LLNL-JRNL-816318

Predicting Energetics Materials' Crystalline Density from Chemical Structure by Machine Learning

P. Nguyen, D. Loveland, J. T. Kim, P. Karande, A. M. Hiszpanski, T. Y.-J. Han

November 3, 2020

Journal of Chemical Information and Modeling

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Predicting Energetics Materials' Crystalline Density from Chemical Structure by Machine Learning

Phan Nguyen,[†,§] Donald Loveland,[‡,§] Joanne T. Kim,[¶] Piyush Karande,[†] Anna M. Hiszpanski,[*,‡] and T. Yong-Jin Han[*,‡]

[†]*Computational Engineering Division, Lawrence Livermore National Laboratory, Livermore, California 94550, United States*

[‡]*Materials Science Division, Lawrence Livermore National Laboratory, Livermore, California 94550, United States*

[¶]*Computing Scholar Program, Lawrence Livermore National Laboratory, Livermore, California 94550, United States*

[§]*Co-first authors*

E-mail: hiszpanski2@llnl.gov; han5@llnl.gov

Abstract

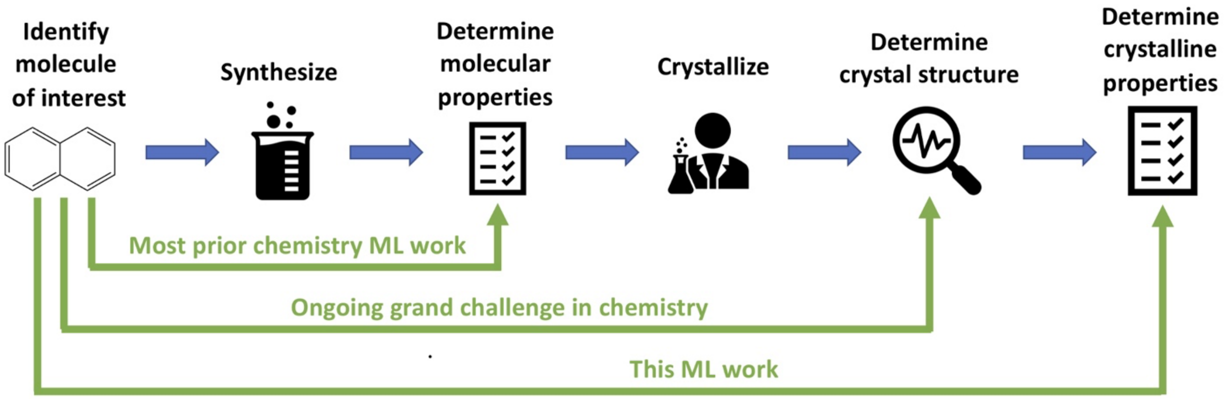
To expedite new molecular compound development, a long-sought goal within the chemistry community has been to predict molecules' bulk properties of interest a priori to synthesis from a chemical structure alone. In this work, we demonstrate that machine learning methods can indeed be used to directly learn the relationship between chemical structures and bulk crystalline properties of molecules, even in the absence of any crystal structure information or quantum mechanical calculations. We focus specifically on a class of organic compounds categorized as energetic materials called high explosives (HE) and predicting their crystalline density. An ongoing challenge within the chemistry machine learning community is deciding how best to featurize molecules as inputs into machine learning models—whether expert handcrafted features or learned molecular representations via graph-based neural network models—yield better results and why. We evaluate both types of representations in combination with a number of machine learning models to predict the crystalline densities of HE-like molecules curated from the Cambridge Structural Database, and we report the performance and pros and cons of our methods. Our message passing neural network (MPNN) based models with learned molecular representations generally perform best, outperforming current state-of-the-art methods at predicting crystalline density and performing well even when testing on a dataset not representative of the training data. However, these models are traditionally considered black boxes and less easily interpretable. To address this common challenge, we also provide a comparison analysis between our MPNN-based model and models with fixed feature representations that provides insights as to what features are learned by the MPNN to accurately predict density.

Introduction

The discovery of new molecular compounds, including energetics, pharmaceutics, organic semiconductors, and food additives, is a labor-intensive and costly Edisonian process, driven by cycles of informed design, synthesis, crystallization, characterization, and property testing. The ability to predict molecular compounds' bulk crystalline properties from a chemical structure alone and a priori to synthesis has been desired for decades to reduce new compounds' development time. However, this goal remains largely elusive as the bulk properties of molecular compounds are often significantly impacted by the crystal structure they adopt, and accurately and efficiently predicting molecular crystal structures using quantum mechanics-based computational approaches is challenging due to the significant influence that crystallization conditions and molecular conformational flexibility (i.e., ability to change 3-dimensional structure) have on crystal structure.[1] Thus, the determination of molecules' crystal structures and their crystalline properties has remained a largely experimental endeavor and bottleneck in new molecular compounds discovery pipelines.

In this work, we demonstrate that machine learning (ML) approaches can be used to directly learn the relationship between chemical structures and bulk crystalline properties of molecular compounds and to make predictions even in the absence of crystal structure information. While a variety of ML approaches have been demonstrated for predicting molecular-level properties, including energy levels and lipophilicity,[2–8] use of ML approaches to predict *bulk crystalline* properties of molecular compounds are far less explored with only a handful of examples.[8–13] Figure 1 provides an overview of the typical process involved with new molecular compounds' development and highlights how our ML approach can provide a shortcut to the time-consuming synthesis, crystallization, and characterization steps that are normally required to know the crystalline properties of a new molecule. We specifically focus our studies on predicting the crystalline density of a class of energetic materials called high explosives (HE), since density of molecular HE directly relates to detonation velocity—an important performance metric when evaluating molecular HE candidates. Furthermore, the development and testing of new HE is particularly hazardous work that would especially benefit from early prioritization and minimization of samples to be synthesized and studied.

Aside from synthesizing and experimentally determining the density of HE candidates or attempting to adopt ML approaches, as we do here, the current best method of attaining a density approximation is via quantum mechanics-based density functional theory (DFT) calculations. Specifically, studying small datasets of known HE molecules, Rice et al.[14] and Qiu et al.[15] separately showed that density of energetics can be approximated by dividing a molecule's molecular weight (which is known precisely from the chemical structure alone) by its molecular volume, approximated using electron density isosurfaces calculated with DFT. Generally within the HE commu-



Identify molecule of interest
Synthesize
Determine molecular properties
Crystallize
Determine crystal structure
Determine crystalline properties
Most prior chemistry ML work
Ongoing grand challenge in chemistry
This ML work

Figure 1: Schematic of the typical process and labor involved with researching new molecular compounds (blue arrows). Highlighted in green are various possible shortcuts that machine learning can enable to expedite new molecule development. By directly predicting crystalline properties from a molecular structure alone, as we do in this work, many of the time-consuming steps normally involved with synthesis, crystallization, and characterization may be prioritized.

nity, if a density prediction error is less than 0.03 g/cc, the prediction is considered "excellent"; predictions with errors within 0.03 and 0.05 g/cc are considered "informative;" predictions with error between 0.05 and 0.10 g/cc are "barely usable;" and predictions with errors greater than 0.10 g/cc are "deceptive."[16,17] For 180 C-, H-, N-, and O-containing HE molecules, Rice et al.'s quantum mechanical-based approach yielded 41.1% of predictions that were excellent within 0.03 g/cc and 21.7% that were informative within 0.03 and 0.05 g/cc.[14,18] However, this approach thereby yields over a third of predictions with errors of 0.05 g/cc or greater.[14,18] As noted by Politzer et al., one likely source for these high errors is the fact that the DFT calculations used to approximate the molecular volume are performed on single molecules (i.e., molecules in the gas phase), thereby omitting intermolecular interactions typically present and affecting density in crystalline materials.[18] Politzer et al. introduced an electrostatic correction factor that helps to improve the accuracy of these DFT-based density predictions.[18] However, still none of the approaches published thus far provide accurate predictions and are over 0.1 g/cc too low for important test cases like 1,3,5-triamino-2,4,6-trinitrobenzene (TATB)—a particularly useful HE molecule given both its simultaneously high performance (stemming from its high density of 1.93 g/cc[19]) and high insensitivity to mechanical shock.[20]

In this work, we describe a curated dataset of HE-like molecules and report and compare machine-learned models to predict the densities of these molecules from chemical structure alone—without crystal structure information or requiring any DFT calculations. In devising these models, we evaluated a number of different machine-readable molecular representations, or featurization methods, and a number of different model architectures—both of which significantly affect the predictive performance of ML models but in ways that are difficult to predict beforehand. We note that certain combinations do not need to be evaluated due to incompatibilities between specific molecular featurization types and ML models, and we quickly determined that the longest-standing and popular ASCII string-based molecular representation method called Simplified Molecular Input Line Entry System or SMILES strings[21] does not perform well for this crystalline density prediction task. We instead focus on more traditional and easily interpretable models (specifically random forest and partial least squares regression) with expert-crafted and predefined molecular-level summarizations of features, as well as more recently developed graph-based models (i.e., message-passing neural networks (MPNNs)) that encode instead only atom and bond information into a machine-learned molecular representation of relevant molecular features and that are therefore less easily interpretable. We report here the performance and pros and cons of each to aide in further ML model development for molecular compounds. In particular, across tests on more than 10000 HE-like molecules with our best model—the largest HE-relevant test set reported to date—61% of predictions were "excellent" and 22% were "informative", and the model performs well even in the density prediction of notoriously difficult molecules like TATB, yielding a prediction of 1.95 g/cc that is only 0.02 g/cc higher than the experimentally reported value.

Materials and Methods

Dataset

A challenge to machine learning for HE molecules has been the lack of readily available large datasets; datasets of approximately 300 HE molecules are considered large for the HE community.[14,22] To address this issue, we curated a dataset of energetic-like molecules from the Cambridge Structural Database (CSD),[23] a repository containing over a million published and experimentally-derived organic, inorganic, and metal–organic small-molecule crystal structures. We sub-selected from the CSD database molecules that either are known HE or are similar to this family of compounds by imposing the following restrictions: 1) molecules must be composed of only carbon, hydrogen, nitrogen, and oxygen, 2) molecules must contain a nitrogen-oxygen bond of any type (e.g., single, double, triple), and 3) the crystal structure cannot be solvated or one of co-crystals (i.e., the crystal structure contains only one type of molecule).

From this subset, we then removed a molecule if its 3D structure could not be correctly constructed from its crystal structure file. In addition, we removed any molecule that was missing a published density value. We note that a molecule in CSD may contain a calculated density as well as a published density provided by the authors of the source publication for the molecule. We found that these values differed in some cases, and so we chose to treat the published density values as ground truth; additional details of the dataset may be found in the Supporting Information. Furthermore, for any molecules that had more than one crystal structure meeting all the above criteria and therefore had multiple densities associated with it, we took only the highest density, which commonly (but not exclusively) corresponds to a lower energy and more stable form.[24] Using the molecules available in CSD as of December 2018, this filtering process resulted in dataset of 10251 energetic or energetic-like molecules for our regression analyses. We provide a list of these molecules' CSD reference codes as a list in the Supporting Information.

The distribution of crystalline densities for the filtered molecules comprising our dataset is shown in Figure 2. The distribution is concentrated at intermediate densities of approximately 1.35 g/cc with a relatively smaller number of values at the low and high extremes, the latter of which is relevant to HE studies[25,26] for its correlation with detonation velocity. Due to the imbalance of density values and the importance of high density datapoints, we apply stratified k-fold cross validation when fitting our models to ensure that the folds are representative of the whole population of densities of the HE-related dataset, similar to what would be trained in a production environment, thereby allowing for a fair prediction assessment and accurate determination of performance in the regimes of interest.

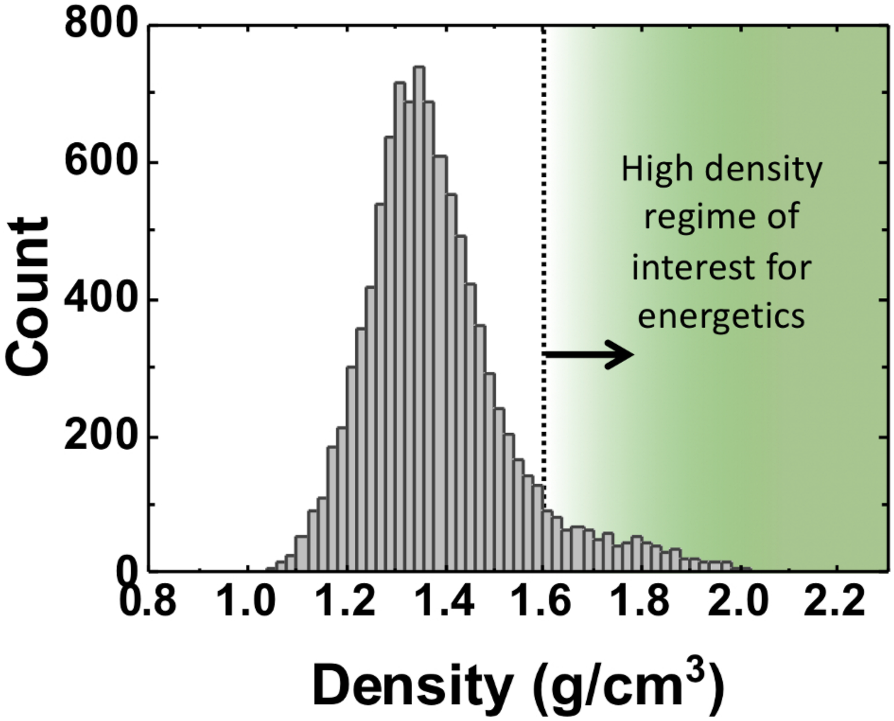


Figure 2: Distribution of published crystal densities in the curated HE-related dataset.

Featurization methods

The assumption that molecular structure can map to quantitative molecular compound properties requires that molecules be encoded with features that are accessible to ML models. To transform molecules into valid inputs for prediction models, various featurizations have been developed that attempt to appropriately summarize relevant characteristics while encoding that information in model-specific, small, or fixed dimensional spaces.[27] For instance, the SMILES representation[21] was developed to encode the structure of a chemical species into short ASCII strings, making it suitable for text-based models. Molecular fingerprints, such as ECFP[28] and E3FP,[29] are another commonly used approach that represent molecules with fixed-length bit vectors in which the vector components correspond to the presence or absence of some feature. While the features that result from constructing these fingerprints are not easily interpretable, the fingerprints are suitable for most ML approaches that require vectorized inputs. Additionally, this approach allows molecules to be easily compared despite having different numbers and types of atoms and bonds because all molecules' vector representations are the same length and molecules with similar chemistries tend to map to vectors having many overlapping bit elements. Briefly, these methods apply an iterative process in which a hash function is used to aggregate information about the neighborhood of an atom as defined by its graph representation. Initially, the neighborhood consists of the nearest neighbors of an atom, and at each iteration, the neighborhood expands to include the neighbors of the neighbors considered so far. E3FP extends this approach by leveraging 3D information about a molecule to define the neighborhood of an atom,[29] and we consider E3FP as a means of molecular representation in one of our models.

In addition to E3FP fingerprints, we also evaluated the 2D molecular descriptors that are available in RDKit[30] as a means of featurizing molecules for input into our models. RDKit is a widely-used open source toolkit for cheminformatics that provides functionalities for working with molecules and calculating a variety of molecular descriptors from input molecules. RDKit's earliest adopters were in the pharmaceutical and drug discovery communities, and this history is reflected in many of the molecular featurizations included in RDKit, which are pharmacologically relevant (i.e., Lipinski's rule of five[31]). In general, the features included in RDKit encompass different chemical and mathematical aspects of molecules that are potentially relevant to estimating molecular compound characteristics. These features include basic information, such as the types and numbers of atoms and other chemical entities, as well as higher-level features, such as graph theory-based calculations that synthesize information about groups of bonds and surface area-related measurements that take into account, for example, partial charges and other physical properties. Details about these descriptors may be found in their respective publications.[30,32–40] We note that while 3D-type descriptors are also available in RDKit, our results (further discussed in the Results and Discussion section) demonstrate that the 2D molecular descriptors alone are sufficient in making reasonably accurate density predictions.

To obtain the feature representations of our 10k dataset of molecules, we use deepchem[41] release 2.3.0 with RDKit[30] release 2019.09.1. Any package options and function parameters for accessing the molecules and features with these packages have been left at their default values. Starting with 111 features, we preprocessed the data by removing any perfectly correlated features and any constant variables, leaving 98 features for our models to learn from for density estimation. In addition, we log-transformed the "information on polynomial coefficients" or Ipc feature.[37] Ipc is an example of one of the hand-engineered molecular descriptors in RDKit's standard list of 2D molecular descriptors; first proposed in 1977, it provides a measure of the information content of a molecule's graph representation based on its characteristic polynomial and number of possible matchings, which spans several orders of magnitude. A complete list of the features and additional exploratory details may be found in the Supporting Information.

In general, the RDKit descriptors represent manually derived features that incorporate chemical domain knowledge to explicate additional molecular properties from a more basic set of features. While this approach is suitable for traditional ML approaches that require a predetermined set of engineered features, these features may not necessarily be important to predicting a target property of interest. In contrast, neural network-based methods are able to automatically derive hidden but relevant feature representations that are conducive to highly accurate predictions as part of the model-fitting procedure. For instance, message passing neural networks (described in the following section) are able to learn an internal representation of a molecule using only node- and bond-level information. The potentially high accuracy, low-level feature space requirement, and lack of an explicit featurization step make these approaches appealing for property prediction. Thus, we also consider using such a more generalized neural network-based model for crystalline density prediction. With this graph-based model, we utilize RDKit's atom and bond descriptors to provide node- and bond-level information; this information, along with the graph structure itself, together represent the featurization of a molecule.

Methods

While machine learning approaches hold great promise for predicting the properties of still unrealized molecular materials, a consistent challenge remains as to what models and molecular featurization methods—fixed or learned representations—will yield the best results. There are two primary reasons for this ambiguity. First, not all featurizations are compatible with all models and vice versa, which makes it challenging to decouple the effects of featurizations and model methods on the overall model performance. Second, the performance of models varies greatly depending on the problem—specifically the dataset and parameters being predicted. Thus, in creating a model to predict bulk crystalline properties of molecular materials from their chemical structures alone, we began by evaluating a number of possible molecular featurization methods and models, as outlined in Figure 3.

Among the potential methods we examine, we focus on several regression-based methods, which are compatible with the molecular-level featurization methods of RDKit and the E3FP fingerprints. Because of the large number of molecular features in the RDKit dataset and possible complex relationships between features and targeted property (i.e., crystalline density), we consider approaches that can discover potential nonlinear dependencies between the features and density values while handling the high-dimensionality and correlations of the features. For example, support vector machines[42] are supervised learning models that can be adapted for regression.[43] With an appropriate kernel function, these models can scale to high dimensional data, handle nonlinear relationships, and use the similarities to a subset of the training samples of a model to make accurate predictions.

One prominent regression-based method that addresses potential modeling issues such as high-dimensionality and correlations of the features and outlier target values is random forests (RF) regression,[44] an ensemble learning method that has been used successfully in many prediction tasks. RF is a supervised learning algorithm that uses feature and sample bagging[45] to learn ensembles of decision trees for classification and regression tasks. By using bagging to learn from subsets of features and samples in a dataset, RF produces decision trees that are robust to high-dimensionality and overfitting, can handle outliers and non-linearities without data scaling and transformations. Additionally, by analyzing changes in model performance when different features are used at the various branching points in decision trees, RF models can also provide a measure of variables' relative importance for achieving good model performance.[44,45] Thus, in the context of our current problem, an RF model can provide scientific insights by elucidating what molecular features are most important for predicting and tuning crystalline density.

Partial least squares regression (PLSR) is another popular statistical method that has been used to model systems and solve regression problems.[46,47] In contrast to RF, PLSR is a dimensionality reduction that maximizes the covariance between the sets of latent variables of the predictor and response spaces of a dataset and removes latent sources of variance with little predictive value. As a result, PLSR can produce parsimonious models that consider the predictor-response variation while reducing the number of predictors that are used, accounting for multicollinearity among the predictors, and producing variable importance in projection (VIP) scores[47,48] that summarize the importance of the predictors in the latent variable construction.

In recent years, neural network-based methods have also seen increased usage and success in chemical prediction problems.[12,13,49–55] One framework, the Message Passing

Neural Network (MPNN), encompasses several neural network-based approaches that rely on message passing algorithms for graph-structured data to make target property predictions.[49] In general, these approaches are able to derive their own features from molecular graphs and supplemental information about the graph's nodes and edges through a message passing algorithm and aggregation framework to predict molecular properties, thereby eliminating the need for complicated feature engineering while achieving high prediction accuracy. However, neural network-based methods tend to be computationally expensive to fit, and the internal representations that are learned by these methods generally do not admit interpretable models. Here, we consider the MPNN variant and implementation by Yang et al. in which messages are directed and associated with edges or bonds instead of undirected and associated with vertices or atoms.[50] By using a directed graph, this variant prevents messages from being instantly passed back to a source node from its original target in order to reduce noise in the resulting models.

In this work, we consider a number of different fixed or learned molecular featurizations in combination with the above-described modeling methods to predict the crystalline densities of energetic materials from their chemical structure alone. Specifically, we developed and evaluated: 1) a support vector regression (SVR) model[42,43] using E3FP fingerprints, a common fixed molecular representation; 2) RF- and PLSR-based models with RDKit molecular-level features, also fixed molecular representations; and 3) an MPNN-based model, which utilizes RDKit atom- and bond-level features to describe network nodes (atoms) and edges (bonds) but yields a learned overall molecular representation. Before fitting the RF and PLSR-based models, we preprocess and filter the densities as described in the Dataset section. We also normalize the filtered features to have a mean of 0 and a standard deviation of 1. In the case of PLSR, we also apply a power transformation[56] to the features. For the MPNN model, the implementation by Yang et al. uses a set of basic features for each atom and bond that is appropriate to the message passing framework, and we use this implementation as-is, including any one-hot encoding and other preprocessing steps that are applied to the features; a summary of these features may be found in Ref. 50 and in the Supporting Information.

To fairly assess the models, we apply stratified k-fold cross-validation. In particular, we use 5 stratified folds with bins defined by boundaries between 1.0 and 2.0 at increments of 0.05 to handle the density imbalance and ensure that each fold is representative of the distribution of densities; a list of the molecules and their groupings are provided in the Supporting Information. For each method, we summarize its overall performance by computing the averages of the R^2 score and root mean squared error (RMSE) across the stratified folds.

As an alternative to stratified splitting, scaffold splitting may also be used to evaluate a method's ability to generalize to structurally different molecules;[57] results based on this approach are provided in the Supporting Information.

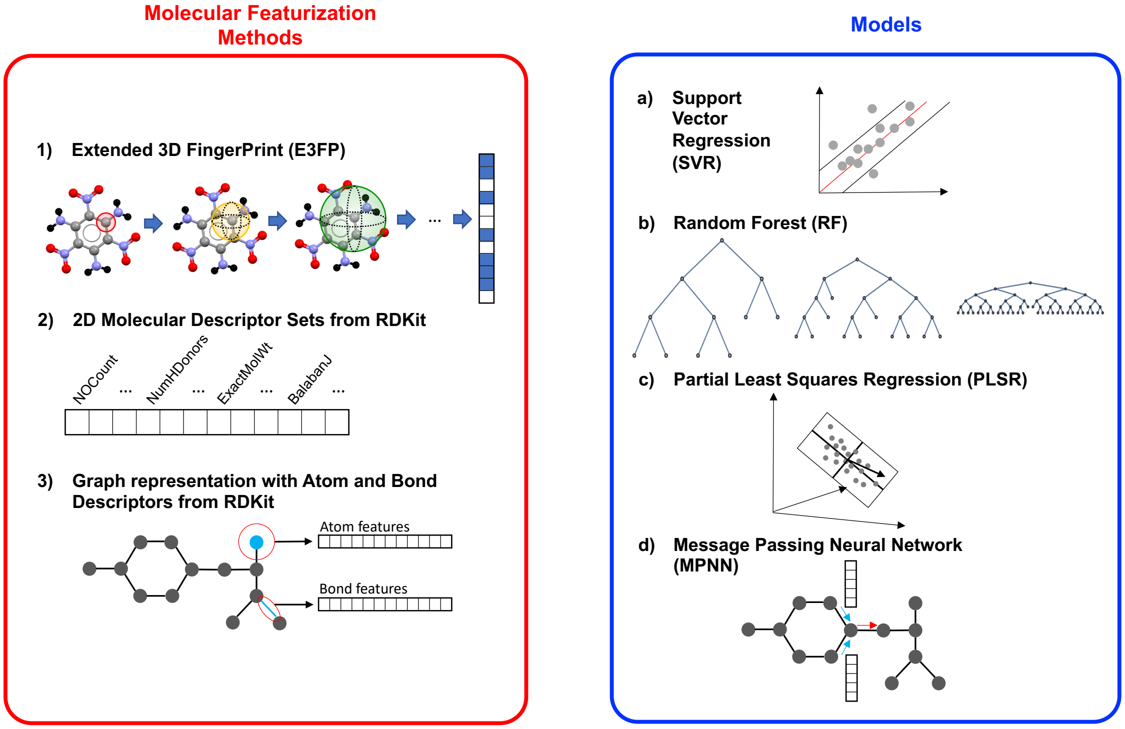

Molecular Featurization Methods
1) Extended 3D FingerPrint (E3FP)
...
2) 2D Molecular Descriptor Sets from RDKit
NOCount
...
NumHDonors
...
ExactMolWt
...
BalabanJ
...
3) Graph representation with Atom and Bond Descriptors from RDKit
Atom features
Bond features
Models
a) Support Vector Regression (SVR)
b) Random Forest (RF)
c) Partial Least Squares Regression (PLSR)
d) Message Passing Neural Network (MPNN)

Figure 3: Overview of density regression models.

Results and Discussion

HE-related density prediction

Using the HE-related dataset and models described in Figures 2 and 3 and in the Materials and Methods section, we first evaluate the goodness of fit for the various combinations of featurizations and methods to predict the densities of the HE-related molecules in CSD. In Figure 4, the predicted densities are plotted against the true densities for each feature-method combination. Generally, the predicted values tend to be close to their corresponding true values for all of the models. This result demonstrates that ML models can indeed be trained to predict crystalline properties of molecules from their chemical structure alone—remarkably even in the absence of any crystal structure information.

Table 1 shows the R^2 and root mean square error (RMSE) values for the RDKit molecular features each of the feature-method combinations that we considered. Compared with the other methods, the E3FP/SVR combination performs poorly with an R^2 value of 0.683, suggesting that the E3FP fingerprint may not capture information that is as relevant for density predictions as the other three models do. The RF- and PLSR-based models both use RDKit's standard set of 2D molecular features that are pre-computed before training the models—a *fixed* molecular representation—and yield R^2 values of 0.878 and 0.900, respectively. That these two models have comparable and good performance despite using different regression methods demonstrates that the featurization method they utilize—RDKit's 2D molecular descriptors—adequately captures the necessary molecular information for predicting density. Interestingly, the MPNN-based model, which utilizes a *learned* molecular representation, has an even slightly better performance, yielding an R^2 value of 0.914. While the performance difference between these three models appears small, that the MPNN utilizing a learned molecular representation performs so well is encouraging because it suggests that handcrafted molecular features like those included in RDKit may not be necessary for predicting crystalline properties. As one may appreciate by perusing RDKit's references, such handcrafted features are often developed by experts over decades and include domain knowledge tailored to specific topics, such as drug design. Though neural network-based methods like MPNNs have their own drawbacks (i.e., they are computationally expensive and the model complexity hinders human interpretability), such approaches may be particularly adept when appropriate handcrafted features have not yet been developed or identified.

To further discriminate between our best models, we examined their performance more closely in the context of our problem of interest: predicting the density of HE-related molecules. As discussed in the introduction, the HE community typically considers density predictions with errors less than 0.03 g/cc as "excellent" and those with errors within 0.03 and 0.05 g/cc as "informative;" prediction with errors larger than 0.05 g/cc are "barely usable" or even "deceptive".[16,17] Figure 5a shows the distribution of absolute errors in this context for the RF-, PLSR-, and MPNN-based models. Notably, the PLSR-based model has 56% of predictions as "excellent" with errors less than 0.03 g/cc and 76% as "informative" with errors less than 0.05 g/cc, and the MPNN model performs even better, yielding 61% of predictions as "excellent" with errors less than 0.03 g/cc and 83% with errors less than 0.05 g/cc. For context, the DFT-based density prediction method with electrostatic correction factors introduced by Politzer et al., which we discussed in the introduction, produced only 50% of its density predictions as "excellent."[18] We find it rather remarkable that a machine learned model using a learned molecular representation and without explicitly knowing any information about the molecular volume or intermolecular interactions can perform so well at predicting density.

As discussed earlier, HE compounds tend to have high densities (i.e., 1.6 g/cc or higher), but as illustrated in Figure 2 , this high-density regime has significantly fewer data points from which models can learn. Thus, in evaluating our models, we were also interested in understanding the distributions of models' errors as a function of the molecules' true, experimentally validated densities, paying particular attention to the error distribution in the high-density, low-data regime of interest. Figure 5b illustrates this concept and shows the median error within bins defined by the true densities, with a bin width of 0.01. Interestingly, we see that while the median error of the RF- and PLSR-based models increases at densities greater than 1.5 g/cc, the MPNN has a relatively constant error that is less than 0.05 g/cc across most of the density regime, including the data-sparse, high-density regime. Of particular note is TATB, an important test HE molecule with a density of 1.93 g/cc. Existing density prediction methods have thus far been challenged in accurately predicting the density of this molecule, producing errors in excess of 0.1 g/cc; here, the MPNN model performs exceptionally well, yielding a prediction of 1.95 g/cc and error of 0.02 g/cc, while our remaining models produce errors greater than 0.09 g/c for this molecule. Though it remains to be understood precisely why the MPNN is better able to generalize and predict in this high density regime, this result is very encouraging for neural network-based methods like the MPNN and suggests that the learned molecular representations may capture more subtleties in molecules' chemistries than the handcrafted RDKit molecular features, as discussed further below.

Table 1: R^2 and RMSE values of different featurization and method combinations for HE-related density prediction.

Feature	Input information	Feature processing	Method	R^2	RMSE
E3FP	atomic, 3D positions	Precomputed	SVR	0.683	0.085
RDKit (molecular)	physicochemical/mathematical	Precomputed	RF	0.878	0.053
RDKit (molecular)	physicochemical/mathematical	Precomputed	PLSR	0.900	0.048
RDKit (atom/bond)	atomic/bond, molecule graph	Learned	MPNN	0.914	0.044

Feature importance

In practice, one of the primary goals in fitting the density regression models is using them to accurately predict the densities for novel molecules designed by domain experts, but understanding which features are relevant and actively contribute to the predicted densities is also valuable to help build fundamental scientific understanding, particular as it applies to designing novel molecules. While the MPNN-based model has better predictive performance than the RF- and PLSR-based models, its algorith-

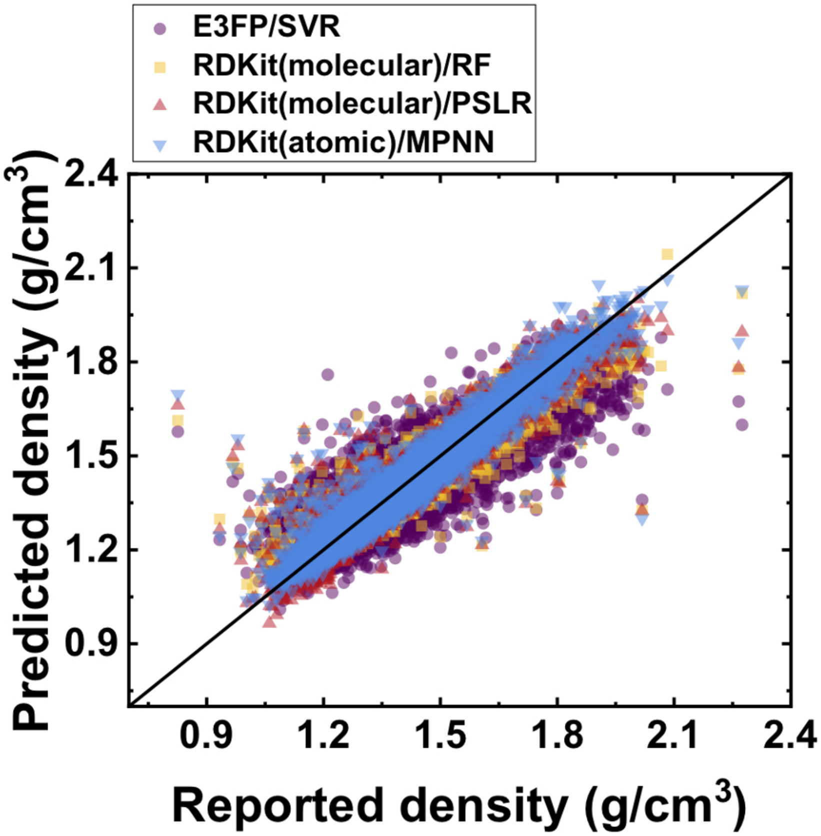


Figure 4: Predicted vs. true densities. In general, the approaches using RDKit with RF, PLSR and MPNN perform better than the baseline E3FP/SVR approach and produce density predictions that are close to the true density values. However, errors tend to be large for extreme values of the true density.

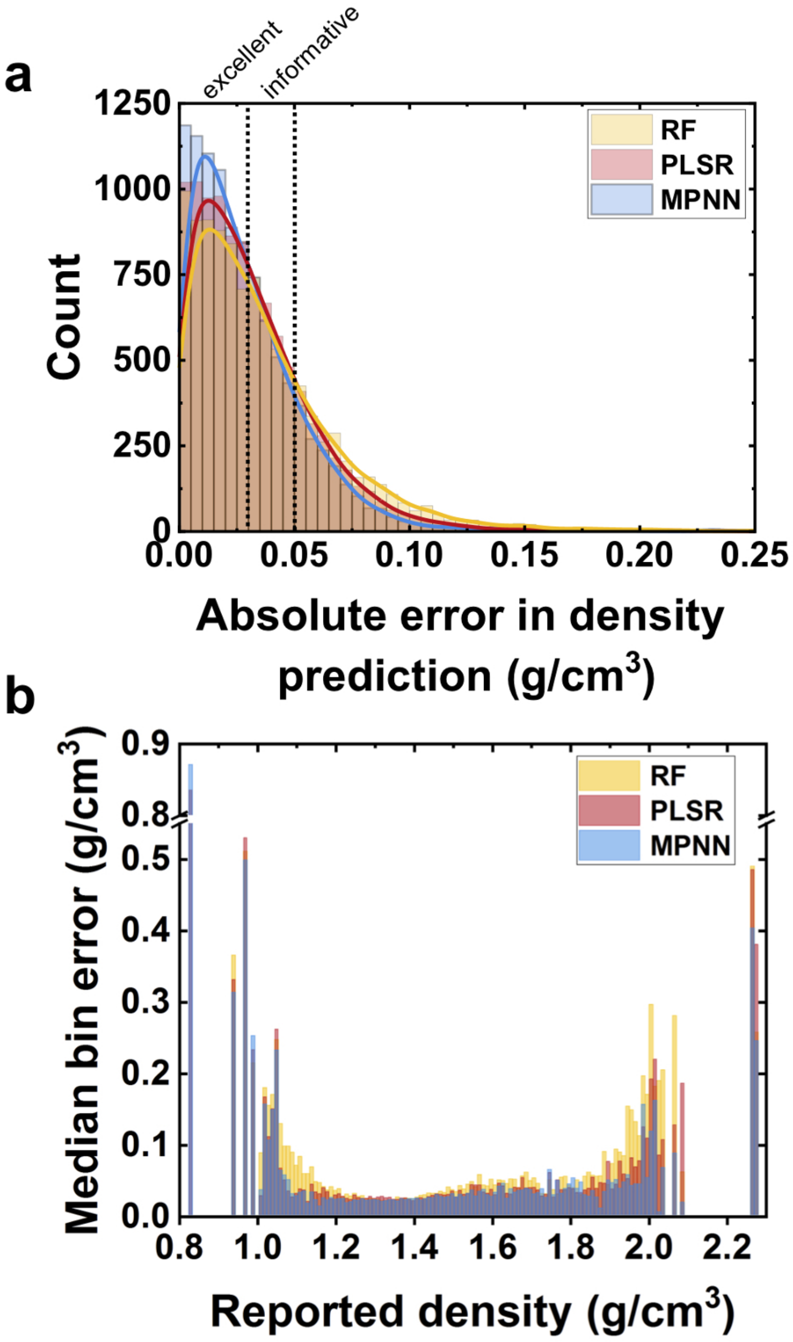


Figure 5: (a) Distributions of predicted density errors and (b) median predicted density error within bins defined by the true densities of the HE-related molecules at 0.01g/cc intervals for RF, PLSR, and MPNN. For clarity, the few errors above 0.25g/cc are not shown. The distributions are concentrated around small values, with the MPNN-based model tending to make the smallest errors and the RF-based model making the largest errors. However, all three approaches contain large outliers and tend to make errors that are larger for values at the tails of the density distribution than for the intermediate density values.

mic structure makes it difficult to interpret in the context of higher level features a scientist could control, i.e. important functional groups, as the primary focus has been on specific node importance.[58] In contrast, the RF- and PLSR-based methods have well-known heuristics and procedures for determining measures of variable importance and natively accept highly interpretable and actionable features. To determine the most important variables in our models for density prediction, we consider Gini and permutation importances[44,45] for our RF model and VIP scores[47,48] for our PLSR model.

Table 2 shows the 20 RDKit molecular features with the highest Gini, permutation, and VIP scores for the RF- and PLSR-based models, which indicates their importance to the accurately predicting density of HE-like molecules; the rankings of the remaining features may be found in the Supporting Information. While the precise rank-order of feature importance varies between the three approaches, many features are common among the three list, and in fact, each approach has the same set of top 6 predictors: NO Count, VSA_EState8, SlogP_VSA5, TPSA, SMR_VSA5, and MolLogP. We explain these features individually below.

- **NO count**: The NO count provides a count of the number of nitrogen and oxygen atoms present on the molecule. The inclusion of this feature in RDKit is reflective of its pharmaceutical origins since nitrogen and oxygen atoms are commonly present in pharmaceutics, and in fact, one of Lipinkski's rule of five[31] to evaluate the likelihood of a given molecule being a pharmaceutic states that no more than 10 nitrogen and oxygen atoms, which act as hydrogen bond acceptors, can be present. Serendipitously, nitrogen and oxygen atoms are also commonly occurring in energetic molecules, most often as nitro (-NO_2) or amine (-NH_2) functional groups. Hydrogen bonding is an important type of intermolecular interaction between hydrogen atoms and highly electronegative atoms, like oxygen and nitrogen, that can affect how molecules crystallize, which in turn affects their density.
- **VSA_EState8**: The E-state or "electrotopological state" index of an atom sum-

marizes the differences in electronegativity between the atom and the other atoms in a molecule, scaled by their physical distances, and this value can be interpreted as the level of accessibility of an atom to interactions.[59] In RDKit, the VSA_EState values accumulates the total E-state values over atoms with a particular van der Waals surface area. In particular, the VSA_EState8 value of a molecule is equal to the sum of the E-state values of atoms in the molecule with a van der Waals surface area between 6.45 and 7.00.

- **MolLogP**: The $\log P$ value or "octanol-water partition coefficient" provides a measure of the lipophilicity.[39] This parameter is an important criterion used by medicinal chemists to screen potential pharmaceutical candidates as it provides an indication of how likely the compound is to reach the intended target tissue in the human body. In RDKit, the MolLogP value is based on the method by Wildman and Crippen[39] and calculated as the sum of the logP contributions over the atoms of the molecule, the values of which have been estimated a priori by least-squares fitting.

- **SlogP_VSA5**: The SLogP_VSA values also summarize information related to the octanol-water partition coefficient. In this case, the SLogP_VSA considers the amount of a molecule's surface area that can be attributed to atoms with certain logP values. In particular, SLogP_VSA5 is equal to the sum of the van der Wals surface areas of atoms with estimated $\log P$ contribution values (based on the method by Wildman and Crippen[39]) that are between 0.10 and 0.15.

- **SMR_VSA5**: MR or "molar refractivity" is a descriptor that reflects the polarizability of a molecule. Like the $\log P$ value, the MR value can be estimated as the sum of the individual MR contributions that can be attributed to each atom, each of which has been estimated beforehand.[39] The SMR_VSA values then summarize the amount of a molecule's surface area that can be attributed to atoms with certain MR values. In particular, SMR_VSA5 is equal to the sum of the van der Wals surface areas of atoms with estimated MR contribution values (based on the method by Wildman and Crippen[39]) that are between 2.45 and 2.75.

- **TPSA**: The TPSA,[60] or topological polar surface area, value is an estimate of the molecular polar surface area, a property that has been shown to correlate with drug transport properties. Analogous to the MR and $\log P$ calculations by Wildman and Crippen,[39] Ertl et al. calculate the TPSA value by taking sum of the polar surface areas over the fragments of a molecule, each of which are estimated a priori.[60]

The fact that the handcrafted molecular features designed to capture pertinent details for pharmaceutics are relevant and important in predicting the crystalline density of HE-like molecules is surprising; however, this observation suggests that these features capture more fundamental attributes about molecules that are more broadly applicable. Indeed, looking more holistically at the molecular features included in RDKit, we classified them according to categories that describe the type of information they capture (see Supporting Information), and on the basis of the features' relative importance in predicting density according to their category classifications, features that pertain to a combination of electronic and topological information appear to be most important. We can rationalize this result in the context of DFT-based methods of density predictions, which rely entirely on accurately approximating a molecule's volume from an electron density isosurface; the hand-crafted electronic-topological features are conceptually similar in that they also capture information about the molecules' shape and electron distribution.

Although MPNNs are not as readily interpretable for scientific application, comparative studies between the MPNN and RF models can help shed insights as to why the MPNN achieves better performance for density regression. During the readout phase of the MPNN model, each updated node feature is aggregated, and this aggregation is transformed into a fixed length vector. The final density prediction is calculated through a weighted sum of the readout vector, similar to classical linear regression. Consequently, the weights connecting the final layer to the output node can be interpreted as coefficients of a linear model and provide feature importance to the newly

Table 2: Top 20 predictors based on random forest Gini and permutation importance and partial least squares regression VIP scores.

rank	RF (Gini)	RF (permutation)	PLSR (VIP)
1	VSA_EState8	TPSA	VSA_EState8
2	SlogP_VSA5	VSA_EState8	SlogP_VSA5
3	TPSA	SMR_VSA5	TPSA
4	SMR_VSA5	SlogP_VSA5	NO Count
5	MolLogP	NO Count	SMR_VSA5
6	NO Count	MolLogP	MolLogP
7	PEOE_VSA6	EState_VSA10	PEOE_VSA7
8	EState_VSA7	FractionCSP3	Chi2n
9	EState_VSA10	PEOE_VSA6	RingCount
10	PEOE_VSA7	EState_VSA7	EState_VSA10
11	Chi2n	VSA_EState6	PEOE_VSA6
12	VSA_EState3	Chi2n	NumHAcceptors
13	FractionCSP3	Kappa2	Chi1n
14	MaxPartialCharge	Kappa3	Chi3n
15	VSA_EState6	VSA_EState3	Chi4n
16	Chi1n	Chi0n	BalabanJ
17	Chi3n	NumHAcceptors	EState_VSA7
18	NumHAcceptors	HallKierAlpha	VSA_EState3
19	Chi0n	Chi1n	Chi0n
20	MinEStateIndex	MolMR	MolMR

learned features. By analyzing the correlation between the top features identified by RF and MPNN, where the RF features are interpretable, we can start to understand what the MPNN is learning as well as determine if it has extracted new information missed by the RDKit feature engineering process.

In Figure 6, we plot the absolute values of the pairwise correlations between the top MPNN and RDKit features which have a 0.005 g/cc or higher impact on the density prediction, determined by final layer weights and permutation importance, respectively. There are two immediate takeaways we find from this experiment: 1) the newly learned features from the last layer of the MPNN model seems to correlate well with many of the hand-engineered RDKit features, and 2) the MPNN model has learned multiple new features that are not captured by the RDKit feature set.

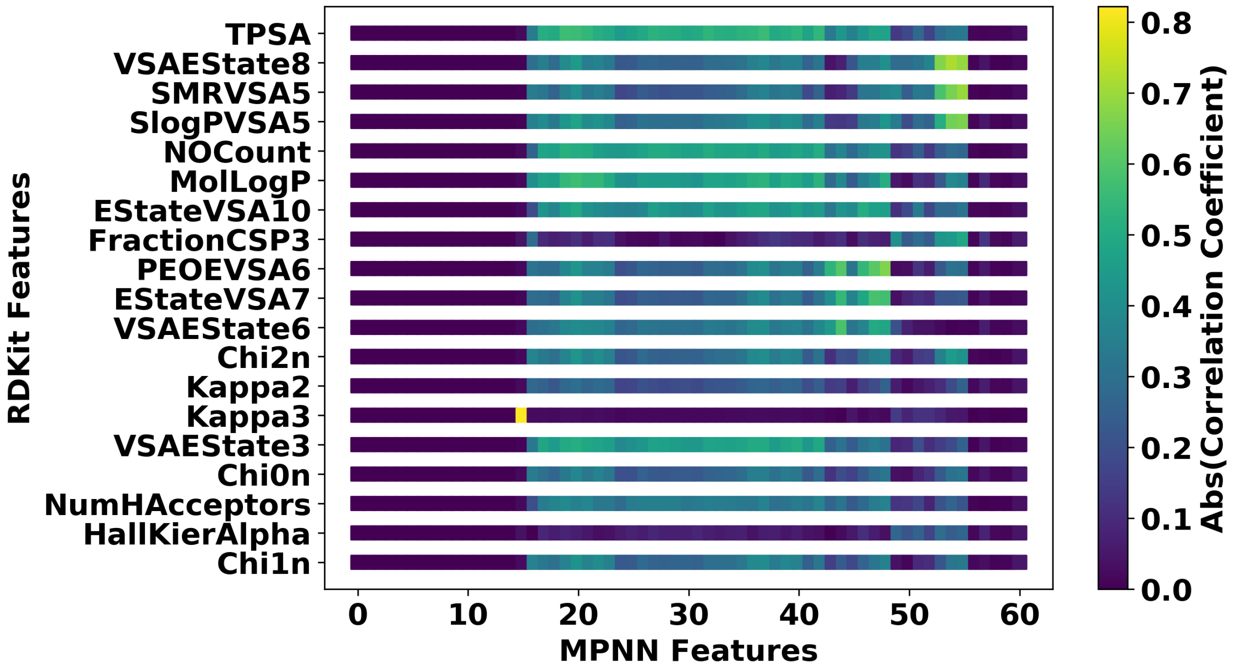


Figure 6: Absolute values of Pearson correlation coefficients between MPNN features and RDKit features where the RDKit features are sorted (in descending order) according to importance determined by RF. Only features with an impact of 0.005g/cc or higher on the final predicted density are considered. Yellow and green indicate a strong correlation (positive or negative) while purple indicates no correlation.

Although deciphering the exact salient information embedded within the MPNN's final layer remains challenging, our analysis makes clear that the MPNN's graph level features are capable of re-deriving many of the relevant hand-engineered properties when given enough data. Moreover, despite being independent of the RDKit feature ranking, the maximum absolute value of the correlation coefficients generally trend with this ranking, as seen by the prominence of yellow and lighter green indicative of high correlations near the top of Figure 6, underscoring the prominence of these features for density estimation. Together, these results indicate chemical awareness within the graph-based NN model, which is crucial to both understanding how to better design molecules and developing trust in applying deep learning to molecular data. In addition, we attribute the increase in MPNN performance, relative to both RF and PLSR, to the MPNN features that are not represented within the RDKit feature set. While we are unable to decipher what these new features represent in the context of chemical information, the isolation of these relevant new features offers a starting point for further analysis and interpretation.

Testing model performance on an out-of-distribution dataset

A key challenge to many ML models' practical utility is that their performance does not translate well when molecules are provided as inputs that are very different than the molecules and data used in model training. Thus, to further demonstrate the utility of our models, we chose to also test our models' performance on an out-of-distribution dataset. We chose as our test a small dataset of 109 chemically diverse energetic molecules originally assembled and reported by Huang and Massa.[61] Treating the Huang and Massa dataset as our test dataset, we removed from our 10k dataset molecules also present in the Huang and Massa dataset and re-trained our models. The Huang and Massa dataset represents an out-of-distribution test dataset for our models in two ways: 1) This dataset has much more chemical diversity than our 10k training dataset and includes non-traditional HE. For example, 21% of the test dataset consists of molecules having fluorine atoms, which are completely absent from our 10k training dataset, and a handful of molecules in the test dataset do not have a N-O bond of any kind, a common feature of most HE molecules and a selection requirement in creating our 10k training dataset. 2) As shown in Figure 7, the distributions of densities for our training and testing datasets are concentrated around different values; the Huang and Massa dataset is centered at high densities of approximately 1.9 g/cc, whereas the 10k dataset is centered at densities of approximately 1.35 g/cc. Also, we removed high-density datapoints from our training dataset, which already has few high-density examples, that were common to the test dataset, thereby furthering the differences between the training and test datasets.

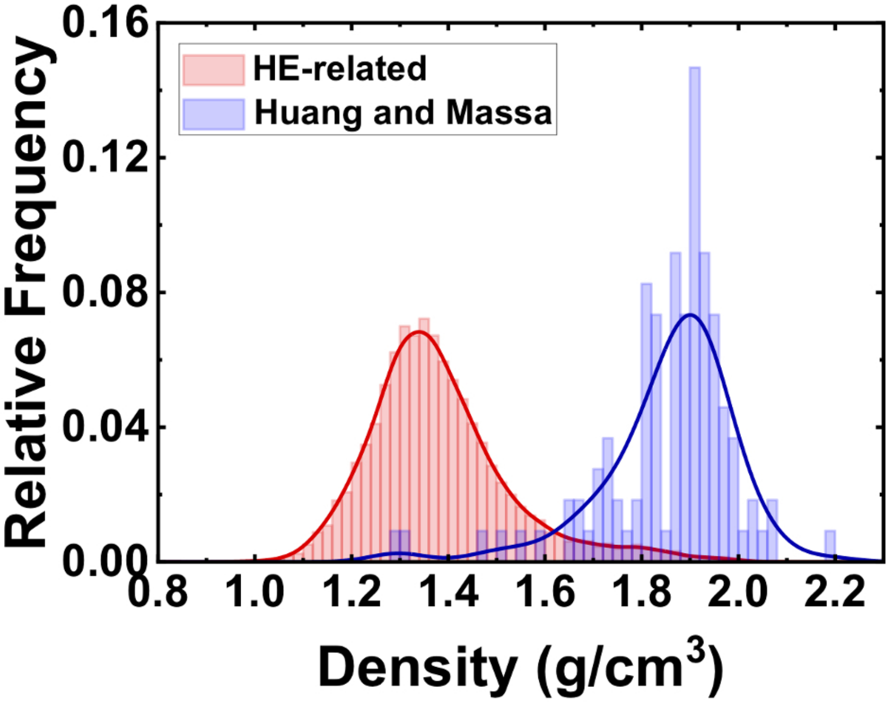


Figure 7: Distributions of the HE-related (red) and Huang and Massa (blue) crystalline density values used as training and test datasets, respectively. The test dataset values are concentrated at the high-density values where the training dataset is sparsely distributed and the trained models are expected to perform poorly (see Figure 5).

Figure 8 shows the predicted densities for Huang and Massa dataset plotted against the true densities using the same feature-method combinations considered previously in the HE-related density prediction analysis. Table 3 reports the corresponding R^2 and RMSE values. As expected given the challenges discussed above with this test, the performance of our models on this high-density, chemically diverse dataset is lower (R^2 decreased; RMSE increased) than the performance we reported earlier on the stratified k-fold splits of our 10k HE-related dataset. In particular, the E3FP/SVR combination—previously our worst performing model—highly underpredicts densities, producing errors that are essentially unreliable. Interestingly, while the RF- and PLSR-based models, both of which utilized fixed molecular representations from RDKit, had comparable performance when predicting on the 10k HE-related dataset, their performance on this out-of-distribution dataset diverges. The RF-based approach also largely underpredicts densities in the out-of-distribution dataset, yielding an R^2 of 0.206, while the PLSR-based model yields an R^2 of 0.708. This result highlights the potential impact and importance of model selection, particularly when the data that a model is expected to predict on is uncharacteristic of the distribution of data that the model was trained on; in this case, two models with nearly equivalent performance when testing on in-distribution data have contrasting performance when testing on out-of-distribution data. Finally, the MPNN-based model, previously our best performing model, performs slightly worse than the PLSR-based approach, yielding an R^2 of 0.624. Though both the PLSR- and MPNN-based models tend to underpredict densities for this test dataset, their performance on the out-of-distribution dataset is still remarkably reasonable, especially in light of the distributional disparities observed in Figure 7. This demonstrates that they are learning patterns from low-to-intermediate density molecules that are still useful to predicting high density HE, but these results underscore the need for improved methodologies and datasets for more thorough practical applications.

Table 3: R^2 and MSE values scores of different featurization and method combinations using the HE-related dataset for training and Huang and Massa dataset for testing.

Featurization	Method	R^2	RMSE
E3FP	SVR	-2.402	0.254
RDKit (molecular)	RF	0.206	0.123
RDKit (molecular)	PLSR	0.708	0.074
RDKit (atom/bond)	MPNN	0.624	0.084
RDKit (atom/bond ordinal encoding)	MPNN	0.798	0.062

Since the distributions of densities of the HE-like dataset and the Huang and Massa

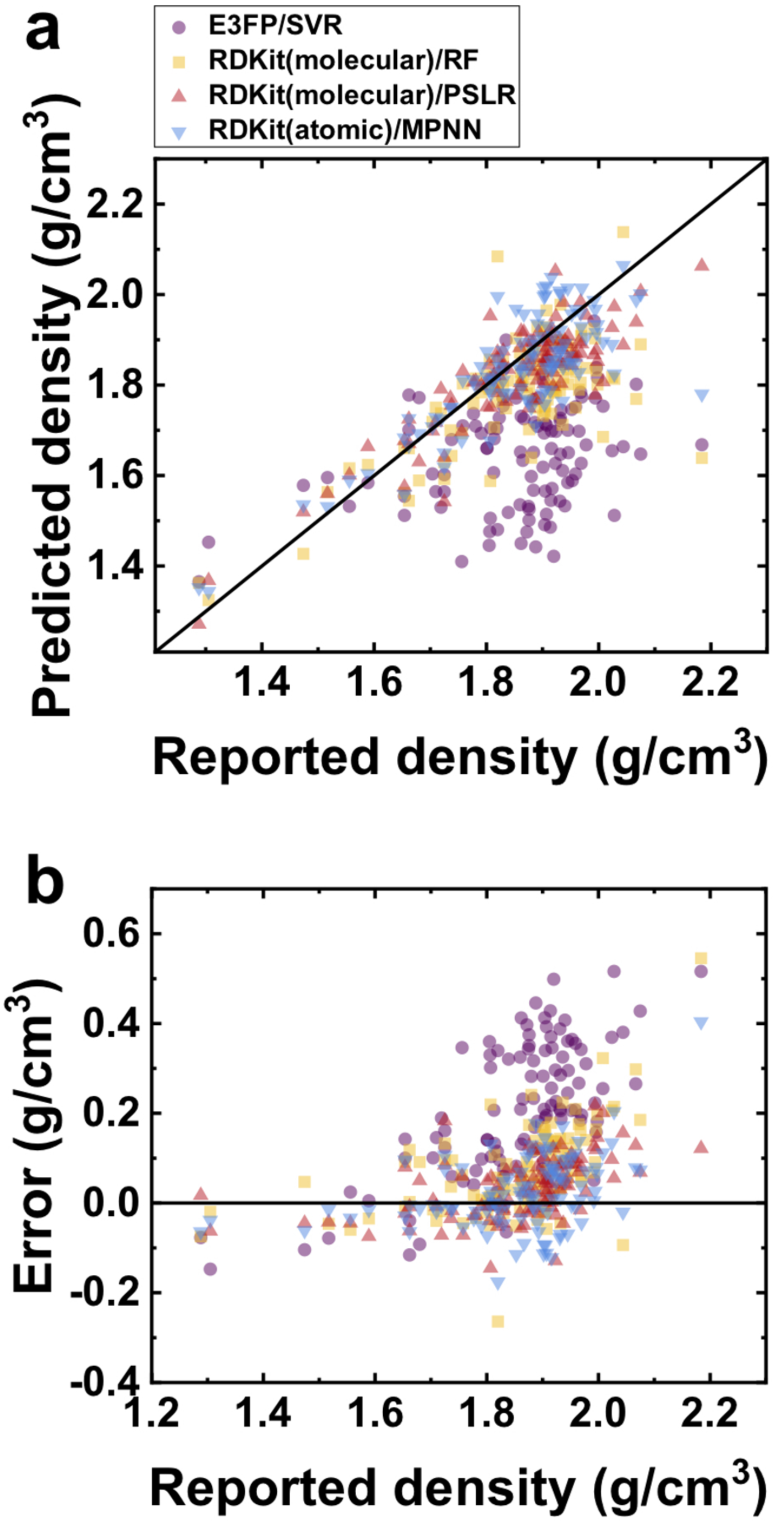


Figure 8: (a) Predicted and (b) error (true – predicted) vs. true packing density values using the HE-related dataset for training and Huang and Massa dataset for testing.

dataset are highly pronounced, the large differences in scores between the two datasets may be a result of overfitting to the relatively smaller densities of the HE-like dataset. To assess overfitting when predicting the high density values, we now recompute the R^2 and RMSE values in Tables 1 and 3 by limiting their computations to true density values that are above a certain threshold. Here, we limit the test densities to values that are above 1.589 g/cc to retain 95% (104 molecules) of the Huang and Massa dataset. The R^2 and RMSE values for the trimmed test datasets are shown in Table 4. We again see that R^2 values are smaller and the RMSE values are larger on the test dataset. However, the changes to these values are more severe with the SVR and RF-based approaches, so these two approaches may be highly overfitting to the HE-related dataset. The R^2 and RMSE values for MPNN also change by large amounts, but not to the same extent. Of the four approaches, the RDKit/PLSR combination appears to be the most robust, with the smallest changes in RMSE at 0.005 g/cc and in R^2 at 0.160, indicating its relatively better generalizability in the high-density range.

Table 4: Trimmed R^2 and MSE values scores using test densities above 1.589 g/cc for different featurization and method combinations in the cross-validation analysis on the HE-related densities (HE) and when using the Huang and Massa densities (HM) for testing.

Featurization	Method	R^2 (HE)	R^2 (HM)	RMSE (HE)	RMSE (HM)
E3FP	SVR	-0.823	-5.738	0.150	0.259
RDKit (molecular)	RF	0.387	-0.551	0.087	0.125
RDKit (molecular)	PLSR	0.595	0.435	0.070	0.075
RDKit (atom/bond)	MPNN	0.660	0.27	0.064	0.086

To remedy MPNN's poor generalizability, as seen in the large decrease in RMSE between the HE-related and Huang and Massa datasets, we also consider an ordinal encoding on the originally one-hot encoded feature vectors (see Supporting Information). This different encoding scheme drastically reduces the dimensionality of the features, a common technique to reduce overfitting, and significantly improves MPNN's performance in the high-density regime. As a result of this modification, MPNN outperforms PLSR based on the R^2 and RMSE values (0.798 and 0.062, respectively) and appears to be more in line with projections based on the training dataset, indicating an increase in generalization.

Conclusions

In this work, we demonstrate that machine learning models can be created to predict a bulk crystalline property—specifically density –of energetic molecular materials using information only from the chemical structure of the molecule and without knowing any information about the crystal structure. Our best models outperform current state-of-the-art methods, including more computationally expensive DFT-based methods, and yield good prediction even on high-density molecules known to be challenging, like TATB. We find that both models utilizing handcrafted features and learned molecular representations perform well, with the learned molecular representation model (MPNN-based) even slightly outperforming those utilizing handcrafted feature methods. This result should be encouraging for the chemical informatics community as a whole, as it suggests that current machine learning methods capture at least as much, if not more, information than what we can master via subject matter expert-informed feature engineering.

The ability to predict crystalline properties from chemical structure is particularly powerful given that computationally predicting molecular crystal structures is still challenging. In this sense, the ability to predict the properties of crystalline molecular materials may also aide such computational efforts to directly predict crystal structures since accurate property prediction, like density, can significantly reduce the potential crystal structure variable search space. However, more immediately, such models may be utilized by domain experts to quickly screen HE candidates, and longer term, may be used in combination with generative models for expedited and computer-aided design of new molecular materials.

Acknowledgement

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and supported by LDRD 19-SI-001. Release number: LLNL-JRNL-816318.

References

(1) Price, S. L. Control and prediction of the organic solid state: a challenge to theory and experiment. *Proc. R. Soc. A* **2018**, *474*, 20180351.

(2) Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* **2018**, *4*, 268–276.

(3) Wu, Z.; Ramsundar, B.; Feinberg, E.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **2018**, *9*, 513–530.

(4) Lu, C.; Liu, Q.; Sun, Q.; Hsieh, C.-Y.; Zhang, S.; Shi, L.; Lee, C.-K. Deep learning for optoelectronic properties of organic semiconductors. *J. Phys. Chem. C* **2020**, *124*, 7048–7060.

(5) Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P. Graph networks as a universal machine learning framework for molecules and crystals. *Chem. Mater.* **2019**, *31*, 3564–3572.

(6) Lusci, A.; Pollastri, G.; Baldi, P. Deep architectures and deep Learning in chemoinformatics: the prediction of aqueous solubility for drug-like molecules. *J. Chem. Inf. Model.* **2013**, *53*, 1563–1575.

(7) Yang, J.; De, S.; Campbell, J. E.; Li, S.; Ceriotti, M.; Day, G. M. Large-scale computational screening of molecular organic semiconductors using crystal structure prediction. *Chem. Mater.* **2018**, *30*, 4361–4371.

(8) Elton, D. C.; Boukouvalas, Z.; Butrico, M. S.; Fuge, M. D.; Chung, P. W. Applying machine learning techniques to predict the properties of energetic materials. *Sci. Rep.* **2018**, *8*, 9059.

(9) Afzal, M. A. F.; Sonpal, A.; Haghighatlari, M.; Schultz, A. J.; Hachmann, J. A deep neural network model for packing density predictions and its application in the study of 1.5 million organic molecules. *Chem. Sci.* **2019**, *10*, 8374–8383.

(10) Bier, I.; Marom, N. Machine Learned Model for Solid Form Volume Estimation Based on Packing-Accessible Surface and Molecular Topological Fragments. *J. Phys. Chem. A* **2020**, *124*, 10330–10345.

(11) Barnes, B. C.; Elton, D. C.; Boukouvalas, Z.; Taylor, D. E.; Mattson, W. D.; Fuge, M. D.; Chung, P. W. Machine Learning of Energetic Material Properties. **2018**,

(12) Barnes, B. C. Deep learning for energetic material detonation performance. *AIP Conf. Proc.* **2020**, *2272*, 070002.

(13) Casey, A. D.; Son, S. F.; Bilionis, I.; Barnes, B. C. Prediction of Energetic Material Properties from Electronic Structure Using 3D Convolutional Neural Networks. *J. Chem. Inf. Model.* **2020**, *60*, 4457–4473.

(14) Rice, B. M.; Hare, J. J.; Byrd, E. F. C. Accurate predictions of crystal densities using quantum mechanical molecular volumes. *J. Phys. Chem. A* **2007**, *111*, 10874–10879.

(15) Qiu, L.; Xiao, H.; Gong, X.; Ju, X.; Zhu, W. Crystal density predictions for nitramines based on quantum chemistry. *J. Hazard. Mater.* **2007**, *141*, 280—288.

(16) Kim, C. K.; Cho, S. G.; Kim, C. K.; Park, H.-Y.; Zhang, H.; Lee, H. W. Prediction of densities for solid energetic molecules with molecular surface electrostatic potentials. *Journal of Computational Chemistry* **2008**, *29*, 1818–1824.

(17) Goh, E. M.; Cho, S. G.; Park, B. S. *J. Def. Tech. Res. 9*, 91.

(18) Politzer, P.; Martinez, J.; Murray, J. S.; Concha, M. C.; Toro-Labbé, A. An electrostatic interaction correction for improved crystal density prediction. *Molecular Physics* **2009**, *107*, 2095–2101.

(19) Cady, H. H.; Larson, A. C. The crystal structure of 1,3,5-triamino-2,4,6-trinitrobenzene. *Acta Crystallogr.* **1965**, *18*, 485–496.

(20) Rice, S. F.; Simpson, R. L. *The unusual stability of TATB (1,3,5-triamino-2,4,6-trinitrobenzene): A review of the scientific literature*; 1990.

(21) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31–36.

(22) Mathieu, D. Sensitivity of energetic materials: theoretical relationships to detonation performance and molecular structure. *Ind. Eng. Chem. Res.* **2017**, *56*, 8191–8201.

(23) Groom, C. R.; Bruno, I. J.; Lightfoot, M. P.; Ward, S. C. The Cambridge structural database. *Acta Crystallogr., Sect. B: Struct. Sci., Cryst. Eng. Mater.* **2016**, *72*, 171–179.

(24) Lee, E. H. A practical guide to pharmaceutical polymorph screening & selection. *Asian J. Pharm. Sci.* **2014**, *9*, 163–175.

(25) Zhang, W.; Zhang, J.; Deng, M.; Qi, X.; Nie, F.; Zhang, Q. A promising high-energy-density material. *Nat. Commun.* **2017**, *8*, 181.

(26) Zhang, C.; Jiao, F.; Li, H. Crystal engineering for creating low sensitivity and highly energetic materials. *Cryst. Growth Des.* **2018**, *18*, 5713–5726.

(27) Guha, R.; Willighagen, E. A survey of quantitative descriptions of molecular structure. *Curr. Trends Med. Chem.* **2012**, *12*, 1946–1956.

(28) Rogers, D.; Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742–754.

(29) Axen, S. D.; Huang, X.-P.; Cáceres, E. L.; Gendelev, L.; Roth, B. L.; Keiser, M. J. A simple representation of three-dimensional molecular structure. *J. Med. Chem.* **2017**, *60*, 7393–7409.

(30) Landrum, G. RDKit: open-source cheminformatics. http://www.rdkit.org, 2006.

(31) Lipinski, C. A. Lead- and drug-like compounds: the rule-of-five revolution. *Drug Discovery Today: Technol.* **2004**, *1*, 337–341.

(32) Todeschini, R.; Consonni, V. *Handbook of Chemoinformatics*; John Wiley & Sons, Ltd, 2008; Chapter VIII.2, pp 1004–1033.

(33) Todeschini, R.; Consonni, V. *Handbook of molecular descriptors*; John Wiley & Sons, 2008; Vol. 11.

(34) Gasteiger, J.; Marsili, M. Iterative partial equalization of orbital electronegativity—a rapid access to atomic charges. *Tetrahedron* **1980**, *36*, 3219 – 3228.

(35) Balaban, A. T. Highly discriminating distance-based topological index. *Chem. Phys. Lett.* **1982**, *89*, 399 – 404.

(36) Bertz, S. H. The first general index of molecular complexity. *J. Am. Chem. Soc.* **1981**, *103*, 3599–3601.

(37) Bonchev, D.; Trinajstić, N. Information theory, distance matrix, and molecular branching. *J. Chem. Phys.* **1977**, *67*, 4517–4533.

(38) Hall, L. H.; Kier, L. B. *Reviews in Computational Chemistry*; John Wiley & Sons, Ltd, 2007; pp 367–422.

(39) Wildman, S. A.; Crippen, G. M. Prediction of physicochemical parameters by atomic contributions. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 868–873.

(40) Labute, P. A widely applicable set of descriptors. *J. Mol. Graphics Modell.* **2000**, *18*, 464 – 477.

(41) Ramsundar, B.; Eastman, P.; Walters, P.; Pande, V.; Leswing, K.; Wu, Z. *Deep Learning for the Life Sciences*; O'Reilly Media, 2019.

(42) Drucker, H.; Burges, C. J. C.; Kaufman, L.; Smola, A.; Vapnik, V. Support vector regression machines. Proceedings of the 9th International Conference on Neural Information Processing Systems. Cambridge, MA, USA, 1996; p 155–161.

(43) Cortes, C.; Vapnik, V. Support-vector networks. *Machine learning* **1995**, *20*, 273–297.

(44) Breiman, L. Random Forests. *Machine Learning* **2001**, *45*, 5–32.

(45) Breiman, L.; Friedman, J.; Stone, C. J.; Olshen, R. A. *Classification and regression trees*; Chapman and Hall/CRC, 1984.

(46) Höskuldsson, A. PLS regression methods. *J. Chemom.* **1988**, *2*, 211–228.

(47) Wold, S.; Sjöström, M.; Eriksson, L. PLS-regression: a basic tool of chemometrics. *Chemom. Intell. Lab. Syst.* **2001**, *58*, 109 – 130.

(48) Chong, I.-G.; Jun, C.-H. Performance of some variable selection methods when multicollinearity is present. *Chemom. Intell. Lab. Syst.* **2005**, *78*, 103 – 112.

(49) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural message passing for quantum chemistry. Proceedings of the 34th International Conference on Machine Learning-Volume 70. 2017; pp 1263–1272.

(50) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. Analyzing learned molecular representations for property prediction. *J. Chem. Inf. Model.* **2019**, *59*, 3370–3388.

(51) Xie, T.; Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys. Rev. Lett.* **2018**, *120*, 145301.

(52) Chandrasekaran, N.; Oommen, C.; Kumar, V. R. S.; Lukin, A. N.; Abrukov, V. S.; Anufrieva, D. A. Prediction of detonation velocity and n-o composition of high energy c-h-n-o explosives by means of artificial neural networks. *Propellants, Explos., Pyrotech.* **2019**, *44*, 579–587.

(53) Kuzminykh, D.; Polykovskiy, D.; Kadurin, A.; Zhebrak, A.; Baskov, I.; Nikolenko, S.; Shayakhmetov, R.; Zhavoronkov, A. 3D molecular representations based on the wave transform for convolutional neural networks. *Mol. Pharmaceutics* **2018**, *15*, 4378–4385.

(54) Ma, J.; Sheridan, R. P.; Liaw, A.; Dahl, G. E.; Svetnik, V. Deep neural nets as a method for quantitative structure–activity relationships. *J. Chem. Inf. Model.* **2015**, *55*, 263–274.

(55) Sanyal, S.; Balachandran, J.; Yadati, N.; Kumar, A.; Rajagopalan, P.; Sanyal, S.; Talukdar, P. MT-CGCNN: integrating crystal graph convolutional neural network with multitask learning for material property prediction. *arXiv preprint arXiv:1811.05660* **2018**,

(56) Yeo, I.-K.; Johnson, R. A. A new family of power transformations to improve normality or symmetry. *Biometrika* **2000**, *87*, 954–959.

(57) Bemis, G. W.; Murcko, M. A. The Properties of Known Drugs. 1. Molecular Frameworks. *J. Med. Chem.* **1996**, *39*, 2887–2893.

(58) Pope, P. E.; Kolouri, S.; Rostami, M.; Martin, C. E.; Hoffmann, H. Explainability methods for graph convolutional neural networks. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019; pp 10764–10773.

(59) Kier, L. B.; Hall, L. H. An electrotopological-state index for atoms in molecules. *Pharm. Res.* **1990**, *7*, 801–807.

(60) Ertl, P.; Rohde, B.; Selzer, P. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. *J. Med. Chem.* **2000**, *43*, 3714–3717.

(61) Huang, L.; Massa, L. Applications of energetic materials by a theoretical method (discover energetic materials by a theoretical method). *Int. J. Energ. Mater. Chem. Propul.* **2013**, *12*, 197–262.

Supporting information

Predicting Energetics Materials' Crystalline Density from Chemical Structure by Machine Learning

Supplementary information

Phan Nguyen,[†,§] Donald Loveland,[‡,§] Joanne T. Kim,[¶] Piyush Karande,[†] Anna M. Hiszpanski,[*,‡] and T. Yong-Jin Han[*,‡]

†*Computational Engineering Division, Lawrence Livermore National Laboratory, Livermore, California 94550, United States*

‡*Materials Science Division, Lawrence Livermore National Laboratory, Livermore, California 94550, United States*

¶*Computing Scholar Program, Lawrence Livermore National Laboratory, Livermore, California 94550, United States*

§*Co-first authors*

E-mail: hiszpanski2@llnl.gov; han5@llnl.gov

CSD/RDKit additional details

CSD densities

In our construction of the HE-related dataset from the molecules in CSD, we noted in the Materials and Methods section of the main text that we used the published density values that were available in CSD. However, the CSD Python API contain several other approaches to obtain a density value for a given molecule. Listing 1 shows an example containing three such approaches, the last of which was used for our dataset.

In Figure S1, pair plots of the densities for the three approaches are shown. We see that most of the molecules have similar densities across the three approaches, but there are some large differences for some molecules. More specifically, the first (`crystal_calculated_density`) and second approaches (`entry_calculated_density`) are the same for almost all of the molecules, but there are densities that are different by a multiple of two. Comparing the second and third approaches (`entry_published_density`), we see that the densities are largely the same, and any differences between the two approaches are on the order of 10^{-1}.

Since the `entry_published_density` density entries were obtained from their respective publications, we chose to use these values for our dataset, but given the similarities between the three approaches, we do not expect any fitted models to drastically change if another approach were to be used. However, one should still be aware of these differences when using the CSD densities for other analyses. Moreover, these differences, especially the doubling of some of the density values between `entry_calculated_density` and `crystal_calculated_density` should prompt a need for explanations as to how these values were obtained or calculated.

```
from ccdc import io

csd_reader = io.EntryReader('CSD')
mol_name = 'JUVMOC'

crystal_calculated_density = csd_reader.crystal(mol_name).calculated_density
entry_calculated_density = csd_reader.entry(mol_name).calculated_density
entry_published_density = csd_reader.entry(mol_name)._entry.editors_info().published_calculated_density().value()
```

Listing 1: Example code for different approaches to obtain densities from CSD.

RDKit features

In Table S1, a list of the 98 RDKit filtered features that were used with the RF and PLSR-based density regression models is shown. We also classified the features into 6 feature types: electronic/topological combination, bond information, lipophilicity, atoms/group in-

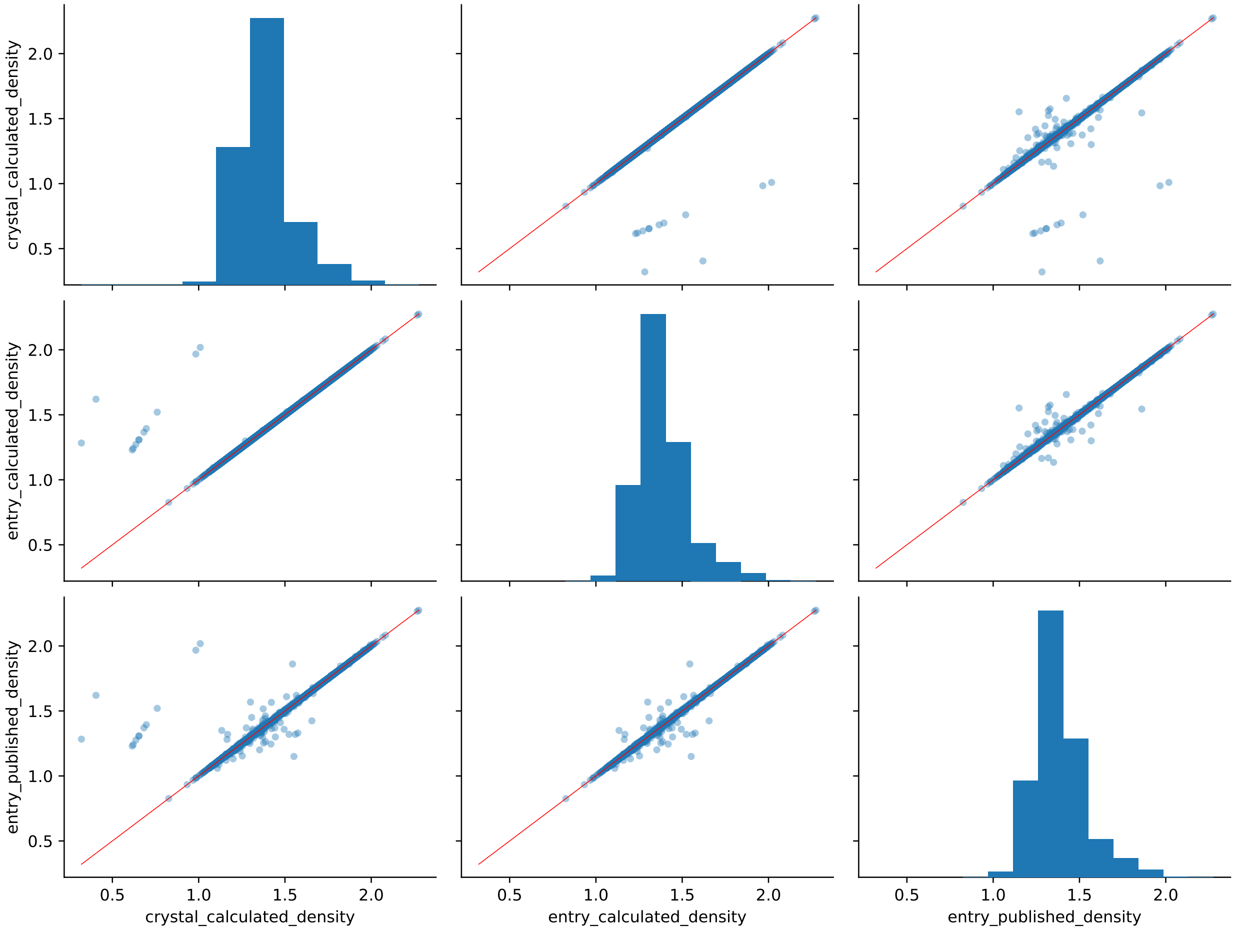


Figure S1: Pair plot of crystal densities based on different approaches available in the CSD Python API and shown in Listing 1. While many of the values are the same across the three methods, there are still inconsistencies that need to be scrutinized.

formation, connectivity/topology, and electronic

For convenience, Table S2 summarizes the list of features used in MPNN, replicated from Ref. 1. Additional details about how these features are incorporated into MPNN may be found in Ref. 1.

Feature correlations

An important step in regression problems is data pre-processing, and certain methods may benefit from specific preparation steps before applying the method to the data to fit a model. The RDKit molecular features described in the Materials and Methods section of the main text were subject to several pre-processing steps based on exploratory data analyses. In Figure S2, a heatmap of the correlations between the RDKit molecular features with the constant variables removed and before filtering for perfectly correlated variables is shown. The features have also been clustered using hierarchical clustering. We see that there are many groups in which the features are highly correlated or anti-correlated with each other, and in some cases, some variables are perfectly correlated with each other. Since the presence of high correlations of the features can adversely affect the performance and interpretation of many approaches, we retained only one of the features of any group of correlated predictors in the dataset, and we selected methods such as random forests and partial least squares regression that can handle potential multicollinearity in a dataset.

Additional pre-processing steps were also performed based on the distributions of values of each feature. Figure S3 shows these distributions for all of the RDKit molecular features. We first note the Ipc feature was log-transformed and replaced with logIpc, since the original values span large orders of magnitude ($\sim 10^0 - 10^{80}$) that can lead to numerical problems during model-fitting. We also see that several predictors are constant and therefore have no predictive value, so they were removed from the dataset. Lastly, many of the features are skewed, so a power transform may be applied in combination with certain methods.

Table S1: RDKit features and classification after filtering to exclude constant or perfectly correlated variables.

Type	Count	Features
Electronic/ topological combination	55	TPSA, Chi0, Chi0n, Chi1, Chi1n, Chi2n, Chi3n, Chi4n, MaxEStateIndex, MinEStateIndex, MinAbsEStateIndex, VSA_EState1, VSA_EState2, VSA_EState3, VSA_EState4, VSA_EState5, VSA_EState6, VSA_EState7, VSA_EState8, VSA_EState9, EState_VSA1, EState_VSA10, EState_VSA11, EState_VSA2, EState_VSA3, EState_VSA4, EState_VSA5, EState_VSA6, EState_VSA7, EState_VSA8, EState_VSA9, SMR_VSA1, SMR_VSA10, SMR_VSA2, SMR_VSA3, SMR_VSA4, SMR_VSA5, SMR_VSA6, SMR_VSA7, SMR_VSA9, PEOE_VSA1, PEOE_VSA10, PEOE_VSA11, PEOE_VSA12, PEOE_VSA13, PEOE_VSA14, PEOE_VSA2, PEOE_VSA3, PEOE_VSA4, PEOE_VSA5, PEOE_VSA6, PEOE_VSA7, PEOE_VSA8, PEOE_VSA9, HallKierAlpha
Bond information	11	NumRotatableBonds, NumAliphaticCarbocycles, NumAliphaticHeterocycles, NumAliphaticRings, NumAromaticCarbocycles, NumAromaticHeterocycles, NumAromaticRings, NumSaturatedCarbocycles, NumSaturatedHeterocycles, NumSaturatedRings, FractionCSP3
Lipophilicity	10	SlogP_VSA1, SlogP_VSA10, SlogP_VSA11, SlogP_VSA2, SlogP_VSA3, SlogP_VSA4, SlogP_VSA5, SlogP_VSA6, SlogP_VSA8, MolLogP
Atom/ group information	8	MolWt, HeavyAtomMolWt, HeavyAtomCount, NHOHCount, NOCount, NumHAcceptors, NumHDonors, RingCount
Connectivity/topology	7	BalabanJ, BertzCT, LabuteASA, Ipc, Kappa1, Kappa2, Kappa3
Electronic	6	MaxPartialCharge, MinPartialCharge, MaxAbsPartialCharge, MinAbsPartialCharge, NumValenceElectrons, NumRadicalElectrons, MolMR

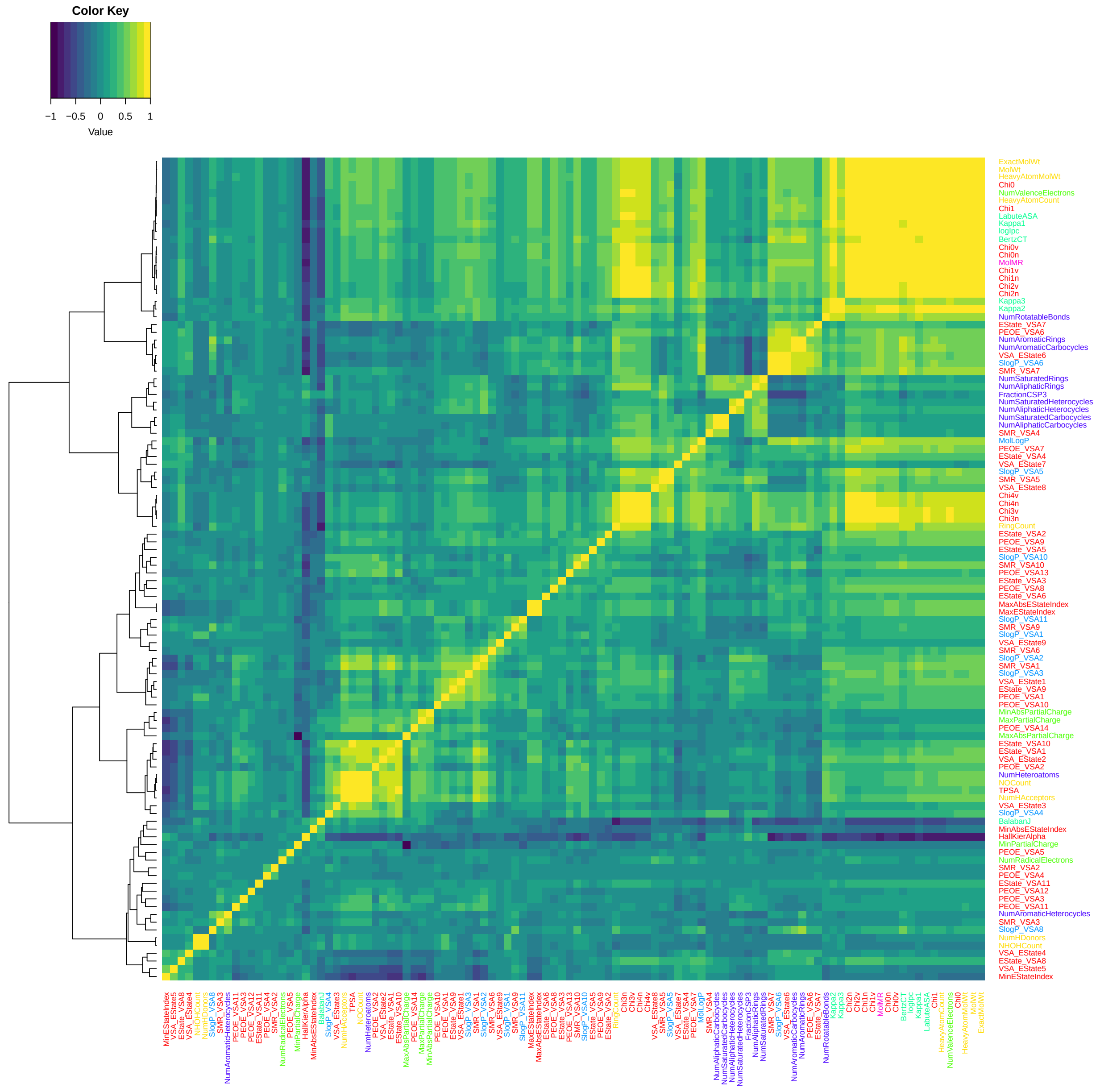


Figure S2: Correlations between RDKit molecular features for the HE-related dataset, colored by feature classification and clustered with hierarchical clustering. There are many clusters of highly correlated variables, some of which are perfectly correlated.

Table S2: RDKit features used in MPNN.

Type	Feature	Description	Size
atom	atom type	type of atom (ex. C, N, O), by atomic number	100
	# bonds	number of bonds the atom is involved in	6
	formal charge	integer electronic charge assigned to atom	5
	chirality	unspecified, tetrahedral CW/CCW, or other	4
	# H	number of bonded hydrogen atoms	5
	hybridization	sp, sp2, sp3, sp3d, or sp3d2	5
	aromaticity	whether this atom is part of an aromatic system	1
	atomic mass	mass of the atom, divided by 100	1
bond	bond type	single, double, triple, or aromatic	4
	conjugated	whether the bond is conjugated	1
	in ring	whether the bond is part of a ring	1
	stereo	none, any, E/Z or cis/trans	6

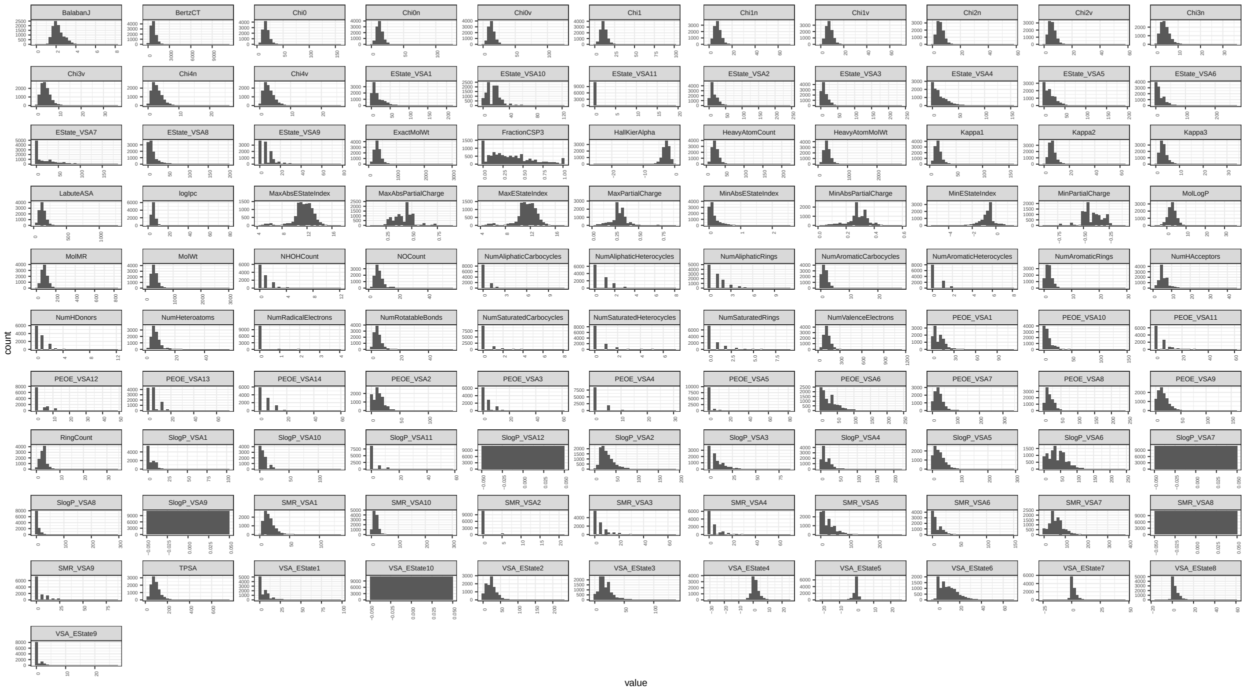


Figure S3: Distributions of values for each RDKit molecular feature for the HE-related dataset.

Oxygen balance

Oxygen balance is an important characteristic of energetic materials that could have been used as an additional filtering criterion for our HE-related dataset. In Figure S4, the oxygen balance values are plotted against the densities for the molecules of our dataset. Since the two values appear to be strongly correlated, filtering based on oxygen balance would eliminate many molecules with low-to-moderate density values, which are still important for training models to correctly screen for candidate HE materials. Any models fit to the reduced dataset would only produce reliable predictions in the high-density range and therefore cannot be used to discriminate between low and high density materials.

Feature importance ranking

In Table 2 of the main text, the top 20 predictors based on random forest Gini and permutation importance and partial least squares regression VIP scores where shown. Table S3 shows the rankings of all of the predictors for each variable importance approach.

Table S3: Variable importance rankings based on random forest Gini and permutation importance and partial least squares regression VIP scores.

rank	RF (Gini)	RF (permutation)	PLSR (VIP)
1	VSA_EState8	TPSA	VSA_EState8
2	SlogP_VSA5	VSA_EState8	SlogP_VSA5
3	TPSA	SMR_VSA5	TPSA
4	SMR_VSA5	SlogP_VSA5	NOCount
5	MolLogP	NOCount	SMR_VSA5
6	NOCount	MolLogP	MolLogP
7	PEOE_VSA6	EState_VSA10	PEOE_VSA7
8	EState_VSA7	FractionCSP3	Chi2n
9	EState_VSA10	PEOE_VSA6	RingCount

rank	RF (Gini)	RF (permutation)	PLSR (VIP)
10	PEOE_VSA7	EState_VSA7	EState_VSA10
11	Chi2n	VSA_EState6	PEOE_VSA6
12	VSA_EState3	Chi2n	NumHAcceptors
13	FractionCSP3	Kappa2	Chi1n
14	MaxPartialCharge	Kappa3	Chi3n
15	VSA_EState6	VSA_EState3	Chi4n
16	Chi1n	Chi0n	BalabanJ
17	Chi3n	NumHAcceptors	EState_VSA7
18	NumHAcceptors	HallKierAlpha	VSA_EState3
19	Chi0n	Chi1n	Chi0n
20	MinEStateIndex	MolMR	MolMR
21	Chi4n	EState_VSA5	Kappa3
22	PEOE_VSA8	SlogP_VSA6	MaxPartialCharge
23	MolMR	Kappa1	Kappa2
24	Kappa2	PEOE_VSA8	EState_VSA1
25	EState_VSA1	PEOE_VSA7	LabuteASA
26	HallKierAlpha	MinEStateIndex	FractionCSP3
27	SlogP_VSA6	EState_VSA6	HallKierAlpha
28	Kappa3	SlogP_VSA4	log_Ipc
29	PEOE_VSA2	EState_VSA1	Kappa1
30	EState_VSA6	MaxPartialCharge	VSA_EState6
31	SlogP_VSA4	PEOE_VSA2	MinEStateIndex
32	EState_VSA5	Chi3n	Chi1
33	Kappa1	RingCount	NumAliphaticRings
34	VSA_EState2	EState_VSA4	EState_VSA4
35	EState_VSA8	VSA_EState9	NumValenceElectrons

rank	RF (Gini)	RF (permutation)	PLSR (VIP)
36	MinAbsPartialCharge	VSA_EState2	HeavyAtomCount
37	EState_VSA4	BertzCT	VSA_EState2
38	LabuteASA	Chi4n	SlogP_VSA6
39	BalabanJ	BalabanJ	BertzCT
40	VSA_EState4	EState_VSA8	MolWt
41	SMR_VSA7	SMR_VSA3	Chi0
42	MaxEStateIndex	SMR_VSA6	HeavyAtomMolWt
43	VSA_EState7	NumRotatableBonds	EState_VSA6
44	SMR_VSA3	SMR_VSA7	VSA_EState7
45	BertzCT	MinAbsPartialCharge	MinAbsPartialCharge
46	MinPartialCharge	LabuteASA	PEOE_VSA2
47	SMR_VSA1	SMR_VSA1	SlogP_VSA4
48	PEOE_VSA13	VSA_EState7	SlogP_VSA3
49	MinAbsEStateIndex	PEOE_VSA9	PEOE_VSA8
50	SlogP_VSA10	SlogP_VSA3	NumAromaticCarbocycles
51	SMR_VSA10	SlogP_VSA10	VSA_EState5
52	NumValenceElectrons	SMR_VSA10	NumAromaticRings
53	PEOE_VSA9	MaxEStateIndex	SlogP_VSA10
54	VSA_EState5	MinPartialCharge	SMR_VSA1
55	log_Ipc	PEOE_VSA13	SMR_VSA7
56	PEOE_VSA14	NumAromaticRings	PEOE_VSA14
57	SlogP_VSA3	PEOE_VSA14	NumAromaticHeterocycles
58	VSA_EState9	NumAromaticCarbocycles	MaxEStateIndex
59	SMR_VSA6	log_Ipc	PEOE_VSA13
60	Chi1	SlogP_VSA8	NumAliphaticHeterocycles
61	Chi0	VSA_EState4	NumSaturatedRings

rank	RF (Gini)	RF (permutation)	PLSR (VIP)
62	SlogP_VSA2	NumAromaticHeterocycles	NumRotatableBonds
63	HeavyAtomMolWt	EState_VSA3	VSA_EState4
64	EState_VSA2	VSA_EState1	NumAliphaticCarbocycles
65	MolWt	PEOE_VSA3	SMR_VSA4
66	NumAromaticCarbocycles	NumAliphaticRings	EState_VSA3
67	EState_VSA3	NumValenceElectrons	EState_VSA8
68	PEOE_VSA3	EState_VSA2	NumSaturatedCarbocycles
69	MaxAbsPartialCharge	Chi1	MinPartialCharge
70	RingCount	Chi0	EState_VSA5
71	NumRotatableBonds	HeavyAtomMolWt	SlogP_VSA8
72	VSA_EState1	SMR_VSA4	NumRadicalElectrons
73	PEOE_VSA1	MinAbsEStateIndex	SMR_VSA10
74	HeavyAtomCount	MolWt	NumSaturatedHeterocycles
75	PEOE_VSA11	SlogP_VSA2	EState_VSA9
76	EState_VSA9	VSA_EState5	SlogP_VSA2
77	PEOE_VSA10	PEOE_VSA11	EState_VSA2
78	NumAromaticRings	PEOE_VSA10	PEOE_VSA11
79	SMR_VSA4	NumAliphaticHeterocycles	VSA_EState9
80	SlogP_VSA8	NumHDonors	NHOHCount
81	NumAromaticHeterocycles	EState_VSA9	NumHDonors
82	NumAliphaticRings	HeavyAtomCount	PEOE_VSA1
83	SlogP_VSA1	NumSaturatedRings	PEOE_VSA9
84	PEOE_VSA12	PEOE_VSA1	VSA_EState1
85	NHOHCount	MaxAbsPartialCharge	MinAbsEStateIndex
86	NumHDonors	PEOE_VSA12	MaxAbsPartialCharge
87	NumAliphaticHeterocycles	NHOHCount	SMR_VSA2

rank	RF (Gini)	RF (permutation)	PLSR (VIP)
88	NumSaturatedRings	NumSaturatedHeterocycles	SMR_VSA3
89	SMR_VSA9	SlogP_VSA1	SMR_VSA6
90	PEOE_VSA5	NumAliphaticCarbocycles	SlogP_VSA1
91	NumSaturatedHeterocycles	PEOE_VSA4	PEOE_VSA3
92	PEOE_VSA4	SlogP_VSA11	SlogP_VSA11
93	NumAliphaticCarbocycles	SMR_VSA9	SMR_VSA9
94	NumSaturatedCarbocycles	SMR_VSA2	PEOE_VSA10
95	SlogP_VSA11	PEOE_VSA5	PEOE_VSA4
96	NumRadicalElectrons	NumSaturatedCarbocycles	PEOE_VSA12
97	SMR_VSA2	NumRadicalElectrons	PEOE_VSA5
98	EState_VSA11	EState_VSA11	EState_VSA11

Elton et al. additional analyses

While we curated and utilized a large 10k dataset, Elton et al. recently reported ML models to predict properties of HE,[2] including density, using a small dataset previously curated by Huang and Massa[3] that consists of 109 energetic molecules. To featurize HE molecules in their dataset, Elton et al. created 21 custom features that incorporated domain knowledge, such as oxygen balance and configurations of bonds involving of C, N, O, and H, and considered more standardized feature sets, including sum over bonds (SoB), Coulomb matrices, and fingerprints. In their analyses, they showed that SoB features in combination with kernel ridge regression (KRR) had the highest average performance across the different target properties that were considered. Since their features and methods exhibited encouraging results for molecular property prediction from small datasets, we also applied our methods to the same smaller 109 HE dataset to compare the approaches. Specifically, using this dataset, we now compare the density prediction performance of our approaches against that of the

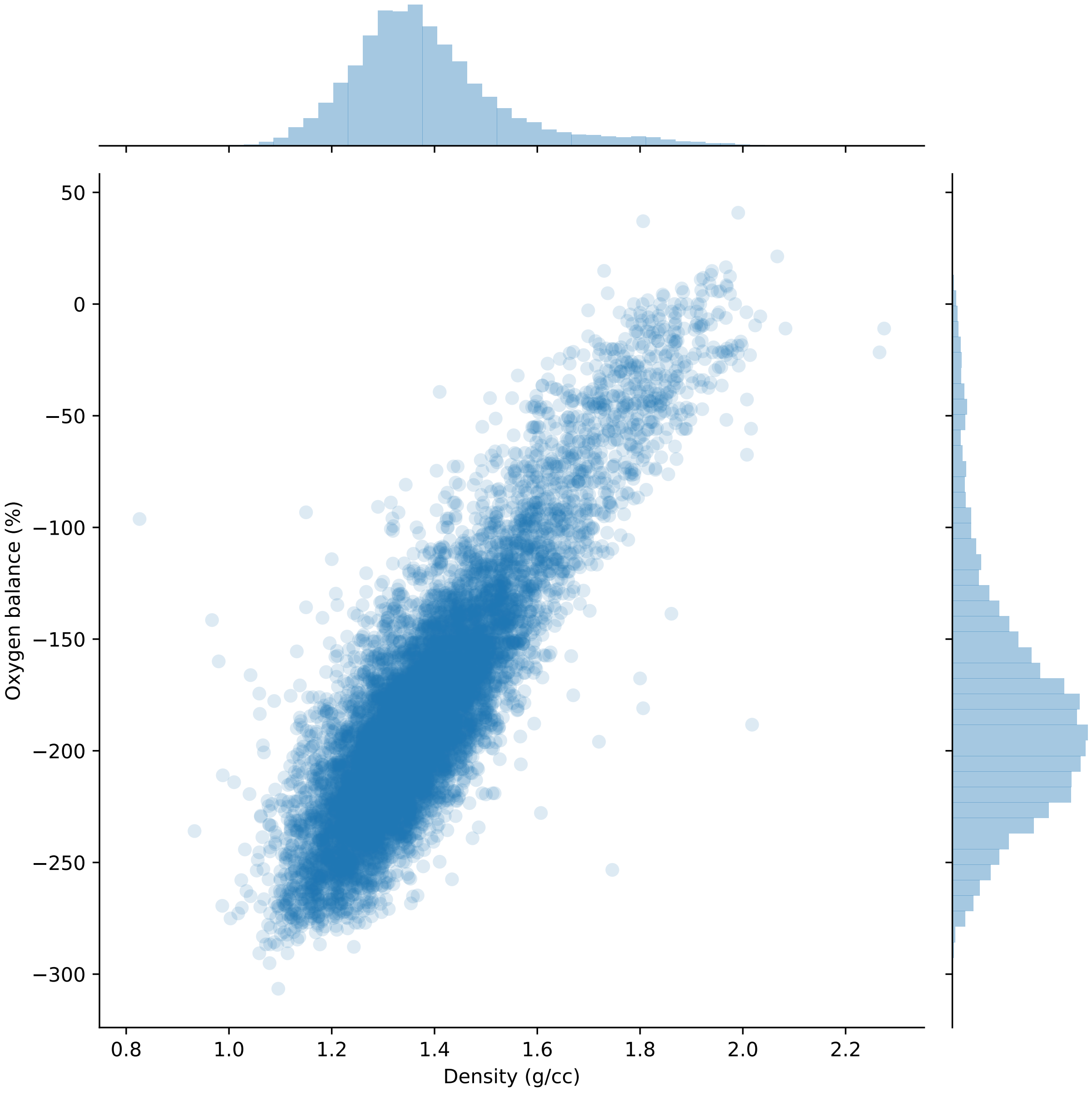


Figure S4: Oxygen balance vs. density for the molecules of the HE-related dataset.

SoB features with KRR.

To compare our methods and results against those of Elton et al.[2] on the dataset by Huang and Massa,[3] we only consider their results for density prediction with SoB features and kernel ridge regression (KRR), which had the highest average Pearson R scores across all predicted target properties. In the analysis by Elton et al., method performance was evaluated by using 5-fold cross-validation on 80-20 train-test splits of the data. For each method and set of features, this cross-validation procedure was repeated 20 times, and the average of various performance metrics were taken across the 20 runs, including the R^2 and Pearson R scores. However, the combination of the small dataset and lack of stratification in the cross-validation procedure may produce dissimilar training and test datasets with high variance in any performance metrics, and averaging may obscure this variance and the true performance of a method.

To fit and assess the density regression models, we implement the same training procedure by Elton et al., using 5-fold cross-validation with an 80-20 train-test split. We repeat this procedure multiple times, and for each run, we compute the mean R^2 and Pearson R across the folds of that run. Since these statistics may be highly variable across runs on a small dataset, rather than compute an average of metrics over different cross-validation runs, we instead consider the distribution of performance metrics over those runs. Due to the computational demands of fitting MPNN-based models, we only use RF and PLSR to model density as a function of the RDKit features in this analysis. In addition, Elton et al. imposed that hydrogen atoms were explicitly present in a molecular topology before computing any features, whereas RDKit assumes by default that these atoms are implicitly known for any feature computations. To enable an equitable comparison, we first make the hydrogen atoms explicit for the RDKit featurization in the following analysis.

Comparison

Figure S5 shows the distribution of R^2 and Pearson R values for 200 runs using the RDKit molecular features with RF and PLSR as well as the SoB features with KRR. Table S4 shows the corresponding means and standard deviations of these distributions. Generally, for every approach considered, the distributions of both accuracy measures tend towards positive values, but the distributions are highly variable, with some containing negative scores that are indicative of poor performance, so the perceived accuracy of an approach can heavily depend on the train-test split of the data.

Of the three approaches, KRR with the SoB features had the highest average Pearson R value, while PLSR with the RDKit features had the highest average R^2 value. However, the large overlap between the distributions for all of the methods with both performance metrics and the lack of robustness between train-test split iterations make it difficult to ascertain a best-performing method for critical applications to novel molecules. Moreover, the large variance in scores and potential poor performance illustrate the difficulties with learning and deploying models based on small datasets in materials discovery and other chemical prediction problems. Nevertheless, the tendency of the Pearson R and R^2 distributions for all of the methods to incline towards larger values suggests that the volume of information that is covered by the featurizations but restricted by the small Huang and Massa dataset still contains information that is relevant to estimating crystal densities. In addition, the models that were considered were able to elicit suitable dependencies to reasonably predict densities from the features over many train-test splits, but the analysis, interpretation, and utility of the fitted models should be subject to additional scrutiny.

Implicit vs. explicit H

In their analyses, Elton et al. assumed that hydrogen atoms should be explicitly present in any molecular representations from which any features were to be computed. However, this step was never fully justified by the authors, and the RDKit documentation cautions that

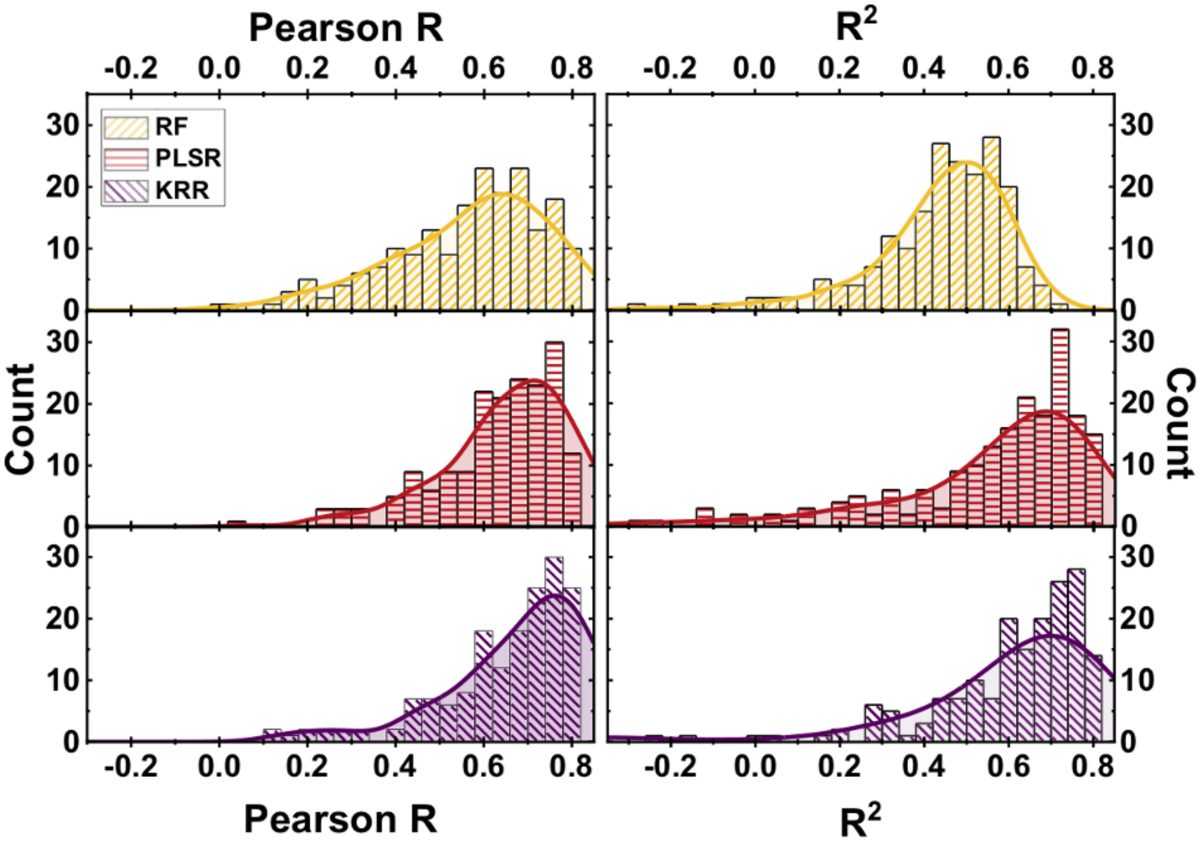


Figure S5: Distributions of Pearson R values and R^2 over 200 train-test split iterations of the Huang and Massa density dataset using the RDKit molecular features with RF and PLSR as well as the SoB features with KRR. While the Pearson R distributions tend toward high values, they exhibit large variance and are similar across the three approaches, making it uncertain which method and features should be preferred on the small dataset based on these metrics alone. With respect to R^2 scores, the PLSR- and KRR-based approaches tend to perform similarly across the iterations, while RF tends to perform worse.

Table S4: Mean and standard deviation of the Pearson R and R^2 values for various combinations of methods and features.

score	method	features	mean	sd
R^2	RF	RDKit	0.443	0.156
R^2	PLSR	RDKit	0.563	0.242
R^2	KRR	SoB	0.550	0.331
Pearson R	RF	RDKit	0.570	0.176
Pearson R	PLSR	RDKit	0.650	0.146
Pearson R	KRR	SoB	0.669	0.179

much of the code for RDKit assumes that any hydrogen atoms are implicit in the molecular topology. However, the documentation also notes that the hydrogen atoms may need to be added in order to obtain realistic geometries.

In Figures S6 and S7, we compare the effectiveness of using implicit and explicit hydrogen atoms in the feature computations based on the Pearson R values and R^2 values, respectively. As with the analysis in the main text, we compute these values over multiple train-test splits and aggregate the values into distributions rather than take the average of the values. In general, the distributions for both metrics suggest that forcing the hydrogens to be explicit in the molecular representations helps to improve the accuracy for PLSR and KRR over different train-test splits, but the difference in the distribution of scores for RF appears to be negligible. Nevertheless, the potential improvements shown in the PLSR and KRR cases suggest that the explicitness of the hydrogen atoms may be an important consideration when fitting a model and should be treated as another parameter that may need to be tuned.

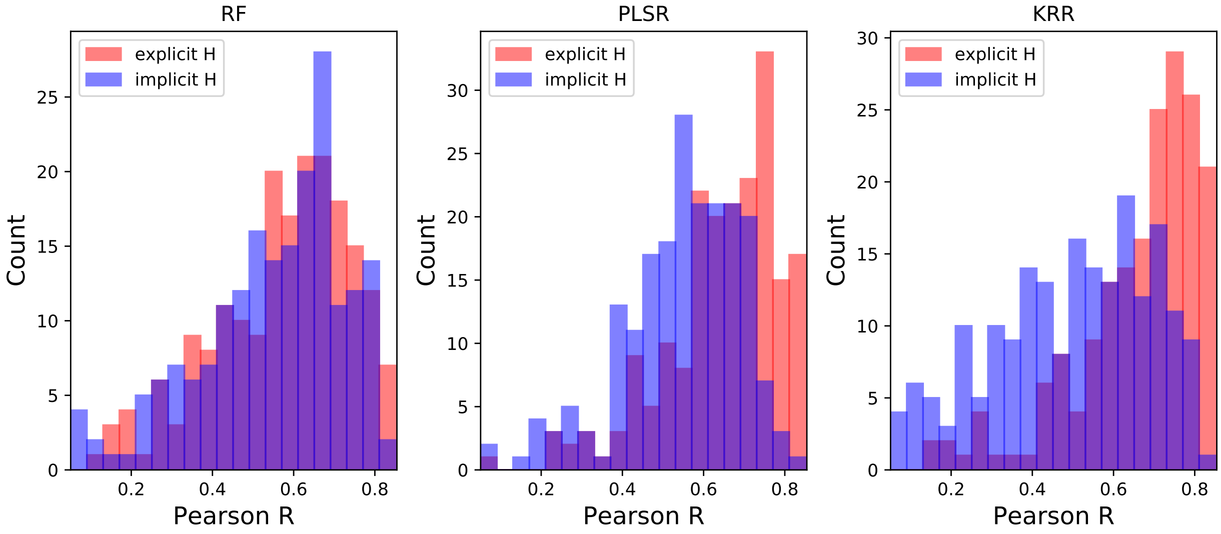


Figure S6: Comparison of Pearson R distributions using implicit and explicit hydrogen atoms in feature computations for the RDKit molecular features with RF and PLSR as well as the SoB features with KRR. Each distribution is based on 200 train-test split iterations of the Huang and Massa density dataset.

We now revisit the HE-related dataset and examine the effect of making the hydrogens explicit in the RDKit features. Table S5 shows the R^2 values for RF, PLSR, and MPNN when

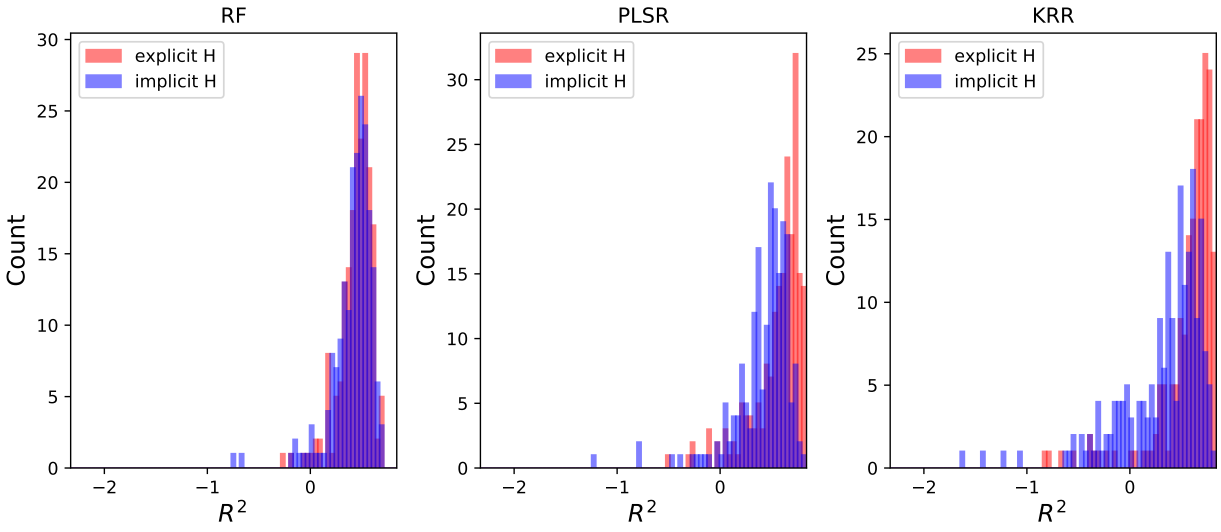


Figure S7: Comparison of R^2 distributions using implicit and explicit hydrogen atoms in feature computations for the RDKit molecular features with RF and PLSR as well as the SoB features with KRR. Each distribution is based on 200 train-test split iterations of the Huang and Massa density dataset.

the hydrogen atoms are implicit and explicit in the featurizations. Unlike the applications to the Huang and Massa dataset, there is no considerable improvement in making the hydrogen atoms explicit; the R^2 are almost the same in the implicit and explicit cases for the three methods. Similar observations can also be made about the prediction errors. In Figure S8 and S9, the distribution of errors for the predicted densities and the median error within bins defined by the true densities with a bin width of 0.01 are shown. As with the R^2 scores, the differences between the implicit and explicit cases are marginal. Therefore, while the explicitness of the hydrogen atoms should be a consideration when fitting a model in certain problems, using the RDKit features based on the default implicit setting does not appear to be detrimental to the accuracy of a method.

Ordinal feature MPNN

A common technique to help reduce overfitting and improve generalization in a machine learning model is to decrease the dimensionality of the feature set. This is commonly seen in

Table S5: R^2 scores when using the RDKit features with implicit and explicit hydrogen atoms for HE-related density prediction. The differences between the implicit and explicit scores are marginal for each method.

Featurization	Method	R^2 (implicit H)	R^2 (explicit H)
RDKit (molecular)	RF	0.878	.888
RDKit (molecular)	PLSR	0.900	.895
RDKit (atom/bond)	MPNN	0.914	0.912

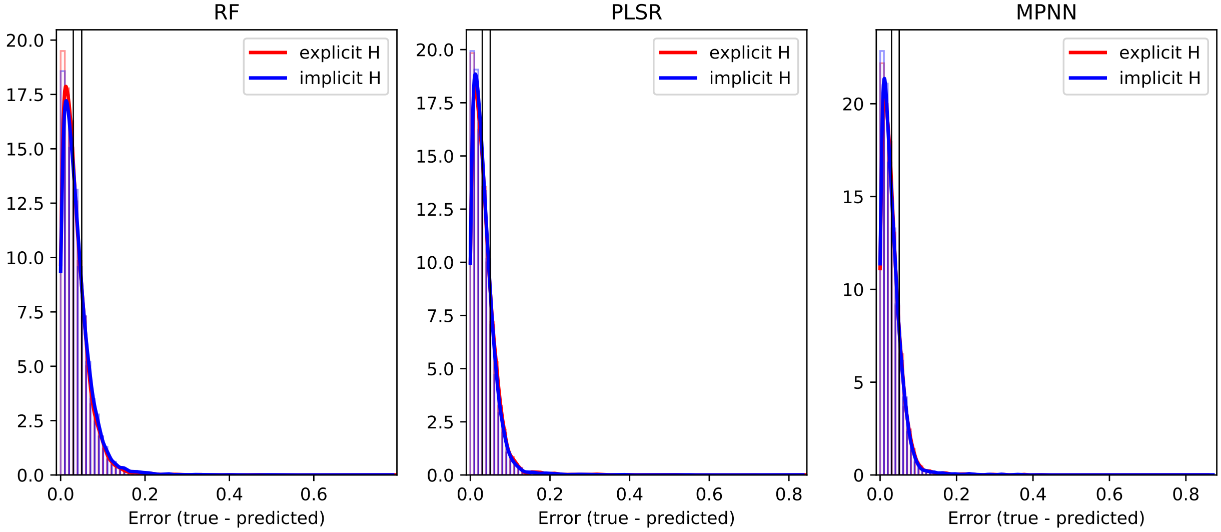


Figure S8: Distributions of predicted density errors when using the RDKit features with implicit and explicit hydrogen atoms for HE-related density prediction.

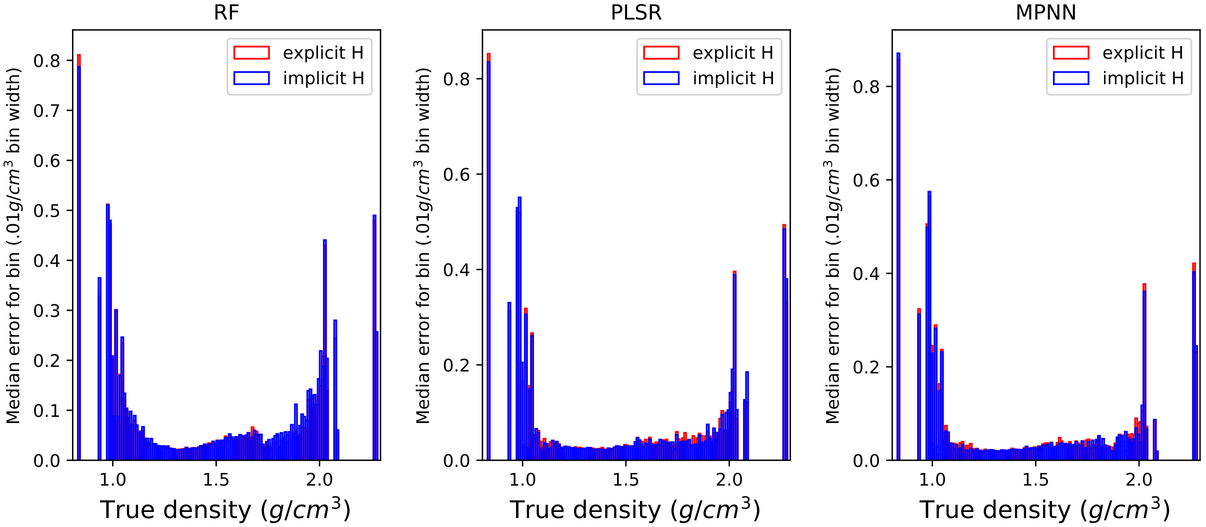


Figure S9: Median predicted density error within bins defined by the true densities of the HE-related molecules at 0.01g/cc intervals when using the RDKit features with implicit and explicit hydrogen atoms for HE-related density prediction.

techniques such as LASSO which aim to find a sparse feature representation which concurrently minimizes a given loss function. We perform a manual feature reduction by ordinally encoding the features described in Table S2 that possess a clear ordered relationship across categories. Compared to the 127 features created from the original one-hot encoding, this ordinal encoding creates a set of just 7 features, where each feature is simply an integer mapping to a single discrete category. The edge features are not ordinally encoded due to the lack of an obvious ordering within a categorical bond type feature. While this reduction limits the feature size, we believe that it also introduces an inductive bias that helps better contextualize categorical features' discrete categories

In order to verify that the ordinal encoding is not overfitting to the high-density data, the new feature set is also used to re-train models using the HE-related dataset. In order to fairly assess the two methods, an ensemble of 5 independently initialized models are averaged to reach a final predicted value for each molecule. As seen in Figures S10 and S11, the ordinal encoding method performs nearly identically to the previous one-hot encoding method, indicating that the generalization is not hurting our predictive performance in other regions of the target space while giving better performance in the high-density regime.

MPNN y-scrambling

To test for the possibility of chance correlation with our dataset and models, we perform y-scrambling with our HE-like dataset and train with MPNN. In particular, we use one of the predetermined folds as our test dataset, we treat the remaining folds as a training dataset, and we permute the density values of the training dataset before fitting models with MPNN. We then evaluate the fitted model on the test fold. We repeat this multiple times, and we compare the resulting RMSE values to those of the original experiment.

Figure S12 shows the results of this experiment with fold 0 of our dataset used for testing. Over all of the random permutations that we perfomed, the test RMSE of this fold (0.047)

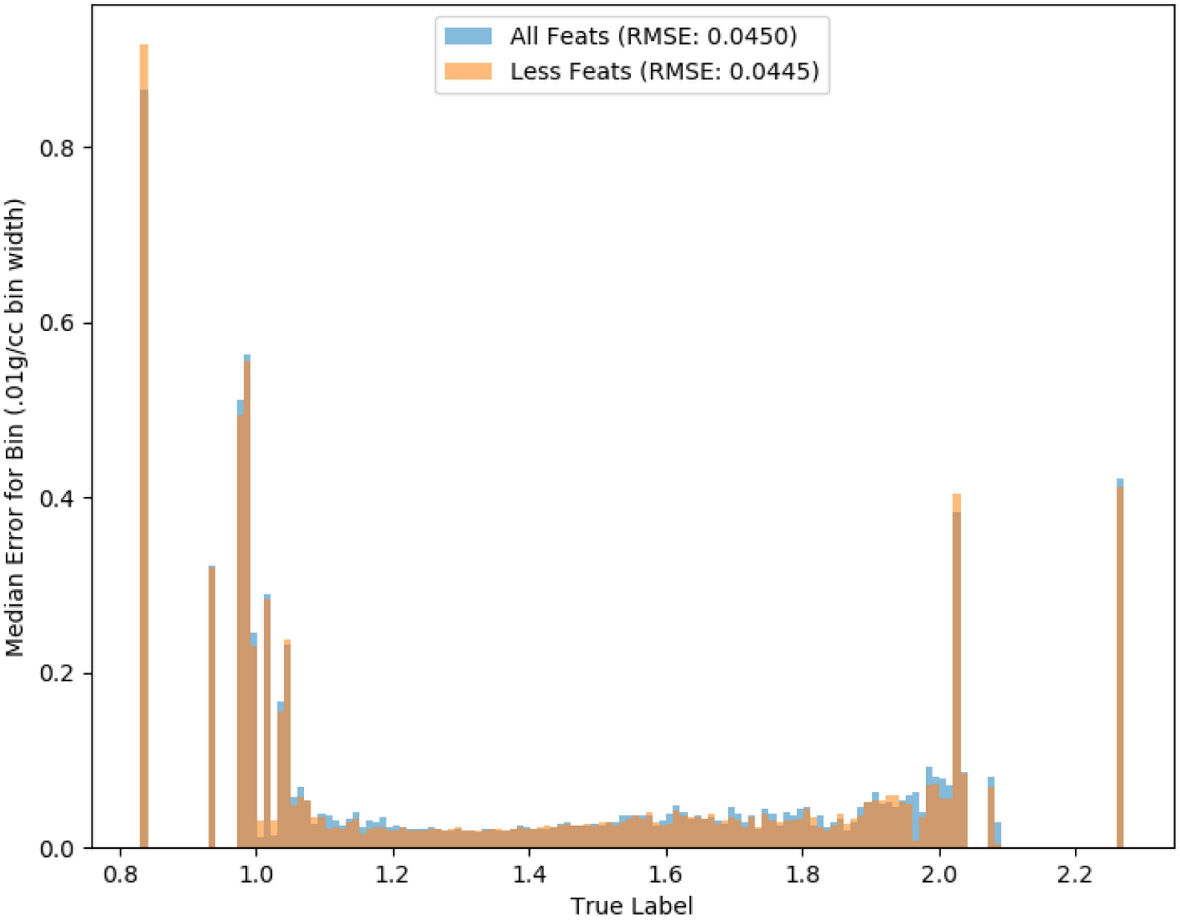
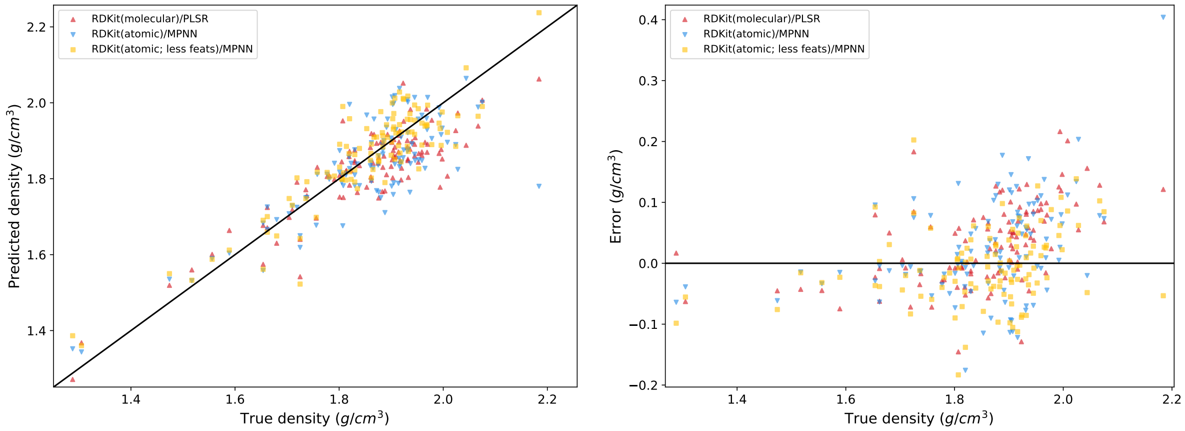


Figure S10: Median predicted density error within bins defined by the true densities of the HE-related molecules at 0.01g/cc intervals for MPNN with the original and ordinally encoded features.



(a) (b)

Figure S11: (a) Predicted and (b) error (true – predicted) vs. true packing density values using the HE-related dataset for training and Huang and Massa dataset for testing.

was always considerably less than the test RMSEs obtained by training on scrambled labels, indicating very little, if any, risk of chance correlation with our approaches.

Scaffold splitting

In the Results and Discussion section of the main text, we evaluated the performance of our approaches using k-fold cross-validation with stratified splits based on the density values. An alternative approach that has been applied to assess method performance is scaffold splitting. Instead of using stratified splits based on the outputs of interest, scaffold splitting attempts to separate structurally different molecules into different subsets based on their two-dimensional structural frameworks.[4] In doing so, this approach is expected to better simulate realistic experimental conditions in which the interest is in evaluating novel structures. However, creating the folds in this manner is also expected to provide a greater challenge for models to learn and generalize from the data.

We now evaluate the performance of our approaches using scaffold splitting with the functions available in deepchem and RDKit. As was done with stratified splitting, we summarize the overall performance of a method by computing the averages of the R^2 and RMSE

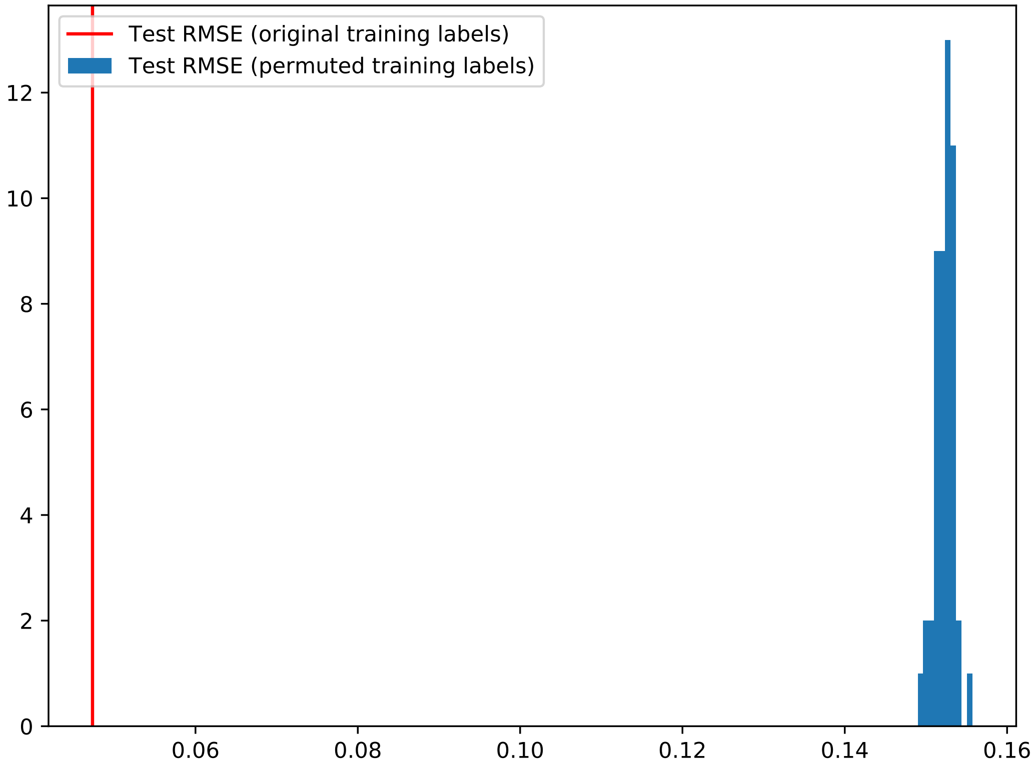


Figure S12: y-scrambling experiment with the HE-like dataset. The blue distribution shows the RMSE values on the unscrambled test set after fitting MPNN models that were trained on permutated density values of the training dataset. The red line shows the RMSE on the test set when MPNN was trained on the original training density values.

scores across the stratified folds. Table S6 compares these scores for the RF-, PLSR-, and MPNN-based methods with stratified and scaffold splits. Even though the methods are trained on splits designed to contain molecules that are structurally different from those contained in the test split, they still achieve good overall performance on unseen data. With each approach, there is a decrease in performance across when switching from stratified splitting to scaffold splitting, but the changes are small; the largest changes occur with random forests, with a decrease of .029 in R^2 and increase of 0.004 g/cc in RMSE. Therefore, these approaches also demonstrate an ability to predict densities of molecules containing novel structural characteristics with a reasonably low amount of error.

We also consider the individual predictions and errors made by these models. Figure S13 compares the predicted and true density values using scaffold splitting, and Figure S14 shows the error distribution and median errors at 0.01 g/cc intervals. Qualitatively, the results are largely similar to those observed using stratified splitting (see Figures 4 and 5 of the main text). There is a shift towards larger errors at the low end of the error distributions, but as with the R^2 and RMSE scores, the shift is small, so the comparisons made between the approaches with the stratified split in the main text are also applicable here.

Table S6: R^2 and RMSE values of different featurization, method combinations, and data-splitting procedures for HE-related density prediction.

Feature	Method	Split	R^2	RMSE
RDKit (molecular)	RF	Stratified	0.878	0.053
RDKit (molecular)	RF	Scaffold	0.849	0.057
RDKit (molecular)	PLSR	Stratified	0.900	0.048
RDKit (molecular)	PLSR	Scaffold	0.888	0.048
RDKit (atom/bond)	MPNN	Stratified	0.914	0.044
RDKit (atom/bond)	MPNN	Scaffold	0.902	0.045

Score details

In main and supplementary texts, some of our approaches were evaluated using the R^2 score, available in the scikit-learn package. In the supplementary text, we also use the Pearson R

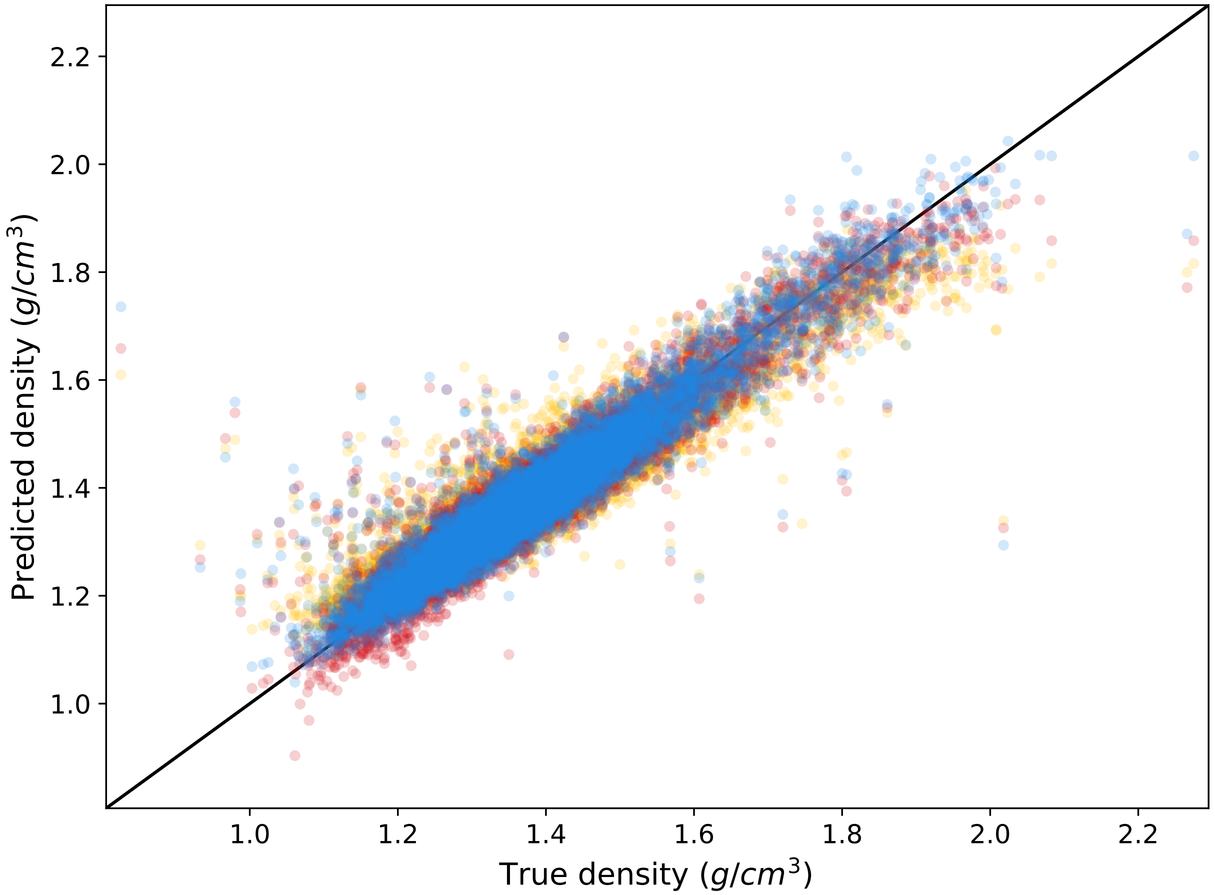
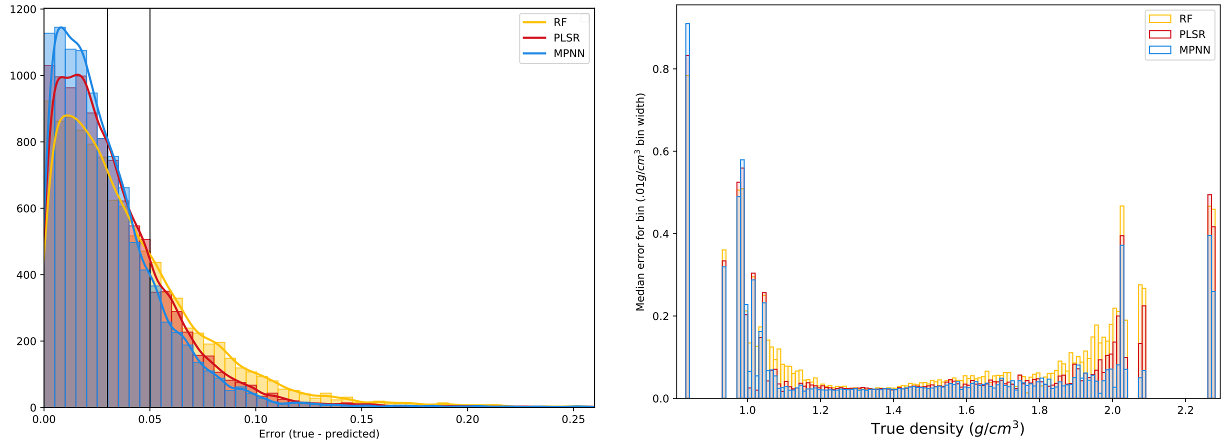


Figure S13: Predicted vs. true densities using scaffold splitting



(a)

(b)

Figure S14: ((a)) Distributions of predicted density errors and ((b)) predicted density error within bins defined by the true densities of the HE-related molecules at 0.01g/cc intervals for RF, PLSR, and MPNN using scaffold splitting. For clarity, the few errors above 0.25g/cc are not shown.

score, available in the Molecular Machine Learning Toolkit by Elton et al.,[2] respectively. In scikit-learn, the R^2 score is defined as

$$R^2 = 1 - \frac{\sum_i \left(y_i - y_i^{\text{pred}}\right)^2}{\sum_i \left(y_i - \bar{y}\right)^2}.$$

In the Molecular Machine Learning Toolkit, the Pearson R score is defined as

$$\text{Pearson } R = \frac{\left(\sum_i \left(y_i - \bar{y}\right)\left(y_i^{\text{pred}} - \bar{y}^{\text{pred}}\right)\right)^2}{\sqrt{\sum_i \left(y_i - \bar{y}\right)^2 \sum_j \left(y_j^{\text{pred}} - \bar{y}^{\text{pred}}\right)^2 + 10^{-9}}},$$

where y_i and y_i^{pred} are the true and predicted values of the i^{th} sample, respectively, and $\bar{y}$ and $\bar{y}^{\text{pred}}$ are the mean of the true and predicted values, respectively. As defined in the toolkit, this score is equivalent to the square of the Pearson correlation, with a small correction to account for potentially constant targets or predictions. For comparison purposes, we also use this defintion.

We note that as used here and in other publications[2,5,6] to evaluate models with cross-validation or when applied test datasets, the R^2 score[7] is not the same as the coefficient of determination. Depending on the field and context, other names for this score include q^2_{ext},[8] Q^2,[9,10] Q^2_{F2},[11] and Q^2_{CV}.[12]

References

(1) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. Analyzing learned molecular representations for property prediction. *J. Chem. Inf. Model.* **2019**, *59*, 3370–3388.

(2) Elton, D. C.; Boukouvalas, Z.; Butrico, M. S.; Fuge, M. D.; Chung, P. W. Applying

machine learning techniques to predict the properties of energetic materials. *Sci. Rep.* **2018**, *8*, 9059.

(3) Huang, L.; Massa, L. Applications of energetic materials by a theoretical method (discover energetic materials by a theoretical method). *Int. J. Energ. Mater. Chem. Propul.* **2013**, *12*, 197–262.

(4) Bemis, G. W.; Murcko, M. A. The Properties of Known Drugs. 1. Molecular Frameworks. *J. Med. Chem.* **1996**, *39*, 2887–2893.

(5) Casey, A. D.; Son, S. F.; Bilionis, I.; Barnes, B. C. Prediction of Energetic Material Properties from Electronic Structure Using 3D Convolutional Neural Networks. *J. Chem. Inf. Model.* **2020**, *60*, 4457–4473.

(6) Barnes, B. C. Deep learning for energetic material detonation performance. *AIP Conf. Proc.* **2020**, *2272*, 070002.

(7) Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.

(8) Tropsha, A.; Gramatica, P.; Gombar, V. K. The Importance of Being Earnest: Validation is the Absolute Essential for Successful Application and Interpretation of QSPR Models. *QSAR Comb. Sci.* **2003**, *22*, 69–77.

(9) Wold, S.; Eriksson, L.; Clementi, S. *Chemometric Methods in Molecular Design*; John Wiley & Sons, Ltd, 1995; Chapter 5, pp 309–338.

(10) Barnes, B. C.; Elton, D. C.; Boukouvalas, Z.; Taylor, D. E.; Mattson, W. D.; Fuge, M. D.; Chung, P. W. Machine Learning of Energetic Material Properties. **2018**,

(11) Consonni, V.; Ballabio, D.; Todeschini, R. Evaluation of model predictive ability by external validation techniques. *J. Chemom.* **2010**, *24*, 194–201.

(12) Huang, Q.; Jin, H.; Liu, Q.; Wu, Q.; Kang, H.; Cao, Z.; Zhu, R. Proteochemometric Modeling of the Bioactivity Spectra of HIV-1 Protease Inhibitors by Introducing Protein-Ligand Interaction Fingerprint. *PLoS One* **2012**, *7*, 1–8.