of sound in the material, average Ghosh [78] electronegativity, and the sum of the Pyykko [79] triple bond covalent radii. Overall, we see a strong reliance on descriptors of atomic radius—which as we noted in the TPOT discussion makes intuitive sense.

Finally, the SISSO model (Eq. 1) offers the most direct interpretation, as it is simply an equation. Immediately, we see that a variety of descriptors related to volume are important. This result is highly intuitive and is not surprising when we consider that we are predicting volume.

The overall good performance of SISSO for this application is promising, as it is one of the most accurate models, while being by far the most interpretable. This represents a key advantage to symbolic regression, as if you can find an accurate model, then it will be easy to understand and analyze the results. Moreover, we note that are not alone in the literature when it comes to leveraging SISSO to generate models of perovskite properties—the last several years have seen success in the creation of models of perovskite properties with this tool. The work of Xie et al. [61] achieved good success in predicting the octahedral tilt in ABO_3 perovskites, the work of Bartel et al. [60] resulted in the creation of a new tolerance factor for ABX_3 perovskite formation, and Ihalage and Hao [59] leveraged descriptors generated by SISSO to predict the formation of quaternary perovskites with formula $(A_{1-x}A'_x)BO_3$ and $A(B_{1-x}B'_x)O_3$.

2D material bandgap

The 2D material bandgap models did not achieve the same performance as for the perovskite systems (see Figure 2). Even in the case of TPOT, SISSO, and XGBoost, which still had the best test-set performance by most metrics, there were a few outliers. Specifically, we find that the test-set MAE for the models ranged between 0.273 to 0.89 eV (Roost) relative to the PBE DFT calculations reported by the 2DMatPedia. Putting this number in perspective, we note the recent work of Tran et al [80], which benchmarked the bandgap predictions of several popular DFT functions for many of the systems in the C2DB; the work identified that the PBE functional exhibited a MAE of 1.50 eV relative to the G_0W_0 method. Other investigators have studied the prediction of 2D material bandgaps: Rajan et al. [81] also achieved a test-set MAE of 0.11 eV on a dataset of 23,870 MXene systems (which, as far as we are aware, has not been made publicly available) using a Gaussian Process regression approach, with DFT-calculated properties including the average M–X bond length, volume per atom, MXene phase, and heat of formation, and compositional properties including the mean Van der Waals radius, standard deviation of periodic table group number, standard deviation of the ionization energy, and standard deviation of the meting temperature. Zhang et al. [82] improved on this error slightly, achieving a test-set MAE of 0.10 eV on the C2DB dataset (around 4000 entries) [14] with both Support-Vector Regression and Random Forest approaches, albeit using descriptors such as the Fermi-energy density of states and total energy of the system (requiring further DFT work for additional prediction). In contrast to both approaches, which used DFT-calculated values that would need to be obtained for new systems to be predicted, the only DFT-calculated value we leverage in our feature set is a bulk bandgap tabulated on the Materials Project [18]. Thus, although our TPOT model had a slightly higher MAE, we note that this would not require further DFT work to generate new predictions.

In addition, we note that although we considered the bandgap of the corresponding bulk material, we did not consider the crystal structure of the corresponding bulk material. As this the electronic properties are heavily influenced by the structure, future work should evaluate the effect of crystal structure on bandgap models in order to ensure robustness across a wide array of structures.

As 2D systems are still relatively novel, we note that much more work has been performed in the 3D materials space, particularly in the leveraging of neural networks to predict bandgaps. The recent Atomistic LIne Graph Neural Network (ALIGNN) [83] reported a test-set MAE of 0.218 eV for the prediction of bulk materials hosted by Materials Project [18] (which as of October 2021 has over 144,000 inorganic systems). The Materials Graph Network (MEGNet) architecture [84] achieved a test-set MAE of 0.32 eV on the bulk systems of the Materials Project. Although these neural network models are on 3D systems, we note that they do not leverage DFT properties (which we re-iterate would cause any resulting model to require a DFT calculation for future prediction) and had access to much larger datasets than the training set we obtained after filtering the 2DMatPedia entries (see Sect. "Data Filtering"). Overall, although the systems we investigate are not 3D bulk systems, we believe this puts the TPOT MAE for the bandgap of 2D systems in perspective.

In all 4 models we trained, many of the incorrect predictions occur where the DFT bandgap is 0 eV (which represented 27% of the training set values). Because of this, we tried simplifying the bandgap problem, by training an XGBoost model to predict whether the system was a metal (see Supporting Information section 6.5.3), and showed that we could achieve good results — for the sake of trying a variety of approaches, we also incorporated a purely structural fingerprint, the Sine Matrix Eigenspectrum (see Supporting Information Section 6.5.1). As this descriptor resulted in some rather large vectors (of length 40, the maximum number of atoms in any system) with little direct physical intuition, we do not directly include it for the purposes of this section. Ultimately, that the Sine Matrix Eigenspectrum provides a useful model indicates the incorporation of structural features can provide useful information to predict the bandgap.

If we take a closer look at the Roost model, we can see a poor generalization to the test set (see Table 3). This indicates that we have likely caused it to overfit (which could have been improved for example through the use of early stopping). Given that Roost is a purely compositional model, this reinforces our conclusion that structural descriptors are necessary to the prediction of the bandgap of these systems.

Future work on this problem may achieve better performance on the bandgap problem by incorporating other structural features (e.g., investigating the bond strengths of the different elements in the system). We also note the very good performance that recent neural network approaches have had on the 3D bandgap problem [83, 84], likely due to their choices in representation of the structure of the 3D systems. Similar to how the Roost model achieves good success when compositional descriptors are appropriate, we may find good success in leveraging neural network approaches when structural features are required. We note here that Deng et al. [31] achieved good results on a variety of molecule properties by incorporating various graph representations from different neural network architectures. Hence, future work in this domain may benefit from the incorporation of the information-dense structural fingerprints that may be obtained from neural network-based approaches.

2D material exfoliation energy

We observed some of the worst model performance (across all models) in the case of the 2D material exfoliation energy. Despite being a larger dataset than either the perovskite (144 total, 129 in the training set) or bandgap (1,412 total, 1,270 in the training set), the 3,049 entries in the training set (out of 3,388 total) for the exfoliation energy proved insufficient to achieve good results for any of the models. Moreover, neither the compositional nor the structural features were sufficient to adequately describe the system.

When we predict exfoliation energies, we're predicting the interaction between layers in an exfoliable material. Overall, finding better methods of cheaply approximating these weak interactions may provide better results in the prediction of exfoliation. Additionally, as the number of datasets which contain exfoliation energies increases (such as the 2DMatPedia [11], C2DB [14, 15], and JARVIS [17]), further insight into this problem will be possible, and more-complex (albeit less interpretible) models will become feasible.

Additionally, in order to obtain more-accurate predictions of exfoliation energy, data generated via a more thorough computational treatment may be required. We illustrate this by examining an outlier in the training set at 9.9 J / m^2, which all four models heavily under-predicted (see Supporting Information Figure 8) (7–8 eV in the case of XGBoost and TPOT, and over 9 eV in the case of Roost and SISSO). Upon closer examination of this system, we find that it is actually a pair of layers containing N atoms (Figure 4A). The 2DMatPedia [11] reports that this system (2dm-id 5985) was not directly sourced by a simulated exfoliation from a bulk structure, but instead was obtained by substituting the atoms in a hypothetical 2D Sb structure (Figure 4B). The Sb structure (2dm-id 4275) was obtained by a simulated exfoliation from a structure obtained from materials project (Figure 4C). The parent bulk material (mp-567409) is reported by the Materials Project [18] to be a monoclinic crystal which undergoes a favorable decomposition (energy above hull is reported as 0.121 eV/atom) to a triclinic system. That being said, as this is a hypothetical 2D system, comparison with the hypothetical 3D bulk system was necessary for the calculation of exfoliation energy. As the prediction of crystal structure is a very challenging field with few easy approximations [85], this may have contributed further to the extreme value of the exfoliation energy. Indeed, as Zhou et al report [11], the decomposition energy lends itself better to assessing whether a material is truly stable. Indeed, despite the extremely high exfoliation energy of this hypothetical 2D N system, it is reported by the 2DMatPedia to have a decomposition energy of 0 eV/atom. This too seems somehwat high, as systems containing N-N bonds tend to be high-energy materials, typically undergoing strongly exothermic decomposition to inert, gaseous N_2 [86]. With this in conjunction with the observation that our models all predict exfoliation energies significantly lower than the tabulated values, we have reason to believe that this system would be far easier to exfoliate than the 10 eV exfoliation energy implies. Moreover, this system may have a strong energetic preference to decompose further into N_2, which additional DFT work could reveal. Overall, this underscores the importance of obtaining high-quality data, and filtering that high-quality data, for the training of interpretable models.

Future outlook

As ML is further integrated into materials discovery workflows, we anticipate that the numerous successes neural networks have presented [1–3, 32–41, 87–89] will continue to propel them onto the cutting edge of chemical property prediction. This comes with the challenge of honing our techniques for their interpretation, an area which has seen much interest in recent years, and where there is still plenty of opportunity for further development [90, 91]. We also expect AutoML techniques such as TPOT will continue gaining traction in materials discovery, due to the amount of success and attention they have recently had [43–48]. This too presents the challenge of interpretability if highly complex pipelines are generated (see Sect. 2.1 and 3.1). We note here that part of the value that AutoML techniques bring is the ability to make advanced techniques accessible to a wider audience of researchers by lowering the barrier of entry. Hence, we expect that the problem of interpretation may be compounded for AutoML (and especially NAS) systems: the ability to automatically extract some level of interpretation from the generated pipelines is important for automation to make ML truly accessible to non-experts. Overall, we expect that as neural network models and AutoML algorithms continue to grow in capability

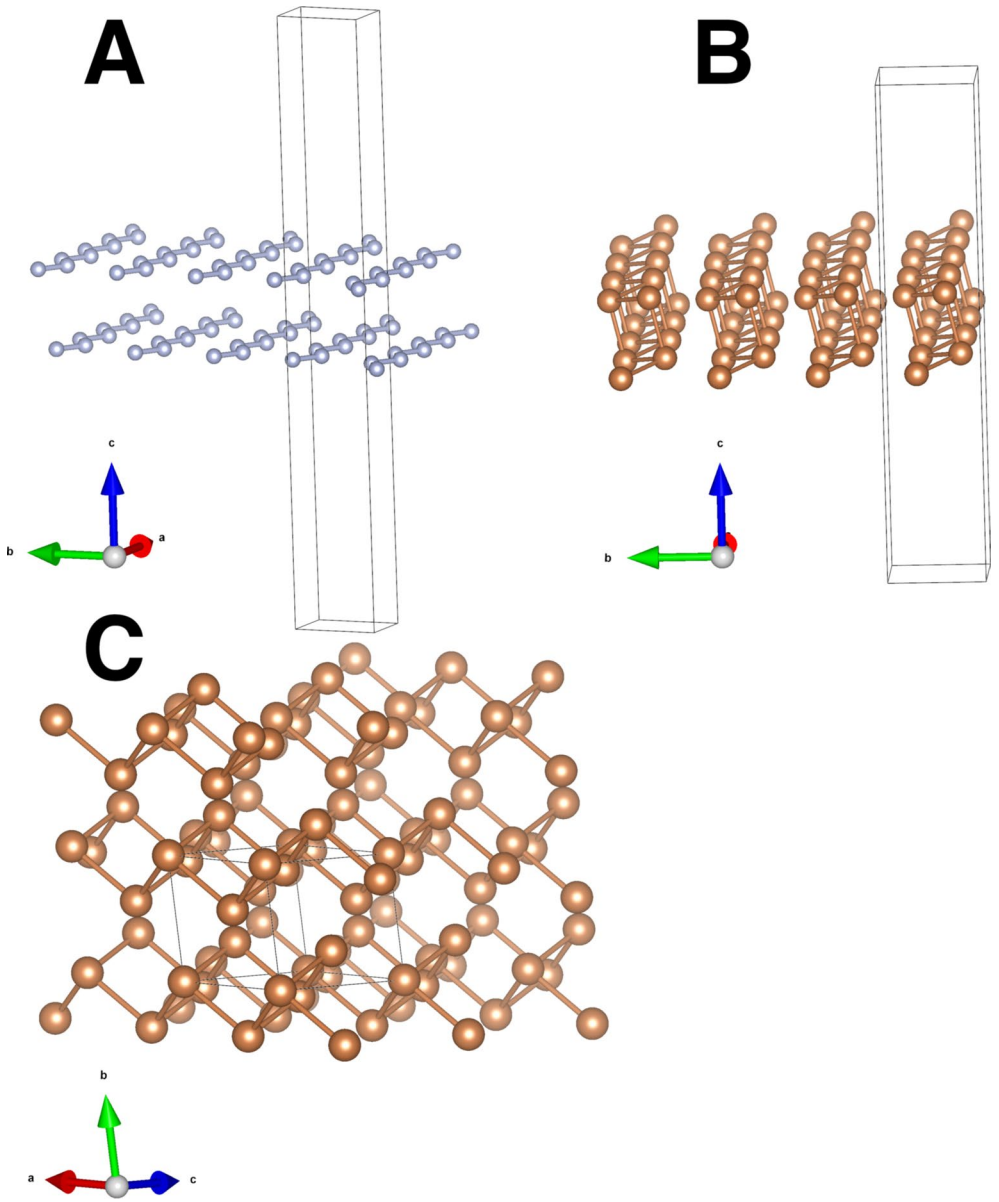

Figure 4: Illustrations of (A) a N-containing system (2dm-id 5985) which persisted as a large outlier across all exfoliation models in the training set, (B) the Sb structure (2dm-id 4275) the N-containing system was derived from, and (C) the bulk structure from Materials Project (mp-567409) from which the exfoliation of the Sb system was simulated.

and complexity, work in developing the tools and techniques needed to interpret them will see a greater attention.

In contrast with challenge of interpreting neural networks or the pipelines found by AutoML systems, symbolic regression tools like Eureqa and SISSO yield an exact equation describing the model and are thus easier to interpret. This makes it easier to achieve key insights with physical interpretations — such as the very intuitive way in which SISSO is able to describe the systems. Overall, despite its reduced ability to predict the exfoliation energy of a material when compared to the models of TPOT, XGBoost, and Roost, we note the mathematical equations returned by SISSO provide a direct relationship between the target properties and model predictions. Additionally, in the case of the exfoliation energy, we believe that we may see further improvements by including richer structural information. We base this on the observation that the Roost model performed poorly on both of this problems – recalling that Roost is only provided the chemical formula of the system, this could indicate that compositional descriptors alone are insufficient to describe these properties. Indeed, it is well known that structure and energy are intimately related (the fundamental assumption of geometry optimization techniques is that energy is a function of

atomic position), hence it can be inferred that exfoliation energy and structure are similarly related. In the case of bandgaps, we note that there is also a strong dependence on structure; Chaves et al [68] notes that the number of layers in a 2D material can strongly influence the bandgap, reporting differences of up to several eV can occur between the bulk and monolayer form of a material.

Interoperability is still a challenge in the materials discovery ecosystem. Although it is possible to easily convert between different chemical file formats (e.g., by OpenBabel [92]), and packages such as Pymatgen [93], Atomic Simulation Environment (ASE) [94], and RDKit [95] can easily convert to each others' format, we note that there is a challenge of calculating features using a variety of different packages. Some tools expect Pymatgen objects (e.g., XenonPy), others expect ASE objects, whereas others require RDKit objects (e.g., all of the descriptors in the RDKit library) to perform a calculation of features, thus creating some standard for the interoperability of these packages would be beneficial. Additionally, further efforts should be made to report the sources of data used by featurization packages. We note that MatMiner [96] is exemplary in this regard: each of the featurization classes it defines has a "citation" method returning the appropriate source to credit. Mendeleev [97] is another good example of this; within its documentation, a table lists citations for many (though not all) of the elemental properties it can return. Overall, by placing a stronger focus on i) interoperability and ii) data provenance, the Python materials modeling ecosystem can be made stronger—and therefore help accelerate materials discovery.

Moreover, we note that as models continue to grow in complexity, it will continue to be more important to evaluate their complexity if they are to be used for practical applications. Although we did not benchmark the models in this study for time, if one were to deploy a model in a production setting, it would be important to understand the CPU / GPU and memory requirements for training and inference to be possible.

All of the models we have investigated in this work required sufficient training data to avoid overfitting. Although techniques such as cross-validation, early stopping (in the case of neural networks and XGBoost), and train/test splitting can help guard against (and detect) overfitting, having a sufficiently large dataset is of the utmost importance to achieve truly generalizable models. As a result, there is a critical need for data management approaches that satsify the set of FAIR principles. This crucial need for effective data management has led to the incorporation of data storage tooling in popular chemistry packages including Pymatgen [93], ASE [94], and RDKit [95]. Moreover, advances in both computational capacity and techniques has given rise to studies performing the high-throughput screening of chemical systems [98–100]. This has resulted in the development of tools focusing on the provenance of data, such as the Automated Interactive Infrastructure and Database for Computational Science (AiiDA) system [101, 102].

Overall, we have identified a series of key issues should see more attention as the digital ecosystem surrounding materials modeling continues to develop. First, interpretability of models allows us to derive physical understanding from the available data. This is a key benefit of symbolic regression tools like SISSO, which result in the creation of human-readable equations describing the model. Additionally, increasing the accessibility of ML techniques through automation (such as in the field of AutoML) will allow a wider range of researchers the ability to benefit from advances in modeling techniques. Data management and data provenance are another major issues, which allow us to better understand which datasets can be combined (e.g., when combining DFT datasets, the methodologies should be consistent between them), and to help us understand if something intrinsic to the training data is affecting model performance. These data management goals are core focus of platforms such as Exabyte [103], which provides an all-in-one solution for i) storing material data and metadata, ii) storing the methodology required to derive a property from a material, and iii) providing the means to automatically perform calculations, and iv) automatically extracting calculation results and storing them for the user. This focus on providing a tool that manages materials, workflows, and calculations has allowed Exabyte to be a highly successful platform, which has led to studies involving automated phonon calculations [104], high-throughput screening of materials for their band structure [105, 106]. Future capabilities of the platform are slated to include a categorization scheme for computational models to provide even more metadata to track the provenance of calculated material properties [107].

Conclusion

In this work, we have performed a series of benchmarks on a diverse set of ML algorithms: gradient boosting (XGBoost), AutoML (TPOT), deep learning (Roost), and symbolic regression (SISSO). These models were used to predict (i) the volume of perovskites, (ii) the DFT bandgap of 2D materials, and (iii) the exfoliation energy of 2D materials. We identify that TPOT, SISSO, and XGBoost tend to produce more-accurate models than Roost, but Roost works well in systems where compositional descriptors are enough to predict the target property. Finally, although SISSO was unable to find an accurate model for the exfoliation energy, it provides a human-readable equation describing the model, facilitating an easier interpretation compared to the other algorithms. We believe that interpretability will remain a key challenge to address as complex techniques (i.e., neural networks and AutoML) become more mainstream within the digital materials modeling ecosystem. Overall, as tools improving the accessibility of machine-learning continue

to be developed, data provenance and model interpretability will become even more important, as it is a critical part of ensuring the accessibility of these techniques. By working to ensure that a wider audience of researchers can achieve insight from the rich digital ecosystem of materials design, materials discovery can be accelerated.

Methodology

Data sources

Crystal structures for the perovskite systems were obtained from the "Stable Inorganic Perovskites" dataset published by Körbel, Marques, and Botti [70], as hosted by NOMAD [19]. This dataset contains a total of 144 DFT-relaxed inorganic perovskites identified via a high-throughput screening strategy. Using this dataset, we develop a model of perovskite volume. As we rely on the use of compositional descriptors for these systems, we have scaled the volume of the perovskite unit cell by the number of formula units, such that the volume has units of $Å^3$ / formula unit.

Structures for 2D materials were obtained from the 2DMatPedia [11], a large database containing a mixture of 6,351 real and hypothetical 2D systems. This database was generated via a DFT-based high-throughput screening approach, which investigated bulk structures hosted by the Materials Project database [18] to find systems which may plausibly form 2D structures. Among other things, the 2DMatPedia provides DFT-calculated exfoliation energies and bandgaps, along with a DFT-optimized structure for each material. We use this dataset to develop models for the bandgap and exfoliation energy of 2D materials. Although the dataset reports exfoliation energies in units of eV, to facilitate comparison with other works focusing on 2D material exfoliation energy, we have converted these into units of J / m^2. Bandgaps are reported in units of eV.

Because datasets may change and evolve over time, we note that all datasets used in this work were accessed during the time period between June and December of 2021. Further details on the datasets can be found in our supporting information section 6.8.

Feature engineering

To facilitate the development of ML algorithms capable of rapidly predicting material properties, we focus primarily on features that do not require further (computationally intensive) DFT calculations. A variety of chemical featurization libraries were used to generate compositional and structural descriptors for the systems we investigated, and they are listed in Sects. "Compositional Descriptors" and "Structural descriptors," respectively. Features with values of NaN (which occurred when a feature could not be calculated) were assigned a value of 0.

In the case of the 2D material bandgap, we include the DFT-calculated bandgap of the respective bulk material; we note that these values are tabulated on the Materials Project and can be looked up, thus circumventing the need for further DFT work. We also note that the 2D and 3D material bandgaps are highly correlated with one-another. In effect, our model becomes a correction on top of the 3D bandgap term and furthermore reduces its applicability to systems with a corresponding 3D parent to be derived from. We acknowledge this produces a slightly less interesting result, but ultimately included it given the difficulty of the bandgap prediction problem.

Compositional descriptors

Compositional (i.e., chemical formula-based) descriptors were calculated via the open-source XenonPy packaged developed by Yamada et al [108]. XenonPy uses tabulated elemental data from Mendeleev [97], Pymatgen [93], the CRC Handbook of Chemistry and Physics [109], and Magpie [110] in order to calculate compositional features. XenonPy does this by combining the elemental descriptors (e.g., atomic weight, ionization potential, etc.) in various ways to form a single composition-weighted value. For example, three compositional descriptors may be obtained with XenonPy by taking the composition-weighted average, sum, or maximum elemental value of the atomic weight. Leveraging the full list of compositional features implemented in XenonPy results to 290 compositional descriptors, which are explained in greater detail within their publication [108].

The 290 compositional descriptors were used for the perovskite volume prediction, 2D material bandgap, and 2D material exfoliation energy prediction problems. We note that these descriptors were not used in the Roost model, as it directly takes the composition for its input.

Structural descriptors

Some structural descriptors were calculated using MatMiner [96], an open-source Python package geared toward data-mining material properties. Leveraging MatMiner, the following 9 descriptors were calculated: Average bond length, average bond angle, Global Instability Index (GII) [111], Ewald Summation Energy [112], a Shannon Information Entropy-based Structural Complexity (both per atom and per cell), and the number of symmetry operations available to the system. In the case of the average bond length and average bond angle, bonds were determined using Pymatgen's implementation of the JMol [113] AutoBond algorithm. This list of bonds was also used to calculate an average Coordination Number (CN) over all atoms in the unit cell. Finally, we also took the perimeter:area ratio of the 2D material's repeating unit.

The structural descriptors were used for the 2D material bandgap and 2D material exfoliation energy problems. We did

not use these descriptors in the case of the perovskite volume prediction problem, and we note that they were not used as inputs to the Roost model.

Data filtering

The data filtering methodology was chosen based on the problem at-hand. The perovskite volume prediction problem did not utilize any data filtering. In the case of the 2D material bandgap and exfoliation energy prediction problems, the data obtained from the 2DMatPedia were required to satisfy all of the following criteria:

1. No elements from the f-block, larger than U, or noble gases were allowed.
2. Decomposition energy must be below 0.5 eV/atom.
3. Exfoliation energy must be strictly positive.

Additionally, in the case of the 2D material bandgap, data were required to have a parent material defined on the Materials Project. This was done because we use the Materials Project's tabulated DFT bandgap of the bulk system as a descriptor for the bandgap of the corresponding 2D system.

ML models

For each dataset investigated, 10% of the given dataset was randomly selected to be held out as a testing set. The same train/test split was used for all 4 models considered (XGBoost, TPOT, Roost, and SISSO). To facilitate a transparent comparison between models, in all cases we report the MAE, RMSE, Maximum Error, and R^2 score of the test set.

Gradient boosting with XGBoost

For details on how XGBoost works, we refer the reader to the XGBoost publication by Chen and Guestrin [24] and to the package's documentation located at the following URL: https://xgboost.readthedocs.io/en/stable/. When training XGBoost models, 20% of randomly selected data were held out as an internal validation set. This was used to adopt an early-stopping strategy, where if the model RMSE did not improve after 50 consecutive rounds, training was halted early. When training, XGBoost was configured to optimize its RMSE.

Hyperparameters were optimized via the open-source Optuna [114] framework. The hyperparameter space was sampled using the Tree-structured Parzen Estimator (TPE) approach [115, 116]. To accelerate the hyperparameter search, we leveraged the Hyperband [117] approach for model pruning, using the validation set RMSE to determine whether to prune a model. Hyperband's budget for the number of trees in the ensemble was set to range between 1 and 256 (corresponding with the maximum number of estimators we allowed an XGBoost model to have). The search space for hyperparameters is found in Table 4.

The variable names here (e.g., `learning_rate`) correspond with the variable names listed in the documentation of XGBoost. Additionally, Optuna was used to select a standardization strategy, choosing between Z-score normalization (i.e., subtracting the mean and dividing by the standard deviation) or Min/Max scaling (i.e., scaling the data such that it has minimum 0 and maximum 1). To prevent test-set leakage, the chosen standardizer was fit only with the internal training set, i.e., the portion of the training set that was not held out as an internal validation set. Optuna performed 1000 trials to minimize the validation set RMSE. We report the results of the final optimized model.

AutoML with TPOT

The AutoML tool TPOT was leveraged with a population size of 100 pipelines, with training proceeding for a total of 100 generations. The default maximum evaluation time of 5 minutes per model was set. As TPOT is an actively maintained open-source repository, for the purposes of future replication we enumerate this configuration's set of allowable components in Table 55. The models listed in this table could be combined in any order any number of times. Models were selected such that their 10-fold cross-validated RMSE was optimized. TPOT also conducts its own internal optimization of model hyperparameters, thus we did not perform our own hyperparameter optimization of the TPOT pipelines.

Neural networks with Roost

The Roost Neural Network (NN) architecture was leveraged using the "example.py" script provided with its source code. Roost is a message-passing graph neural network which leverages the material stoichiometry instead of the material structure for its inputs. For details on the specific architecture (e.g., the number of message-passing layers, activation functions, etc.) we refer the reader to the original paper by Goodall et.

Table 4: Ranges of hyperparameters screened with Optuna for all XGBoost runs. The search was inclusive of the listed minima and maxima. Hyperparameters use the same variable naming convention as in the XGBoost documentation.

Hyperparameter	Minimum	Maximum
`Learning_rate`	0	2
`Min_split_loss`	0	2
`Max_depth`	0	256
`Min_child_weight`	0	10
`Reg_lambda`	0	2
`Reg_alpha`	0	2

al. for more details on how it is formulated [34]. Our work used the reference implementation found on the GitHub page reported by the original Roost publication.

Models are trained for a total of 512 epochs with the default settings. In the case of Roost models, the only feature provided is the composition of the system, given through the chemical formula.

Symbolic regression with SISSO

The first step of using SISSO is reducing the number of primary features down from a list of hundreds down to the tens. This is done due to the exponential computational cost of SISSO with respect to the number of features and the number of rungs being considered. To perform this down selection we first generate a rung 1 feature space including all of the primary features and operators that are used in the SISSO calculation. We then check how often each of the primary features appear in the ten thousand generated features that are most correlated to the target property. Additionally, we add units to all of the preselected primary features to ensure all generated expressions are valid.

In many cases, it was easy to infer what the abstract units are for the XenonPy descriptors. In a few cases where the units weren't as clear, we compared the reported elemental values of those units to those of known sources (e.g., the NIST WebBook [118] or the CRC Handbook [109]) in order to determine the units. Finally, although it was generally easy to determine where the source of a feature was, sometimes we were unable to determine a source. In these cases, we refer to the features as a "XenonPy" feature (for example, "$r_{XenonPy}$").

The optimal number of terms (up to 3) and rung (up to 2), i.e., the number of times operators is recursively applied to the feature space, is determined using a five-fold cross-validation scheme. For all models, we allow for an external bias term to be non-zero and use a SIS selection size of 500. The resulting descriptors were then evaluated using the same external test set for each of the other methods. To take advantage of SISSO's ability to generate new composite descriptors and operate in large feature spaces, additional features were included in the SISSO calculations. A full list of features used in the SISSO work can be found in the linked GitHub repository.

Author contributions

JD and TB were responsible for performing the calculations and preparing the manuscript. TP performed and advised on the SISSO calculations. MS, RB, SB, and TB guided and conceptualized the project. All authors contributed in revising and writing the manuscript.

Funding

This research was supported by the US Department of Energy (DoE) Small Business Innovation Research (SBIR) program (grant no. DE-SC0021514). Computational resources were provided by Exabyte Inc. T.P. and M.S. were funded by the NOMAD CoE (Novel Materials Discovery Center of Excellence, European Union's Horizon 2020 research and innovation program, grant agreement N° 951786), the project TEC1p (European Research Council, grant agreement N° 740233), and the project FAIRmat (FAIR Data Infrastructure for Condensed-Matter Physics and the Chemical Physics of Solids, German Research Foundation, project N° 460197019). T.P. would like to thank the Alexander von Humboldt Foundation for their support through the Alexander von Humboldt Postdoctoral Fellowship Program.

Data availability

Copies of the datasets used in this work can be found at Exabyte's GitHub (https://github.com/Exabyte-io/Scientific-Projects/tree/Updates_29_09_22/DigitalEcosystem/raw_data) in the form of serialized Python objects (pkl files).

Code availability

Jupyter (Python) notebooks are available on Exabyte's GitHub (https://github.com/Exabyte-io/Scientific-Projects/tree/Updates_29_09_22), which contains code to reproduce our results and figures.

Declarations

Conflicts of interest James Dean carried out this work while employed by Exabyte Inc. Timur Bazhirov is currently employed by Exabyte Inc.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1557/s43578-023-01164-w.

References

1. C. Draxl, M. Scheffler, Big data-driven materials science and Its FAIR data infrastructure, in *Handbook of materials modeling: methods: theory and modeling*. ed. by W. Andreoni, S. Yip (Springer International Publishing, Cham, 2020), pp.49–73. https://doi.org/10.1007/978-3-319-44677-6_104
2. A..C. Mater, M..L. Coote, Deep learning in chemistry. J. Chem. Info. Model. **59**(6), 2545–2559 (2019). https://doi.org/10.1021/acs.jcim.9b00266
3. K..T. Butler, D..W. Davies, H. Cartwright, O. Isayev, Aon Walsh, Machine learning for molecular and materials science.

Nature **559**(7715), 547–555 (2018). https://doi.org/10.1038/s41586-018-0337-2

4. L. Bornmann, R. Mutz, Growth rates of modern science: a bibliometric analysis based on the number of publications and cited references. J. Assoc. Info. Sci. Technol. **66**(11), 2215–2222 (2015). https://doi.org/10.1002/asi.23329
5. J. Derek, *Price little science* (Columbia University Press, New York, 1963). https://doi.org/10.7312/pric91844
6. J. Derek, *Price science since Babylon* (Yale University Press, New Haven, 1975)
7. D..J. de Solla, Price, networks of scientific papers. Science **149**(3683), 510–515 (1965). https://doi.org/10.1126/science.149.3683.510
8. National Science and Technology Council. Materials Genome Initiative for Global Competitiveness. Government, White House Office of Science and Technology Policy, United States of America, (2011)
9. Subcommittee on the Materials Genome Initiative Committee on Technology. Materials Genome Initiative Strategic Plan. Government, National Science and Technology Council, United States of America, (2021)
10. J..J. de Pablo, N..E. Jackson, M..A. Webb, L.-Q. Chen, J..E. Moore, D. Morgan, R. Jacobs, T. Pollock, D..G. Schlom, E..S. Toberer, J. Analytis, I. Dabo, D..M. DeLongchamp, G..A. Fiete, G..M. Grason, G. Hautier, Y. Mo, K. Rajan, E..J. Reed, E. Rodriguez, V. Stevanovic, J. Suntivich, K. Thornton, J.-C. Zhao, New frontiers for the materials genome initiative. Comput. Mater. **5**(1), 1–23 (2019). https://doi.org/10.1038/s41524-019-0173-4
11. J. Zhou, L. Shen, M..D. Costa, K..A. Persson, S..P. Ong, P. Huck, Y. Lu, X. Ma, Y. Chen, H. Tang, Y..P. Feng, 2DMatPedia, an open computational database of two-dimensional materials from top-down and bottom-up approaches. Sci. Data **6**(1), 86 (2019). https://doi.org/10.1038/s41597-019-0097-3
12. S. Curtarolo, W. Setyawan, G..W. Hart, M. Jahnatek, R..V. Chepulskii, R..H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, M..J. Mehl, H..T. Stokes, D..O. Demchenko, D.. Morgan, AFLOW: an automatic framework for high-throughput materials discovery. Comput. Mater. Sci. **58**, 218–226 (2012a). https://doi.org/10.1016/j.commatsci.2012.02.005
13. S. Curtarolo, W. Setyawan, S. Wang, J. Xue, K. Yang, R..H. Taylor, L..J. Nelson, G..L..W. Hart, S. Sanvito, M. Buongiorno-Nardelli, N. Mingo, O. Levy, AFLOWLIB.ORG: a distributed materials properties repository from high-throughput ab initio calculations. Comput. Mater. Sci. **58**, 227–235 (2012b). https://doi.org/10.1016/j.commatsci.2012.02.002
14. M.N. Gjerding, A. Taghizadeh, A. Rasmussen, S. Ali, F. Bertoldo, T. Deilmann, N.R. Knøsgaard, M. Kruse, A.H. Larsen, S. Manti, T.G. Pedersen, U. Petralanda, T. Skovhus, M.K. Svendsen, J.J. Mortensen, T. Olsen, K.S. Thygesen, Recent progress of the computational 2D materials database (C2DB). 2D Mater. **8**(4), 044002 (2021). https://doi.org/10.1088/2053-1583/ac1059
15. S. Haastrup, M. Strange, M. Pandey, T. Deilmann, P.S. Schmidt, N.F. Hinsche, M.N. Gjerding, D. Torelli, P.M. Larsen, A.C. Riis-Jensen, J. Gath, K.W. Jacobsen, J.J. Mortensen, T. Olsen, K.S. Thygesen, The computational 2D materials database: high-throughput modeling and discovery of atomically thin crystals. 2D Mater. **5**(4), 042002 (2018). https://doi.org/10.1088/2053-1583/aacfc1
16. D..D. Landis, J..S. Hummelshøj, S. Nestorov, J. Greeley, M. Dułak, T. Bligaard, J..K. Nørskov, Karsten W. Jacobsen, The computational materials repository. Comput. Sci. Eng. **14**(6), 51–57 (2012). https://doi.org/10.1109/MCSE.2012.16
17. K. Choudhary, K..F. Garrity, A..C..E. Reid, B.. DeCost, A..J. Biacchi, A..R. Hight Walker, Z. Trautt, J. Hattrick-Simpers, A..G. Kusne, A. Centrone, A. Davydov, J. Jiang, R. Pachter, G. Cheon, E. Reed, A. Agrawal, X. Qian, V. Sharma, H. Zhuang, S..V. Kalinin, B..G. Sumpter, G. Pilania, P. Acar, S. Mandal, K. Haule, D. Vanderbilt, K. Rabe, F. Tavazza, The joint automated repository for various integrated simulations (JARVIS) for data-driven materials design. Comput. Mater. **6**(1), 1–13 (2020). https://doi.org/10.1038/s41524-020-00440-1
18. A. Jain, S.P. Ong, G. Hautier, W. Chen, W.D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, K.A. Persson, Commentary: the materials project: a materials genome approach to accelerating materials innovation. APL Mater. **1**(1), 011002 (2013). https://doi.org/10.1063/1.4812323
19. C. Draxl, M. Scheffler, The NOMAD laboratory: from data sharing to artificial intelligence. J. Phys. Mater. **2**(3), 036001 (2019). https://doi.org/10.1088/2515-7639/ab13bb
20. S. Kirklin, J..E. Saal, B. Meredig, A. Thompson, J..W. Doak, M. Aykol, S. Rühl, C. Wolverton, The open quantum materials database (OQMD): assessing the accuracy of DFT formation energies. Comput. Mater. **1**(1), 1–15 (2015). https://doi.org/10.1038/npjcompumats.2015.10
21. T. van der Ploeg, P.C. Austin, E.W. Steyerberg, Modern modelling techniques are data hungry: a simulation study for predicting dichotomous endpoints. BMC Med. Res. Methodol. **14**(1), 137 (2014). https://doi.org/10.1186/1471-2288-14-137
22. L. Mason, J. Baxter, P. Bartlett, M. Frean, *Boosting algorithms as gradient descent. Advances in neural information processing systems* (MIT Press, Cambridge, 2000)
23. T. Hastie, R. Tibshirani, J.H. Friedman, *The elements of statistical learning: data mining, inference, and prediction. Springer series in statistics* (Springer, New York, 2009)
24. T. Chen, C. Guestrin. X.G. Boost, A Scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016. https://doi.org/10.1145/2939672.2939785
25. H. Liang, K. Jiang, T.-A. Yan, G.-H. Chen, XGBoost: an optimal machine learning model with just structural features to discover MOF adsorbents of Xe/Kr. ACS Omega **6**(13), 9066–9076 (2021). https://doi.org/10.1021/acsomega.1c00100

26. N.A. Husna, A. Bustamam, A. Yanuar, D. Sarwinda, O. Hermansyah, The comparison of machine learning methods for prediction study of type 2 diabetes mellitus's drug design. AIP Conf. Proc. **2264**(1), 030010 (2020). https://doi.org/10.1063/5.0024161
27. P.D. Ivatt, M.J. Evans, Improving the prediction of an atmospheric chemistry transport model using gradient-boosted regression trees. Atmospheric Chem. Phys. **20**(13), 8063–8082 (2020). https://doi.org/10.5194/acp-20-8063-2020
28. C.W. Schmidt, Tox 21: new dimensions of toxicity testing. Environ. Health Perspect. **117**(8), A348–A353 (2009). https://doi.org/10.1289/ehp.117-a348
29. D.L. Mobley, J.P. Guthrie, FreeSolv: a database of experimental and calculated hydration free energies, with input files. J. Comput.-Aided Mol. Design **28**(7), 711–720 (2014). https://doi.org/10.1007/s10822-014-9747-x
30. M. Kuhn, I. Letunic, L.J. Jensen, P. Bork, The SIDER database of drugs and side effects. Nucleic Acids Res. **44**(D1), D1075-1079 (2016). https://doi.org/10.1093/nar/gkv1075
31. D. Deng, X. Chen, R. Zhang, Z. Lei, X. Wang, F. Zhou, XGraphBoost: extracting graph neural network-based features for a better prediction of molecular properties. J. Chem. Info. Model. **61**(6), 2697–2705 (2021). https://doi.org/10.1021/acs.jcim.0c01489
32. J. Behler, M. Parrinello, Generalized neural-network representation of high-dimensional potential-energy surfaces. Phys. Rev. Lett. **98**(14), 146–401 (2007). https://doi.org/10.1103/PhysRevLett.98.146401
33. T. Xie, J.C. Grossman, Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. Phys. Rev. Lett. **120**(14), 145–301 (2018). https://doi.org/10.1103/PhysRevLett.120.145301
34. R.A. Goodall, A.A. Lee, Predicting materials properties without crystal structure: deep representation learning from stoichiometry. Nat. Commun. **11**(1), 6280 (2020). https://doi.org/10.1038/s41467-020-19964-7
35. J. Behler, Four generations of high-dimensional neural network potentials. Chem. Rev. **121**(16), 10037–10072 (2021). https://doi.org/10.1021/acs.chemrev.0c00868
36. Kun Yao, John E. Herr, David W. Toth, Ryker Mckintyre, John Parkhill, The TensorMol-0.1 model chemistry: a neural network augmented with long-range physics. Chemical Science **9**(8), 2261–2269 (2018). https://doi.org/10.1039/C7SC04934J
37. J. Westermayr, M. Gastegger, P. M.arquetand, Combining SchNet and SHARC: the SchNarc machine learning approach for excited-state dynamics. J. Phys. Chem. Lett. **11**(10), 3828–3834 (2020). https://doi.org/10.1021/acs.jpclett.0c00527
38. K.T. Schütt, P.-J. Kindermans, H.E. Sauceda, S. Chmiela, A. Tkatchenko, K.-R. Müller, SchNet: a continuous-filter convolutional neural network for modeling quantum interactions. Mach. Learn. (2017). https://doi.org/10.48550/arXiv.1706.08566
39. A. Toniato, P. Schwaller, A. Cardinale, J. Geluykens, T. Laino, Unassisted noise reduction of chemical reaction datasets. Nat. Mach. Intel. **3**(6), 485–494 (2021). https://doi.org/10.1038/s42256-021-00319-w
40. A.C. Vaucher, P. Schwaller, J. Geluykens, V.H. Nair, A. Iuliano, T. Laino, Inferring experimental procedures from text-based representations of chemical reactions. Nat. Commun. **12**(1), 2573 (2021). https://doi.org/10.1038/s41467-021-22951-1
41. J. Panteleev, H. Gao, L. Jia, Recent applications of machine learning in medicinal chemistry. Bioorganic Med. Chem. Lett. **28**(17), 2807–2815 (2018). https://doi.org/10.1016/j.bmcl.2018.06.046
42. Y. Liang, S. Li, C. Yan, M. Li, C. Jiang, Explaining the black-box model: a survey of local interpretation methods for deep neural networks. Neurocomputing **419**, 168–182 (2021). https://doi.org/10.1016/j.neucom.2020.08.011
43. P. Gijsbers, E. LeDell, J. Thomas, S. Poirier, B. Bischl, J, Vanschoren. An open source autoML benchmark. arXiv:1907.00909*[cs, stat]*, (2019)
44. Q. Yao, M. Wang, Y. Chen, W. Dai, Y-F. Li, W-W. Tu, Q. Yang, Y. Yu, Taking Human out of Learning applications: a survey on automated machine learning. arXiv:1810.13306*[cs, stat]*, December (2019)
45. X. He, K. Zhao, X. Chu, AutoML: a survey of the state-of-the-art. Knowl.-Based Syst. **212**, 106622 (2021)
46. T.T. Le, F. Weixuan, J.H. Moore, Scaling tree-based automated machine learning to biomedical big data with a feature set selector. Bioinformatics **36**(1), 250–256 (2020). https://doi.org/10.1093/bioinformatics/btz470
47. R.S. Olson, R.J. Urbanowicz, P.C. Andrews, N.A. Lavender, L.C. Kidd, J.H. Moore, *Automating biomedical data science through tree-based pipeline optimization. Applications of evolutionary computation lecture notes in computer science* (Springer International Publishing, Cham, 2016), pp.123–137
48. R.S. Olson, N. Bartley, R.J. Urbanowicz, J.H. Moore, Evaluation of a Tree-based Pipeline Optimization Tool for Automating Data Science. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, GECCO '16, pp. 485–492, New York, (2016b). Association for Computing Machinery. https://doi.org/10.1145/2908812.2908918
49. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
50. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, *PyTorch: an imperative style, high-performance deep learning*

library. Advances in neural information processing systems (Curran Associates Inc., New York, 2019), pp.8024–8035

51. M. Amir Haeri, M.M. Ebadzadeh, G. Folino, Statistical genetic programming for symbolic regression. Appl. Soft Comput. **60**, 447–469 (2017). https://doi.org/10.1016/j.asoc.2017.06.050
52. K.E. Kinnear, W.B. Langdon, L. Spector, P.J. Angeline, *Una-May O'Reilly. Advances in genetic programming* (MIT Press, Cambridge, 1994)
53. Michael Schmidt, Hod Lipson, Distilling free-form natural laws from experimental data. Science **324**(5923), 81–85 (2009). https://doi.org/10.1126/science.1165893
54. D.R. Stoutemyer, Can the Eureqa symbolic regression program, computer algebra and numerical analysis help each other? arXiv:1203.1023*[cs]*, (2012)
55. J. Dean, M.G. Taylor, G. Mpourmpakis, Unfolding adsorption on metal nanoparticles: connecting stability with catalysis. Sci. Adv. **5**(9), eaax5101 (2019). https://doi.org/10.1126/sciadv.aax5101
56. Kaiyang Tan, Mudit Dixit, James Dean, Giannis Mpourmpakis, Predicting metal-support interactions in oxide-supported single-atom catalysts. Indust. Eng. Chem. Res. **58**(44), 20236–20246 (2019). https://doi.org/10.1021/acs.iecr.9b04068
57. R. Ouyang, S. Curtarolo, E. Ahmetcik, M. Scheffler, L.M. Ghiringhelli, SISSO: a compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates. Phys. Rev. Mater. **2**(8), 083 (2018). https://doi.org/10.1103/PhysRevMaterials.2.083802
58. R. Ouyang, E. Ahmetcik, C. Carbogno, M. Scheffler, L.M. Ghiringhelli, Simultaneous learning of several materials properties from incomplete databases with multi-task SISSO. J. Phys.: Mater. **2**(2), 024–002 (2019). https://doi.org/10.1088/2515-7639/ab077b
59. A. Ihalage, Y. Hao, Analogical discovery of disordered perovskite oxides by crystal structure information hidden in unsupervised material fingerprints. Comput. Mater. **7**(1), 1–12 (2021). https://doi.org/10.1038/s41524-021-00536-2
60. C.J. Bartel, C. Sutton, B.R. Goldsmith, R. Ouyang, C.B. Musgrave, L.M. Ghiringhelli, M. Scheffler, New tolerance factor to predict the stability of perovskite oxides and halides. Sci. Adv. **5**(2), eaav0693 (2019). https://doi.org/10.1126/sciadv.aav0693
61. Stephen R. Xie, Parker Kotlarz, Richard G. Hennig, Juan C. Nino, Machine learning of octahedral tilting in oxide perovskites by symbolic classification with compressed sensing. Comput. Mater. Sci. **180**, 109–690 (2020). https://doi.org/10.1016/j.commatsci.2020.109690
62. C.M. Acosta, R. Ouyang, A. Fazzio, M. Scheffler, L.M. Ghiringhelli, C. Carbogno, Analysis of topological transitions in two-dimensional materials by compressed sensing. arXiv:1805.10950*[cond-mat]*, May 2018
63. C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1**(5), 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x
64. S. Zeng, P. Kar, U.K. Thakur, K. Shankar, A review on photocatalytic co2 reduction using perovskite oxide nanomaterials. Nanotechnology **29**(5), 052001 (2018). https://doi.org/10.1088/1361-6528/aa9fb1
65. P. Roy, N.K. Sinha, S. Tiwari, A. Khare, A review on perovskite solar cells: evolution of architecture, fabrication techniques, commercialization issues and status. Solar Energy **198**, 665–688 (2020)
66. F. Xue, C. Zhang, Y. Ma, Y. Wen, X. He, Y. Bin, X. Zhang, Integrated memory devices based on 2d materials. Adv. Mater. **34**(48), 2201880 (2022). https://doi.org/10.1002/adma.202201880
67. M. Long, P. Wang, H. Fang, H. Weida, Progress, challenges, and opportunities for 2d material based photodetectors. Adv. Funct. Mater. **29**(19), 1803807 (2019). https://doi.org/10.1002/adfm.201803807
68. A. Chaves, J.G. Azadani, H. Alsalman, D.R. da Costa, R. Frisenda, A.J. Chaves, S.H. Song, Y.D. Kim, D. He, J. Zhou, A. Castellanos-Gomez, F.M. Peeters, Z. Liu, C.L. Hinkle, S.-H. Oh, P.D. Ye, S.J. Koester, Y.H. Lee, P. Avouris, X. Wang, T. Low, Bandgap engineering of two-dimensional semiconductor materials. 2D Mater. Appl. **4**(1), 29 (2020). https://doi.org/10.1038/s41699-020-00162-4
69. M.A. Islam, P. Serles, B. Kumral, P.G. Demingos, T. Qureshi, A. Meiyazhagan, A.B. Puthirath, M.S.B. Abdullah, S.R. Faysal, P.M. Ajayan, D. Panesar, C.V. Singh, T. Filleter, Exfoliation mechanisms of 2D materials and their applications. Appl. Phys. Rev. **9**(4), 041301 (2022). https://doi.org/10.1063/5.0090717
70. S. Körbel, M.A.L. Marques, S. Botti, Stability and electronic properties of new inorganic perovskites from high-throughput ab initio calculations. J. Mater. Chem. C **4**(15), 3157–3167 (2016). https://doi.org/10.1039/C5TC04172D
71. C. Draxl, M. Scheffler, NOMAD: The FAIR concept for big-data-driven materials science. arXiv:1805.05039*[cond-mat, physics:physics]*, May 2018
72. P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees. Mach. Learn. **63**(1), 3–42 (2006). https://doi.org/10.1007/s10994-006-6226-1
73. M. Rahm, R. Hoffmann, N.W. Ashcroft, Atomic and Ionic Radii of Elements 1–96. Chem. European J. **22**(41), 14625–14632 (2016). https://doi.org/10.1002/chem.201602949
74. Martin Rahm, Roald Hoffmann, N.. W. Ashcroft, Corrigendum: atomic and ionic radii of elements. Chem. European J. **23**(16), 4017–4017 (2017). https://doi.org/10.1002/chem.201700610
75. A.K. Rappe, C.J. Casewit, K.S. Colwell, W.A. Goddard, W.M. Skiff, UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations. J. Am. Chem. Soc. **114**(25), 10024–10035 (1992). https://doi.org/10.1021/ja00051a040

76. M.W. Gaultois, T.D. Sparks, C.K.H. Borg, R. Seshadri, W.D. Bonificio, D.R. Clarke, Data-driven review of thermoelectric materials: performance and resource considerations. Chem. Mater. **25**(15), 2911–2920 (2013). https://doi.org/10.1021/cm400893e
77. D. Jha, L. Ward, A. Paul, W.-K. Liao, A. Choudhary, C. Wolverton, A. Agrawal, ElemNet: deep learning the chemistry of materials from only elemental composition. Sci. Rep. **8**(1), 17593 (2018). https://doi.org/10.1038/s41598-018-35934-y
78. D.C. Ghosh, A new scale of electronegativity based on absolute radii of atoms. J. Theoretical Comput. Chem. **04**(01), 21–33 (2005). https://doi.org/10.1142/S0219633605001556
79. P. Pyykkö, S. Riedel, M. Patzschke, Triple-bond covalent radii. Chem. European J. **11**(12), 3511–3520 (2005). https://doi.org/10.1002/chem.200401299
80. F. Tran, J. Doumont, L. Kalantari, P. Blaha, T. Rauch, P. Borlido, S. Botti, M.A.L. Marques, A. Patra, S. Jana, P. Samal, Bandgap of two-dimensional materials: thorough assessment of modern exchange–correlation functionals. J. Chem. Phys. **155**(10), 104–103 (2021)
81. A.C. Rajan, A. Mishra, S. Satsangi, R. Vaish, H. Mizuseki, K.-R. Lee, A.K. Singh, Machine-learning-assisted accurate band gap predictions of functionalized MXene. Chem. Mater. **30**(12), 4031–4038 (2018). https://doi.org/10.1021/acs.chemmater.8b00686
82. Y. Zhang, X. Wenjing, G. Liu, Z. Zhang, J. Zhu, M. Li, Bandgap prediction of two-dimensional materials using machine learning. PLOS ONE **16**(8), e0255637 (2021). https://doi.org/10.1371/journal.pone.0255637
83. K. Choudhary, Brian DeCost. Atomistic line graph neural network for improved materials property predictions. arXiv:2106.01829*[cond-mat]*, (2021)
84. C. Chen, W. Ye, Y. Zuo, C. Zheng, S.P. Ong, Graph networks as a universal machine learning framework for molecules and crystals. Chem. Mater. **31**(9), 3564–3572 (2019). https://doi.org/10.1021/acs.chemmater.9b01294
85. A.R. Oganov, *Modern methods of crystal structure prediction* (Wiley-VCH, Weinheim, 2011)
86. D. Kumar, A.J. Elias, The explosive chemistry of nitrogen. Resonance **24**(11), 1253–1271 (2019). https://doi.org/10.1007/s12045-019-0893-2
87. P. Schwaller, R. Petraglia, V. Zullo, V.H. Nair, R.A. Haeuselmann, R. Pisoni, C. Bekas, A. Iuliano, T. Laino, Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. Chem. Sci. **11**(12), 3316–3325 (2020). https://doi.org/10.1039/C9SC05704H
88. P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C.A. Hunter, C. Bekas, A.A. Lee, Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. ACS Central Sci. **5**(9), 1572–1583 (2019). https://doi.org/10.1021/acscentsci.9b00576
89. Philippe Schwaller, Théophile. Gaudin, Dávid. Lányi, Costas Bekas, Teodoro Laino, Found in translation: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. Chem. Sci. **9**(28), 6091–6098 (2018). https://doi.org/10.1039/C8SC02339E
90. F.-L. Fan, J. Xiong, M. Li, G. Wang, On interpretability of artificial neural networks: a survey. IEEE Trans. Radiat. Plasma Med Sci (2021). https://doi.org/10.1109/TRPMS.2021.3066428
91. Y. Zhang, P. Tiňo, A. Leonardis, K. Tang, A survey on neural network interpretability. IEEE Trans. Emerg. Topics Comput. Intell. **5**(5), 726–742 (2021b). https://doi.org/10.1109/TETCI.2021.3100641
92. N.M. O'Boyle, M. Banck, C.A. James, C. Morley, T. Vandermeersch, G.R. Hutchison, Open babel: an open chemical toolbox. J. Cheminformatics **3**(1), 33 (2011). https://doi.org/10.1186/1758-2946-3-33
93. S.P. Ong, W.D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V.L. Chevrier, K.A. Persson, G. Ceder, Python Materials Genomics (pymatgen): a robust, open-source python library for materials analysis. Comput. Mater. Sci. **68**, 314–319 (2013). https://doi.org/10.1016/j.commatsci.2012.10.028
94. A.H. Larsen, J.J. Mortensen, J. Blomqvist, I.E. Castelli, R. Christensen, M. Dulak, J. Friis, M.N. Groves, B. Hammer, C. Hargus, E.D. Hermes, P.C. Jennings, P.B. Jensen, J. Kermode, J.R. Kitchin, E.L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J.B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K.S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, K.W. Jacobsen, The atomic simulation environment— a python library for working with atoms. J. Phys.: Condensed Matter. **29**(27), 273–002 (2017). https://doi.org/10.1088/1361-648X/aa680e
95. G. Landrum, P. Tosco, B. Kelley, *sriniker, gedeck, NadineSchneider, Riccardo Vianello, Ric, Andrew Dalke, Brian Cole, AlexanderSavelyev, Matt Swain, Samo Turk, Dan N, Alain Vaucher, Eisuke Kawashima, Maciej Wójcikowski, Daniel Probst, guillaume godin, David Cosgrove, Axel Pahl, JP, Francois Berenger, strets123, JLVarjo, Noel O'Boyle, Patrick Fuller* (Gianluca Sforna, and DoliathGavid. RDKit, Jan Holst Jensen, 2021)
96. Logan Ward, Alexander Dunn, Alireza Faghaninia, N.E.R. Zimmermann, S. Bajaj, Q. Wang, J. Montoya, J. Chen, K. Bystrom, M. Dylla, K. Chard, M. Asta, K.A. Persson, G. Jeffrey Snyder, I. Foster, A. Jain, Matminer: an open source toolkit for materials data mining. Comput. Mater. Sci. **152**, 60–69 (2018). https://doi.org/10.1016/j.commatsci.2018.05.018
97. Łukasz Mentel. Mendeleev – A Python resource for properties of chemical elements, ions and isotopes, ver. 0.9.0, (2014)
98. Bingbing Zhang, Xiaodong Zhang, Yu. Jin, Ying Wang, Wu. Kui, Ming-Hsien. Lee, First-Principles High-Throughput Screening Pipeline for Nonlinear Optical Materials: Application to Borates. Chemistry of Materials **32**(15), 6772–6779

(2020). https://doi.org/10.1021/acs.chemmater.0c02583. (**ISSN 0897-4756**)

99. Lorenz M. Mayr, Dejan Bojanic, Novel trends in high-throughput screening. Current Opinion in Pharmacology **9**(5), 580–588 (2009). https://doi.org/10.1016/j.coph.2009.08.004
100. James Dean, Michael J. Cowan, Jonathan Estes, Mahmoud Ramadan, Giannis Mpourmpakis, Rapid prediction of bimetallic mixing behavior at the nanoscale. ACS Nano **14**(7), 8171–8180 (2020). https://doi.org/10.1021/acsnano.0c01586
101. M. Uhrin, S.P. Huber, J. Yu, N. Marzari, G. Pizzi, Workflows in AiiDA: engineering a high-throughput, event-based engine for robust and modular computational workflows. Comput. Mater. Sci. **187**, 110–086 (2021). https://doi.org/10.1016/j.commatsci.2020.110086
102. S.P. Huber, S. Zoupanos, M. Uhrin, L. Talirz, L. Kahle, R. Häuselmann, D. Gresch, T. Müller, A.V. Yakutovich, C.W. Andersen, F.F. Ramirez, C.S. Adorf, F. Gargiulo, S. Kumbhar, E. Passaro, C. Johnston, A. Merkys, A. Cepellotti, N. Mounet, N. Marzari, B. Kozinsky, G. Pizzi, AiiDA 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. Sci. Data **7**(1), 300 (2020). https://doi.org/10.1038/s41597-020-00638-4
103. T. Bazhirov, Data-centric online ecosystem for digital materials science. arXiv:1902.10838*[cond-mat, physics:physics]*, (2019)
104. T. Bazhirov, E. X. Abot, Fast and accessible first-principles calculations of vibrational properties of materials. arXiv:1808.10011*[cond-mat, physics:physics]*, (2018)
105. P. Das, M. Mohammadi, T. Bazhirov, Accessible computational materials design with high fidelity and high throughput. arXiv:1807.05623*[cond-mat, physics:physics]*, (2018)
106. P. Das, T. Bazhirov, Electronic properties of binary compounds with high fidelity and high throughput. J. Phys.: Conf. Series **1290**, 012–011 (2019). https://doi.org/10.1088/1742-6596/1290/1/012011
107. A. Zech, T. Bazhirov, CateCom: a practical data-centric approach to categorization of computational models. J. Chem. Inf. Model. **62**(5), 1268–1281 (2022). https://doi.org/10.1021/acs.jcim.2c00112
108. H. Yamada, C. Liu, S. Wu, Y. Koyama, S. Ju, J. Shiomi, J. Morikawa, R. Yoshida, Predicting materials properties with little data using shotgun transfer learning. ACS Central Sci. **5**(10), 1717–1730 (2019). https://doi.org/10.1021/acscentsci.9b00804
109. J.R. Rumble, T.J. Bruno, M.J. Doa, *CRC handbook of chemistry and physics: a ready-reference book of chemical and physical data* (CRC Press, Boca Raton, 2021)
110. L. Ward, A. Agrawal, A. Choudhary, C. Wolverton, A general-purpose machine learning framework for predicting properties of inorganic materials. Comput. Mater. **2**(1), 1–7 (2016). https://doi.org/10.1038/npjcompumats.2016.28
111. A. Salinas-Sanchez, J.L. Garcia-Muñoz, J. Rodriguez-Carvajal, R. Saez-Puche, J.L. Martinez, Structural characterization of R2BaCuO5 (r = y, lu, yb, tm, er, ho, dy, gd, eu and sm) oxides by x-ray and neutron diffraction. J. Solid State Chem. **100**(2), 201–211 (1992). https://doi.org/10.1016/0022-4596(92)90094-C
112. P.P. Ewald, Die Berechnung optischer und elektrostatischer Gitterpotentiale. Annalen der Physik **369**(3), 253–287 (1921). https://doi.org/10.1002/andp.19213690304
113. Jmol development team. Jmol, (2016)
114. T. Akiba, S. Sano, T. Yanase, T. Ohta, M. K. Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pp. 2623–2631, New York, (2019). Association for Computing Machinery. https://doi.org/10.1145/3292500.3330701
115. J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, *Algorithms for hyper-parameter optimization. Advances in neural information processing systems* (Curran Associates Inc., New York, 2011)
116. J. Bergstra, D. Yamins, D. Cox, Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on Machine Learning*, pp 115–123. PMLR, (2013)
117. L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, A. Talwalkar, Hyperband: a novel bandit-based approach to hyperparameter optimization. J. Mach. Learn. Res. **18**(1), 6765–6816 (2017)
118. NIST Chemistry WebBook, *NIST standard reference database number 69* (National Institute of Standards and Technology, Gaithersburg, 2021)