JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

pubs.acs.org/jcim Article

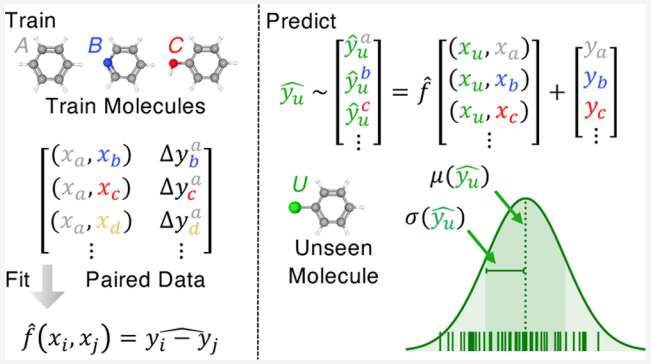
Pairwise Difference Regression: A Machine Learning Meta-algorithm for Improved Prediction and Uncertainty Quantification in Chemical Search

Michael Tynes,* Wenhao Gao, Daniel J. Burrill, Enrique R. Batista, Danny Perez, Ping Yang,* and Nicholas Lubbers*

Cite This: *J. Chem. Inf. Model.* 2021, 61, 3846−3857 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Machine learning (ML) plays a growing role in the design and discovery of chemicals, aiming to reduce the need to perform expensive experiments and simulations. ML for such applications is promising but difficult, as models must generalize to vast chemical spaces from small training sets and must have reliable uncertainty quantification metrics to identify and prioritize unexplored regions. *Ab initio* computational chemistry and chemical intuition alike often take advantage of differences between chemical conditions, rather than their absolute structure or state, to generate more reliable results. We have developed an analogous comparison-based approach for ML regression, called pairwise difference regression (PADRE), which is applicable to arbitrary underlying learning models and operates on pairs of input data points. During training, the model learns to predict differences between all possible pairs of input points. During prediction, the test points are paired with all training set points, giving rise to a set of predictions that can be treated as a distribution of which the mean is treated as a final prediction and the dispersion is treated as an uncertainty measure. Pairwise difference regression was shown to reliably improve the performance of the random forest algorithm across five chemical ML tasks. Additionally, the pair-derived dispersion is both well correlated with model error and performs well in active learning. We also show that this method is competitive with state-of-the-art neural network techniques. Thus, pairwise difference regression is a promising tool for candidate selection algorithms used in chemical discovery.



INTRODUCTION

Machine learning (ML)-driven optimization algorithms are of growing interest in the design and discovery of materials, largely due to the relatively low cost of ML model evaluation compared to theoretical calculations or physical experiments.[1−8] Two main problems confront this application of ML: (1) building a model that can generalize well to a vast chemical search space from a comparatively small set of observations and (2) identifying regions of chemical space where model uncertainty is high and thus more data is needed. To illustrate the first problem, consider that the well-known density functional theory (DFT) database QM-9[9] contains data on roughly 10^5 molecules but is still 10^6 times smaller than the set of possible druglike molecules (GDB-17)[10] from which its constituents were drawn. Far less data is typically available for more specialized chemical applications, despite problem spaces being similarly large. The second problem is usually solved using an uncertainty quantification (UQ) metric defined for model predictions. Many but not all ML model types natively support UQ [*e.g.*, Gaussian processes and random forests (RFs)], and developing new UQ metrics is an area of active research.[11−14] Popular approaches to uncertainty quantification include model ensemble disagreement[15,16] and distance from the observed training points as measured by the model, for example, the kernel within a Gaussian process,[17] the RF distance function,[18,19] and the recently developed neural network (NN) latent space distance method.[11]

It is common practice for computational chemists to take the difference of estimates with highly correlated errors in order to take advantage of the cancellation of these errors. This is possible because a given level of chemical theory introduces assumptions about chemical systems which can affect simulations of different systems in similar ways. Taking the difference of a quantity between calculations at the same level

Received: June 14, 2021
Published: August 4, 2021

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

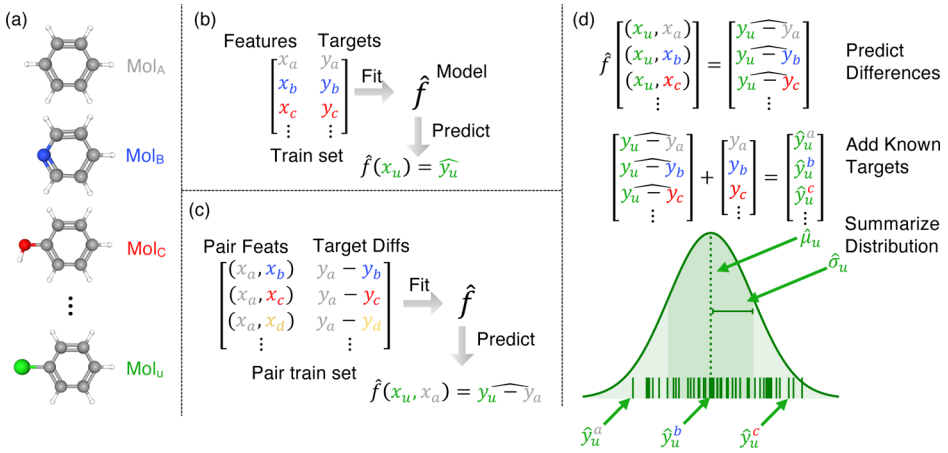


Figure 1. Illustration of PADRE. For (b–d), quantities with hats ˆ are estimates under the estimator $\hat{f}$. (a) Set of seen molecules (labeled {A, B, C, ...}) used for training a regressor to predict the properties of a single unseen molecule U. (b) Classical regression approach wherein a model is fit to predict the reference targets y_i from the feature vectors $\boldsymbol{x}_i$ describing the molecules. (c) Construction of a pairwise training set from which a model learns to predict differences in target values from pairs of feature vectors. (d) Process of pairing an unseen feature vector with all seen feature vectors, giving a set of difference predictions under the model. This set is then converted by the addition of known quantities into a distribution of target predictions for the unseen feature vector.

of theory partially cancels out this bias, which can lead to more accurate predictions. This principle was outlined in a classic work[20] which formulates counterpoise corrections for determining interaction energies in chemical systems. Similar principles are used to isolate differential chemical effects in complex reaction conditions. For example, ref 21 introduced the notion of differential binding energy as a method of studying the contributions of structural variations within a set of ligands. In another example, ref 22 employed a difference approach to calculate free energies for selective metallic extraction by removing uncertainties related to solvation. In some cases, such as in the separation process design (*e.g.*, ref 23), the difference in chemical behavior is of primary interest, and the absolute response magnitude is of secondary importance. The stability of differential (delta) quantities has led to their popularity for modeling applications, for example, in the case of reaction enthalpies[24] and bond dissociation energies.[25]

A data science approach to molecular differences has also been widely employed in drug discovery in the form of matched molecular pair analysis (MMPA).[26−28] In MMPA, molecules that differ structurally only in one key substructure are called a matched pair and represent a transformation from one substructure to another. A regressor then learns how these substructure transformations lead to changes in target properties, thus deriving design rules for molecules. MMPA's success is often partially attributed to error cancellation similar to that discussed above. In image processing, a pairwise approach to data augmentation was developed in ref 29 to improve classification by a deep NN. In this approach, the average of two images is presented to the network. This combination of images can be interpreted as a form of highly correlated input noise which regularizes the network prediction. Pairwise ML is an established technique in the information retrieval literature wherein a model learns to rank data items by relevance to a query.[30] Pairwise analysis is also established in the field of metric learning, where the goal is to learn a distance function between data points.[31,32] Two-legged (or "Siamese") NNs,[33] a family of metric learning network architectures originally developed for signature verification,[34] have been used to measure the similarity between drug candidates[35] and to directly predict the biological activity of molecules.[36] A pairwise difference NN of a similar architecture to Siamese networks was used in ref 37 to find drug candidates with optimal properties. We contrast this method with ours in the Discussion section.

Motivated by these successful examples, we explore whether the same principles can be applied to a large set of data all together to improve the performance of ML regression algorithms by targeting the differences of chemical properties between all possible pairs of molecules. Like bagging[38] or boosting,[39,40] this approach is a meta-procedure that can be applied to a base learner. It is not, however, an ensembling procedure but rather a re-framing of the regression problem. We find that this procedure leads to a natural non-parametric uncertainty quantification metric when using the pairwise model to make predictions on individual data points. We investigated the performance of this method in five chemical regression tasks and found that it consistently offers a predictive advantage over standard regression. Further, we find that the uncertainty metric is both well correlated with model error and useful for uncertainty-driven candidate selection (active learning).[41] Moreover, we explicitly compare our results against the recent methods developed in ref 1 and find that it performs competitively on the same data.

METHODS

High-Level Description. Here, we explain our approach for pairwise difference regression (PADRE). Suppose we wanted to build an ML model that can learn to predict theoretically calculated or experimentally measured properties (henceforth, called "reference" properties) for a set of molecules {A, B, C, D} and can generalize to predict the same properties about a new, unknown molecule U. This problem setup is illustrated in Figure 1a. The classical regression approach to this problem is simply to train the model to A through D and then predict on U. One disadvantage of this approach is that the model will be forced to learn about the systematic errors introduced by the theoretical or experimental method as well as the underlying

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

"true" signal. Instead, consider that the model might learn to predict differences in properties from the set of differences between molecule descriptions {A−B, A−C, A−D, B−C, ...}. For one, these differences may be more reflective of the underlying chemical properties owing to cancellation of systematic errors. Moreover, the set of differences manifestly contains more samples than the original dataset. In this light, the transformation to a set of difference could be viewed as a data augmentation scheme. Such schemes often improve the performance of models, particularly when applied to small datasets.[42]

To enable ML of pairwise differences, we convert the original n training points to n^2 points formed from pairwise information (visualized in Figure 1b,c). The model is able to witness the features from two molecules simultaneously, with the task of predicting the difference between their regression targets.

The obvious question raised by this procedure is how to recover prediction of the target property on a new molecule. Our solution to this problem is visualized in Figure 1d. Intuitively, a direct target prediction for a new molecule U may be recovered by first predicting the difference between U and any previously seen molecule M and then adding back in the reference target for M from the training set. Because there are many such known molecules in the training set, we can conduct this prediction procedure for all known molecules M and an unknown molecule U to form a distribution of predictions for U. This distribution is a result of a simple procedural algorithm rather than the explicit modeling of probabilities. Nonetheless, we explore the notion that the mean of this distribution forms a reasonable predictor and that the standard deviation is related to the uncertainty of the ML procedure.

Mathematical Formulation. We briefly review the standard ML formalism for regression. Given a set of n observations indexed by i of feature vectors $\boldsymbol{x}_i \in \mathbf{R}^m$ and the associated target reference values $y_i \in \mathbf{R}$, a model $\hat{f}$ is trained to predict y_i from $\boldsymbol{x}_i$. The feature vectors can be stored as the rows of a matrix $X \in \mathbf{R}^{m\times n}$ such that the whole model can be written as $\mathbf{R}^n \ni \hat{\mathbf{y}} = \hat{f}(\mathbf{X})$ with $\hat{f}$ evaluated rowwise.

In PADRE, this ML formalism is slightly augmented. First, we convert a training dataset of n observations into a dataset of n^2 pairwise observations by promoting features and targets from single-indexed quantities to double-indexed quantities. To do this, let us introduce a pairwise index of tuples p

$$p \in \{(i, j),\ i \in \{1...n\},\ j \in \{1...n\}\} \tag{1}$$

such that p_i denotes the first value of the index and p_j denotes the second value of the index. The pairwise features $\tilde{\mathbf{x}}_p$ should depend on from both the features $\mathbf{x}_{pi}$ and $\mathbf{x}_{pj}$. We build them over the concatenation (denoted by the direct sum $\oplus$) of the two feature sets, as well as the explicit differences between these features

$$\tilde{\mathbf{x}}_p = \mathbf{x}_{p_i} \oplus \mathbf{x}_{p_j} \oplus (\mathbf{x}_{p_i} - \mathbf{x}_{p_j}) \tag{2}$$

Note that this is a particular choice of pairwise featurization and that many definitions are possible, including the rather extreme choice of the direct product which would square the number of features. The PADRE feature matrix $\tilde{\mathbf{X}}$ can be written as

$$\tilde{\mathbf{X}} = \begin{pmatrix} \tilde{\mathbf{x}}_0^{\mathrm{T}} \\ \tilde{\mathbf{x}}_1^{\mathrm{T}} \\ \vdots \\ \tilde{\mathbf{x}}_n^{\mathrm{T}} \\ \tilde{\mathbf{x}}_{n+1}^{\mathrm{T}} \\ \vdots \\ \tilde{\mathbf{x}}_{n^2}^{\mathrm{T}} \end{pmatrix}_{n^2\times 3m} = \begin{pmatrix} (\mathbf{x}_1 \oplus \mathbf{x}_1 \oplus \mathbf{x}_1 - \mathbf{x}_1)^{\mathrm{T}} \\ (\mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \mathbf{x}_1 - \mathbf{x}_2)^{T} \\ \vdots \\ (\mathbf{x}_1 \oplus \mathbf{x}_n \oplus \mathbf{x}_1 - \mathbf{x}_n)^{\mathrm{T}} \\ (\mathbf{x}_2 \oplus \mathbf{x}_1 \oplus \mathbf{x}_2 - \mathbf{x}_1)^{\mathrm{T}} \\ \vdots \\ (\mathbf{x}_n \oplus \mathbf{x}_n \oplus \mathbf{x}_n - \mathbf{x}_n)^{\mathrm{T}} \end{pmatrix}_{n^2\times 3m} \tag{3}$$

with each single training point $\boldsymbol{x}_i \in \mathbf{R}^m$.

The targets for pairwise difference regression $\tilde{y}_p$ are formed as

$$\tilde{y}_p = \Delta y_{ij} = y_{p_i} - y_{p_j} \tag{4}$$

or, in vector form,

$$\tilde{\mathbf{y}} = \begin{pmatrix} (y_1 - y_1) \\ (y_1 - y_2) \\ \vdots \\ (y_1 - y_n) \\ (y_2 - y_1) \\ \vdots \\ (y_n - y_n) \end{pmatrix}_{n^2\times 1} \tag{5}$$

Then, a pairwise model $\hat{f}$ can be chosen as any learning algorithm or architecture and fit using standard techniques to approximate $\tilde{\mathbf{y}} = \tilde{f}(\tilde{\mathbf{X}})$. To form predictions on an individual unknown data point $\boldsymbol{x}_u$, we form the empirical distribution of predictions $\hat{y}_u$ over the training data indexed by i

$$\hat{y}_u \sim \widehat{\Delta y_{ui}} + y_i = \tilde{f}\left(\tilde{\mathbf{x}}_{(u,i)}\right) + y_i \tag{6}$$

In other words, to predict $\hat{y}_u$ for an unseen data point x_u, we pair x_u with every point in the training set and use the regressor to generate n predictions. This leads to a mean prediction, compared to the training data, of

$$\hat{\mu}_u = \mathrm{mean}[\hat{y}_u] \tag{7}$$

$$= \frac{1}{N}\sum_{i=1}^{N} \hat{\tilde{y}}_{(u,i)} + y_i \tag{8}$$

$$= \frac{1}{N}\sum_{i} \widehat{\Delta y_{ui}} + y_i \tag{9}$$

and a standard deviation of predictions $\hat{\sigma}_u$ through the equation

$$\hat{\sigma}_u = \mathrm{Var}[\hat{y}_u]^{1/2} \tag{10}$$

which can be evaluated analogously using the distribution of points examined during training. We note that while these definitions of $\hat{\mu}$ and $\hat{\sigma}$ are suggestive of a Gaussian distribution, in this context, they are simply descriptive measures of the location and scale of empirical distributions without heavy tails. Figure S1 shows that the PADRE distributions defined in eq 6 satisfy this condition for the experiments presented below.

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

Table 1. Summary of Datasets[a]

dataset	target(s)	feature type	$n_{features}$	n_{points}	n_{search}	n_{test}	source
Fe-DFT	dE	fingerprint	6029	1148	1000	145	this work
Fe-DFTB	dE	fingerprint	11656	12726	10000	2726	this work
redox	log P, ΔG	RAC-155	155	548	500	48	ref 1
lanthanide	log K	RDK	102	6577	5000	1577	ref 56

[a] $n_{features}$ indicates the number of descriptors used as ML inputs for each task. n_{points} indicates the total number of data items in each dataset. n_{search} is the number of data items available to the algorithm (called the search set) from which training sets of various sizes were drawn. n_{test} indicates the number of held-out test data items not present in the search set.

Implementation. Expressed in an array programming language, PADRE can be implemented very compactly. The pseudocode for the algorithm is available in the Supporting Information. Our implementation is in numpy[43] and follows the pseudocode closely.

To compare the performance of pairwise and classical regression, we conduct a series of experiments with a fixed model architecture: an RF. We briefly review the RF regression algorithm. Regression RFs grow an ensemble of independent decision trees T_i, with $i = 1, ..., n_{trees}$. Each T_i is a regressor that is fit by recursively splitting bootstrapped samples of the training set into groups of higher purity; that is, the two partitions created by each split have lower variance in their target values than the pre-partition set. This is accomplished with a recursive greedy algorithm that finds the optimal feature value on which to split the data at each level of the tree. This strategy is repeated recursively until a maximum tree depth or a minimum number of samples within a partition is reached. The final data partitions are called leaves, and the leaves collectively partition the entire feature space into axis-perpendicular regions which are associated with the average value of the (bootstrapped) training data they contain. Each tree T_i can now be considered a function that maps any element of the feature space to these average values. The overall RF prediction is then defined as the average over the tree predictions, $1/n_{trees} \sum_{i=1}^{n_{trees}} T_i(x)$. For a more complete overview, we recommend to refer to refs 44 and 45.

We chose to use RFs because they are consistently strong learners on diverse tasks (ref 46 even claims that RFs are the best classifiers, although this view is disputed[47]), are well known to perform reasonably even when the number of features exceeds the number of observations by orders of magnitude,[48] are relatively fast to fit,[49] and are among the models most robust to hyperparameter choices.[50] A set of experiments reported in the Supporting Information and shown in Figures S2 and S3 indicates that default hyperparameters are among the best hyperparameter choices for the tasks presented herein. Thus, RFs provide a suitable baseline against which to examine the influence of choosing to use pairwise difference regression over simple regression with all else being equal. We use an RF with 50 estimators and default hyperparameters implemented in scikit-learn.[51] As an uncertainty metric for our baseline RF, we use the standard deviation over the tree ensemble predictions, a commonly used uncertainty quantification approach.[15,16,19,52,53]

We avoided both NNs due to their relative sensitivity to hyperparameters and Gaussian processes due to their pronounced susceptibility to the curse of dimensionality.[54] We also avoided linear models because under our current featurization scheme, they would yield a non-informative UQ metric that is independent from the testing set, which we show in the Discussion section.

Datasets. We compared classical regression and PADRE on five tasks from four datasets of metal−ligand complexes, of which three are computational and one is experimental. The first two datasets ("Fe-DFT" and "Fe-DFTB") were generated in-house and consist of binding energies (dE) of iron−ligand complexes computed with DFT and density functional tight binding (DFTB), respectively.[55] The details of the generation of these datasets are available in the Supporting Information, and the datasets are available in the Supporting Information. The third dataset ("redox") is publicly available and contains DFT-estimated octanol−water partition coefficient (log P) and free energy of oxidation (ΔG) values for transition metal complexes designed for redox flow batteries.[1] The fourth dataset ("lanthanide") contains experimentally determined binding affinities (log K = [ML]/[M][L]) for lanthanide−ligand complexes[56] obtained from PubChem.[57] These datasets are summarized in Table 1.

The redox and lanthanide datasets were featurized with the schemes in their publications. The Fe-* datasets were featurized using molecular fingerprints (FPs) of the ligands generated with the cheminformatics package RDKit[58] along with the charge and spin of the metal center, the number of coordinating atoms in the ligand, and the number of non-water ligands in the complex. We concatenated three types of FPs: RDKit topological FPs[59] (FP size = 2^{12}), Morgan FPs[60] (with depth = 10), and atom pair FPs.[61] With these settings, there are an extremely large number of FP features. However, many of them are redundant. As such, we detect groups of features that are redundant with each other (100% correlated across all items within a dataset) and remove all but one feature from these groups, yielding the number indicated in Table 1. We note that $n_{features}$ is close in size to or even exceeds n_{points} in the case of Fe-DFT and Fe-DFTB, respectively. While some regressors would struggle in this case, RFs are well known to perform reasonably even when the number of features exceeds the number of observations by orders of magnitude.[48]

COMPUTATIONAL EXPERIMENTS AND RESULTS

PADRE Performance on Random Training Sets. For each regression task, an initial search space of n_{search} data items was chosen at random, and the remaining n_{test} were held out for testing. From the n_{search} data items, a training set of size n_{train} = 50 was chosen at random. (The numerical values of n_{search} and n_{test} by task can be found in Table 1.) Both a simple RF and a pairwise difference random forest (PADRE-RF) were fit to the initial n_{train} = 50 data points, and model performance metrics were collected on the train, search, and test sets. Then, k_{select} = 10 points were chosen from the search space at random and added to the train set, and the processes of fitting, metric collection, and selection from the search space were repeated out to n_{train} = 200. This experiment was repeated n_{cv} = 100 times using Monte Carlo cross-validation (MC-CV), that is,

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

using different train/test split seeds to establish the mean and standard error of the mean for performance.

To show the predictive advantage of PADRE-RF over RF concisely and without units, we computed the ratios of each predictor's test set mean absolute error (MAE) for each task. Figure 2 shows these ratios for each task as a function of

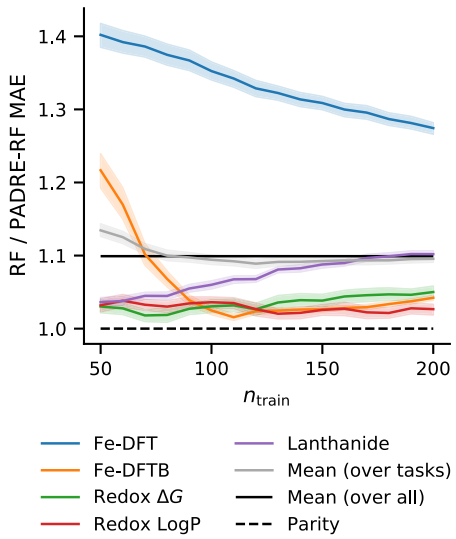


Figure 2. Test set MAE ratio for RF/PADRE-RF by task and n_{train}. Ratios were computed within each MC-CV split. The solid lines indicate the geometric mean of these ratios and error bands represent bootstrap 68% confidence intervals.

training set size n_{train}. From this figure, we see that PADRE consistently offers an advantage over RF because the value of RF error/PADRE error is always positive. With respect to train size, PADRE's advantage is roughly constant for the Redox tasks, decreases for the Fe-* tasks, and increases for the lanthanide task. Over all of the evaluated tasks and train sizes, the RF's error is 1.1 times the magnitude of PADRE-RF's error. The taskwise averages of these ratios over n_{train} are shown in Table 2, from which we see that PADRE-RF offers an

Table 2. Model Performance on Test Set with Random Train Sets[a]

task	units	RF MAE	PADRE-RF MAE	ratio
Fe-DFT	kcal/mol	11.5	8.64	1.33
Fe-DFTB	kcal/mol	12.4	11.7	1.06
redox ΔG	eV	0.52	0.503	1.03
redox log P		1.44×10^{-3}	1.40×10^{-3}	1.03
lanthanide		2.13	2.0	1.10
overall				1.09

[a]Results shown are the test set MAE for each model, RF and PADRE-RF, along with the ratio of MAE for RF to PADRE-RF, for each task. MAE values are averaged over $n_{cv} = 100$ splits and over $n_{train} = 50$ to $n_{train} = 200$. Average is arithmetic for MAE and geometric for ratios.

advantage for all tasks, albeit a slim one for the redox log P and redox ΔG tasks. In the case of the Fe-DFT task, the improvement was larger than 30%. The raw MAE, RMSE, and R^2 values for each task are shown in Figures S3–S5.

Uncertainty Quantification Performance. We assess the utility of the PADRE dispersion $\hat{\sigma}$ (eq 10) as an uncertainty metric using two methods: Spearman's rank–rank correlation coefficient ρ and confidence curves.

To illustrate the motivation for using Spearman's ρ, consider that an uncertainty metric is useful insofar as it can reliably identify the unseen points on which the model is performing relatively well and points on which it is doing relatively poorly. In other words, a good uncertainty metric should reliably rank-order unseen points in terms of the magnitude of their error. This means that an uncertainty metric may not have a linear relationship with error magnitude and yet be useful, for example, to guide an active learning procedure. Figure 3a shows $\hat{\sigma}$ against $|\hat{y}_{pair} - y|$. Although points are not clustered around a line of best fit, we see that points with the smallest and largest $\hat{\sigma}$ are among the points with the smallest and largest absolute error values, respectively. This fact is made more clear by Figure 3b, which shows the ranks of $\hat{\sigma}$ against the ranks of $|\hat{y}_{pair} - y|$ (where the rank of a list element is simply the element's position in the sorted version of the list). The ordinary Pearson correlation of these ranks defines Spearman's ρ, which are shown in Figure 3c. ρ ranges between 0.2 and approximately 0.65 for all tasks in, indicating that $\hat{\sigma}$ is a useful proxy for error across all tasks examined. Comparisons with the RF ensemble disagreement are shown in Figures S4 and S5. A definition and further discussion of rank and Spearman's ρ are available in the Supporting Information.

Confidence curves also measure the ability of an uncertainty metric to identify the most uncertain points in a dataset. Rather than examining the rank order of points, the confidence curve measures the aggregate effect of using the uncertainty metric as a filter on predictions. They are constructed for a dataset of size n by identifying the $k_{discard} < n$ points with the highest uncertainty values, excluding these points, and then measuring model error on the remaining $n - k_{discard}$ points. Ideally, model error will decrease monotonically with $k_{discard}$,[62−64] indicating that the uncertainty metric can identify on which points the model is performing well and which points it is performing poorly.

Mathematically, let $\hat{\sigma}_i$ be the set of uncertainty values for a set of predictions $\hat{y}_i$ over a test dataset with $i = 1, ..., n_{test}$. Re-sort i such that $\hat{\sigma}_i$ is sorted in decreasing order. The confidence curve for the MAE is defined as

$$c_{MAE}(k_{discard}) = \mathbf{E}[|\hat{y} - y| | \hat{\sigma} < \hat{\sigma}_{k_{discard}}] \quad (11)$$

Figure 4 shows the confidence curves for all five tasks. We observe that c_{MAE} has the desired monotonically decreasing behavior for all tasks. For the lanthanide and redox ΔG tasks, the error falls by more than half when discarding half the points. In general, the confidence curves are consistent with Figure 3C.

Figure 5 shows the confidence curves for ΔG and log P for PADRE compared to those reported using an NN-based approach in Janet *et al.*[11] for the redox dataset, evaluated using the same training set and search set, alongside the base RF learner. The base RF performs worst in both tasks. For ΔG, the NN model performs slightly better, but PADRE-RF performs nearly as well and is at parity with the NN for $k_{discard} = 10$ and $k_{discard} = 20$. In the log P task, the NN model has a lower base error ($k_{discard} = 0$) by a factor of approximately 2.5, but the uncertainty quantification does not provide effective filtering on the data; there, the PADRE-RF model meets and even exceeds the performance of the NN method for about $k_{discard} \geq 10$. Although the PADRE-RF's overall error is larger, $\hat{\sigma}$ is very effective at determining which points have high error.

PADRE Performance in Active Learning. Active learning is a common candidate selection algorithm in chemis-

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

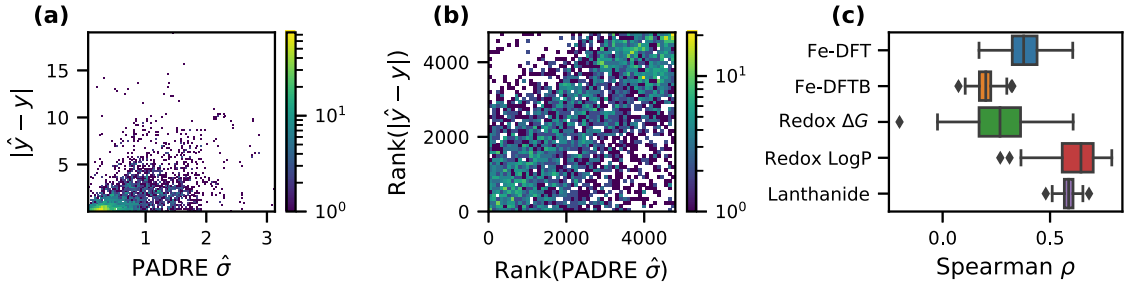


Figure 3. Validation of PADRE $\hat{\sigma}$ as an uncertainty metric. (a) 2D Histogram of prediction error against $\hat{\sigma}$ for the unitless redox log P task, with color indicating bin occupancy counts. (b) 2D Histogram of the rank of prediction error against the rank of $\hat{\sigma}$ for the redox log P task, with color indicating bin occupancy counts. (c) Associated Spearman's correlation coefficient ρ for all tasks. The color in (c) is merely stylistic. Each plot is given using n_{train} = 200.

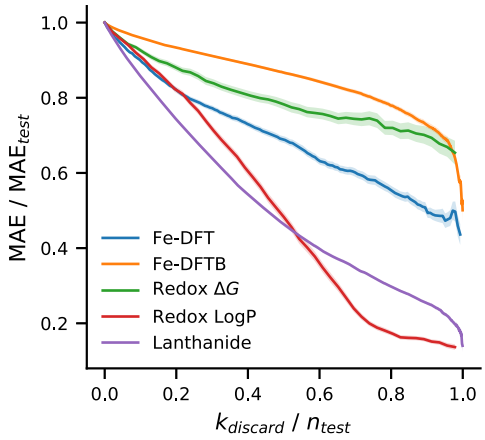


Figure 4. PADRE-RF MAE confidence curves for all tasks at n_{train} = 200, 0-1 rescaled and averaged over n_{cv} = 100. Error bands are empirical SEM.

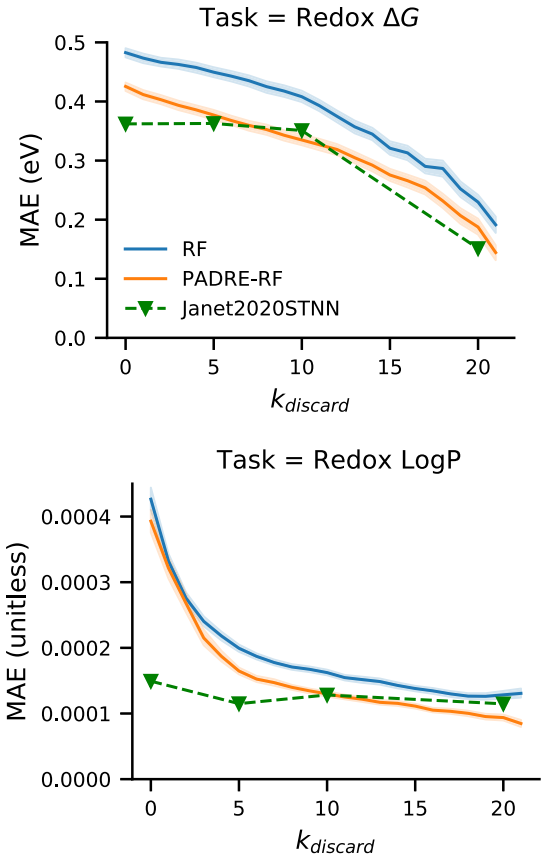


Figure 5. Confidence curves on the test data points in the redox dataset. Janet2020STNN corresponds to values for the single-task NN reported in ref 1. The curves for RF and PADRE-RF were generated using the exact data and data splitting description described in ref 1 and averaged over n_{cv} = 100 random splits (error bands are 68% bootstrap CI). The curves across methods are similar for ΔG. On log P, despite beginning with higher prediction error, PADRE-RF shows a much larger decrease compared to the relatively constant curve of the NN method.

try[15,16,65−68] and in the computer science ML literature[18,41,52] in which the goal is to select the unlabeled points that, if labeled and included in the training set, would lead to the largest reduction of model error. This approach often relies on an uncertainty metric that can identify the unseen points on which the model performance is the poorest, which we saw was true of the PADRE $\hat{\sigma}$ (eq 10) in the previous section.

We performed active learning experiments in the same fashion as those for random training sets of increasing size, except that after the initial n_{train} = 50 random points were selected from the search set, the k_{select} = 10 were not chosen randomly but instead were the 10 points for which the uncertainty metric was the largest. The MAE for both RF and PADRE-RF is shown in Figure 6 as a function of training set size, alongside the analogous results for random training set selection. We examine the performance on the search set because this quantifies the model performance on a pool of molecules that it is actively searching through, but note that the test set performance (shown in Figure S3) is typically similar. Overall, PADRE-RF significantly outperforms the base RF for each task. In general, active learning outperforms random learning as expected, although not by a large margin on the DFT and DFTB datasets. However, this is not the case for the lanthanide dataset, wherein active learning performed worse than random learning.

This unexpected result on the lanthanide dataset required further investigation, particularly given that the confidence curve for the lanthanide dataset was well behaved (*i.e.*, data with high error was effectively flagged by $\hat{\sigma}$). To examine this more carefully, we conducted an oracle learning experiment for diagnostic purposes. Our oracle-based learning is simply active learning where the k_{select} points are those with the highest values of $|\hat{y} - y|$, that is, active learning with an oracle with perfect knowledge of the true error. In other words, the oracle knows the worst-performing molecules and feeds them to the model for training. The results of this experiment are shown in Figure 7 for the lanthanide task and, for comparison to a model that behaved as expected, the redox ΔG task.

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

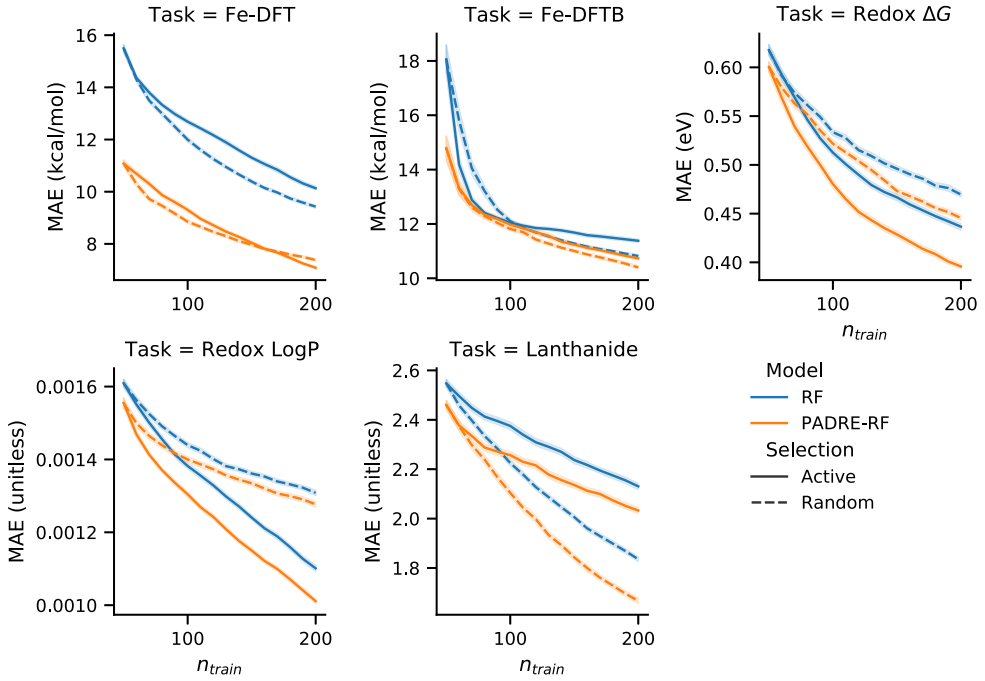


Figure 6. Learning curves for each task, showing every combination of active/random sampling from the search space and RF and PADRE-RF regression. Solid lines are means over n_{cv} = 100, error bands are bootstrap 68% confidence intervals.

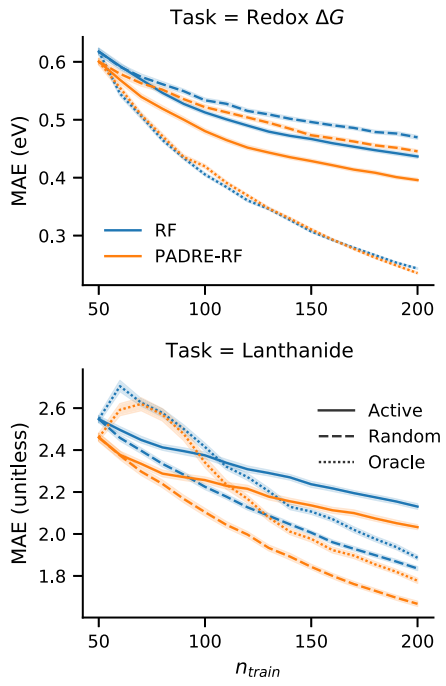


Figure 7. Learning curves on the search set for the redox ΔG and lanthanide tasks for both RF and PADRE-RF and for active, random, and oracle selection. Solid lines are means over n_{cv} = 100 and error bands are bootstrap 68% confidence intervals. Oracle selection corresponds to selecting points from the search space with the highest true prediction error. This leads to consistent and large reductions of generalization error for the redox ΔG task but an initial increase in error for the lanthanide task, indicating possible label noise.

For redox ΔG, the oracle learning is well behaved and performs even better than active learning, as one would expect. However, for the lanthanide dataset, the oracle-based learning scheme actually increases in error over the first few selection generations. This implies that selection of poorly performing search space molecules did not help but actually hurt the model generalization. There are several possible explanations. Tentatively, these points could be noisy or mislabeled. The lanthanide dataset was constructed from experimental data collected from a vast variety of sources in the literature. It is quite possible that a small percentage of these points are mislabeled. They could also be in some regard strong outliers in feature space, so that learning to these points does not help the model generalize to new molecules. Finally, these points could also be somehow harder to learn for the RF. Each of these possibilities, or a combination thereof, could explain (1) why these points have high error and (2) why training on them hurts generalization performance, regardless of whether PADRE-RF or RF is applied. Regardless of the cause, this phenomenon points to a larger limitation of uncertainty-driven active learning: adding molecules with high error to the training set does not always improve model generalization.

Overall, our experiments show that PADRE-RF-driven active learning outperforms ordinary RF-driven active learning by 11% (Table 3) averaged across tasks and train sizes. Notably, in all tasks but the lanthanide dataset, the benefit of using

Table 3. Model Performance on Search Set with Active-Learning-Selected Train Sets[a]

task	random learning	active learning
Fe-DFT	1.33	1.39
Fe-DFTB	1.06	1.04
lanthanide	1.07	1.05
redox ΔG	1.03	1.08
redox log P	1.03	1.07
overall	1.10	1.12

[a]Results show the ratios in the MAE of RF predictions to PADRE-RF predictions on the search set in active and random learning contexts for n_{train} = 200 averaged (geometrically) over n_{cv} = 100 splits.

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

PADRE appears to grow as n_{train} increases past 100, as can be seen in Figure 8. A more detailed comparison of performance metrics across model types and selection strategies is shown in Figures S5–S7.

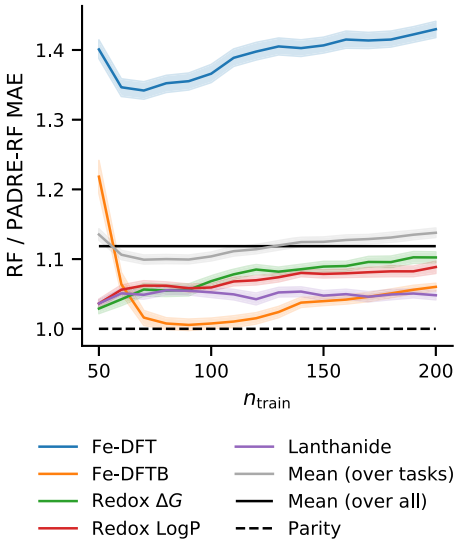


Figure 8. Search set MAE ratio for RF/PADRE-RF by task and n_{train}. Ratios were computed within each MC-CV split. Solid lines indicate the geometric mean of these ratios and error bands represent bootstrap 68% confidence intervals.

DISCUSSION

Overall, the above experiments indicate that the PADRE reformulation is useful for reducing model error constructing an uncertainty metric. We have focused on one-to-one comparisons with a base RF on both new and established datasets. There are many other factors that could be introduced and analyzed in pursuit of best performance.

PADRE itself could be generalized with several hyperparameters. First, the particular pairwise featurization (eq 2) is simple and effective but is not the only possible choice. For example, a direct product of features could be used, although this would square the number of features. Simply using the differences between pairs of feature vectors $\boldsymbol{x}_i - \boldsymbol{x}_j$ is possible, but we note that using this alone would assume that features that are common to any two data points are irrelevant to predicting their difference. In data where this is the case, a linear model may suffice. PADRE need not rely on fixed length feature vectors and could be implemented in a message-passing or graph-convolutional NN, allowing it to benefit from the notable performance gains in these areas (*e.g.*, refs 69 and 70). However, in this case, care should be taken to consider the drawbacks of pairwise separable models as discussed in the following paragraphs. Second, in treating very large datasets, one might not pair every training example together, but rather pair each example with a fixed number of other examples which can be chosen randomly or according to a scheme intended to identify the pairs most useful for prediction. The uncertainty metric itself need not be constructed from the variance of predictions: another possibility is to use the inter-quartile range along with the median as a predictor. In early tests, we tried this but did not find any significance difference in comparison with the mean and variance approach.

When constructing a PADRE model, it may be important to avoid pairwise separability of predictions for a few reasons. Formally, we study the structure of a separable pairwise model f that can be written as

$$\widehat{\Delta y}_{ij} = f(\mathbf{x}_i, \mathbf{x}_j) = g(\mathbf{x}_i) - g(\mathbf{x}_j) = \hat{y}_i - \hat{y}_j \tag{12}$$

Models with this property include linear regression with the pairwise featurization presented in eq 2 and two-legged NNs such as the network presented in ref 37. Pairwise separability undercuts the utility of the pairwise uncertainty metric defined in eq 10 and prevents the model from fully exploiting pairwise information to improve prediction. To see this clearly, first consider the overall prediction $\hat{\mu}_u$ for an unseen data point $\mathbf{x}_u$. With i as an index of a training set of size n, from eq 7, we have

$$\hat{\mu}_u = \frac{1}{n}\sum_i [f(\boldsymbol{x}_u, \boldsymbol{x}_i) + y_i] \tag{13}$$

$$= \frac{1}{n}\sum_i [g(\boldsymbol{x}_u) - g(\boldsymbol{x}_i) + y_i] \tag{14}$$

$$= g(\boldsymbol{x}_u) - \frac{1}{n}\sum_i [g(\boldsymbol{x}_i) - y_i] \tag{15}$$

We see that the central model predictions $\hat{\mu}$ do not leverage pairwise information but rather use g to predict $\hat{y}_u$ and then the procedure adds a learned offset from the training data which corrects for the shift of g compared to the training data. We can define this offset as $b = 1/n\sum_i[g(\mathbf{x}_i) - y_i]$ and write $\mu_u = g(\mathbf{x}_u) - b$. The value of $g(\mathbf{x})$ is corrected by b such that the average of $\hat{\mu}$ is equal to the average of y over the training set. Further examining the property of a separable model on $\hat{\sigma}$ reveals another important structure. Taking a few simple steps from the definition of $\hat{\sigma}_u$ in eq 10 and invoking separability

$$\hat{\sigma}_u = \frac{1}{n-1}\sum_i [f(\boldsymbol{x}_u, \boldsymbol{x}_i) + y_i - \hat{\mu}_u]^2 \tag{16}$$

$$= \frac{1}{n-1}\sum_i [g(\boldsymbol{x}_u) - g(\boldsymbol{x}_i) + y_i - g(x_u) - b]^2 \tag{17}$$

$$= \frac{1}{n-1}\sum_i [g(\boldsymbol{x}_i) - y_i + b]^2 \tag{18}$$

we see that when f is separable, $\hat{\sigma}_u$ is a constant determined by the training set and thus cannot be used to measure model uncertainty on unseen data points. Furthermore, the pairwise version of a mean squared error loss, $\mathcal{L}_p$

$$\mathcal{L}_p = \frac{1}{n^2}\sum_{ij} [f(\boldsymbol{x}_i, \boldsymbol{x}_j) - (y_i - y_j)]^2 \tag{19}$$

can be rewritten for a separable model as

$$\mathcal{L}_p = \frac{2}{n}\sum_i [\hat{\mu}_i - y_i]^2 \tag{20}$$

which demonstrates that for a separable model, the mean squared error loss is also separable, ultimately implying that it can be written into a mean squared loss which acts only on single data items. In other words, for a separable model, the pairwise mean squared error loss is equivalent to a mean squared error loss that disregards the average value of $g(\mathbf{x})$ predictions on the training set, and this average is restored

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

using the constant b after training. We provide a proof for eq 20 in the Supporting Information. From eq 20, we conclude that any effect of PADRE for separable models can only be attributed to the dynamics of learning or the inability to find a global minimum in the loss function. In this regard, it is reasonable to conjecture that the effects may be similar to batch normalization in deep learning;[71] this loss dynamically accounts for shifts of mean prediction during training. Batch normalization was shown to improve network training and in some cases dramatically.

Further work can be conducted to evaluate PADRE in the context of molecular search. The method examined in our experiments, active learning, is an extreme example of the exploration−exploitation tradeoff, which favors only exploration. Bayesian optimization[72] represents a general way of combining prediction and uncertainty information to drive the search procedure toward systems which exhibit a specific target property value—often the minimum or maximum of a property of interest. This method is widely employed in chemistry[73−76] and we hope to employ PADRE in a similar fashion to optimize chemical properties in future work. Another aspect of automated search is the parallel selection of new data; in some cases, selecting the points of highest uncertainty simultaneously results in the selection of highly correlated structures.[77] A remedy is to examine the covariance of possible selections. Sets of points with a high covariance may represent more diverse choices for active learning or Bayesian optimization. Such a model covariance can be produced in PADRE using the empirical distribution of predictions, eq 6. This could compliment other available techniques.[11,19,72]

We have also not fully explained why PADRE improves prediction performance. Although we hypothesize that the performance boost is due to a combination of systematic error cancellation and data augmentation, it is possible that one of these factors contributes more than the other or that other unconsidered factors are involved.

The main limitation of PADRE as described in this work is that it increases the expense of training, both in terms of memory and time constraints, because it transforms a regression problem on an $n \times m$ feature matrix into a regression problem on an $n^2 \times 3$ m feature matrix. It may be possible to address this limitation by sub-sampling from the set of all possible pairs. PADRE also incurs the cost of the storage of the training features and targets with the model, although similar costs are faced by other popular methods such as Gaussian processes and kernel ridge regression which also store training points. It is possible to only store a subsample of the training data, and there may exist subsets of the training data analagous to support vectors that give an optimal PADRE predictor.

CONCLUSIONS

In this work, we introduced a reformulation of a regression problem into the problem of predicting pairwise differences between data points, which we term PADRE. It can be simply and cleanly described with a few equations. It is a meta-algorithm that builds upon a base learning algorithm, which we have explored using the RF model.

Through five regression tasks on four datasets, we have demonstrated that PADRE improves model performance. We then showed that PADRE has a natural dispersion $\hat{\sigma}$ that is effective as a metric for uncertainty; points with a large dispersion are more likely to have large error, as quantified through Spearman's rank−rank coefficient and confidence curves. In a head-to-head comparison with a state-of-the-art NN method for uncertainty quantification,[1] a PADRE RF performs comparably. Having demonstrated this, we applied PADRE for active learning and conclude that the uncertainty metric tends to select points that improve model performance. Overall, our results show that PADRE is useful for ML problems involving chemical search. We are hopeful that it will be useful for the Bayesian property optimization for molecules and materials. PADRE may be more broadly useful in solving arbitrary ML problems where training data is limited or where uncertainty quantification is valuable.

In our discussion, we mentioned many variations that are possible and could be applied in future work and discussed scaling considerations and their possible solutions. We also identified the class of separable pairwise models, for which the PADRE uncertainty quantification would not be effective. In terms of future applications, an important area for ML in chemistry is chemical search, in particular Bayesian optimization through the combinatoric space of feasible chemical compounds for a given application.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.1c00670.

Visualizations of PADRE distributions and statistical normality tests, pseudocode description of PADRE algorithm, plots of RF and PADRE-RF performance for various hyperparameter choices, Fe-DFT and Fe-DFTB dataset generation details, discussion of Spearman's ρ, uncertainty−error correlation and prediction MAE, RMSE, and R^2 for all tasks, model types, and train sizes (PDF)

Ligand SMILES strings, complex descriptions, and binding energies for the Fe-DFT and Fe-DFTB datasets (ZIP)

AUTHOR INFORMATION

Corresponding Authors

Michael Tynes − *Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States; Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States;* orcid.org/0000-0002-5007-1056; Email: mtynes@lanl.gov

Ping Yang − *Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States;* orcid.org/0000-0003-4726-2860; Email: pyang@lanl.gov

Nicholas Lubbers − *Computer, Computational, and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States;* Email: nlubbers@lanl.gov

Authors

Wenhao Gao − *Computer, Computational, and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States; Department of Chemical Engineering, Massachusetts Institute of Technology,*

Cambridge, Massachusetts 02139, United States; orcid.org/0000-0002-6506-8044

Daniel J. Burrill − *Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States; Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States*

Enrique R. Batista − *Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States; Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States;* orcid.org/0000-0002-3074-4022

Danny Perez − *Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, United States;* orcid.org/0000-0003-3028-5249

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.1c00670

Notes

The authors declare no competing financial interest.

The Fe-DFT and Fe-DFTB datasets are available in the Supporting Information. The descriptions of all quantum mechanical calculations for computing binding energies in these datasets are available in the Supporting Information. All ML experiments were conducted in python (v3.8.5). The FP descriptors for the Fe-DFT and Fe-DFTB datasets were produced with the RDKit software package[58] (v2020.03.01) according to the description in the Methods section of this work. The Redox[1] and Lanthanide[56] datasets are available for download with their respective publications. The Redox dataset is published without features, but these can be reconstructed using the freely available molSimplify software package[78,79] (v1.5.0 was used for this work). The lanthanide dataset is published with all relevant features. The pseudocode for the core PADRE transformation is available in the Supporting Information written using a syntax compatible with numpy[43] (numpy v1.19.1 was used in this work). The RF implementation used is that of scikit-learn[51] (v0.23.2). All the source code related to this work is currently considered proprietary by Traid National Security, LLC, which operates Los Alamos National Laboratory.

■ ACKNOWLEDGMENTS

This study is based on the work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, Separation Science Program under contract number 2022LANLE3M1. Los Alamos National Laboratory (LANL) is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (contract no. 89233218CNA000001). M.T. acknowledges LANL's Center for Nonlinear Studies (CNLS). W.G. acknowledges the Applied Machine Learning (AML) program at LANL's Information Science and Technology Institute (ISTI). Computational experiments were conducted on LANL's CCS-7 Darwin cluster.

■ REFERENCES

(1) Janet, J. P.; Ramesh, S.; Duan, C.; Kulik, H. J. Accurate Multiobjective Design in a Space of Millions of Transition Metal Complexes with Neural-Network-Driven Efficient Global Optimization. *ACS Cent. Sci.* **2020**, *6*, 513−524.

(2) Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine Learning for Molecular and Materials Science. *Nature* **2018**, *559*, 547−555.

(3) Gubernatis, J. E.; Lookman, T. Machine Learning in Materials Design and Discovery: Examples From the Present and Suggestions for the Future. *Phys. Rev. Mater.* **2018**, *2*, 120301.

(4) Moosavi, S. M.; Jablonka, K. M.; Smit, B. The Role of Machine Learning in the Understanding and Design of Materials. *J. Am. Chem. Soc.* **2020**, *142*, 20273−20287.

(5) Reyes, K. G.; Maruyama, B. The Machine Learning Revolution in Materials? *MRS Bull.* **2019**, *44*, 530−537.

(6) Braham, E. J.; Cho, J.; Forlano, K. M.; Watson, D. F.; Arròyave, R.; Banerjee, S. Machine Learning-directed Navigation of Synthetic Design Space: A Statistical Learning Approach to Controlling the Synthesis of Perovskite Halide Nanoplatelets in the Quantum-Confined Regime. *Chem. Mater.* **2019**, *31*, 3281−3292.

(7) Balachandran, P. V. Machine Learning Guided Design of Functional Materials with Targeted Properties. *Comput. Mater. Sci.* **2019**, *164*, 82−90.

(8) Lima, A. N.; Philot, E. A.; Trossini, G. H. G.; Scott, L. P. B.; Maltarollo, V. G.; Honorio, K. M. Use of Machine Learning Approaches for Novel Drug Discovery. *Expert Opin. Drug Discov.* **2016**, *11*, 225−239.

(9) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; Von Lilienfeld, O. A. Quantum Chemistry Structures and Properties of 134 Kilo Molecules. *Data Sci.* **2014**, *1*, 140022.

(10) Ruddigkeit, L.; Van Deursen, R.; Blum, L. C.; Reymond, J.-L. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *J. Chem. Inf. Model.* **2012**, *52*, 2864−2875.

(11) Janet, J. P.; Duan, C.; Yang, T.; Nandy, A.; Kulik, H. J. A Quantitative Uncertainty Metric Controls Error in Neural Network-Driven Chemical Discovery. *Chem. Sci.* **2019**, *10*, 7913−7922.

(12) Vishwakarma, G.; Sonpal, A.; Hachmann, J. Metrics for Benchmarking and Uncertainty Quantification: Quality, Applicability, and Best Practices for Machine Learning in Chemistry. *Trends Chem.* **2021**, *3*, 146−156.

(13) Simm, G. N.; Reiher, M. Error-Controlled Exploration of Chemical Reaction Networks with Gaussian Processes. *J. Chem. Theory Comput.* **2018**, *14*, 5238−5248.

(14) Noh, J.; Gu, G. H.; Kim, S.; Jung, Y. Uncertainty-Quantified Hybrid Machine Learning/Density Functional Theory High Throughput Screening Method for Crystals. *J. Chem. Inf. Model.* **2020**, *60*, 1996−2003.

(15) Smith, J. S.; Nebgen, B.; Lubbers, N.; Isayev, O.; Roitberg, A. E. Less Is More: Sampling Chemical Space with Active Learning. *J. Chem. Phys.* **2018**, *148*, 241733.

(16) Reker, D.; Schneider, G. Active-learning Strategies in Computer-Assisted Drug Discovery. *Drug Discov. Today* **2015**, *20*, 458−465.

(17) Noack, M. M.; Doerk, G. S.; Li, R.; Streit, J. K.; Vaia, R. A.; Yager, K. G.; Fukuto, M. Autonomous Materials Discovery Driven by Gaussian Process Regression with Inhomogeneous Measurement Noise and Anisotropic Kernels. *Sci. Rep.* **2020**, *10*, 17663.

(18) Kee, S.; del Castillo, E.; Runger, G. Query-by-Committee Improvement with Diversity and Density in Batch Active Learning. *Inf. Sci.* **2018**, *454*−*455*, 401−418.

(19) Reker, D.; Schneider, P.; Schneider, G. Multi-Objective Active Machine Learning Rapidly Improves Structure−Activity Models and Reveals New Protein-Protein Interaction Inhibitors. *Chem. Sci.* **2016**, *7*, 3919−3927.

(20) Boys, S. F.; Bernardi, F. The Calculation of Small Molecular Interactions by the Differences of Separate Total Energies. Some Procedures with Reduced Errors. *Mol. Phys.* **1970**, *19*, 553−566.

(21) Morgan, B.; Scholtz, J. M.; Ballinger, M. D.; Zipkin, I. D.; Bartlett, P. A. Differential Binding Energy: a Detailed Evaluation of the Influence of Hydrogen-Bonding and Hydrophobic Groups on the Inhibition of Thermolysin by Phosphorus-Containing Inhibitors. *J. Am. Chem. Soc.* **1991**, *113*, 297−307.

(22) Keith, J. M.; Batista, E. R. Theoretical Examination of the Thermodynamic Factors in the Selective Extraction of Am3+ From Eu3+ by Dithiophosphinic Acids. *Inorg. Chem.* **2012**, *51*, 13−15.

(23) Rufford, T. E.; Smart, S.; Watson, G. C. Y.; Graham, B. F.; Boxall, J.; Diniz da Costa, J. C.; May, E. F. The Removal of CO2 and N2 From Natural Gas: A Review of Conventional and Emerging Process Technologies. *J. Petrol. Sci. Eng.* **2012**, *94*−*95*, 123−154.
(24) Noor, E.; Haraldsdóttir, H. S.; Milo, R.; Fleming, R. M. T. Consistent Estimation of Gibbs Energy Using Component Contributions. *PLoS Comput. Biol.* **2013**, *9*, No. e1003098.
(25) John, P. C. S.; Guan, Y.; Kim, Y.; Kim, S.; Paton, R. S. Prediction of Organic Homolytic Bond Dissociation Enthalpies at Near Chemical Accuracy with Sub-Second Computational Cost. *Nat. Commun.* **2020**, *11*, 2328.
(26) Tyrchan, C.; Evertsson, E. Matched Molecular Pair Analysis in Short: Algorithms, Applications and Limitations. *Comput. Struct. Biotechnol. J.* **2017**, *15*, 86−90.
(27) Kramer, C.; Fuchs, J. E.; Whitebread, S.; Gedeck, P.; Liedl, K. R. Matched Molecular Pair Analysis: Significance and the Impact of Experimental Uncertainty. *J. Med. Chem.* **2014**, *57*, 3786−3802.
(28) Dossetter, A. G.; Griffen, E. J.; Leach, A. G. Matched Molecular Pair Analysis in Drug Discovery. *Drug Discov. Today* **2013**, *18*, 724−731.
(29) Inoue, H.. Data Augmentation by Pairing Samples for Images Classification. **2018**, arXiv preprint arXiv:1801.02929.
(30) Liu, T.-Y. *Learning to Rank for Information Retrieval*; Springer Verlag: Berlin-Heidelberg, 2011.
(31) Kulis, B. Foundations and Trends in Machine Learning. *Metric Learning: A Survey*; Now Publishers, 2012; Vol. *5*, pp 287−364.
(32) Kaya, M.; Bilge, H. Ş. Deep Metric Learning: A Survey. *Symmetry* **2019**, *11*, 1066.
(33) Chicco, D. Siamese Neural Networks: An Overview. *J. Artif. Neural Network* **2021**, 73−94.
(34) Bromley, J.; Bentz, J. W.; Bottou, L.; Guyon, I.; LeCun, Y.; Moore, C.; Säckinger, E.; Shah, R. Signature Verification Using a "Siamese" Time Delay Neural Network. *Int. J. Pattern Recogn. Artif. Intell.* **1993**, *07*, 669−688.
(35) Jeon, M.; Park, D.; Lee, J.; Jeon, H.; Ko, M.; Kim, S.; Choi, Y.; Tan, A.-C.; Kang, J. ReSimNet: Drug Response Similarity Prediction Using Siamese Neural Networks. *Bioinf* **2019**, *35*, 5249−5256.
(36) Fernández-Llaneza, D.; Ulander, S.; Gogishvili, D.; Nittinger, E.; Zhao, H.; Tyrchan, C. Siamese Recurrent Neural Network with a Self-Attention Mechanism for Bioactivity Prediction. *ACS Omega* **2021**, *6*, 11086−11094.
(37) Jiménez-Luna, J.; Pérez-Benito, L.; Martínez-Rosell, G.; Sciabola, S.; Torella, R.; Tresadern, G.; De Fabritiis, G. DeltaDelta Neural Networks for Lead Optimization of Small Molecule Potency. *Chem. Sci.* **2019**, *10*, 10911−10918.
(38) Breiman, L. Bagging Predictors. *Mach. Learn.* **1996**, *24*, 123−140.
(39) Friedman, J. H. Greedy Function Approximation: a Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189−1232.
(40) Sutton, C. D. Classification and Regression Trees, Bagging, and Boosting. *Handbook of Statistics* **2005**, *24*, 303−329.
(41) Settles, B. *Active Learning Literature Survey; Technical Report TR1648*; University of Wisconsin-Madison Department of Computer Sciences: Madison, Wisconsin, 2009.
(42) Shorten, C.; Khoshgoftaar, T. M. A Survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60.
(43) Harris, C. R.; Millman, K. J.; van der Walt, S. J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N. J.; Kern, R.; Picus, M.; Hoyer, S.; van Kerkwijk, M. H.; Brett, M.; Haldane, A.; del Río, J. F.; Wiebe, M.; Peterson, P.; Gérard-Marchant, P.; Sheppard, K.; Reddy, T.; Weckesser, W.; Abbasi, H.; Gohlke, C.; Oliphant, T. E. Array Programming with NumPy. *Nature* **2020**, *585*, 357−362.
(44) Hastie, T.; Tibshirani, R.; Friedman, J.. *The Elements of Statistical Learning*; Springer Series in Statistics; Springer New York: New York, New York, 2001; Vol. *1*.
(45) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5−32.
(46) Fernández-Delgado, M.; Cernadas, E.; Barro, S.; Amorim, D. Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? *J. Mach. Learn. Res.* **2014**, *15*, 3133−3181.
(47) Wainberg, M.; Alipanahi, B.; Frey, B. J. Are Random Forests Truly the Best Classifiers? *J. Mach. Learn. Res.* **2016**, *17*, 1−5.
(48) Díaz-Uriarte, R.; De Andres, S. A. Gene Selection and Classification of Microarray Data Using Random Forest. *BMC Bioinf.* **2006**, *7*, 3.
(49) Biau, G. Analysis of a Random Forests Model. *J. Mach. Learn. Res.* **2012**, *13*, 1063−1095.
(50) Probst, P.; Boulesteix, A.-L.; Bischl, B. Tunability: Importance of Hyperparameters of Machine Learning Algorithms. *J. Mach. Learn. Res.* **2019**, *20*, 1−32.
(51) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825−2830.
(52) Seung, H. S.; Opper, M.; Sompolinsky, H. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*; Haussler, D., Ed.; Association for Computing Machinery; New York, NY: Pittsburgh, Pennsylvania, July 27-29, 1992; pp 287−294.
(53) Borisov, A.; Tuv, E.; Runger, G. In *Active Learning and Experimental Design Workshop In Conjunction with AISTATS, 16 May 2010*; Guyon, I., Cawley, G., Dror, G., Lemaire, V., Statnikov, A., Eds.; JMLR Workshop and Conference Proceedings; PMLR: Sardinia, Italy, 2011; Vol. *16*; pp 59−69.
(54) Bengio, Y.; Delalleau, O.; Le Roux, N. *The Curse of Dimensionality for Local Kernel Machines*; Technical Report, 1258, p 12.
(55) Bannwarth, C.; Ehlert, S.; Grimme, S. GFN2-xTB—An Accurate and Broadly Parametrized Self-Consistent Tight-Binding Quantum Chemical Method with Multipole Electrostatics and Density-Dependent Dispersion Contributions. *J. Chem. Theory Comput.* **2019**, *15*, 1652−1671.
(56) Chaube, S.; Goverapet Srinivasan, S.; Rai, B. Applied Machine Learning for Predicting the Lanthanide-Ligand Binding Affinities. *Sci. Rep.* **2020**, *10*, 14322.
(57) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; Zaslavsky, L.; Zhang, J.; Bolton, E. E. PubChem in 2021: New Data Content and Improved Web Interfaces. *Nucleic Acids Res.* **2021**, *49*, D1388−D1395.
(58) RDKit: Open-Source Cheminformatics. http://www.rdkit.org. (accessed: Feb 2, 2020).
(59) RDKit: Open-Source Cheminformatics. http://www.rdkit.org/docs/GettingStartedInPython.html#topological-fingerprints. (accessed: Feb 2, 2020).
(60) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742−754.
(61) 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.* **1985**, *25*, 64−73.
(62) Scalia, G.; Grambow, C. A.; Pernici, B.; Li, Y.-P.; Green, W. H. Evaluating Scalable Uncertainty Estimation Methods for Deep Learning-Based Molecular Property Prediction. *J. Chem. Inf. Model.* **2020**, *60*, 2697−2717.
(63) Shaker, M. H.; Hüllermeier, E. Lecture Notes in Computer Science. *Proceedings of the International Symposium on Intelligent Data Analysis 2020*; Berthold, M. R., Feelders, A., Krempl, G., Eds.; 2020; Vol. *12080*, pp 444−456. April 27-29 2020
(64) Huang, X.; Yang, J.; Li, L.; Deng, H.; Ni, B.; Xu, Y. Evaluating and Boosting Uncertainty Quantification in Classification. **2019**, arXiv preprint arXiv:1909.06030.
(65) Mukherjee, A.; Su, A.; Rajan, K. Deep Learning Model for Identifying Critical Structural Motifs in Potential Endocrine Disruptors. *J. Chem. Inf. Model.* **2021**, *61*, 2187−2197.
(66) Barrett, R.; White, A. D. Investigating Active Learning and Meta-Learning for Iterative Peptide Design. *J. Chem. Inf. Model.* **2021**, *61*, 95−105.

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857

(67) Konze, K. D.; Bos, P. H.; Dahlgren, M. K.; Leswing, K.; Tubert-Brohman, I.; Bortolato, A.; Robbason, B.; Abel, R.; Bhat, S. Reaction-Based Enumeration, Active Learning, and Free Energy Calculations To Rapidly Explore Synthetically Tractable Chemical Space and Optimize Potency of Cyclin-dependent Kinase 2 Inhibitors. *J. Chem. Inf. Model.* **2019**, *59*, 3782−3793.
(68) Carleo, G.; Cirac, I.; Cranmer, K.; Daudet, L.; Schuld, M.; Tishby, N.; Vogt-Maranto, L.; Zdeborová, L. Machine Learning and the Physical Sciences. *Rev. Mod. Phys.* **2019**, *91*, 045002.
(69) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. In *Proceedings of the 34th International Conference on Machine Learning*; Precup, D., Teh, Y. W., Eds.; Proceedings of Machine Learning Research; PMLR: Sydney, Australia, 06−11 Aug, 2017; *70*, pp 1263−1272.
(70) 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.
(71) Ioffe, S.; Szegedy, C. In *Proceedings of the 32nd International Conference on Machine Learning*; Bach, F., Blei, D., Eds.; Proceedings of Machine Learning Research, PMLR: Lille, France, 07−09 Jul, 2015; Vol. *37*; pp 448−456.
(72) Jones, D. R.; Schonlau, M.; Welch, W. J. Efficient Global Optimization of Expensive Black-Box Functions. *J. Global Optim.* **1998**, *13*, 455−492.
(73) Shields, B. J.; Stevens, J.; Li, J.; Parasram, M.; Damani, F.; Alvarado, J. I. M.; Janey, J. M.; Adams, R. P.; Doyle, A. G. Bayesian Reaction Optimization as a Tool for Chemical Synthesis. *Nature* **2021**, *590*, 89−96.
(74) Häse, F.; Roch, L. M.; Aspuru-Guzik, A. Gryffin: An Algorithm for Bayesian Optimization for Categorical Variables Informed by Physical Intuition with Applications to Chemistry. **2020**, arXiv preprint arXiv:2003.12127.
(75) Griffiths, R.-R.; Hernández-Lobato, J. M. Constrained Bayesian Optimization for Automatic Chemical Design Using Variational Autoencoders. *Chem. Sci.* **2020**, *11*, 577−586.
(76) Häse, F.; Roch, L. M.; Kreisbeck, C.; Aspuru-Guzik, A. Phoenics: a Bayesian Optimizer for Chemistry. *ACS Cent. Sci.* **2018**, *4*, 1134−1145.
(77) Zhan, D.; Qian, J.; Cheng, Y. Pseudo Expected Improvement Criterion for Parallel EGO Algorithm. *J. Global Optim.* **2017**, *68*, 641−662.
(78) Ioannidis, E. I.; Gani, T. Z. H.; Kulik, H. J. molSimplify: A Toolkit for Automating Discovery in Inorganic Chemistry. *J. Comput. Chem.* **2016**, *37*, 2106−2117.
(79) Nandy, A.; Duan, C.; Janet, J. P.; Gugler, S.; Kulik, H. J. Strategies and Software for Machine Learning Accelerated Discovery in Transition Metal Chemistry. *Ind. Eng. Chem. Res.* **2018**, *57*, 13973−13986.

https://doi.org/10.1021/acs.jcim.1c00670
J. Chem. Inf. Model. 2021, 61, 3846−3857