Hydration free energies from kernel-based machine learning: Compound-database bias

Rauer, C.; Bereau, T.

Link to publication

Clemens Rauer, and (iD) Tristan Bereau

COLLECTIONS

Paper published as part of the special topic on Machine Learning Meets Chemical PhysicsMACH2020

View Online   Export Citation   CrossMark

ARTICLES YOU MAY BE INTERESTED IN

Hydration free energies from kernel-based machine learning: Compound-database bias

View Online   Export Citation   CrossMark

Clemens Rauer[1] and Tristan Bereau[1,2,a]   (iD)

AFFILIATIONS

[1] Max Planck Institute for Polymer Research, 55128 Mainz, Germany

[2] Van 't Hoff Institute for Molecular Sciences and Informatics Institute, University of Amsterdam, Amsterdam 1098 XH, The Netherlands

Note: This paper is part of the JCP Special Topic on Machine Learning Meets Chemical Physics.
[a] Author to whom correspondence should be addressed: t.bereau@uva.nl

ABSTRACT

We consider the prediction of a basic thermodynamic property—hydration free energies—across a large subset of the chemical space of small organic molecules. Our *in silico* study is based on computer simulations at the atomistic level with implicit solvent. We report on a kernel-based machine learning approach that is inspired by recent work in learning electronic properties but differs in key aspects: The representation is averaged over several conformers to account for the statistical ensemble. We also include an atomic-decomposition ansatz, which offers significant added transferability compared to molecular learning. Finally, we explore the existence of severe biases from databases of experimental compounds. By performing a combination of dimensionality reduction and cross-learning models, we show that the rate of learning depends significantly on the breadth and variety of the training dataset. Our study highlights the dangers of fitting machine-learning models to databases of a narrow chemical range.

Published under license by AIP Publishing. https://doi.org/10.1063/5.0012230

I. INTRODUCTION

Applications of machine-learning (ML) models to atomic and molecular systems have had tremendous impact in our ability to tackle a more systematic exploration of chemical compound space.[1–4] Much of these developments stem from a combination of apt representations that incorporate the relevant symmetries together with flexible and expressive interpolation machines.[5–12] Work in the last few years has been devoted to the learning of electronic properties of molecules: atomization energies,[13,14] multipole moments,[15,16] the electron density,[17] or the wave function.[18]

In contrast, applications of machine learning have not transpired as much to biomolecular systems and soft matter, where configurational averages can lead to significant entropic effects.[19–22] Recent examples include developments in coarse-grained force fields,[23–26] optimizing collective variables,[27–29] as well as compound screening and optimization.[30,31]

Predicting thermodynamic properties across chemical space is of high industrial and technological relevance. Strong interests in drug design, for instance, predictions of water–octanol partitioning or protein-ligand binding, are illustrated by decades-old contributions.[32,33] Computationally efficient predictions of protein-ligand binding traditionally entail the statistical scoring of a docked ligand in a protein pocket.[34] Virtual drug discovery adopted early on a framework to correlate the molecular structure with physicochemical as well as biochemical properties.[35] More recent applications have leveraged the use of modern machine-learning techniques to improve predictive capabilities.[36] The field remains plagued by limited and noisy reference experimental data, despite efforts at improving transferability.[37] This overall hinders ML models in reaching satisfying generalization across chemical space.

At the crux of ML generalization is the breadth and variety of the chemical space spanned by the training set.[38,39] Chemical space is overwhelmingly large—supposedly up to 10^{60} drug-like molecules—making any exhaustive treatment unconceivable.[40,41] Any compound dataset, typically in the range 10^3–10^7 molecules, thereby stands as a minuscule subsampling of chemical space. How

uniform—or at least representative—can such a subsampling be? Experimental datasets suffer from biases due to both practical interests in specific interactions (e.g., hydrogen bonds) and historic developments in synthetic chemistry.[42] Recent successes in ML applications for electronic properties, on the other hand, have largely stemmed from *dense* subsets of chemical space, incorporating a rich, representative coverage over a small neighborhood. Databases such as the GDB algorithmically enumerate molecules that ought to be chemically stable,[43] unlike experimentally available compounds that are scarcely populated in chemical space.

In this work, we focus on a basic yet fundamental thermodynamic property: hydration free energies (HFEs). HFEs quantify the free energy required to transfer a solute molecule from vacuum to bulk-liquid water. We point out the existence of several recent deep ML models for hydration free energies, including AIMNet based on a density-based solvation model,[44] DeepChem,[45] which works on functional class fingerprints, and Delfos targeting different solutes and solvents.[46] The present report consists of a comparative study of the performance of kernel-based ML models against three databases:

- QM9 is based on the algorithmically grown GDB database.[47] QM9 consists of 134k molecules with up to nine heavy atoms, including chemical elements C, O, N, and F. For this study, we removed all molecules containing fluorine. We restrict our study to 4000 randomly selected compounds.
- eMolecules consists of more than 20×10^6 commercially available compounds.[48] We limit the set to up to nine heavy atoms and elements C, O, and N. From the resulting 34 517 molecules, we randomly selected 4000.
- FreeSolv consisting of around 500 molecules with experimentally available HFEs.[49] Further limiting this set to up to nine heavy elements C, O, and N reduces to 259 compounds.

Various methods exist to predict HFEs *in silico*, the main workhorse being molecular dynamics (MD) together with physics-based force fields, coupled with rigorous free-energy calculation techniques (e.g., alchemical transformations).[50] Although explicit-solvent MD simulations remain the best compromise in terms of accuracy and transferability, they remain computationally expensive, preventing us from easily generating large databases. Furthermore, setting up a protocol for accurate explicit-solvent simulations requires extreme care at odds with a screening study.[51] Instead, we turn to *implicit*-solvent MD simulations to generate our reference free energies. Implicit-solvent simulations run in the gas phase and add a Poisson–Boltzmann solvation term to the Hamiltonian.[52] They display larger errors compared to explicit-solvent simulations (2.6 kcal/mol vs 1.3 kcal/mol[53]) but at a significantly lower computational cost.

To enhance the generalization of the ML model, we explore two aspects. First, rather than feeding the representation of a single conformer, our representation averages over *several* snapshots—a proxy for the underlying configurational average. Physically, any arbitrary configuration is devoid of any statistical weight, and it is only the configurational average that ought to link to the free energy. Second, we probe the ability to learn *atomic* contributions of the free energy via an additive decomposition ansatz. Despite the absence of physical justification, we propose it in an effort to reduce the underlying

interpolation space and will empirically test its ability to improve transferability.

We first describe the theoretical setting, in particular, the kernel-based ML modeling. We describe the formalism behind the atomic-decomposition ansatz and test its transferability. Atom-decomposed ML models will be compared to simple linear regression as a baseline to better grasp the requirements on the training set. Finally, we compare the learning performance in the three databases, operate cross-learning between databases to probe their transferability, and study the breadth and variety of the spanned chemical space through dimensionality reduction.

II. METHODS

A. Linear model

To later assess the quality of our ML models, we first propose the use of linear regression as a baseline. We express the molecular free energy of hydration (HFE), G, as a linear combination of atomic contributions, weighted by the number of corresponding atoms in the compound. For molecule i, this would correspond to

$$G_i = \sum_j N_i^j g^j, \tag{1}$$

where N_i^j is the number of atoms of type j in molecule i and g^j is the contribution of atom type j to the molecular free energy. We can then write the linear system as a matrix equation $\boldsymbol{G} = N\boldsymbol{g}$, where N is the matrix of atomic contributions and $\boldsymbol{g}$ is the unknown vector of atomic contributions.

We consider two models with different numbers of atomic parameters: (i) four chemical elements (C, O, N, and H) and (ii) 39 atom types of the GAFF force field, as assigned by the force-field generating ANTECHAMBER program.[54]

B. Kernel-ridge regression

We use kernel-ridge regression (KRR) to learn the mapping $\boldsymbol{Q} \mapsto G$, where $\boldsymbol{Q}$ denotes an input molecular representation and G its corresponding HFE. The two quantities can be linked via a kernel, $\hat{K}_{ij} = \hat{K}(\boldsymbol{Q}_i, \boldsymbol{Q}_j) = \mathrm{Cov}(\boldsymbol{G}_i, \boldsymbol{G}_j)$, that encodes the similarity between inputs $\boldsymbol{Q}_i$ and $\boldsymbol{Q}_j$. Training a kernel model is equivalent to solving the set of linear equations $\boldsymbol{G} = \hat{K}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ is the vector of weight coefficients. This vector is optimized by inversion of the Tikhonov-regularized problem

$$\boldsymbol{\alpha} = (\hat{K} + \lambda \mathbb{1})^{-1} \boldsymbol{G}, \tag{2}$$

with hyperparameter λ and the identity matrix $\mathbb{1}$. Prediction for the compound $\boldsymbol{Q}^*$ is subsequently obtained through an expansion of the kernel evaluated on the training set

$$G(\boldsymbol{Q}^*) = \sum_{i=1}^N \alpha_i \hat{K}(\boldsymbol{Q}_i, \boldsymbol{Q}^*), \tag{3}$$

where index i runs over all N training points.

We use a Gaussian kernel with the Euclidean norm

$$K(\boldsymbol{Q}_i, \boldsymbol{Q}_j) = \exp\left(-\frac{\|\boldsymbol{Q}_i - \boldsymbol{Q}_j\|_2^2}{2\sigma^2}\right), \tag{4}$$

where hyperparameter σ defines the width of the kernel distribution. The molecular representation, Q, used in this work is the Spectrum of London and Axilrod-Teller-Muto (ATM) potential (SLATM).[55,56] It encodes each atom i of a molecule via a histogram of one-, two-, and three-body terms in the neighborhood of atom i, stored in one vector. Each level of interaction encodes, respectively, (i) the nuclear charge Z_i, (ii) the spectrum of radial distribution of the London potential $\rho_i^{ij}(r)$, and (iii) the spectrum of angular distribution of the ATM potential $\rho_i^{ijk}(\theta)$, where r is the distance, θ is the angle, and $i \ne j \ne k$. The representation is invariant to translation, rotation, and permutation. A molecular representation is thereby built through the concatenation of each atom vector. Both molecular and atomic variants of this representation exist and are used here. SLATM describes a single molecular configuration, while free energies inherently result from a configurational average. Therefore, we adapt the representation used to be not just the SLATM vector of one conformation but also the *average* of the SLATM vectors over 30 Boltzmann-weighted snapshots sampled every 100 ps from the gas-phase simulations used to calculate the HFEs (see Sec. II D). Working with the average, rather than a concatenation, allows us to bypass any ordering issue of the conformations in the representation vector.

C. Atomic-decomposition ansatz

The above-mentioned KRR scheme aims at the prediction of HFEs for an entire molecule at once—one pair of molecules per entry in the kernel matrix $\hat{K}$. As an alternative, we explore the possibility to learn *atomic* contributions to the HFE. We generalize our linear models (Sec. II A) such that each atomic contribution may not be strictly limited in resolution to a chemical element alone but more broadly its local environment, which we refer to here as an *atom-in-molecule* contribution. This approach will benefit from a smaller interpolation space, thereby facilitating learning.[57,58] Expressing the HFE of a molecule from atomic contributions de facto assumes a decomposition,

$$G(Q) = \sum_{l=1}^{n_{\text{atoms}}} g(q_l), \tag{5}$$

where q_l is the atom-in-molecule representation of atom l and $g(q_l)$ is its contribution to the molecular HFE, G. We will refer to Eq. (5) as the atomic-decomposition ansatz. Effectively, the aSLATM representation of q_l generalizes the concept of atom types in Eq. (1).

We aim at establishing a second mapping $q \mapsto g$ via a *local* kernel $\hat{k}$. Equation (5) links the two kernels, $\hat{K}$ and $\hat{k}$. The target properties available for the global kernel will allow us to infer an ML model for the local kernel, despite the lack of atomic target properties. We rewrite Eq. (5) as a set of N molecules with a corresponding set of M atomic contributions by introducing the mapping matrix $\hat{L}$

$$G = \hat{L}g. \tag{6}$$

The coefficient $\hat{L}_{ij}$ is 1 if molecule i contains atomic contribution j, and 0 otherwise. In case that molecule i contains n identical atomic contributions j, this would lead to a coefficient $\hat{L}_{ij} = n$, reducing the size of the matrix. This bookkeeping matrix allows us to link the

global and local kernels $\hat{K} = \hat{L}\hat{k}\hat{L}^\top$.[58] Training an ML model takes advantage of both this relationship between kernels and the linear system of equation $G = \hat{K}\alpha$,

$$\alpha = \left(\hat{L}\hat{k}\hat{L}^\top + \lambda \mathbb{1}\right)^{-1} G. \tag{7}$$

Once trained, predictions of both atomic contributions and molecular free energies are, respectively, given by

$$g^* = \left(\hat{L}\hat{k}^*\right)^\top \alpha, \tag{8}$$

$$G^* = \hat{L}^* \left(\hat{L}\hat{k}^*\right)^\top \alpha, \tag{9}$$

where $\hat{k}^*$ and $\hat{K}^*$ refer to the local and global kernels between the training and test data, respectively.

In the following, $\hat{k}$ takes the same form as $\hat{K}$ [see Eq. (4)]. For the atomic representation, we use the atomic variant of SLATM, denoted herein aSLATM.

The hyperparameters that need optimization consist of the regularization term λ in Eq. (2) as well as the length-scale normalization σ of the Gaussian kernel [Eq. (4)]. A systematic grid search led to the values $\sigma = 100$ and $\lambda = 10^{-8}$. We found the latter being very much insensitive to the accuracy of the kernel over a wide range of values. For all KRRs, the results shown are averaged over five independent runs using random train-test splits.

D. Computer simulations

Computer simulations were carried out in GROMACS 2016.4.[59] We generated initial molecular configurations using rdkit starting from their simplified molecular-input line-entry system (SMILES) string.[60] The force field was generated from the ANTECHAMBER program package with AM1-BCC atomic charges.[54,61] We ran gas-phase molecular dynamics simulations using Langevin dynamics at $T = 300$ K for 3 ns, of which we omit the first 100 ps for equilibration. We average over 29 evenly spaced snapshots of the trajectory and compute the HFE from the molecular mechanics Poisson–Boltzmann surface area (MM-PBSA), as implemented in g_mm/pbsa.[62,63] The polar contribution was obtained using the Poisson–Boltzmann solver with the vacuum, solute, and solvent dielectric constants set to $\varepsilon = 1$, 2, and 80, respectively. The non-polar solvation energy was calculated using the solvent accessible surface area method using a probe radius $r = 1.4$ Å^3, surface-tension parameter $\gamma = 0.022\,677\,8$ kJ mol^{-1} Å^{-2}, and an offset $\Delta G_{\text{corr}} = 3.849\,82$ kJ mol^{-1}.

III. RESULTS

A. Reference free energies

We first benchmark our implicit-solvent calculations against experimental HFEs. We focus on a subset of 355 molecules from the FreeSolv database, consisting solely of chemical elements C, H, O, and N (herein denoted CHON). Figure 1 shows a correlation of the HFEs from both simulations and experiments. The mean absolute error (MAE) of the implicit solvation HFEs of these molecules is 1.29 ± 1.24 kcal/mol. The standard deviation is heavily impacted
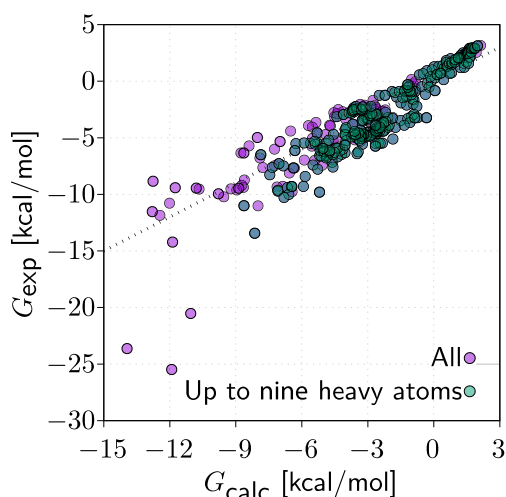
FIG. 1. Correlation of experimental, G_{exp}, and implicit-solvent, G_{calc}, hydration free energies (HFEs) for 355 CHON molecules from the FreeSolv database. The subset of points in green focuses on compounds up to nine heavy atoms. The dotted line indicates perfect agreement.

by outliers of large molecular weight. However, the databases we work with in this study contain mostly compounds up to nine heavy atoms, which feature a lower standard deviation (highlighted in Fig. 1)—MAE of 1.24 ± 0.86 kcal/mol.

We further compare the consistency of the calculated implicit-solvent and experimental free-energy datasets by comparing learning curves of ML models. KRR was applied to learn reference free energies obtained by each dataset. We use the FreeSolv dataset with up to nine heavy atoms. For both learning procedures, we rely on the same averaged aSLATM vectors obtained from the conformational ensemble of the simulations. The results for both sets of HFEs are shown in Fig. 2. The implicit-solvation free energies yield better learning performance than the experimental values. Despite a
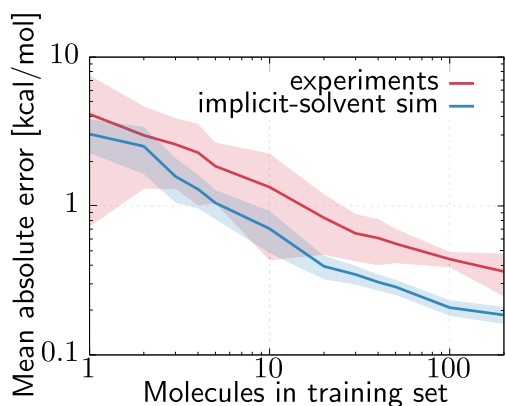
virtually identical slope, the experimental predictions are shifted *up* by roughly 0.3 kcal/mol. We point at two possible reasons: (i) This shift can be affected by the conformational sampling being identical for both curves, leading to more consistency for the calculated implicit-solvent free energies, as conformational sampling and free energies were obtained from the same set of simulations. (ii) However, it could also be caused by the experimental free energies being more heterogeneous. All in all, the results show that implicit-solvent calculations offer a reasonable proxy for the experimental values, which we rely on in the rest of this work.

In the supplementary material, we provide CSV files containing all reference free energies calculated as the basis of this work.

B. Atomic-decomposition ansatz

We set out to compare the learning performance of HFEs from the molecular representation and the atomic-decomposition ansatz exposed in Sec. II C. As a challenging test, we specifically focus on learning both types of transferability simultaneously: across databases and compound size. We first picked 663 molecules out of QM9, with 86 and 577 compounds featuring 7 and 8 heavy atoms, respectively. Using this training set, we optimized both a global and a local ML model to predict implicit-solvent HFEs. We apply the two ML models to predict the implicit-solvent HFEs of a different set of 351 molecules taken out of the FreeSolv database, which feature a broad range of molecular weights. Figure 3 displays the mean absolute error of the two ML models as a function of the number of heavy atoms in the predicted (test) compounds. Overall, we find that the global ML model $\hat{K}$ leads to significantly larger errors—up to a factor of 5 in error compared to the atomic kernel for the smallest and largest compounds. The atomic-decomposed ML model, on the other hand, features a slight improvement around seven and eight heavy atoms—used for the training set. Furthermore, it displays a remarkably flat behavior (shown on a logarithmic scale), indicating robust transferability across molecular weight. We do observe an increase in the error for molecules toward 15 heavy atoms, likely hinting at a lack of coverage of sufficient chemical environments in the training. The effect may be compounded with significant errors

FIG. 2. Learning curves for the HFE of the FreeSolv dataset using both the experimental and the calculated implicit-solvent free energies. Both curves are averaged over five independent simulations.
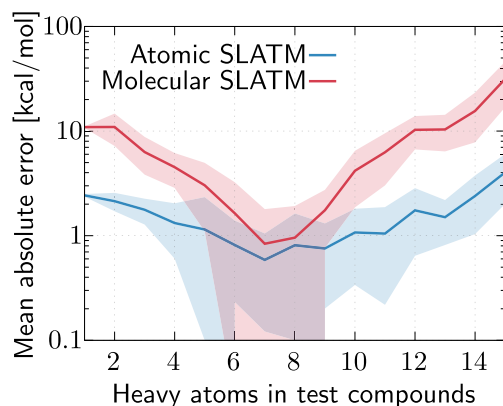

FIG. 3. Atomic and Molecular SLATM representations using a KRR model learned on 1000 QM9 compounds with seven and eight heavy atoms. Predictions on the FreeSolv database.

of the implicit-solvent method for large compounds, suggesting a lack of coherence across the molecular weight.

The results empirically justify the atomic-decomposition ansatz: the ML model using the aSLATM atom-in-molecule representation reaches better performance across molecular weights, despite the lack of formal justification for Eq. (5). Indeed, the free energy is an ensemble property of the *entire* system. A decomposition of G into finer components may be considered, for instance, via the thermodynamic integration (TI) formula.[64] TI couples the two Hamiltonians of the solute without and with the solvent

$$G = \int_0^1 d\lambda \left\langle \frac{\partial U}{\partial \lambda} \right\rangle_\lambda, \tag{10}$$

where U is the potential energy of the system, λ is a coupling parameter between the two Hamiltonians, and $\langle \cdot \rangle_\lambda$ denotes an average over the canonical ensemble at the coupling parameter λ. For the most explicit-solvent atomistic force fields, the non-bonded interactions of U are pairwise decomposable. In our case, the use of MM-PBSA clouds a simple decomposition due to the solvation term. However, establishing links between TI and an atom-decomposed ML model may help to shed light on key contributions of the free energy.

The present results indicate that any error due to the decomposition ansatz must be smaller than the accuracy of learning reached from this small dataset. The behavior of the atomic-decomposed ML model as a function of the training-set size is probed next.

C. Atomic decomposition: Rate of learning

In order to test the performance of our atomic-decomposed ML model, we take as a training set 4000 random molecules sampled from the QM9 database, for which we have calculated the HFE using MM/PBSA.

As a baseline model, we predict the HFEs using linear regression, with both the four-element CHON model and the 39 GAFF atom types. Both models were trained on the HFEs of 2000 randomly selected molecules. We report the mean absolute errors for the held-out 2000 compounds in Table I. The four-element CHON model has an MAE of 1.80 ± 1.98 kcal/mol, while the refined 39-atom-type GAFF model yields 1.06 ± 0.02 kcal/mol. As such, splitting chemical environments in the 39 atom types defined by GAFF almost yield chemical accuracy (1 kcal/mol).

These linear models offer us a way to better assess the performance of the atomic-decomposed ML model. Figure 4 compares the different regressions in terms of their out-of-sample predictions, displaying the MAE as a function of the training-set size. The atom-decomposed ML model needs 40 and 300 molecules to outperform the CHON and GAFF linear models, respectively. With 2500 molecules in the training set, the atom-decomposed ML model

TABLE I. Mean absolute error for both CHON and GAFF linear regression models. Both models were trained and tested on a 50% hold-out split of 4000 QM9 molecules.

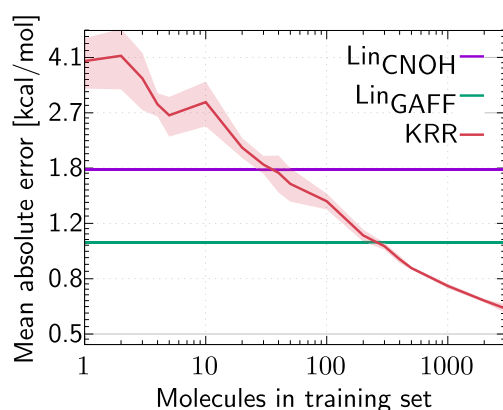| Model name	Parameters	MAE (kcal/mol)
CHON elements	4	1.80 ± 1.98
GAFF atom types	39	1.06 ± 0.02

FIG. 4. Learning curve of the hydration free energies for 4000 randomly chosen molecules of the QM9 database. The two lines show the prediction error for the CHON and GAFF linear models. KRR is the atomic-decomposed ML model.

yields an MAE of only 0.7 kcal/mol. It illustrates how offering a richer description consistently improves the performance: from chemical elements (CHON) to a select list of force-field-based atom types (GAFF) and to the atom-in-molecule aSLATM representation. The latter can be seen as a continuous generalization of the others, offering a more accurate mapping between local chemical environment and free-energy contribution.

D. Database bias

If learning HFEs across a subset of QM9 requires $\sim 10^2$ training compounds, how transferable is this result to other databases? We set the stage for this question by comparing atom-decomposed learning in three different databases, which differently try to span chemical space: QM9, eMolecules, and FreeSolv. We will more directly address the question further down in Sec. III E.

Figure 5 shows independent learning curves for the three databases. While QM9 and eMolecules show similar learning performance—the latter being slightly more performant—we
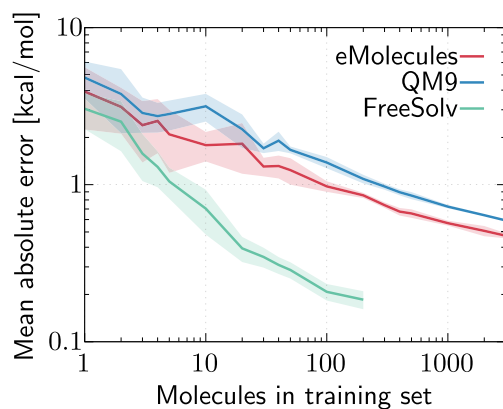

FIG. 5. Learning curves for the HFE ML models for the three databases QM9, eMolecules, and FreeSolv. All curves are averaged over five independent runs.

observe surprisingly different behavior for FreeSolv: the learning curve reaches chemical accuracy after less than 10 compounds and only 0.3 kcal/mol after 200 training molecules. The three ML models feature identical architectures and representations; the results suggest a significant bias in the nature of the subsets of chemical space they span.

To probe the difference in the spanned chemistries of the three databases, we performed dimensionality reduction. We used the UMAP algorithm.[65] UMAP builds a fuzzy topological representation in the original high-dimensional space and identifies a low-dimensional embedding by means of a cross-entropy measure. The UMAP parameters consisted of the number of neighbors (15), the minimum distance (0.1), the dimensionality of the embedding (1 or 2), and the metric (Euclidean).

To compare the three databases, we need to project them onto the same reduced subspace. A one-dimensional (1D) projection was first trained from the atomic SLATM representation on QM9 as

FIG. 6. (a) Probability distribution, $p(\varphi_1)$, of the 1D UMAP projection of the three databases. (b) Probability distribution, $p(\varphi_1, \varphi_2)$, and one-dimensional cuts of the 2D UMAP projection of the three databases QM9 (QM), eMolecules (EM), and FreeSolv (FS).

we assumed QM9 to cover chemical space more broadly than the other two since it represents a subset of the GDB. After training, we map all three databases onto this 1D projection, subsequently called φ_1. Figure 6(a) shows the probability distribution, $p(\varphi_1)$, as a measure of coverage in that subspace. The results are striking: While QM9 shows a remarkably flat distribution across the range of φ_1, eMolecules displays larger fluctuations, while FreeSolv yields high peaks, indicating significant localization within the subspace. This localization translates into significant bias in the chemical space spanned by the database—most atomic environments cluster at few points within the φ_1 subspace. This behavior explains the exceptional learning behavior shown in Fig. 5. The intermediate regime of eMolecules (between broad and spiked) indicates slight but noticeable localizations in chemical space, translating in learning performance that is slightly more favorable than QM9. As such, the commercially available database shows less diversity than the algorithmically generated database.

The same trend applies when mapping to a two-dimensional (2D) mapping of the chemical space spanned by these databases. A similar training procedure is applied as for the 1D case. Figure 6(b) shows the 2D probability distribution, $p(\varphi_1, \varphi_2)$, as well as 1D cuts thereof. The sharp peaks of FreeSolv subsist in *both* dimensions. The 2D space more explicitly illustrates the presence of "islands" in chemical space—most strongly pronounced for FreeSolv—but we also find significant differences between QM9 and eMolecules.

E. Cross-learning

To further probe the overlap between subsets of chemical space spanned by the three datasets, we performed cross-learning experiments, in which we train on one dataset and predict on another. For FreeSolv, all 259 molecules were used for the training and test sets, while for QM9 and eMolecules, we considered all 4000 molecules for training and up to 3000 randomly selected compounds for testing. Figure 7 and Table II report the cross-learning results for our ML models, which we analyze in the following.

Figure 7(a) reports cross-learning trained on the FreeSolv database. The FreeSolv → FreeSolv is identical to what was shown in Fig. 5, and cross-learning to the other databases shows a significant deterioration of the accuracy: QM9 and eMolecules saturate to roughly 1.9 kcal/mol and 1.3 kcal/mol, respectively. We recall the results from Fig. 4: the CHON and GAFF linear models reach an accuracy of 1.80 kcal/mol and 1.06 kcal/mol, respectively. As such, the FreeSolv cross-learning on QM9 is *worse* than a four-parameter linear regression. Beyond the MAEs at the highest training size, what is striking is the apparent plateau behavior after the first decade of training points: Learning improves negligibly from 10 to 10^2 data points. This aspect speaks for the lack of breadth of chemical space—the dataset features few, over-represented chemical environments. Furthermore, the small but noticeable offset between eMolecules and QM9 suggests more difficulties in learning the latter, which further hints at its broader diversity of chemical environments.

Figure 7(b) demonstrates the opposite effect: to what extent the three databases can predict FreeSolv. All three databases eventually lead to similar accuracy, albeit with different learning rates: FreeSolv is more efficient at learning itself than the others, while QM9 and eMolecules show significant offsets. This is another hint at the
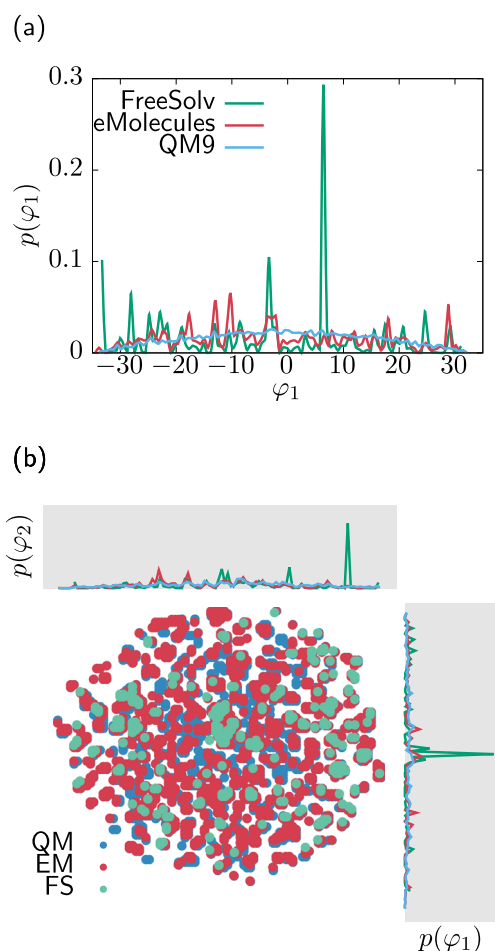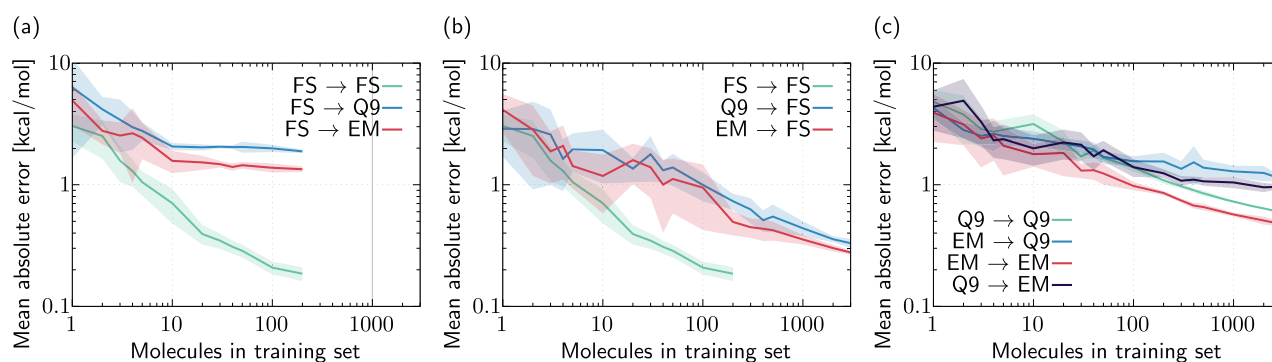
FIG. 7. [(a)-(c)] Learning curves for the cross-learning experiments: training on one dataset and prediction on another. FS, EM, and QM stand for FreeSolv, eMolecules, and QM9, respectively. The legends indicate the type of cross-learning: $X \rightarrow Y$ means training on X and prediction on Y.

TABLE II. Cross-learning machine learning models of the three databases considered in this study. All energies given in kcal/mol.

Training	Prediction		
	QM9	eMolecules	FreeSolv
QM9	0.65 ± 0.02	1.00 ± 0.03	0.36 ± 0.02
eMolecules	1.26 ± 0.05	0.61 ± 0.01	0.35 ± 0.01
FreeSolv	2.04 ± 0.06	1.26 ± 0.05	0.24 ± 0.02

broader diversity of compounds from eMolecules and, to a larger extent, QM9.

Finally, Fig. 7(c) focuses on the comparison between eMolecules and QM9. All curves roughly start at the same offset at low training data. Both self-learning curves (QM9 → QM9 and eMolecules → eMolecules) reach the lowest MAE, thanks to a slightly more favorable rate of learning. When it comes to cross-learning, QM9 shows a slight advantage at learning eMolecules than vice versa. Based on the dimensionality reduction (Fig. 6), we argue that this arises from the broader diversity of chemical environments present in QM9.

IV. CONCLUSIONS

In the present work, we study the machine learning (ML) of hydration free energies (HFEs) across a subset of small organic molecules—those made of chemical elements C, H, O, and N. To probe the effects of database biases on the learning of thermo-dynamic properties, we generated reference HFEs using implicit-solvent computer simulations at an atomistic resolution. We find that an atomic-decomposition ansatz, in which we assume a linear decomposition of the HFE in atomic contributions, offers remark-able transferability, compared to the more straightforward learn-ing of the molecular property. As a baseline, we compare two lin-ear models based on atom types, as often used in force fields. A 39-parameter model based on the GAFF atom types yields 1.06 kcal/mol. Training a better performing atom-decomposed ML

model requires a couple hundred molecules in the training set. The atom-in-molecule environment encoded in the aSLATM represen-tation offers *de facto* a generalization of the concept of force-field atom types.

ML models trained on different databases show significantly different performance. Using dimensionality reduction, we find that FreeSolv and eMolecules, two databases of commercially avail-able compounds, show strong localizations in the chemical space spanned. We can very efficiently train an ML model out of the FreeSolv database, but it fails to generalize to the other databases. Furthermore, cross-learning across databases shows that training a model on FreeSolv and deploying it on QM9 is worse than a four-parameter linear model and shows a severe plateau behav-ior, highlighting the lack of chemical diversity. The combination of cross-learning and dimensionality reduction shows that super-vised learning can help empirically establish which database probes a broader chemical space. It also shows that deploying an ML model on independent databases can help probe its generalization.

SUPPLEMENTARY MATERIAL

See the supplementary material for CSV data files for subsets of the FreeSolv,[49] eMolecules,[48] and QM9[47] databases containing SMILES strings and associated hydration free energies, as calcu-lated from atomistic simulations with implicit solvent. The data file for FreeSolv additionally contains reference experimental free energies.[49]

DATA AVAILABILITY

The data that support the findings of this study are available within the article (and its supplementary material).

REFERENCES

[1] J. Behler, "Perspective: Machine learning potentials for atomistic simulations," J. Chem. Phys. **145**(17), 170901 (2016).

[2] R. Ramakrishnan and O. A. von Lilienfeld, "Machine learning, quantum chemistry, and chemical space," Rev. Comput. Chem. **30**, 225–256 (2017).

[3] B. Sanchez-Lengeling and A. Aspuru-Guzik, "Inverse molecular design using machine learning: Generative models for matter engineering," Science **361**(6400), 360–365 (2018).

[4] O. A. von Lilienfeld, "Quantum machine learning in chemical compound space," Angew. Chem., Int. Ed. **57**(16), 4164–4169 (2018).

[5] A. P. Bartók, R. Kondor, and G. Csányi, "On representing chemical environments," Phys. Rev. B **87**(18), 184115 (2013).

[6] K. T. Schütt, F. Arbabzadah, S. Chmiela, K. R. Müller, and A. Tkatchenko, "Quantum-chemical insights from deep tensor neural networks," Nat. Commun. **8**(1), 13890 (2017).

[7] H. Huo and M. Rupp, "Unified representation for machine learning of molecules and crystals," arXiv:1704.06439, 13754 (2017).

[8] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K.-R. Müller, "Machine learning of accurate energy-conserving molecular force fields," Sci. Adv. **3**(5), e1603015 (2017).

[9] A. Glielmo, P. Sollich, and A. De Vita, "Accurate interatomic force fields via machine learning with covariant kernels," Phys. Rev. B **95**(21), 214302 (2017).

[10] A. Grisafi, D. M. Wilkins, G. Csányi, and M. Ceriotti, "Symmetry-adapted machine learning for tensorial properties of atomistic systems," Phys. Rev. Lett. **120**(3), 036002 (2018).

[11] T. Bereau, R. A. DiStasio, Jr, A. Tkatchenko, and O. A. von Lilienfeld, "Non-covalent interactions across organic and biological subsets of chemical space: Physics-based potentials parametrized from machine learning," J. Chem. Phys. **148**(24), 241706 (2018).

[12] N. Thomas, T. Smidt, S. Kearnes, L. Yang, Li Li, K. Kohlhoff, and P. Riley, "Tensor field networks: Rotation-and translation-equivariant neural networks for 3D point clouds," arXiv:1802.08219 (2018).

[13] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, "Fast and accurate modeling of molecular atomization energies with machine learning," Phys. Rev. Lett. **108**(5), 058301 (2012).

[14] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, and K.-R. Müller, "Assessment and validation of machine learning methods for predicting molecular atomization energies," J. Chem. Theory Comput. **9**(8), 3404–3419 (2013).

[15] T. Bereau, D. Andrienko, and O. A. von Lilienfeld, "Transferable atomic multipole machine learning models for small organic molecules," J. Chem. Theory Comput. **11**(7), 3225–3233 (2015).

[16] M. Gastegger, J. Behler, and P. Marquetand, "Machine learning molecular dynamics for the simulation of infrared spectra," Chem. Sci. **8**(10), 6924–6935 (2017).

[17] A. Grisafi, A. Fabrizio, B. Meyer, D. M. Wilkins, C. Corminboeuf, and M. Ceriotti, "Transferable machine-learning model of the electron density," ACS Cent. Sci. **5**(1), 57–64 (2018).

[18] K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, and R. J. Maurer, "Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions," Nat. Commun. **10**(1), 5024 (2019).

[19] T. Bereau, D. Andrienko, and K. Kremer, "Research update: Computational materials discovery in soft matter," APL Mater. **4**(5), 053101 (2016).

[20] A. L. Ferguson, "Machine learning and data science in soft materials engineering," J. Phys.: Condens. Matter **30**(4), 043002 (2017).

[21] T. Bereau, "Data-driven methods in multiscale modeling of soft matter," in *Handbook of Materials Modeling* (Springer International Publishing, 2018), pp. 1–12.

[22] N. E. Jackson, M. A. Webb, and J. J. de Pablo, "Recent advances in machine learning towards multiscale soft materials design," Curr. Opin. Chem. Eng. **23**, 106–114 (2019).

[23] S. T. John and G. Csányi, "Many-body coarse-grained interactions using Gaussian approximation potentials," J. Phys. Chem. B **121**(48), 10934–10949 (2017).

[24] J. Wang, S. Olsson, C. Wehmeyer, A. Pérez, N. E. Charron, G. De Fabritiis, F. Noé, and C. Clementi, "Machine learning of coarse-grained molecular dynamics force fields," ACS Cent. Sci. **5**(5), 755–767 (2019).

[25] H. Chan, M. J. Cherukara, B. Narayanan, T. D. Loeffler, C. Benmore, S. K. Gray, and S. K. R. S. Sankaranarayanan, "Machine learning coarse grained models for water," Nat. Commun. **10**(1), 379 (2019).

[26] A. Moradzadeh and N. R. Aluru, "Transfer-learning-based coarse-graining method for simple fluids: Toward deep inverse liquid-state theory," J. Phys. Chem. Lett. **10**(6), 1242–1250 (2019).

[27] M. M. Sultan and V. S. Pande, "Automated design of collective variables using supervised machine learning," J. Chem. Phys. **149**(9), 094106 (2018).

[28] C. Wehmeyer and F. Noé, "Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics," J. Chem. Phys. **148**(24), 241703 (2018).

[29] Y. B. Varolgunes, T. Bereau, and J. F. Rudzinski, "Interpretable embeddings from molecular simulations using Gaussian mixture variational autoencoders," Mach. Learn.: Sci. Technol. **1**, 015012 (2020).

[30] E. Y. Lee, B. M. Fulan, G. C. L. Wong, and A. L. Ferguson, "Mapping membrane activity in undiscovered peptide sequence space using machine learning," Proc. Natl. Acad. Sci. U. S. A. **113**(48), 13588–13593 (2016).

[31] C. Hoffmann, R. Menichetti, K. H. Kanekal, and T. Bereau, "Controlled exploration of chemical space by machine learning of coarse-grained representations," Phys. Rev. E **100**(3), 033302 (2019).

[32] A. J. Hopfinger and R. D. Battershell, "Application of scap to drug design. 1. Prediction of octanol-water partition coefficients using solvent-dependent conformational analyses," J. Med. Chem. **19**(5), 569–573 (1976).

[33] G. R. Marshall, "Computer-aided drug design," Annu. Rev. Pharmacol. Toxicol. **27**(1), 193–213 (1987).

[34] A. R. Leach, B. K. Shoichet, and C. E. Peishoff, "Prediction of protein-ligand interactions. Docking and scoring: Successes and gaps," J. Med. Chem. **49**(20), 5851–5855 (2006).

[35] H. Kubinyi, *3D QSAR in Drug Design: Volume 1: Theory Methods and Applications* (Springer Science & Business Media, 1993).

[36] F. Ghasemi, A. Mehridehnavi, A. Pérez-Garrido, and H. Pérez-Sánchez, "Neural network and deep-learning algorithms used in QSAR studies: Merits and drawbacks," Drug Discovery Today **23**(10), 1784–1790 (2018).

[37] H. Altae-Tran, B. Ramsundar, A. S. Pappu, and V. Pande, "Low data drug discovery with one-shot learning," ACS Cent. Sci. **3**(4), 283–293 (2017).

[38] A. Lin, D. Horvath, V. Afonina, G. Marcou, J.-L. Reymond, and A. Varnek, "Mapping of the available chemical space versus the chemical universe of lead-like compounds," ChemMedChem **13**(6), 540–554 (2018).

[39] M. Glavatskikh, J. Leguy, G. Hunault, T. Cauchy, and B. Da Mota, "Dataset's chemical diversity limits the generalizability of machine learning predictions," J. Cheminf. **11**(1), 69 (2019).

[40] C. M. Dobson, "Chemical space and biology," Nature **432**(7019), 824 (2004).

[41] J.-L. Reymond, "The chemical space project," Acc. Chem. Res. **48**(3), 722–730 (2015).

[42] R. Menichetti, K. H. Kanekal, and T. Bereau, "Drug–membrane permeability across chemical space," ACS Cent. Sci. **5**(2), 290–298 (2019).

[43] L. Ruddigkeit, R. van Deursen, L. C. Blum, and J.-L. Reymond, "Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17," J. Chem. Inf. Model. **52**(11), 2864–2875 (2012).

[44] R. Zubatyuk, J. S. Smith, J. Leszczynski, and O. Isayev, "Accurate and transferable multitask prediction of chemical properties with an atoms-in-molecules neural network," Sci. Adv. **5**(8), eaav6490 (2019).

[45] S. T. Hutchinson and R. Kobayashi, "Solvent-specific featurization for predicting free energies of solvation through machine learning," J. Chem. Inf. Model. **59**(4), 1338–1346 (2019).

[46] H. Lim and Y. Jung, "Delfos: Deep learning model for prediction of solvation free energies in generic organic solvents," Chem. Sci. **10**(36), 8306–8315 (2019).

[47] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, "Quantum chemistry structures and properties of 134 kilo molecules," Sci. Data **1**, 140022 (2014).

[48] See https://www.emolecules.com for eMolecules.

[49] D. L. Mobley and J. P. Guthrie, "FreeSolv: A database of experimental and calculated hydration free energies, with input files," J. Comput.-Aided Mol. Des. **28**(7), 711–720 (2014).

[50] D. Shivakumar, J. Williams, Y. Wu, W. Damm, J. Shelley, and W. Sherman, "Prediction of absolute solvation free energies using molecular dynamics free energy perturbation and the OPLS force field," J. Chem. Theory Comput. **6**(5), 1509–1519 (2010).

[51] Z. Gaieb, S. Liu, S. Gathiaka, M. Chiu, H. Yang, C. Shao, V. A. Feher, W. P. Walters, B. Kuhn, M. G. Rudolph et al., "D$_3$R grand challenge 2: Blind prediction of protein–ligand poses, affinity rankings, and relative binding free energies," J. Comput.-Aided Mol. Des. **32**(1), 1–20 (2018).

[52] S. Genheden and U. Ryde, "The MM/PBSA and MM/GBSA methods to estimate ligand-binding affinities," Expert Opin. Drug Discovery **10**(5), 449–461 (2015).

[53] A. Nicholls, D. L. Mobley, J. P. Guthrie, J. D. Chodera, C. I. Bayly, M. D. Cooper, and V. S. Pande, "Predicting small-molecule solvation free energies: An informal blind test for computational chemistry," J. Med. Chem. **51**(4), 769–779 (2008).

[54] J. Wang, W. Wang, P. A. Kollman, and D. A. Case, "Antechamber: An accessory software package for molecular mechanical calculations," J. Am. Chem. Soc. **222**, U403 (2001).

[55] B. Huang and O. A. von Lilienfeld, "The "DNA" of chemistry: Scalable quantum machine learning with "amons"," arXiv:1707.04146 (2017).

[56] B. Huang, N. O. Symonds, and O. A. von Lilienfeld, "Quantum machine learning in chemistry and materials," in Handbook of Materials Modeling (Springer International Publishing, 2018), pp. 1–27.

[57] M. Ceriotti, M. J. Willatt, and G. Csányi, "Machine learning of atomic-scale properties based on physical principles," in Handbook of Materials Modeling (Springer International Publishing, 2018), pp. 1–27.

[58] C. Scherer, R. Scheid, D. Andrienko, and T. Bereau, "Kernel-based machine learning for efficient simulations of molecular liquids," J. Chem. Theory Comput. **16**(5), 3194–3204 (2020).

[59] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers," SoftwareX **1-2**, 19–25 (2015).

[60] See http://www.rdkit.org for RDKit: Open-source cheminformatics.

[61] A. W. Sousa da Silva and W. F. Vranken, "ACPYPE-antechamber python parser interface," BMC Res. Notes **5**(1), 367 (2012).

[62] R. Kumari, R. Kumar, Open Source Drug Discovery Consortium, and A. Lynn, "g_mmpbsa—A GROMACS tool for high-throughput MM-PBSA calculations," J. Chem. Inf. Model. **54**(7), 1951–1962 (2014).

[63] N. A. Baker, D. Sept, S. Joseph, M. J. Holst, and J. A. McCammon, "Electrostatics of nanosystems: Application to microtubules and the ribosome," Proc. Natl. Acad. Sci. U. S. A. **98**(18), 10037–10041 (2001).

[64] J. G. Kirkwood, "Statistical mechanics of fluid mixtures," J. Chem. Phys. **3**(5), 300–313 (1935).

[65] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," arXiv:1802.03426 (2018).