SCIENTIFIC REPORTS

OPEN

Applying machine learning techniques to predict the properties of energetic materials

Received: 22 January 2018
Accepted: 1 June 2018
Published online: 13 June 2018

Daniel C. Elton, Zois Boukouvalas, Mark S. Butrico, Mark D. Fuge & Peter W. Chung

We present a proof of concept that machine learning techniques can be used to predict the properties of CNOHF energetic molecules from their molecular structures. We focus on a small but diverse dataset consisting of 109 molecular structures spread across ten compound classes. Up until now, candidate molecules for energetic materials have been screened using predictions from expensive quantum simulations and thermochemical codes. We present a comprehensive comparison of machine learning models and several molecular featurization methods - sum over bonds, custom descriptors, Coulomb matrices, Bag of Bonds, and fingerprints. The best featurization was sum over bonds (bond counting), and the best model was kernel ridge regression. Despite having a small data set, we obtain acceptable errors and Pearson correlations for the prediction of detonation pressure, detonation velocity, explosive energy, heat of formation, density, and other properties out of sample. By including another dataset with ≈300 additional molecules in our training we show how the error can be pushed lower, although the convergence with number of molecules is slow. Our work paves the way for future applications of machine learning in this domain, including automated lead generation and interpreting machine learning models to obtain novel chemical insights.

During the past few decades, enormous resources have been invested in research efforts to discover new energetic materials with improved performance, thermodynamic stability, and safety. A key goal of these efforts has been to find replacements for a handful of energetics which have been used almost exclusively in the world's arsenals since World War II - HMX, RDX, TNT, PETN, and TATB[1]. While hundreds of new energetic materials have been synthesized as a result of this research, many of which have remarkable properties, very few compounds have made it to industrial production. One exception is CL-20[2,3], the synthesis of which came about as a result of development effort that lasted about 15 years[1]. After its initial synthesis, the transition of CL-20 to industrial production took another 15 years[1]. This time scale (20–40 years) from the initial start of a materials research effort until the successful application of a novel material is typical of what has been found in materials research more broadly. Currently, the development of new materials requires expensive and time consuming synthesis and characterization loops, with many synthesis experiments leading to dead ends and/or yielding little useful information. Therefore, computational screening and lead generation is critical to speeding up the pace of materials development. Traditionally screening has been done using either ad-hoc rules of thumb, which are usually limited in their domain of applicability, or by running large numbers of expensive quantum chemistry calculations which require significant supercomputing time. Machine learning (ML) from data holds the promise of allowing for rapid screening of materials at much lower computational cost. A properly trained ML model can make useful predictions about the properties of a candidate material in milliseconds rather than hours or days[4].

Recently, machine learning has been shown to accelerate the discovery of new materials for dielectric polymers[5], OLED displays[6], and polymeric dispersants[7]. In the realm of molecules, ML has been applied successfully to the prediction of atomization energies[8], bond energies[9], dielectric breakdown strength in polymers[10], critical point properties of molecular liquids[11], and exciton dynamics in photosynthetic complexes[12]. In the materials science realm, ML has recently yielded predictions for dielectric polymers[5,10], superconducting materials[13], nickel-based superalloys[14], elpasolite crystals[15], perovskites[16], nanostructures[17], Heusler alloys[18], and the thermodynamic stabilities of half-Heusler compounds[19]. In the pharmaceutical realm the use of ML has a longer history than in other fields of materials development, having first been used under the moniker of quantitative

Department of Mechanical Engineering, University of Maryland, College Park, 20742, United States. Correspondence and requests for materials should be addressed to D.C.E. (email: delton@umd.edu) or P.W.C. (email: pchung15@umd.edu)

structure-property relationships (QSPR). It has been applied recently to predict properties of potential drug molecules such as rates of absorption, distribution, metabolism, and excretion (ADME)[20], toxicity[21], carcinogenicity[22], solubility, and binding affinity[23].

As a general means of developing models for relationships that have no known analytic forms, machine learning holds great promise for broad application. However, the evidence to date suggest machine learning models require data of sufficient quality, quantity, and diversity which ostensibly limits the application to fields in which datasets are large, organized, and/or plentiful. Published studies to date use relatively large datasets, with $N = 10{,}000–100{,}000$ being typical for molecular studies and larger datasets appearing in the realm of materials. When sufficiently large datasets are available, very impressive results can be obtained. For example, recently it has been shown that with enough data ($N = 117{,}000$[24] or $N = 435{,}000$[25]) machine learning can reproduce properties calculated from DFT with smaller deviations from DFT values than DFT's deviation from experiment[24,25].

Compared to other areas of materials research, the application of ML methods to energetic materials has received relatively little attention likely due to the scarcity of quality energetics data. Thus energetics may be a suitable platform to consider the factors that effect machine learning performance when limited to small data. While machine learning has been applied to impact sensitivity[26–29], there is little or no previously published work applying ML to predict energetic properties such as explosive energy, detonation velocity, and detonation pressure. While there is previous work applying ML to heat of formation[30,31] and detonation velocity & pressure[32–35], the studies are restricted to energetic materials in narrow domains, such as series of molecules with a common backbone. Furthermore, the aforementioned studies have been limited to the use of hand picked descriptors combined with linear modeling.

Therefore, in this work we wish to challenge the assumption that large data sets are necessary for ML to be useful by doing the first comprehensive comparison of ML methods to energetics data. We do this using a dataset of 109 energetic compounds computed by Huang & Massa[36]. While we later introduce additional data from Mathieu[37] for most of our work we restrict our study to the Huang & Massa data to demonstrate for the first time how well different ML models & featurizations work with small data. The Huang & Massa data contains molecules from ten distinct compound classes and models trained on it should be relatively general in their applicability. The diverse nature of the Huang & Massa data is important as ultimately we wish our models to be applicable to wide range of candidate energetic molecules.

To obtain the energetic properties in their dataset, Huang & Massa calculated gas phase heats of formation using density functional theory calculations at the B3LYP/6-31 G(d,p) level, and calculated the heat of sublimation using an empirical packing energy formula[38]. These two properties were used to calculate the heat of formation of the solid $\Delta H_f^{\text{solid}}$. They obtained densities primarily from experimental crystallographic databases. Huang & Massa then used their calculated heats of formation and densities as inputs to a thermochemistry code which calculates energetic properties under the Chapman-Jouguet theory of detonation. They validated the accuracy of their method with experimental data[38]. The result is a dataset with nine properties - density, heat of formation of the solid, explosive energy, shock velocity, particle velocity, sound velocity, detonation pressure, detonation temperature, and TNT equivalent per cubic centimeter. Several of these properties have significant correlations between them (a correlation matrix plot is given in Supplementary Fig. S1).

Featurization Methods

In the case of small data featurization is more critical than selecting a model, as with larger data models can learn to extract complex and/or latent features from a general purpose (materials agnostic) featurization. Feature vectors must be of reasonable dimensionality $d \ll N$, where d is the number of dimensions of the feature vector and N is the number of molecules in the training set, to avoid the curse of dimensionality and the so-called "peaking phenomena"[39]. Some featurizations make chemically useful information more transparent than others. More abstract general purpose featurizations, such as SMILES strings & Coulomb matrices (discussed below) only excel with large data and deep learning models, which can learn to extract the useful information. With small data, great gains in accuracy can sometimes be gained by hand selecting features using chemical intuition and domain expertise. For example, the number of azide groups in a molecule is known to increase energetic performance while also making the energetic material more sensitive to shock. While the number of azide groups is implicitly contained in the SMILES string and Coulomb matrix for the molecule, ML models such as neural networks typically need a lot of data to learn how to extract that information. To ensure that azide groups are being used by the model to make predictions with small data, an explicit feature corresponding to the number of such groups can be put in the feature vector.

Oxygen balance.

It is well known that the energy released during an explosion is largely due to the reaction of oxygen with the fuel atoms carbon and hydrogen. Based on this fact, Martin & Yallop (1957) found a linear relationship between detonation velocity and a descriptor called oxygen balance[40]. It can be defined either of two ways:

$$\text{OB}_{1600} \equiv \frac{1600}{m_{\text{mol}}}(n_{\text{O}} - 2n_{\text{C}} - n_{\text{H}}/2) \quad \text{OB}_{100} \equiv \frac{100}{n_{\text{atoms}}}(n_{\text{O}} - 2n_{\text{C}} - n_{\text{H}}/2) \tag{1}$$

here n_{C}, n_{H}, and n_{O} are the number of carbons, hydrogens, and oxygens respectively, m_{mol} is the molecular weight, and n_{atoms} is the number of atoms. An oxygen balance close to zero is sometimes used as requirement in determining if material may be useful as a novel energetic[41]. While it is easy to calculate and provides a useful rule of thumb, oxygen balance has limitations, which will become clear when compared to more sophisticated featurizations. One limitation of oxygen balance is that it neglects the large variation in bond strengths found in different molecules. It also neglects additional sources of energy released in an explosion, such as from nitrogen recombination (formation of N_2), halogen reactions, and the release of strain energy. Furthermore, oxygen balance is built on the assumption that oxygen reacts completely to form CO_2 and H_2O. More sophisticated calculations take into account the formation of CO (which commonly occurs at high temperatures or in low-oxygen compounds) as well as H_2 and trace amounts of unreacted O_2 and solid carbon which may appear. Predicting the proportion of such products requires complex thermochemical codes.

Custom descriptor set. By a "descriptor" we mean any function that maps a molecule to a scalar value. There are many types of descriptors, ranging from simple atom counts to complex quantum mechanical descriptors that describe electron charge distributions and other subtle properties. There are many different algorithms for generating a descriptor set of a specified size from a large pool. Since a brute force combinatorial search for the best set is often prohibitively expensive computationally, usually approximate methods are employed such as statistical tests, greedy forward elimination, greedy backwards elimination, genetic algorithms, etc. For instance, Fayet *et al.* used Student's *t*-test to rank and filter descriptors from a pool of 300 possible ones[30]. A pitfall common to the *t*-test and other statistical tests is that two descriptors that have low ranks on their own may be very useful when combined (eg. through multiplication, addition, subtraction, division). Given many of the difficulties in descriptor set selection, Guyon and Elisseeff recommend incorporating physical intuition and domain knowledge whenever possible[42].

We chose to design our descriptor set based on physical intuition and computational efficiency (we ignore descriptors which require physics computations). The first descriptor we chose was oxygen balance. Next we included the nitrogen/carbon ratio (n_N/n_C), a well known predictor of energetic performance[43]. Substituting nitrogens for carbon generally increases performance, since N=N bonds yield a larger heat of formation/enthalpy change during detonation compared to C-N and C=N bonds[43]. In addition to raw nitrogen content, the way the nitrogen is incorporated into the molecule is important. For instance, the substitution of N in place of C-H has the extra effect of increasing crystal density. Nitrogen catenation, both in N=N and N-N≡N (azide) is known to greatly increase performance but also increase sensitivity. On the other hand, nitrogen in the form of amino groups (NH_2) is known to decrease performance but also decrease sensitivity[43]. The way oxygen is incorporated into a molecule is similarly important - oxygen that is contained in nitro groups (NO_2) release much more energy during detonation than oxygen that is already bonded to a fuel atom. Martin & Yallop (1958) distinguish four oxygen types that are present in energetic materials[40]. Based on all of this, we distinguished different types of nitrogen, oxygen, and flourine based on their bond configurations:

$$\begin{array}{ll}
N{-}N{-}O_2(\text{nitrogen nitro group}) & \mathrm{C} = N{-}O(\text{fulminate group}) \\
C{-}N{-}O_2(\text{carbon nitro group}) & C{-}N = \mathrm{N}(\text{azo group}) \\
O{-}N{-}O_2(\text{oxygen nitro group}) & C{-}N{-}H_2(\text{amine group}) \\
O{-}N = \mathrm{O}(\text{nitrite group}) & C{-}N({-}O){-}C(N{-}\text{oxide nitrogen}) \\
\mathrm{C} = N{-}F(\text{nitrogen}{-}\text{flourine group}) & C{-}F(\text{carbon}{-}\text{flourine group}) \\
C{-}O{-}H(\text{hydroxyl oxygen}) & \mathrm{N} = \mathrm{O}(\text{nitrate or nitrite oxygen}), \\
N{-}O{-}C(N{-}\text{oxide oxygen}) & \mathrm{C} = \mathrm{O}(\text{keton/carboxyl oxygen})
\end{array}$$

We also included raw counts of carbon and nitrogen, hydrogen, and fluorine. We tested using ratios instead of raw counts (n_x/n_{total}) but found this did not improve the performance of the featurization. All together, our custom descriptor set feature vector is:

$$\begin{aligned}\boldsymbol{x}_{\mathrm{CDS}} = \; & [\mathrm{OB}_{100},\, n_{\mathrm{N}}/n_{\mathrm{C}},\, n_{\mathrm{NNO2}},\, n_{\mathrm{CNO2}},\, n_{\mathrm{ONO2}},\, n_{\mathrm{ONO}},\, n_{\mathrm{CNO}},\, n_{\mathrm{CNN}},\, n_{\mathrm{NNN}},\, n_{\mathrm{CNH2}}, \\ & n_{\mathrm{CN(O)C}},\, n_{\mathrm{CNF}},\, n_{\mathrm{CF}},\, n_{\mathrm{C}},\, n_{\mathrm{N}},\, n_{\mathrm{NO}},\, n_{\mathrm{COH}},\, n_{\mathrm{NOC}},\, n_{\mathrm{CO}},\, n_{\mathrm{H}},\, n_{\mathrm{F}},]\end{aligned} \tag{2}$$

Sum over bonds. Based on the intuition that almost all of the latent heat energy is stored in chemical bonds, we introduce a bond counts feature vector. The bond counts vectors are generated by first enumerating all of the bond types in the dataset and then counting how many of each bond are present in each molecule. There are 20 bond types in the Huang & Massa dataset:

N-O, N:O, N-N, N=O, N=N, N:N, N#N, C-N, C-C, C-H, C:N, C:C, C-F, C-O, C=O, C=N, C=C, H-O, H-N, F-N.

We use the SMARTS nomenclature for bond primitives ('-' for single bond, '=' for double bond, '#' for triple bond, and ':' for aromatic bond). Following Hansen, *et al.*[44] we call the resulting feature vector "sum over bonds".

Coulomb matrices. An alternative way of representing a molecule is by the Coulomb matrix featurization introduced by Rupp *et al.*[8,45]. The Coulomb matrix finds its inspiration in the fact that (in principle) molecular properties can be calculated from the Schrödinger equation, which takes the Hamiltonian operator as its input. While extensions of the Coulomb matrix for crystals have been developed[46], it is designed for treating gas phase molecules. The Hamiltonian operator for an isolated molecule can be uniquely specified by the nuclear coordinates $\boldsymbol{R}_i$ and nuclear charges Z_i. Likewise, the Coulomb matrix is completely specified by $\{\boldsymbol{R}_i, Z_i\}$. The entries of the Coulomb matrix $\mathbf{M}$ for a given molecule are computed as:

$$\mathbf{M}_{ij} = \begin{cases} 0.5Z_i^{2.4}, & i = j \\ \dfrac{Z_iZ_j}{|\mathbf{R}_i - \mathbf{R}_j|} & i \neq j \end{cases}, \tag{3}$$

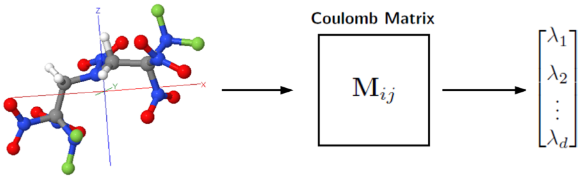
Coulomb Matrix
M_{ij}
$\begin{bmatrix}\lambda_1\\ \lambda_2\\ \vdots\\ \lambda_d\end{bmatrix}$

Figure 1. Transformation of (x, y, z) coordinates and nuclear charges to the Coulomb matrix eigenvalue spectra representation of a molecule.

The diagonal elements of **M** correspond to a polynomial fit of the potential energies of isolated atoms, while the off-diagonals elements correspond to the energy of Coulombic repulsion between different pairs of nuclei in the molecule. It is clear by construction that **M** is invariant under translations and rotations of the molecule. However, Coulomb matrices are not invariant under random permutations of the atom's indices. To avoid this issue, one can use the eigenvalue spectrum of the Coulomb matrix since the eigenvalues of a matrix are invariant under permutation of columns or rows. In this approach, the Coulomb matrix is replaced by a feature vector of the eigenvalues, $\langle\lambda_1, \cdots, \lambda_d\rangle$, sorted in a descending order, as illustrated in Fig. 1.

To obtain fixed length feature vectors, the size is of the vectors is set at the number of atoms in the largest molecule in the dataset $d = d_{\max}$ and the feature vectors for molecules with number of atoms less than $d_{\max}$ are padded with zeros. Using the eigenvalues implies a loss of information from the full matrix. Given this fact, we also compare with the "raw" Coulomb matrix. Since it is symmetric, we take only the elements from the diagonal and upper triangular part of the matrix and then put them into a feature vector (which we call "Coulomb matrices as vec"). The "raw" Coulomb matrix is quite a bit different than other feature vectors as the physical meaning of each specific element in the feature vector differs between molecules. This appears to be problematic, especially for kernel ridge regression (KRR) and the other variants on linear regression, which treat each element separately and for which it is usually assumed that each element has the same meaning in every sample.

Bag of bonds. The bag of bonds featurization was introduced by Hansen *et al*. in 2015[44]. It is inspired by the "bag of words" featurization used in natural language processing. In bag of words, a body of text is featurized into a histogram vector where each element, called a "bag", counts the number of times a particular word appears. Bag of bonds follows a similar approach by having "bags" that correspond to different types of bonds (such as C-O, C-H, etc). Bonds are distinguished by the atoms involved and the order of the bond (single, double, triple). However bag of bonds differs from bag of words in several crucial ways. First, each "bag" is actually a vector where each element is computed as $Z_iZ_j/|\mathbf{R_i} - \boldsymbol{R}_\mathbf{j}|$. The bag vectors between molecules are enforced to have a fixed length by padding them with zeros. The entries in each bag vector are sorted by magnitude from highest to lowest to ensure a unique representation. Finally, all of the bag vectors are concatenated into a final feature vector.

Fingerprinting. Molecular fingerprints were originally created to solve the problem of identifying isomers[47], and later found to be useful for rapid substructure searching and the calculation of molecular similarity in large molecular databases. In the past two decades, fingerprints have been used as an alternative to descriptors for QSPR studies. Fingerprinting algorithms transform the molecular graph into a vector populated with bits or integers. In this work we compare several fingerprints found in RDKit, a popular cheminformatics package–Atom-Pair[48], Topological Torsion[49], Extended Connectivity Fingerprints (ECFPs)[50], E-state fingerprints[51], Avalon fingerprints[52], RDKit graph fingerprints[53], ErG fingerprints[54], and physiochemical property fingerprints[55].

A few general features of fingerprints can be noted. First of all, all the fingerprinting algorithms start with atom level descriptors, each of which encode atom-specific information into an integer or real number. The simplest atom descriptor is the atomic number, but in most fingerprints this is augmented with additional information that describes the local environment of the atom. A second general point is that some fingerprinting algorithms can generate either integer vectors or bit vectors. In extended connectivity fingerprints (ECFPs) for instance, count information on the number of times a feature/fragment appears is lost when bit vectors are used. In RDKit, fingerprints return bit vectors by default. The length of bit vector based fingerprint representations can be tuned (to avoid the curse of dimensionality) through a process called folding. Another point is that the fingerprints we study contain only 2D graph information, and do not contain information about 3D structure/conformation. Comparisons between 2D fingerprints and analogous 3D fingerprints have found that 3D fingerprints generally do not yield superior performance for similarity searching[56], or binding target prediction[57], although they are useful for certain pharmaceutical applications[58].

Other featurizations. Two other featurizations that have been developed for molecules are smooth overlap of atomic positions (SOAP)[59,60], and Fourier series of atomic radial distribution functions[61]. We choose not to investigate these featurizations due to their poorer performance in past comparisons. Another featurization strategy, the random walk graph kernel, may be promising for future exploration but is computationally expensive to properly implement[62]. Very recently, several custom deep learning architectures with built-in featurization have been developed - custom graph convolutional fingerprints[63,64], deep tensor networks[65], message passing neural networks[4], and hierarchically interacting particle neural nets[66]. We attempted to train a graph convolutional fingerprint using the "neural fingerprint" code of Duvenaud *et al.*[63] but were not able achieve accuracy that was competitive with any of the other featurizations (for example the best we achieved for shock velocity was a MAE

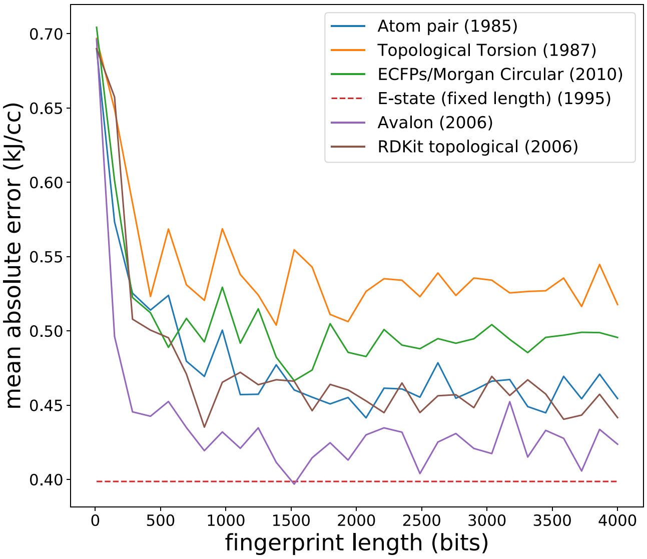

Figure 2. Mean errors in explosive energy obtained with different fingerprinting methods at different fingerprint lengths, using Bayesian ridge regression and leave-5-out cross validation. The E-state fingerprint has a fixed length and therefore appears as a flat line.

of ≈0.35 km/s as opposed to 0.30 km/s for sum over bonds + KRR). Additional data and hyperparameter optimization would likely improve the competitiveness of custom convolutional fingerprinting.

Results

Comparison of fingerprints.

Figure 2 shows fingerprint mean absolute error of explosive energy in out-of-sample test data using kernel ridge regression vs. the fingerprint length in bits. The E-state fingerprint, which has a fixed length, performs the best, with the Avalon fingerprint nearly matching it. The Estate fingerprint is based on electrotopological state (E-state) indices[67], which encode information about associated functional group, graph topology, and the Kier-Hall electronegativity of each atom. E-state indices were previously found to be useful for predicting the sensitivity (h_{50} values) of energetic materials with a neural network[68]. The E-state fingerprint differs from all of the other fingerprints we tested as it is fixed length, containing a vector with counts of 79 E-state atom types. Thus, technically it is more akin to a descriptor set than a fingerprint. We found that only 32 of the atom types were present in the energetic materials we studied (types of C,N,H,O,F), so we truncated it to a length of 32 (ie. threw out the elements that were always zero). The E-state fingerprint can also be calculated as a real-valued vector which sums the E-state indices for each atom, however we found the predictive performance was exactly the same as with the count vector version. It is not immediately clear why E-state performs better than the other fingerprints, but we hypothesize that it is due to the fact that the E-state atom types are differentiated by valence state and bonding pattern, including specifically whether a bond to a hydrogen is present. Thus a NH_2 nitrogen is considered a specific atom type. In the atom-pair, topological torsion, Morgan, and RDKit topological fingerprint the atom types are differentiated soley by atomic number, number of heavy atom neighbors, and number of pi electrons. The Avalon fingerprint, which does almost as well as E-state, likely does so because it contains a large amount of information including atom pairs, atom triplets, graph topology, and bonding configurations.

Comparison of featurizations.

Table 1 shows a comparison of the featurization schemes discussed in the previous sections, using kernel ridge regression with hyperparameter optimization performed separately via grid search with cross validation for each featurization. Hyperparameter optimization was done for all hyperparameters - the regularization parameter α, kernel width parameter and kernel type (Laplacian/L1 vs Gaussian/L2). Fairly dramatic differences are observed in the usefulness of different featurizations. The sum over bonds featurization always performs the best, but additional gains can be made by concatenating featurizations. The gains from concatenation are especially reflected in the correlation coefficient r, which increases from 0.65 to 0.78 after E-state and the custom descriptor set is concatenated with sum over bonds. As expected, the two different oxygen balance formulae perform nearly the same, and Coulomb matrix eigenvalues perform better than the raw Coulomb matrix. Summed bag of bonds (summing each bag vector) performs nearly as well as traditional bag of bonds, but with a much more efficient representation ($\boldsymbol{x} \in \mathbb{R}^{20}$vs $\boldsymbol{x} \in \mathbb{R}^{2527}$).

Comparison of machine learning models.

Table 2 presents a comparison of five different ML models & seven featurization methods for each target property in the Huang & Massa dataset (a complete comparison with 5 additional models and additional evaluation metrics can be found in Supplementary Tables S1, S2, S3, and S4 and in Supplementary Fig. S2). The mean average error was averaged over 20 random train-test splits (with replacement) with a train/test ratio of 4:1. We also calculated 95% confidence intervals (shown in Supplementary Table S) which show the values are well converged and meaningful comparisons can be made. Hyperparameter optimization was performed on all models using grid search. We found that careful hyperparameter optimization,

name	$\mathrm{MAE}_{\mathrm{train}}$	$\mathrm{MAE}_{\mathrm{test}}$	$\mathrm{MAPE}_{\mathrm{test}}$	R^2_{train}	R^2_{test}	$\mathrm{r}_{\mathrm{train}}$	$\mathrm{r}_{\mathrm{test}}$
E-state + CDS + SoB	0.244	0.334	8.93	0.88	0.76	0.88	0.79
CDS + SoB	0.247	0.335	9.32	0.88	0.75	0.88	0.79
E-state + custom descriptor set	0.224	0.345	9.50	0.89	0.75	0.90	0.79
SoB + OB100	0.256	0.358	10.50	0.87	0.61	0.87	0.70
sum over bonds (SoB)	0.280	0.379	10.69	0.84	0.67	0.84	0.71
truncated E-state	0.260	0.414	12.65	0.85	0.66	0.85	0.70
custom descriptor set (CDS)	0.398	0.432	12.92	0.68	0.57	0.68	0.63
Bag of Bonds (BoB)	0.213	0.467	12.60	0.89	0.54	0.90	0.60
Oxygen balance$_{1600}$	0.419	0.489	15.66	0.67	0.41	0.68	0.56
Summed Bag of Bonds	0.262	0.493	13.63	0.85	0.18	0.85	0.56
Coulomb matrix eigenvalues	0.314	0.536	15.73	0.81	0.37	0.82	0.48
Oxygen balance$_{100}$	0.444	0.543	17.46	0.59	0.44	0.62	0.57
Coulomb matrices as vec	0.395	0.672	21.86	0.57	0.05	0.67	0.20

Table 1. Detailed comparison of 13 different featurization schemes for prediction of explosive energy with kernel ridge regression, ranked by $\mathrm{MAE}_{\mathrm{test}}$. These variance in MAEs between folds was less than 0.01 in all cases. Hyperparameter optimization was used throughout with nested 5-fold cross validation. The metrics are averaged over 20 train-test sets using shuffle split with 80/20 splitting.

especially of the regularization parameter, is critical to properly comparing and evaluating models. We found that LASSO regression and Bayesian ridge regression performed nearly as well as ridge regression, and gradient boosted trees performed nearly as well as random forest, so we omitted them from the table. Two key observations can be made. The first is that the sum over bonds featurization is the best for all target properties with the exception of the speed of sound, where bag of bonds does slightly better. The second is that kernel ridge and ridge regression performed best. Other models might become competitive with the addition of more data. The gap between the train and test MAEs indicates overfitting is present.

Incorporating dimensionality reduction. Dimensionality reduction can often improve model performance, especially when $n_{\mathrm{features}} \approx n_{\mathrm{examples}}$[39]. Our first test was done with Coulomb matrices, where we found the error converged with $D = 15$ or more principle components. A similar convergence at $D = 15$ was observed for E-state and for the combined featurization Estate + SoB + CDS. We also experimented with some other dimensionality reduction techniques such as t-SNE, PCA followed by fast independent component analysis (ICA), and spectral embedding (Laplacian eigenmaps). In all cases, however, the error was not improved by dimensionality reduction (see supplementary information for more detail). This indicates that while our feature vectors could be compressed without loosing accuracy, the regularization in our models is capable of handling the full dimensionality of our feature vectors without loss in performance.

Analysis and Discussion

Machine learning performance vs. training data quantity and diversity. To compare our diverse-data case with how machine learning works in a narrow, non-diverse context we fit the dataset of Ravi *et al.* which contains 25 pyrazole-based molecules (Table 3). Ravi *et al.* performed DFT calculations to obtain the heat of formation with respect to products and then used the Kamlet-Jacobs equations to predict detonation velocity and detonation pressure. Table 3 shows how very high accuracy in predicting energetic properties is achieved. Not surprisingly, the custom descriptor set slightly outperforms other featurizations here, since the only thing that is changing within the dataset is the type and number of functional groups attached to the pyrazole backbone. While accurate, the trained models will not generalize beyond the class of molecules they were trained since they cannot capture significant differences between classes[35]. Similarly, insights gained from the interpretation of such models may not generalize and may be contaminated by spurious correlations that often occur in small datasets.

The part of chemical space where a model yields accurate predictions is known as the applicability domain[69]. The concept of applicability domain is inherently subjective as it depends on a choice of accuracy one considers acceptable. Mapping the applicability domain of a model by generating additional test data is typically expensive given the high dimensionality of chemical space. There are several heuristics for estimating an outer limit to the applicability domain of a model, such as using the convex hull of the training data, a principle components based bounding box, or a cutoff based on statistical leverage[69]. While useful, applicability domains determined from such methods all suffer from the possibility that there may be holes inside where the model performs badly. To spot holes, distance based methods can be used, which average the distance to the *k* nearest neighbors in the training data under a given metric and apply a cutoff. There is no universal way of determining a good cutoff, however, so determining a practical cutoff requires trial and error.

Residual analysis. Useful insights into the applicability of our models can be gained by looking at a model's residuals ($y_{\mathrm{pred}} - y_{\mathrm{true}}$) in the test set. We chose to look at residuals in leave-one-out cross validation using the sum over bonds featurization and kernel ridge regression (Fig. 3). We suspected that our model may have difficulty predicting the explosive energy for cubane-derived molecules, since they release a large amount of strain energy

		ρ, $\frac{g}{cc}$	ΔH_f^s, $\frac{kJ}{mol}$	E_e, $\frac{kJ}{cc}$	V_s, $\frac{km}{s}$	V_p, $\frac{km}{s}$	V_{snd}, $\frac{km}{s}$	**P, GPa**	**T, K**	$\frac{TNT_{equiv}}{cc}$
KRR	Estate	0.10	261.02	0.63	0.48	0.13	0.41	4.95	500.19	0.18
	CDS	0.08	198.81	0.50	0.44	0.11	0.37	3.07	462.63	0.17
	SoB	0.07	68.73	0.40	**0.31**	**0.09**	**0.25**	2.90	331.36	**0.11**
	CM eigs	0.09	288.41	0.67	0.67	0.18	0.61	5.67	600.08	0.22
	Bag of Bonds	**0.06**	166.66	0.47	0.33	0.11	0.29	3.38	478.93	0.18
	Estate + CDS + SoB	**0.06**	**71.40**	**0.36**	0.32	0.10	0.29	2.76	359.66	0.13
Ridge	Estate	0.09	269.11	0.58	0.57	0.14	0.45	4.71	491.21	0.19
	CDS	0.07	193.19	0.43	0.39	0.11	0.33	3.23	438.27	0.17
	SoB	**0.06**	82.00	0.37	0.32	0.10	0.29	3.01	**327.43**	**0.11**
	CM eigs	0.09	355.12	0.79	0.60	0.16	0.55	5.82	590.69	0.19
	Bag of Bonds	0.06	163.76	0.48	0.32	0.11	0.31	3.37	472.93	0.19
	Estate + CDS + SoB	0.06	77.31	0.39	0.32	0.10	0.28	2.78	383.07	0.13
SVR	Estate	0.09	207.78	0.60	0.45	0.13	0.35	4.41	476.06	0.17
	CDS	0.07	223.24	0.52	0.34	0.12	0.32	3.21	436.81	0.18
	SoB	0.06	130.78	0.40	**0.31**	0.10	0.28	2.97	331.27	0.14
	CM eigs	0.08	288.41	0.55	0.60	0.15	0.53	4.54	584.44	0.21
	Bag of Bonds	0.07	159.24	0.47	0.35	0.12	0.28	3.34	385.59	0.18
	Estate + CDS + SoB	0.06	129.89	0.37	0.34	0.10	0.28	**2.73**	353.18	0.13
RF	Estate	0.09	252.74	0.59	0.50	0.14	0.39	4.09	488.98	0.19
	CDS	0.07	241.67	0.46	0.36	0.11	0.29	3.34	435.77	0.16
	SoB	0.07	136.91	0.48	0.40	0.12	0.30	3.47	417.46	0.15
	CM eigs	0.09	286.89	0.67	0.62	0.15	0.51	5.52	512.22	0.20
	Bag of Bonds	0.07	172.41	0.46	0.36	0.10	0.29	3.10	418.35	0.16
	Estate + CDS + SoB	0.07	144.18	0.43	0.34	**0.09**	0.26	3.11	401.27	0.15
kNN	Estate	0.08	236.55	0.61	0.49	0.15	0.41	4.30	563.89	0.20
	CDS	0.07	242.99	0.55	0.39	0.13	0.33	3.56	478.50	0.18
	SoB	0.08	184.43	0.54	0.44	0.12	0.36	3.65	427.20	0.17
	CM eigs	0.10	343.48	0.62	0.67	0.15	0.51	5.52	570.55	0.22
	Bag of Bonds	0.08	238.05	0.53	0.40	0.11	0.32	3.58	515.25	0.19
	Estate + CDS + SoB	0.08	171.65	0.54	0.43	0.12	0.35	3.57	442.14	0.17
mean	n/a	0.11	309.75	0.69	0.65	0.15	0.55	4.88	629.20	0.22
		1.86	0.50	3.93	8.47	2.04	6.43	32.13	3568.65	1.43

Table 2. Average mean absolute errors (MAEs) in the test sets for different combinations of target property, model and featurization. Hyperparameter optimization was used throughout with nested 5-fold cross validation. The test MAEs are averaged over 20 test sets using shuffle split with 80/20 splitting. The properties are density, heat of formation of the solid, explosive energy, shock velocity, particle velocity, sound velocity, detonation pressure, detonation temperature, and TNT equivalent per cubic centimeter. The models are kernel ridge regression (KRR), ridge regression (Ridge), support vector regression (SVR), random forest (RF), *k*-nearest neighbors (kNN), and a take-the-mean dummy predictor. The last row gives the average value for each property in the dataset.

		V_{det} (km/s)	**ρ (g/cc)**	**P_{det} (GPa)**
KRR	Estate	0.12, 0.99	0.04, 0.98	199, 0.92
	CDS	**0.07, 0.99**	0.03, 0.99	1.10, 0.99
	SoB	0.08, 0.99	0.03, 0.99	**0.83, 0.99**
Ridge	Estate	0.32, 0.91	0.03, 0.98	2.48, 0.98
	CDS	0.14, 0.99	**0.02, 0.99**	**1.34, 0.99**
	SoB	0.44, 0.86	0.03, 0.99	2.92, 0.96
mean	n/a	1.25, 0.00	0.27, 0.00	12.90, 0.00

Table 3. Mean absolute errors and Pearson correlation coefficients for ML on the dataset of Ravi *et al.*[34], which contains 25 nitropyrazole molecules. 5-fold cross validation was used, so $N_{train} = 20$ and $N_{test} = 5$.

in explosion - something which is not significant in other molecules except for the CL20 group, and which is not explicitly contained in any of our featurizations. In fact, the three worst performing molecules were heptanitrocubane, the linear molecule QQQBRD02, and cubane. The model overestimates the energy of cubane and underestimates the explosive energy of nitrocubane. The mean absolute errors and average residuals of different groups

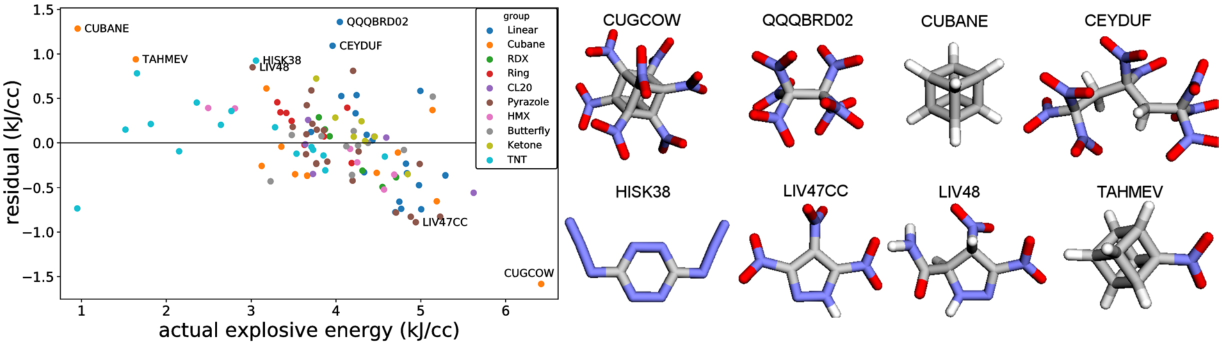

Figure 3. Residuals in leave-one-out cross validation with kernel ridge regression and the sum over bonds featurization (left), and some of the worst performing molecules (right).

group	N_{group}	*r*	R^2	MAE	avg. residual (kJ/cc)
HMX	6	0.968	0.83	0.32	−0.06
Butterfly	10	0.917	0.78	0.18	−0.01
TNT	16	0.854	0.83	0.31	0.10
CL20	6	0.854	0.83	0.22	−0.09
Cubane	12	0.814	0.75	0.58	−0.04
Ring	8	0.548	0.17	0.26	0.20
RDX	6	0.377	0.19	0.28	−0.11
Pyrazole	20	0.254	0.21	0.42	−0.07
Ketone	7	0.099	−0.13	0.25	0.15
Linear	18	0.003	−1.12	0.52	0.00

Table 4. The mean absolute error, Pearson correlation and average residual in different groups, for prediction of explosive energy (kJ/cc) with leave-one-out CV on the entire dataset using sum over bonds and kernel ridge regression. The groups are sorted by *r* value rather than MAE since the average explosive energy differs significantly between groups.

in leave-one-out CV are shown in Table 4. In order to visualize the distribution of data in high dimensional space one can utilize various embedding techniques that can embed high dimensional data into two dimensions. Often one finds that high dimensional data lies close to a lower dimensional manifold. The embeddings shown in the supplementary information (t-SNE, PCA, & spectral) show that the cubane molecules are quite separated from the rest of the data (Supplementary Information Fig. S3). Based on this observation, we tried removing the cubanes. We found that the accuracy of prediction of explosive energy with kernel ridge regression and sum over bonds remained constant (MAE = 0.36 kJ/cc) while the *r* value actually decreased significantly from 0.76 to 0.68.

Learning curves. Insights into the data-dependence of ML modeling can be obtained from plots of cross-validated test error vs number of training examples, which are known as learning curves. While we were able to obtain good learning curves from just the Huang & Massa dataset, to ensure their accuracy we supplemented the Huang & Massa data with 309 additional molecules from the dataset given in the supplementary information of Mathieu *et al*., which includes detonation velocity and detonation pressure values calculated from the Kamlet-Jacobs equations[37]. A few of the molecules are found in both datasets, but most are unique, yielding a total of ≈400 unique molecules. We assume detonation velocity is equivalent to the shock velocity found in the Huang & Massa data. The method of calculation of properties differs between the two datasets, possibly introducing differing bias between the sets, so we shuffle the data beforehand. Figure 4 shows the learning curves for detonation velocity and detonation pressure. The gap between training and test score curves indicates error from overfitting (variance) while the height of the curves indicates the degree of error from the choice of model (bias). As the quantity of data increases, the gap between the training and test curves should decrease. The flattening out of the training accuracy curve indicates a floor in accuracy for the given choice of model & featurization. Even with much more data, it is very likely that creating predictions more accurate than such a floor would require better choices of model and featurization. Empirically, learning curves are known to have a $AN_{\text{train}}^{-\beta}$ dependence[70]. In the training of neural networks typically $1 < \beta < 2$[71], while we found values between $0.15 - 0.30$ for the properties studied. Since different types of models can have different learning curves, we also looked at random forest, where we found similar plots with $\beta \approx 0.20$.

Conclusion and Future Directions

We have presented evidence that machine learning can be used to predict energetic properties out of sample after being trained on a small yet diverse set of training examples ($N_{\text{train}} = 87$, $N_{\text{test}} = 22$). For all the properties tested,

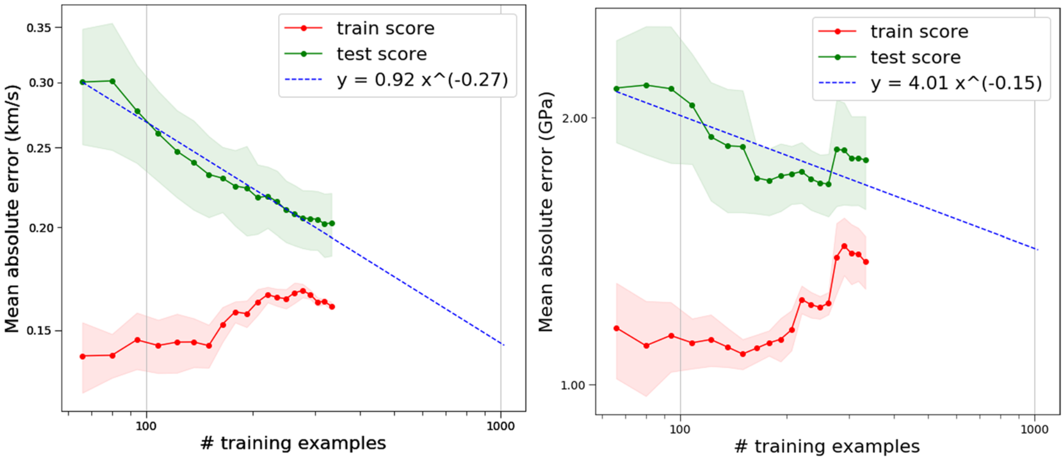

Figure 4. The learning curves for predicting detonation velocity (left) and detonation pressure (right) for the combined ($N = 418$) dataset plotted on a log-log plot. Shaded regions show the standard deviation of the error in 5-fold cross validation.

either kernel ridge or ridge regression were found to have the highest accuracy, and out of the base featurizations we compared the sum over bonds featurization performed best. Small improvements could be gleaned by concatenating featurizations. The best *r* values achieved were 0.94 for heat of formation, 0.74 for density, 0.79 for explosive energy, and 0.78 for shock velocity. With kernel ridge regression and sum over bonds we obtained mean percentage errors of 11% for explosive energy, 4% for density, 4% for detonation velocity and 11% for detonation pressure. By including ≈300 additional molecules in our training we showed how the mean absolute errors can be pushed lower, although the convergence with number of molecules is slow.

There are many possible future avenues of research in the application of machine learning methods to the discovery of new energetic materials. One possible way to overcome limitations of small data is with transfer learning[23,72]. In transfer learning, a model is first trained for a task where large amounts of data is available, and then the model's internal representation serves as a starting point for a prediction task with small data. Another way our modeling might be improved without the need for more data is by including in the training set CNOHF type molecules that are non-energetic, which may require overcoming sampling bias by reweighting the energetic molecules. Additional work we have undertaken investigates how our models can be interpreted to illuminate structure-property relationships which may be useful for guiding the design of new energetic molecules[73]. Finally, a promising future direction of research involves coupling the property predicting models developed here with generative models such as variational autoencoders or generative adversarial networks to allow for molecular generation and optimization.

Methods

We used in-house Python code for featurization and the *scikit-learn* package (http://scikit-learn.org/http://scikit-learn.org/) for machine learning. Parts of our code has been used to establish a library we call the Molecular Machine Learning (MML) toolkit. The MML toolkit is open source and available online at https://github.com/delton137/mmltoolkit.

Model evaluation. There are many different ways to evaluate model performance. The simplest scoring function is the mean absolute error (MAE):

$$\text{MAE} = \frac{1}{N}\sum_{i}^{N}|y_i^{\text{true}} - y_i^{\text{pred}}| \tag{4}$$

A related scoring function is the root mean squared error (RMSE), also called the standard error of prediction (SEP), which is more sensitive to outliers than MAE:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i}(y_i^{\text{true}} - y_i^{\text{pred}})^2} \tag{5}$$

We also report the mean average percent error (MAPE), although this score is often misleadingly inflated when y_i^{true} is small:

$$\text{MAPE} = \frac{100}{N}\sum_{i}^{N}\left|\frac{y_i^{\text{true}} - y_i^{\text{pred}}}{y_i^{\text{true}}}\right| \tag{6}$$

It is also important to look at the Pearson correlation coefficient r:

$$r = \frac{\sum_i (y_i^{\text{true}} - \overline{y}_i^{\text{true}})(y_i^{\text{pred}} - \overline{y}_i^{\text{pred}})}{\sqrt{\sum_i (y_i^{\text{true}} - \overline{y}_i^{\text{true}})^2 \sum_i (y_i^{\text{pred}} - \overline{y}_i^{\text{pred}})^2}} \tag{7}$$

Additionally, we report the coefficient of determination:

$$R^2 = 1 - \frac{\sum_i (y_i^{\text{true}} - y_i^{\text{pred}})^2}{\sum_i (y_i^{\text{true}} - \overline{y}_i^{\text{true}})^2} \tag{8}$$

Unlike the Pearson correlation coefficient r, R^2 can assume any value between $-\infty$ and 1. When reported for test or validation data, this scoring function is often called Q^2. While a bad Q^2 indicates a bad model, a good Q^2 does not necessarily indicate a good model, and models should not be evaluated using Q^2 in isolation[74].

Data gathering. With a the exception of the Coulomb-matrix and bag-of-bonds featurizations, the only input required for our machine learning featurizations is the molecular connectivity graph. For easy encoding of the molecular graph we used Simplified Molecular-Input Line-Entry System (SMILES) strings[75]. SMILES strings are a non-unique representation which encode the molecular graph into a string of ASCII characters. SMILES strings for 56 of the Huang & Massa molecules were obtained from the Cambridge Structure Database Python API[76], and the rest were generated by hand using a combination of the Optical Structure Recognition Application[77], the www.molview.orgwww.molview.org molecule builder, and the Open Babel package[78], which can convert.mol files to SMILES. Since Coulomb matrices require atomic coordinates, 3D coordinates for the molecules were generated using 2D → 3D structure generation routines in the RDKit[53] and Open Babel[78] python packages. After importing the SMILES into RDKit and adding hydrogens to them, an embedding into three dimensions was done using distance geometry, followed by a conjugate gradient energy optimization. Next, a weighted rotor search with short conjugate gradient minimizations was performed using Open Babel to find the lowest energy conformer. Finally, a longer conjugate gradient optimization was done on the lowest energy conformer. For one molecule (the cubane variant 'EAT07') we used the *obgen* utility program of Open Babel to do the coordinate embedding (in retrospect, *obgen* could have been used to yield good enough results for all molecules). All energy minimizations were done with the MMFF94 forcefield[79]. The accuracy of the generated structures was verified by visually comparing the generated structures of 56 of the compounds to x-ray crystallographic coordinates obtained from the Cambridge Structure Database.

Data Availability. The compiled property data from Huang & Massa (2010) and Mathieu (2017) and the generated SMILES strings are included in the Supplementary Information. The full Mathieu (2017) dataset is available free of charge on the ACS Publications website at : DOI:10.1021/acs.iecr.7b02021 Python Jupyter notebooks for reproducing all of our results have been open sourced and are available at https://github.com/delton137/Machine-Learning-Energetic-Molecules-Notebooks. The Python notebooks use the open source Molecular Machine Learning toolkit developed in conjunction with this work, which can be found at https://github.com/delton137/mmltoolkit.

References

1. National Research Council, Division on Engineering and Physical Sciences. Advanced Energetic Materials (National Academies Press, 2004).
2. Nielsen, A. T. *et al*. Synthesis of polyazapolycyclic caged polynitramines. *Tetrahedron* **54**, 11793–11812 (1998).
3. Viswanath, D. S., Ghosh, T. K. & Boddu, V. M. Hexanitrohexaazaisowurtzitane (HNIW, CL-20), 59-100 (Springer Netherlands, Dordrecht, 2018).
4. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry. *arXiv e-prints* 1704.01212 (2017).
5. Mannodi-Kanakkithodi, A., Pilania, G., Huan, T. D., Lookman, T. & Ramprasad, R. Machine learning strategy for accelerated design of polymer dielectrics. *Sci. Rep.* **6**, 20952 (2016).
6. Gomez-Bombarelli, R. *et al*. Design of efficient molecular organic light-emitting diodes by a high-throughput virtual screening and experimental approach. *Nat. Mat.* (2016).
7. Menon, A. *et al*. Elucidating multi-physics interactions in suspensions for the design of polymeric dispersants: a hierarchical machine learning approach. Mol. S st. *Des. Eng.* (2017).
8. Rupp, M., Tkatchenko, A., Muller, K.-R. & von Lilienfeld, O. A. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.* **108**, 058301 (2012).
9. Yao, K., Herr, J. E., Brown, S. N. & Parkhill, J. Intrinsic bond energies from a bonds-in-molecules neural network. *The J. Phys. Chem. Lett.* **8**, 2689–2694 (2017).
10. Pilania, G., Wang, C., Jiang, X., Rajasekaran, S. & Ramprasad, R. Accelerating materials property predictions using machine learning. *Sci. Rep.* **3**, 2810 (2013).
11. Carande, W. H., Kazakov, A., Muzny, C. & Frenkel, M. Quantitative structure-property relationship predictions of critical properties and acentric factors for pure compounds. *J. Chem. Eng. Data* **60**, 1377–1387 (2015).
12. Hase, F., Valleau, S., Pyzer-Knapp, E. & Aspuru-Guzik, A. Machine learning exciton dynamics. *Chem. Sci.* **7**, 5139–5147 (2016).
13. Stanev, V. *et al*. Machine learning modeling of superconducting critical temperature. *arXiv e-prints* 1709.02727 (2017).
14. Conduit, B., Jones, N., Stone, H. & Conduit, G. Design of a nickel-base superalloy using a neural network. *Mater. Des.* **131**, 358–365 (2017).
15. Faber, F. A., Lindmaa, A., von Lilienfeld, O. A. & Armiento, R. Machine learning energies of 2 million elpasolite (abC_2D_6) crystals. *Phys. Rev. Lett.* **117**, 135502 (2016).
16. Schmidt, J. *et al*. Predicting the thermodynamic stability of solids combining density functional theory and machine learning. *Chem. Mater.* **29**, 5090–5103 (2017).
17. Ju, S. *et al*. Designing nanostructures for phonon transport via bayesian optimization. *Phys. Rev. X* **7**, 021024 (2017).

18. Sanvito, S. *et al*. Accelerated discovery of new magnets in the heusler alloy family. Sci. Adv. 3 (2017).
19. Legrain, F., Carrete, J., van Roekeghem, A., Madsen, G. K. & Mingo, N. Materials screening for the discovery of new half-heuslers: Machine learning versus ab-initio methods. J. Phys. Chem. B (2017).
20. Maltarollo, V. G., Gertrudes, J. C., Oliveira, P. R. & Honorio, K. M. Applying machine learning techniques for adme-tox prediction: a review. Expert. *Opin. on Drug Metab. & Toxicol.* **11**, 259–271 (2015).
21. Mayr, A., Klambauer, G., Unterthiner, T. & Hochreiter, S. Deeptox: Toxicity prediction using deep learning. *Front. Environ. Sci.* **3**, 80 (2016).
22. Zhang, L. *et al*. Carcinopred-el: Novel models for predicting the carcinogenicity of chemicals using molecular fingerprints and ensemble learning methods. *Sci. Rep.* **7**, 2118 (2017).
23. 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. Info. Mod.* **55**, 263–274 (2015).
24. Faber, F. A. *et al*. Prediction errors of molecular machine learning models lower than hybrid DFT error. *J. Chem. Theo. Comp.* **13**, 5255–5264 (2017).
25. Ward, L. *et al*. Including crystal structure attributes in machine learning models of formation energies via voronoi tessellations. *Phys. Rev. B* **96**, 024104 (2017).
26. Rice, B. M. & Hare, J. J. A quantum mechanical investigation of the relation between impact sensitivity and the charge distribution in energetic molecules. *J. Phys. Chem. A* **106**, 1770–1783 (2002).
27. Prana, V., Fayet, G., Rotureau, P. & Adamo, C. Development of validated QSPR models for impact sensitivity of nitroaliphatic compounds. *J. Haz. Mat.* **235–236**, 169–177 (2012).
28. 3d-morse descriptors explained. *J. Mol. Graph. Model*. 54, 194–203 (2014).
29. QSPR studies of impact sensitivity of nitro energetic compounds using three-dimensional descriptors. J. Mol. Graph. Model. 36, 10–19 (2012).
30. Fayet, G., Rotureau, P., Joubert, L. & Adamo, C. Development of a QSPR model for predicting thermal stabilities of nitroaromatic compounds taking into account their decomposition mechanisms. *J. Mol. Model.* **17**, 2443–2453 (2011).
31. Turker, L., Gumus, S. & Atalar, T. A DFT study on nitro derivatives of pyridine. *J. Energ. Mater.* **28**, 139–171 (2010).
32. Turker, L. A first-order linear model for the estimation of detonation velocity. *J. Energ. Mater.* **29**, 7–25 (2011).
33. Infante-Castillo, R. & Hernandez-Rivera, S. P. Predicting heats of explosion of nitroaromatic compounds through NBO charges and N-NMR chemical shifts of nitro groups. *Adv. Phys. Chem.* **4**, 304686 (2012).
34. Ravi, P., Gore, G. M., Tewari, S. P. & Sikder, A. K. DFT study on the structure and explosive properties of nitropyrazoles. *Mol. Simul.* **38**, 218–226 (2012).
35. Zeman, S. Sensitivities of High Energy Compounds, 195–271 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007).
36. Huang, L. & Massa, L. Applications of energetic materials by a theoretical method (discover energetic materials by a theoretical method). *Int. J. Ener. Mat. Chem. Prop.* **12**, 197–262 (2013).
37. Mathieu, D. Sensitivity of energetic materials: Theoretical relationships to detonation performance and molecular structure. *Ind. & Eng. Chem. Res.* **56**, 8191–8201 (2017).
38. Huang, L., Massa, L. & Karle, J. Discovery of energetic materials by a theoretical method (DEMTM). Int. *J. Ener. Mat. Chem. Prop.* **10**, 33–44 (2011).
39. Theodoridis, S. & Koutroumbas, K. Pattern Recognition, Fourth Edition, 4th edn., (Academic Press, 2008)
40. Martin, A. R. & Yallop, H. J. Some aspects of detonation. part 1. -detonation velocity and chemical constitution. *Trans. Faraday Soc.* **54**, 257–263 (1958).
41. Klapotke, T. Chemistry of High-Energy Materials (2017).
42. Guyon, I. & Elisseeff, A. An introduction to variable and feature selection. J. Mach. Learn. Res. 3 (2003).
43. Politzer, P. & Murray, J. S. Detonation Performance and Sensitivity: A Quest for Balance (Elsevier Science, 2014).
44. Hansen, K. *et al*. Machine learning predictions of molecular properties: Accurate many-body potentials and nonlocality in chemical space. *The J. Phys. Chem. Lett.* **6**, 2326–2331 (2015).
45. Montavon, G. *et al*. Learning invariant representations of molecules for atomization energy prediction. In Pereira, F., Burges, C. J. C., Bottou, L. & Weinberger, K. Q. (eds.) Advances in Neural Information Processing Systems 25, 440–448 (Curran Associates, Inc., 2012).
46. Faber, F., Lindmaa, A., von Lilienfeld, O. A. & Armiento, R. Crystal structure representations for machine learning models of formation energies. Int. J. Quantum Chem. **115** (2015).
47. Morgan, H. L. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *J. Chem. Documentation* **5**, 107–113 (1965).
48. Carhart, R. E., Smith, D. H. & Venkataraghavan, R. Atom pairs as molecular features in structure-activity studies: definition and applications. *J. Chem. Inf. Comput. Sci.* **25**, 64–73 (1985).
49. Nilakantan, R., Bauman, N., Dixon, J. S. & Venkataraghavan, R. Topological torsion: a new molecular descriptor for sar applications. comparison with other descriptors. *J. Chem. Inf. Comput. Sci.* **27**, 82–85 (1987).
50. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Info. Mod.* **50**, 742–754 (2010).
51. Hall, L. H. & Kier, L. B. Electrotopological state indices for atom types: A novel combination of electronic, topological, and valence state information. *J. Chem. Inf. Comput. Sci.* **35**, 1039–1045 (1995).
52. Gedeck, P., Rohde, B. & Bartels, C. QSAR - how good is it in practice? comparison of descriptor sets on an unbiased cross section of corporate data sets. *J. Chem. Info. Mod.* **46**, 1924–1936 (2006).
53. Landrum, G. RDKit: Open-source cheminformatics. http://www.rdkit.org.
54. Stiefl, N., Watson, I. A., Baumann, K. & Zaliani, A. ErG: 2D pharmacophore descriptions for scaffold hopping. *J. Chem. Info. Mod.* **46**, 208–220 (2006).
55. Kearsley, S. K. *et al*. Chemical similarity using physiochemical property descriptors. *J. Chem. Inf. Comput. Sci.* **36**, 118–127 (1996).
56. Rhodes, N., Clark, D. E. & Willett, P. Similarity searching in databases of flexible 3d structures using autocorrelation vectors derived from smoothed bounded distance matrices. *J. Chem. Info. Mod.* **46**, 615–619 (2006).
57. Nettles, J. H. *et al*. Bridging chemical and biological space: "target fishing" using 2D and 3D molecular descriptors. *J. Medicinal Chem.* **49**, 6802–6810 (2006).
58. Lowis, D. R. HQSAR. a new, highly predictive QSAR technique. *Tripos Tech. Notes* **1**, 3 (1998).
59. Bartok, A. P., Kondor, R. & Csanyi, G. On representing chemical environments. *Phys. Rev. B* **87**, 184115 (2013).
60. Bartok, A. P. *et al*. Machine learning unifies the modeling of materials and molecules. *Sci. Adv.* **3** (2017).
61. von Lilienfeld, O. A., Ramakrishnan, R., Rupp, M. & Knoll, A. Fourier series of atomic radial distribution functions: A molecular fingerprint for machine learning models of quantum chemical properties. *Int. J. Quan. Chem.* **115** (2015).
62. Ferré, G., Haut, T. & Barros, K. Learning molecular energies using localized graph kernels. *J. Chem. Phys.* **146**, 114107 (2017).
63. Duvenaud, D. *et al*. Convolutional networks on graphs for learning molecular fingerprints. In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2, NIPS'15, 2224–2232 (2015).
64. Kearnes, S., McCloskey, K., Berndl, M., Pande, V. & Riley, P. Molecular graph convolutions: moving beyond fingerprints. *J. Comput. Mol. Des.* **30**, 595–608 (2016).
65. Schütt, K. T., Arbabzadah, F., Chmiela, S., Müller, K. R. & Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nat. Comm.* **8**, 13890 (2016).

66. Lubbers, N., Smith, J. S. & Barros, K. Hierarchical modeling of molecular energies using a deep neural network. *arXiv e-print* 1710.00017 (2017).
67. Kier, L. B. & Hall, L. H. An electrotopological-state index for atoms in molecules. *Pharm. Res.* **7**, 801–807 (1990).
68. Prediction of impact sensitivity of nitro energetic compounds by neural network based on electrotopological-state indices. J. Haz. Mat. **166**, 155–186 (2009).
69. Sahigara, F. *et al*. Comparison of different approaches to define the applicability domain of QSAR models. *Mol.* **17**, 4791–4810 (2012).
70. Huang, B. & von Lilienfeld, O. A. Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity. *J. Chem. Phys.* **145**, 161102 (2016).
71. Muller, K. R., Finke, M., Murata, N., Schulten, K. & Amari, S. A numerical study on learning curves in stochastic multilayer feedforward networks. *Neural Comput.* **8**, 1085–1106 (1996).
72. Hutchinson, M. L. *et al*. Overcoming data scarcity with transfer learning. *arXiv e-prints* 1711.05099 (2017).
73. Barnes, B. C. *et al*. "Machine Learning of Energetic Material Properties", Proceedings of the 16th International Detonation Symposium, Cambridge MD, USA, July 2018. Manuscript in preparation.
74. Golbraikh, A. & Tropsha, A. *Beware of q2! J. Mol. Graph. Model.* **20**, 269–276 (2002).
75. Weininger, D. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. J. Chem. Inf. Comp. Sci. **28** (1988).
76. Groom, C. R., Bruno, I. J., Lightfoot, M. P. & Ward, S. C. The Cambridge Structural Database. *Acta Crystallogr. Sect. B* **72**, 171–179 (2016).
77. Filippov, I. V. & Nicklaus, M. C. Optical Structure Recognition Software To Recover Chemical Information: OSRA, An Open Source Solution. *J. Chem. Info. Mod.* **49**, 740–743 (2009).
78. O'Boyle, N. M. *et al*. Open Babel: An open chemical toolbox. *J. Cheminformatics* **3**, 33 (2011).
79. Halgren, T. A. Merck molecular force field. i. basis, form, scope, parameterization, and performance of MMFF94. J. Comp. Chem. **17** (1996).

Acknowledgements

Support for this work is gratefully acknowledged from the U.S. Office of Naval Research under grant number N00014-17-1-2108 and from the Energetics Technology Center under project number 2044-001. Partial support is also acknowledged from the Center for Engineering Concepts Development in the Department of Mechanical Engineering at the University of Maryland, College Park. We thank Dr. Brian C. Barnes for comments on the manuscript. We also thank Dr. Ruth M. Doherty and Dr. William Wilson from the Energetics Technology Center for their encouragement, thoughts on energetic materials, and for proofreading the manuscript.

Author Contributions

P.W.C. and M.F. conceived the research, D.E. wrote the code and most of the manuscript, M.B. prepared the data and helped generate the SMILES strings, Z.B. did part of the machine learning and dimensionality reduction work. All authors reviewed and edited the manuscript.

Additional Information

Supplementary information accompanies this paper at https://doi.org/10.1038/s41598-018-27344-x.

Competing Interests: The authors declare no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.