Digital Discovery

Accepted Manuscript

This article can be cited before page numbers have been issued, to do this please use: F. Grasselli, V. Kapil, S. Bonfanti, K. Rossi and S. Chong, *Digital Discovery*, 2025, DOI: 10.1039/D5DD00102A.

ROYAL SOCIETY OF CHEMISTRY

rsc.li/digitaldiscovery

View Article Online
DOI: 10.1039/D5DD00102A

Journal Name

ARTICLE TYPE

Cite this: DOI: 00.0000/xxxxxxxxxx

Uncertainty in the era of machine learning for atomistic modeling†

Federico Grasselli,[*,a,b] Sanngyu Chong,[c] Venkat Kapil,[d,e,f] Silvia Bonfanti,[g,h] Kevin Rossi[*,i,j]

Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

The widespread adoption of machine learning surrogate models has significantly improved the scale and complexity of systems and processes that can be explored accurately and efficiently using atomistic modeling. However, the inherently data-driven nature of machine learning models introduces uncertainties that must be quantified, understood, and effectively managed to ensure reliable predictions and conclusions. Building upon these premises, in this Perspective, we first overview state-of-the-art uncertainty estimation methods, from Bayesian frameworks to ensembling techniques, and discuss their application in atomistic modeling. We then examine the interplay between model accuracy, uncertainty, training dataset composition, data acquisition strategies, model transferability, and robustness. In doing so, we synthesize insights from the existing literature and highlight areas of ongoing debate.

1 Introduction

Tycho Brahe, 16th century Danish astronomer, is credited for the "great care he took in correcting his observations for instrumental errors",[1] introducing the concept of measurement-theory inconsistency in astronomy, thus turning it into an empirical science. Since then, the ability to assess instrument and model errors as well as quantify the uncertainty and confidence intervals when making predictions has become a pillar of the scientific method and, in fact, discriminates between what is scientific and what is not.

In many cases, chemists and materials scientists draw conclusions based on incomplete or uncertain information, as it is often the case when dealing with expensive, time-consuming, oftentimes noisy measurements. Uncertainty quantification (UQ) provides a framework for systematically incorporating uncertainty in this scientific process, thereby enhancing the reliability, robustness, and applicability of experimental and theoretical results. In the context of materials science, chemistry, and condensed matter physics, researchers optimize materials and properties while accounting for uncertainties, variations, and errors in their measurements and theories (e.g., via replication and sensitivity analysis). This improves the reliability and validity in the models of physical phenomena and design of novel materials and processes.

Nowadays, machine learning and artificial intelligence methods are emerging as a key tools for accelerating the design, engineering, characterization, and understanding of materials, molecules, and reactions at interfaces In the context of atomistic modeling, machine learning facilitates the development of predictive models and interatomic potentials that can simulate material behavior with high accuracy and reduced computational cost compared to traditional methods. Incorporating UQ into these machine learning models is crucial for assessing the reliability of predictions and understanding the limitations of the models. By quantifying uncertainties, researchers can identify areas where the model's predictions are less certain, guide the selection of new data points for training (active learning), understand how to train robust models, and make informed decisions about the deployment of these models in practical applications.

In this perspective, we examine how the integration of machine learning and UQ enhances the predictive capabilities of atomistic simulations and ensures that inherent uncertainties are systematically accounted for, leading to more robust and reliable materials design and discovery. In this context, we acknowledge the recent contributions to the topic by Dai *et al.* and Kulichenko *et al.*.

[a] *Dipartimento di Scienze Fisiche, Informatiche e Matematiche, Università degli Studi di Modena e Reggio Emilia, 41125 Modena, Italy;federico.grasselli@unimore.it*
[b] *CNR NANO S3, 41125 Modena, Italy*
[c] *Laboratory of Computational Science and Modeling, Institute of Materials, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland*
[d] *Yusuf Hamied Department of Chemistry, University of Cambridge, Cambridge CB2 1EW, United Kingdom*
[e] *Department of Physics and Astronomy, University College London, London, United Kingdom*
[f] *Thomas Young Centre and London Centre for Nanotechnology, University College London, London WC1E 6BT, United Kingdom*
[g] *Center for Complexity and Biosystems, Department of Physics "Aldo Pontremoli", University of Milan, Via Celoria 16, 20133 Milano, Italy*
[h] *NOMATEN Centre of Excellence, National Center for Nuclear Research, ul. A. Sołtana 7, 05-400 Swierk/Otwock, Poland*
[i] *Department of Materials Science and Engineering, Delft University of Technology, 2628 CD, Delft, The Netherlands; E-mail: k.r.rossi@tudelft.nl*
[j] *Climate Safety and Security Centre, TU Delft The Hague Campus, Delft University of Technology, 2594 AC, The Hague, The Netherlands*

View Article Online
DOI: 10.1039/D5DD00102A

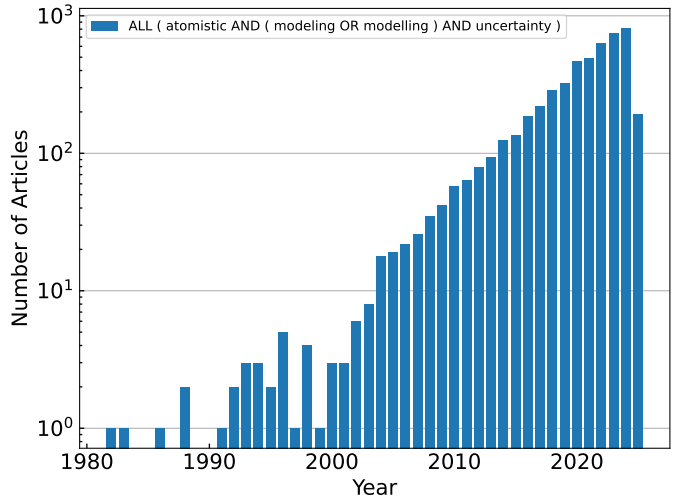

Fig. 1 Frequency (count) of articles per year corresponding to the search query reported in the figure legend: all fields of research, querying atomistic & model(l)ing & uncertainty. All fields of research refers to articles titles, abstracts, keywords, and references. Data retrieved from Scopus accessed on 06 Mar 2025.

By the same token, we remark that, while we aim for a complete discussion, we intentionally focus on a a relatively restricted number of representative works in the literature. A search for the term "Atomistic Modeling (or Modelling)" and "Uncertainty" shows that these appear with an increasing frequency, amounting to more than 5000 literature items (Figure 1), highlighting the significance of the topic under scrutiny.

The focus of our work is then on analysing recent trends in uncertainty estimation methods —such as Bayesian frameworks and ensemble approaches — and their practical application to assess prediction reliability, and guide efficient data acquisition. When possible, we synthesize and unify insights from the literature. Examples include connection interpretation of uncertainty and extrapolation measures as Mahalanobis distances and discussion on geometrical and statistical approaches to define in- and out-of distribution predictions. When consensus is lagging, we highlight key gaps and open research questions. These concern benchmarking of uncertainty quantification methods and models trasnferability, uncertainty propagation for dynamical observables; uncertainty quantification in data-efficient methods including foundation and multi-fidelity approaches. In conclusion, we aim to provide a unified framework and highlight the open questions toward robust, efficient, and interpretable machine learning approaches for atomistic modeling.

2 Uncertainty Estimates

For an uncertainty quantification method to be effective, a number of properties are desirable. In particular, the UQ methods shall be: [2]

1. *accurate*, by realistically modeling the true uncertainty associated with the ML prediction, and aiming to minimize bias and systematic errors;
2. *precise* enough to provide a sufficiently narrow range of possible values;
3. *robust*, against variations in the data or model assumptions, providing reliable results also when tested out-of-domain;
4. *traceable and comprehensive*, by capturing and identifying all the possible sources of uncertainty, which include the choice of hyperparameters and training set data points, or the stochastic optimization of non-deterministic models.
5. *computationally efficient*, requiring only a negligible overhead, possibly also in training, in obtaining the uncertainty values of interest from the ML model

In what follows, we adopt operative definitions of uncertainty based on the variance (second moment) of the distribution of predictions (either theoretical or constructed via ensembles) to quantify the spread of uncertain outcomes. This definition indeed displays the properties listed above. The analysis of first and second moment only may not be fully descriptive for non-Gaussian (e.g., skew, heavy-tailed or multi-modal) distributions. Nonetheless, it provides an interpretable and computationally lightweight measure of variability. Furthermore, it aligns well with Gaussian or near-Gaussian models, such as those that are built from the Laplace approximation (see Sec. 2.1.3). Finally, it supports simple calibration strategies that leverage the comparison of the uncertainty estimate with the second moment of the empirical distribution followed by the residuals $y_i - \tilde{y}_i$ between the reference value for input i and its ML prediction.

Finally, towards a clear and unified discussion, we spell out the notation we will adopt for our successive considerations:

- $\mathbf{x}$ (or $\mathbf{x}_i$ when labeling is needed) generic input/sample
- $\mathbf{X}$ matrix collecting inputs in the training set as rows
- $\mathbf{f}_i$ array of features corresponding to $\mathbf{x}_i$
- $\mathbf{F}$ matrix collecting training-set features as rows
- $\mathbf{w}$ parameters of the model (a.k.a. weights)
- $\tilde{y}_i \equiv \tilde{y}(\mathbf{x}_i)$ machine-learning prediction for input $\mathbf{x}_i$
- y_i reference value corresponding to input $\mathbf{x}_i$
- $\mathscr{D}$ training dataset of input-label pairs $(\mathbf{x}_i, y_i)$, with $i = 1, \ldots, N_{\text{train}}$
- σ_i^2 variance on prediction $\tilde{y}_i$
- α calibration constant (see Sec. 2.4)
- $\mathscr{L}$ training-set loss function
- ℓ_i term of the loss function corresponding to a single instance i of the training set
- $\tilde{\sigma}(\mathbf{x})$ machine-learning prediction, corresponding to input $\mathbf{x}$, for the uncertainty in mean-variance estimates and mean-variance ensembles, see Sec. 2.3.

View Article Online
DOI: 10.1039/D5DD00102A

2.1 Formulae for direct (and simple) uncertainty estimates

In the literature there exist several direct formulae to estimate the ML uncertainty on a given prediction, which make the details much dependent on the specific ML approach, e.g., linear/kernel ridge regression; full or sparse Gaussian process regression (GPR); neural-network (NN) models. Nonetheless, all these direct estimates share a common (Bayesian) interpretation. In fact, for a given new sample $\star$, the general shape of the uncertainty associated to the prediction of is in the form of a Mahalanobis (square) distance:[4]

$$\sigma_\star^2 = \alpha^2 \mathbf{f}_\star^\top \mathbf{G} \mathbf{f}_\star \tag{1}$$

i.e. the (non-Euclidean) norm of properly defined feature vector, $\mathbf{f}_\star$, that the model associates to the new sample (see also Figure 2 for additional insights). For simplicity, we assume the features have been centered, i.e. that $\frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} f_{i,a} = 0, \forall a = 1, \ldots, N_f$. The prefactor α^2 is independent of $\star$ and acts as a tuneable constant that must calibrated on some validation dataset (and also provides the correct units for the variance of the predictions). Why calibration is needed and how to calibrate uncertainty are discussed in Sec. 2.4. The shape of the positive-definite metric tensor $\mathbf{G}$ is model-dependent, but possess some common characteristics:

1. it can be viewed as an inverse covariance matrix of the properly defined features of the input data points in $\mathscr{D}$, i.e. $\mathbf{G} = [\text{cov}(\mathbf{F})]^{-1}$, where $\mathbf{F} \in \mathbb{R}^{N_{\text{train}} \times N_f}$ collects as rows the transpose of the feature vectors $\{\mathbf{f}_i\}_{i=1,\ldots,N_{\text{train}}}$ of the training points;

2. it is therefore strongly dependent on the distribution of input points $\mathbf{x}_i$ in $\mathscr{D}$ and on how much the new point $\star$ is "close" to such distribution in this metric space;

3. it is largely independent of the specific target values y_i in $\mathscr{D}$.*

We report below the specific, model-dependent expression of the features $\mathbf{f}$ and therefore of the metric tensor $\mathbf{G}$.

2.1.1 Linear regression

In a linear regression $\mathbf{f} \equiv \mathbf{x}$, so that $N_f = D$, and

$$\tilde{y}(\mathbf{x}, \mathbf{w}) = \mathbf{x}^\top \mathbf{w} \tag{2}$$

where $\mathbf{w}$ are the weights. In a Bayesian picture, if we assume the weights to be sampled from a zero-mean Gaussian prior, we have:

$$\mathbf{G} = \left(\mathbf{X}^\top \mathbf{X} + \varsigma^2 \mathbf{I}_D \right)^{-1} \tag{3}$$

where ς^2 acts as a regularizer strength and $\mathbf{I}_D$ is the identity matrix of size D equal to the number of components of (any) input $\mathbf{x} \in \mathbb{R}^D$. Therefore, the uncertainty on a prediction $\tilde{y}_\star = \tilde{y}(\mathbf{x}_\star)$ is

$$\sigma_\star^2 = \alpha^2 \mathbf{x}_\star^\top \left(\mathbf{X}^\top \mathbf{X} + \varsigma^2 \mathbf{I}_D \right)^{-1} \mathbf{x}_\star \tag{4}$$

* The only dependence on the specific target quantity and values is through the value of the regularizer that is included to make the inversion of the covariance matrix numerically stable.

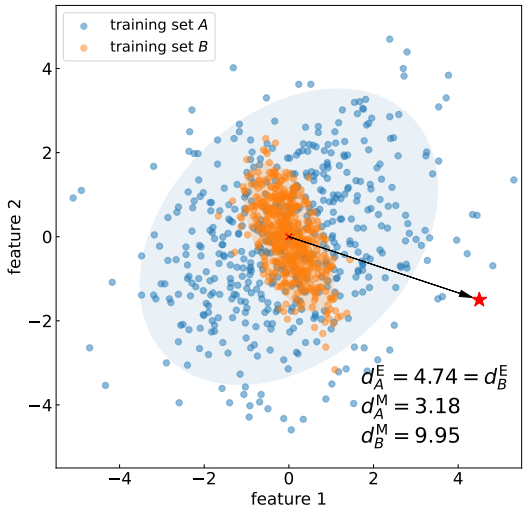

Fig. 2 Mahalanobis distance. The new sample $\star$ has equal Euclidean distance d^E between the distribution of features in training set A (blue), characterized by a large covariance, and the distribution of features in training set B (orange), characterized by a smaller covariance. The shaded ellipses have axes equal to the eigenvalues of the covariance. In striking contrast to Euclidean distance, the Mahalanobis distance of $\star$ from the distribution of features in training set B, d_B^M, is more than three times larger than that from A, d_A^M.

2.1.2 Gaussian process regression

In the GPR problem $\mathbf{f} \equiv \boldsymbol{\phi}(\mathbf{x})$, i.e. the regression exercise has the form

$$\tilde{y}(\mathbf{x}) = [\boldsymbol{\phi}(\mathbf{x})]^\top \mathbf{w}, \tag{5}$$

where $\boldsymbol{\phi}(\mathbf{x})$ maps the D-dimensional input $\mathbf{x}$ into a N_f-dimensional feature space (in general, as a nonlinear function of the input). The components $\phi^a(\mathbf{x})$, with $a = 1, \ldots, N_f$ are often called *basis functions*.

By assuming again that the weights are sampled from a zero-mean Gaussian prior, we have:

$$\mathbf{G} = \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \varsigma^2 \mathbf{I}_{N_f} \right)^{-1} \tag{6}$$

and the uncertainty on a prediction $\tilde{y}_\star = \tilde{y}(\mathbf{x}_\star)$ is

$$\sigma_\star^2 = \alpha^2 \boldsymbol{\phi}(\mathbf{x}_\star)^\top \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \varsigma^2 \mathbf{I}_{N_f} \right)^{-1} \boldsymbol{\phi}(\mathbf{x}_\star) \tag{7}$$

Oftentimes the reported GPR uncertainty formula is:

$$\sigma_\star^2 = k(\mathbf{x}_\star, \mathbf{x}_\star) - \mathbf{k}(\mathbf{x}_\star, \mathbf{X})^\top \left[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \alpha^2 \mathbf{I}_{N_{\text{train}}} \right]^{-1} \mathbf{k}(\mathbf{x}_\star, \mathbf{X}) \tag{8}$$

which makes use of the kernel that, for any pair of inputs $\mathbf{x}_i$ and $\mathbf{x}_j$, is here defined as in Rasmussen and Williams: $k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_w^2 [\boldsymbol{\phi}(\mathbf{x}_i)]^\top \boldsymbol{\phi}(\mathbf{x}_j)$, where σ_w^2 is the variance of the prior distribution of the weights, which can be identified in α^2/ς^2. In some references—such as Tipping, which forms the basis for Appendix A—the factor σ_w^2 is either omitted or absorbed into the feature definition via the rescaling $\boldsymbol{\phi} \leftarrow \sigma_w \boldsymbol{\phi}$. In this Perspective, we have chosen to adhere as closely as possible to the conventions and definitions adopted in the referenced papers, to main-

View Article Online
DOI: 10.1039/D5DD00102A

tain consistency and facilitate comparison. To switch from Eq. (7) to Eq. (8) or vice versa, it is sufficient to use Woodbury's identity, after assuming all the needed matrix inversions are possible. In this sense, the role of the regularizer is crucial: in fact, whenever $\mathbf{\Phi}$ is not full rank, either $\mathbf{\Phi}^\top\mathbf{\Phi}$ is invertible and $\mathbf{\Phi}\mathbf{\Phi}^\top$ is not (case of "tall" matrix $\mathbf{\Phi}$, with $N_{\text{train}} > N_f$), or $\mathbf{\Phi}\mathbf{\Phi}^\top$ is invertible and $\mathbf{\Phi}^\top\mathbf{\Phi}$ is not (case of "broad" matrix $\mathbf{\Phi}$, with $N_{\text{train}} < N_f$). For the relation between the eigenvalues/-vectors of $\mathbf{\Phi}^\top\mathbf{\Phi}$ and $\mathbf{\Phi}\mathbf{\Phi}^\top$, see Tipping.

Mercer's theorem ensures that for any kernel there exist a possibly infinite (i.e. $N_f \to \infty$) set of basis functions. Furthermore, the representation in terms of the functions $\boldsymbol{\phi}$ is also very useful when sparse kernel approximations, such as the Nyström method detailed in Appendix A, are employed.

2.1.3 Neural networks

Expressions analogous to Eq. (1) have appeared for neural networks several decades ago, in the work of MacKay in the early '90s[7–9], introducing Laplace approximation within the context of Bayesian approach to neural networks. The Laplace approximation consists in approximating the posterior distribution of the weights as a multivariate Gaussian distribution, centered around the maximum a posteriori (MAP) *optimal* weights $\mathbf{w}_o$, that are obtained after the NN training:

$$p(\mathbf{w}|\mathscr{D}) \approx \mathscr{N}\left(\mathbf{w}_o, \alpha^2\mathbf{G}\right) \tag{9}$$

where $\alpha^2\mathbf{G}$ is the covariance matrix of the weights close to MAP, and the tuneable parameter α may be interpreted as a noise level on observation[9]. The discrepancy principle would indicate the mean square error of the observations[10] as an empirical estimate of α^2; nonetheless, the latter is often treated as a tuneable parameter, since additional calibrations are often required, as reported in, e.g., MacKay and Imbalzano *et al.*. In the Laplace approximation, the matrix $\mathbf{G}$ is given by the inverse Hessian matrix of loss function $\mathscr{L} = \sum_i \ell_i(\tilde{y}(\mathbf{x}_i, \mathbf{w}), y_i)$, computed at MAP:

$$\mathbf{H}_o = \left.\frac{\partial^2 \mathscr{L}}{\partial \mathbf{w}\, \partial \mathbf{w}^\top}\right|_{\mathbf{w}_o} = \sum_{i=1}^{N_{\text{train}}} \left.\frac{\partial^2 \ell_i}{\partial \mathbf{w}\, \partial \mathbf{w}^\top}\right|_{\mathbf{w}_o} \tag{10}$$

$\mathbf{H}_o$ is routinely approximated by its Gauss-Newton form, which employs only first order derivatives of predictions with respect to the weights and evaluated at MAP, $\boldsymbol{\phi}_i \equiv \left.\frac{\partial \tilde{y}_i}{\partial \mathbf{w}}\right|_{\mathbf{w}_o}^\top$ which can be easily retrieved by backpropagation:

$$\mathbf{H}_o \approx \sum_{i=1}^{N_{\text{train}}} \boldsymbol{\phi}_i \frac{\partial^2 \ell_i}{\partial \tilde{y}_i^2} \boldsymbol{\phi}_i^\top \tag{11}$$

Finally, the distribution of the output corresponding to the input $\star$ becomes:[12]

$$p(y_\star|\mathbf{x}_\star, \mathscr{D}) \approx \mathscr{N}\left(\tilde{y}(\mathbf{x}_\star, \mathbf{w}_o); \alpha^2\, \boldsymbol{\phi}_\star^\top \mathbf{H}_o^{-1} \boldsymbol{\phi}_\star\right) \tag{12}$$

from which the variance $\sigma_\star^2 = \alpha^2\, \boldsymbol{\phi}_\star^\top \mathbf{H}_o^{-1} \boldsymbol{\phi}_\star$ is obtained in the form of a Mahalanobis distance, Eq. (1). In the context of atomistic modeling, this approach has been investigated by Zaverkin and Kästner, and leveraged for active-learning strategies (see also Sec. 3.5). The same results can be obtained in an alternative but equivalent mathematical construction probing how robust a ML model is, to a change in the prediction of an input $\star$, based on a constrained minimization of the loss.[14,15] (see Sec. 3.4 for more details).

Two remarks should be made: first, as it is reasonable to expect, the quality of this approximation depends on how much the posterior distribution of the NN model is close to a multivariate Gaussian; second, the large number of weights, and thus of components in $\boldsymbol{\phi}_i$, in current deep NN typical architectures makes the storage of $\mathbf{H}_o$ unfeasible, even in the Gauss-Newton approximation, due to memory requirements quadratic in the size of the $\boldsymbol{\phi}$ arrays. The context of NN Gaussian processes,[16] and in particular of the Neural Tangent Kernel formalism,[17–19] provides the ideal theoretical framework to justify the first point and to find a viable strategy to overcome the second one. In Bigi *et al.*, the use of a last-layer (LL) approximation,[12,20] was extensively justified, whereby only the derivatives of the predictions with respect to the LL weights $\mathbf{w}^L$, i.e. the LL latent features

$$\mathbf{f}_i \equiv \left.\frac{\partial \tilde{y}_i}{\partial \mathbf{w}^L}\right|_{\mathbf{w}_o}^\top \tag{13}$$

are considered in building $\mathbf{H}_o$ and in evaluating the variance of the prediction for a new input $\star$. For a mean square loss function, the latter becomes:†

$$\sigma_\star^2 = \alpha^2 \mathbf{f}_\star^\top \left(\mathbf{F}^\top\mathbf{F} + \varsigma^2 \mathbf{I}_{N_L}\right)^{-1} \mathbf{f}_\star \tag{14}$$

where a regularizer has been added for numerical stability and where N_L is the number of components of LL latent features, i.e. the number of nodes of the last hidden layer of the NN. A few further remarks:

1. The presented derivation is based on the assumption that no additional nonlinear activation is applied to the product of the LL latent features and LL weights, i.e. that $\tilde{y} = \mathbf{f}^\top\mathbf{w}^L$. Things get more complicated whenever a nonlinear application function φ is instead applied, $\tilde{y} = \varphi(\mathbf{f}^\top\mathbf{w}^L)$, even though the correct distribution of the prediction can in principle still be sampled (e.g., by Monte Carlo integration).

2. The LL latent features $\mathbf{f}_i$ do not explicitly depend on the LL weights $\mathbf{w}^L$, which are the weights reported to change more during training.[17] As such, the LL latent features, as well as the covariance matrix $\mathbf{F}^\top\mathbf{F}$, are expected to be rather constant during training.[18] Furthermore, the different elements of the array of LL latent features are identically distributed at initialization, and centered around zero, for *any* given sample, because the weights (and in particular $\mathbf{w}^{L-1}$) are taken as independent, identically distributed and centered around zero. This implies that the additional enforcement of feature centering should not change the result of the uncertainty estimate.

† To avoid notation overburden, we used the same symbol α^2 in both Eqs. (12) and (14) for the calibration parameters, although there is no reason for them to be equal.

View Article Online
DOI: 10.1039/D5DD00102A

3. Numerical experiments indicate that, analog to linear regressions, while the calibration of α^2 is crucial, the regularizer scarcely affects $\sigma_\star^2$ even when the regularizer strength ς^2 is varied over several orders of magnitude, unless data are scarce or highly collinear. The role of the regularizer is further discussed in Appendix E of Ref.[15].

Wenger *et al.* report on limitations of the NTK perspective. In particular, to leverage the advantages of the NTK formulation, one would need "*architectures that are orders of magnitude wider than they are deep*" (verbatim from Ref. Wenger *et al.*). This is in fact the case for several current architectures of ML models in atomistic learning. ‡ Note, the NTK theory is valid for any N_{train} in the infinite-width limit, but, for finite-width networks, the width must grow sufficiently fast (polynomially) with the number of samples to maintain the NTK approximation valid

Beyond last-layer approximations, it has been shown in the context of atomistic machine learning that full-gradient representations, combined with random projections to reduce memory cost, can offer improved performance.[24] For a broader perspective on gradient features and their connection to the Neural Tangent Kernel, we refer the interested reader also to Holzmüller *et al.*.

2.1.4 Bayesian methods beyond the Laplace approximation

As already discussed, Bayesian UQ scope is to find the probability distribution of the output (a.k.a. posterior predictive distribution) by means of Bayes' rule

$$p(y_\star|\mathbf{x}_\star,\mathscr{D}) = \int \mathrm{d}\mathbf{w} p(y_\star|\mathbf{x}_\star,\mathbf{w}) p(\mathbf{w}|\mathscr{D}) \tag{15}$$

from which one can quantify the uncertainty as the second moment of the distribution. When Laplace approximation is invoked, one can obtain simple expressions like Eq. (12). Whenever this is not possible, an explicit sampling of the posterior

$$\begin{aligned} p(\mathbf{w}|\mathscr{D}) &\propto \exp\left[-\frac{\mathscr{U}(\mathbf{w})}{\mathscr{T}}\right] \\ \mathscr{U}(\mathbf{w}) &= -\ln \underbrace{\prod_{i=1}^{N_{\text{train}}} p(y_i|\mathbf{x}_i,\mathbf{w})}_{\text{likelihood}, p(\mathscr{D}|\mathbf{w})} - \ln \underbrace{p(\mathbf{w})}_{\text{prior}} \end{aligned} \tag{16}$$

is needed. Here, $\mathscr{T}$ represents the posterior "temperature", introduced as an additional hyperparameter. The true Bayesian posterior is obtained when $\mathscr{T}=1$;[7] when $\mathscr{T}<1$ such a "cold" posterior is a sharper distribution.[26] The sampling is usually performed via Markov chain Monte Carlo (MCMC) methods. In order to generate proposals for the Monte Carlo acceptance step, state-of-the-art techniques often leverage Hamiltonian-like dynamics, whereby the parameters $\mathbf{w}$ are evolved according to the "forces" $-\boldsymbol{\nabla}_{\mathbf{w}}\mathscr{U}(\mathbf{w})$.[27] While standard Hamiltonian MCMC methods may be computationally intractable for current NN architectures featuring a huge number of parameters, stochastic-gradient MCMC algorithms have been recently devised and applied to NN ML interatomic potentials,[28,29] which involve data mini-batching and give results comparable to (deep) ensemble methods (see Sec. 2.2). Finally, last-layer variants of variational Bayesian approaches,[30], providing a sampling-free, single-pass model that enhances UQ, have not yet been explored in atomistic machine learning but could offer a promising direction.

‡ For instance, the default fitting NN architecture of DeePMD is made by 3 layers of 240 neurons each.[22] Even in the earliest Behler-Parrinello architectures the number of nodes (40) was much larger than the number of layers (2).[23]

2.2 Ensembles of models

Another class of approaches to quantify the ML uncertainty exists, which is based on the generation of an ensemble of several equivalent models $y^{(m)}(\mathbf{x})$, to compute the empirical mean

$$\bar{y}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M} y^{(m)}(\mathbf{x}) \tag{17}$$

and variance

$$\sigma^2(\mathbf{x}) = \frac{1}{M-1}\sum_{m=1}^{M} [y^{(m)}(\mathbf{x}) - \bar{y}(\mathbf{x})]^2 \tag{18}$$

for the prediction corresponding to any given input of features $\mathbf{x}$.

The ensemble members $y^{(m)}(\mathbf{x})$ are routinely obtained in different ways:

1. by subsampling the entire dataset and then training one model for each of the subsampled datasets $\mathscr{D}^{(m)}$. The size of $\mathscr{D}^{(m)}$ depends on the subsampling technique, but usually amounts to $\frac{M-1}{M}\times N_{\text{train}}$.

2. for models that are not trained via a deterministic approach, stochasticity in the model architecture and training details (e.g., varying the random seed, Monte Carlo dropout[31]) can be exploited to obtain the ensemble.

2.3 Mean-variance estimation models and Mean-variance ensembles

The goal of mean variance (MV) estimation models is to predict the uncertainty $\tilde{\sigma}^2(\mathbf{x})$ affecting a given prediction $\tilde{y}(\mathbf{x})$ together with the prediction itself. Different from ensembles, here the model is trained to directly predict the best value and its variance, rather than building an ensemble to deduce them; see also the region enclosed by the dashed line in Fig. 3(a). MV estimation models are usually trained by using a negative log-likelihood loss function,[32] that, for a single instance $\mathbf{x}_i$, reads

$$\begin{aligned} \ell_i &= -\ln p(y_i|\mathbf{x}_i,\mathbf{w}) \\ &= \frac{1}{2}\left[\frac{(y_i-\tilde{y}(\mathbf{x}_i))^2}{\tilde{\sigma}^2(\mathbf{x}_i)} + \ln\tilde{\sigma}^2(\mathbf{x}_i) + \ln 2\pi\right] \end{aligned} \tag{19}$$

Busk et al.[33] interpret $\tilde{\sigma}^2(\mathbf{x})$, predicted by MV model as an additional model output, as an *aleatoric* uncertainty that may stem from random noise, data inconsistencies, or the model's inability to fit precisely, and, as such, cannot typically be reduced by collecting more data. Incidentally, it should be remarked that, in the

View Article Online
DOI: 10.1039/D5DD00102A

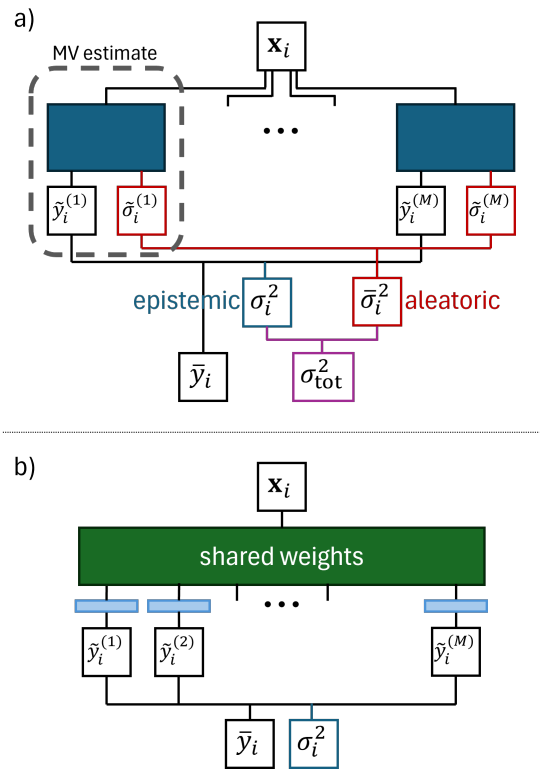

Fig. 3 a) Mean variance ensemble model. b) Shallow ensemble.

case of a dataset that assumes *noiseless* observations, all the deviations of the model's prediction from the observations must be considered as model's bias,[34] since considering the observations to be the ground truth implies that a perfect model should really pass through those points.

Ensembles models and MV estimation models can be combined to give rise to mean-variance *ensembles*, introduced by Lakshminarayanan *et al.* with the name of deep ensembles; see Fig. 3(a). In this approach, a committee of MV estimation models is created, where each member of the committee outputs a prediction and variance pair, $\left(\tilde{y}^{(m)}(\mathbf{x}), \tilde{\sigma}^{2,(m)}(\mathbf{x})\right)$, and then the uncertainty is estimated by summing the variance of the predictions, as in Eq. (18), with the average of the variances of the committee:

$$\sigma_{\text{tot}}^2(\mathbf{x}) = \frac{1}{M-1}\sum_{m=1}^{M}[y^{(m)}(\mathbf{x}) - \bar{y}(\mathbf{x})]^2 + \frac{1}{M}\sum_{m=1}^{M}\tilde{\sigma}^{2,(m)}(\mathbf{x}) \tag{20}$$

which assumes that the two addenda are uncorrelated. Busk *et al.* and Carrete *et al.* interpret the first addendum as the epistemic uncertainty and the second addendum as aleatoric uncertainty.

2.3.1 Shallow ensembles

While the mean-variance ensemble approach is known to provide robust uncertainty estimates and is commonly considered the current state-of-the-art, it often suffers from the large computational cost incurred from training and evaluation of multiple models. Given that the commonly adopted ensemble size is $\geq$ 5–10, it can quickly become prohibitive for sufficiently large and complex models, especially neural networks.

In this latter case, akin to LL approximation motivated in 2.1.3, one could construct "shallow ensembles" (Figure 3(b)), where only the last-layer of the neural network is varied in obtaining an ensemble of models, and rest of the weights are shared across the members. Such an approach effectively mitigates the large computational cost associated with the training of multiple neural network models and their inference. Kellner and Ceriotti present a version of this approach, where a shallow ensemble of models is trained to using an NLL-like loss obtain the empirical mean and variance through Eqs. (17) and (18).

2.3.2 Mixtures of experts

An emergent approach across numerous machine learning domains, towards highly accurate and efficient models, concerns the adoption of ensemble learning and mixture of experts (MoE) strategies.[38] In the context of atomistic modeling these have been explored by Zeni *et al.* for monoelemental systems and Wood *et al.* for efficient universal models for atoms. The formulae discussed in the previous section can be easily extended to mixture-of-experts models, where the prediction for a sample $\star$ is

$$\tilde{y}(\mathbf{x}_\star) = \sum_{k=1}^{K}\pi_k(\mathbf{x}_\star)\,\tilde{y}^{(k)}(\mathbf{x}_\star) \tag{21}$$

Here, $\pi_k(\mathbf{x})$ is an input-dependent coefficient representing the contribution of model k of the mixture. The total number of models is K. The coefficients π_k are normalized so that $\sum_{k=1}^{K}\pi_k(\mathbf{x}) = 1, \forall \mathbf{x}$. Examples include soft-max assignment based on distance:[39]

$$\pi_k(\mathbf{x}) = \frac{e^{s(d_k(\mathbf{x}))}}{\sum_{k'=1}^{K} e^{s(d_{k'}(\mathbf{x}))}} \tag{22}$$

where $s(d_k(\mathbf{x}))$ labels a function of the reciprocal of the distance $d_k(\mathbf{x})$ between the point $\mathbf{x}$ and the centroid of the model dataset k. Other smooth space-partitioning functions based on density have been similarly suggested:[38]

$$\pi_k(\mathbf{x}) = \frac{p^{(k)}(\mathbf{x})}{\sum_{k=1}^{K} p^{(k)}(\mathbf{x})} \tag{23}$$

where $p^{(k)}(\mathbf{x})$ is the probability density to find $\mathbf{x}$ according to model k of the mixture (e.g., in the case of Gaussian mixture models $p^{(k)}$ is the probability density function of a normal distribution defined by the k-th cluster's center and covariance). By assuming that different models of the mixtures are independent among one another and characterized by the uncertainties

$$\sigma_\star^{2,(k)} = \alpha^{2,(k)}\mathbf{f}_\star^\top \mathbf{G}^{(k)}\mathbf{f}_\star, \qquad k = 1, \ldots, K \tag{24}$$

then, standard uncertainty propagation gives:

$$\sigma_\star^2 = \sum_{k=1}^{K}\left[\frac{\partial \tilde{y}_\star}{\partial \tilde{y}_\star^{(k)}}\right]^2 \sigma_\star^{2,(k)} = \sum_{k=1}^{K}\pi_k^2\sigma_\star^{2,(k)} \tag{25}$$

View Article Online
DOI: 10.1039/D5DD00102A

2.4 Calibration: quantification and practicalities

2.4.1 Maximum log-likelihood calibration

Musil *et al.* and Imbalzano *et al.* have shown that the uncertainty $\sigma^2(\mathbf{x})$ estimated by the ensemble-based approach of Eq. (18) can be calibrated *a posteriori* by applying a global (i.e., $\mathbf{x}$-independent) scaling factor α^2, such that $\sigma^2_{\text{calib.}}(\mathbf{x}) \leftarrow \alpha^2\sigma^2(\mathbf{x})$. α^2 is chosen to maximize the log-likelihood of the predictive distribution over a validation set of N_{val} points, and is given by:

$$\alpha^2 = \frac{1}{N_{\text{val}}}\sum_{i=1}^{N_{\text{val}}}\frac{|y_i - \bar{y}(\mathbf{x}_i)|^2}{\sigma^2(\mathbf{x}_i)} \tag{26}$$

It is crucial to remark that Eq. (26) is a biased estimator in the number of models composing the ensemble, M. Whenever M is small, Eq. (26) should be replaced by the bias-corrected formula

$$\alpha^2 = -\frac{1}{M} + \frac{M-3}{M-1}\frac{1}{N_{\text{val}}}\sum_{i=1}^{N_{\text{val}}}\frac{|y_i - \bar{y}(\mathbf{x}_i)|^2}{\sigma^2(\mathbf{x}_i)} \tag{27}$$

from which it is also seen that at least $M = 4$ members are needed. With proper (straightforward) changes, this approach can be easily extended to the other UQ techniques outlined in the previous sections. Notice that tracing back a clear distinction between epistemic and aleatoric components of the uncertainty, as outlined in Eq. (20), may be problematic after calibration.

2.4.2 Expected vs observed uncertainty parity plots

Another common approach to check UQ calibration involves constructing parity plots, typically on a log-log scale, to compare the estimated variance with the observed distribution of (squared) residuals, sometimes binned according to the model's estimated variance.[2,15,16,37,42] A proxy to summarize these *reliability plots* is the so-called expected normalized calibration error (ENCE), recently introduced by Levi *et al.*, defined by:

$$\text{ENCE} = \frac{1}{N_{\text{bins}}}\sum_{b=1}^{N_{\text{bins}}}\frac{|\text{RMV}(b) - \text{RMSE}(b)|}{\text{RMV}(b)}$$

$$\text{RMV}(b) \equiv \sqrt{\frac{1}{|b|}\sum_{i\in b}\sigma^2(\mathbf{x}_i)} \tag{28}$$

$$\text{RMSE}(b) \equiv \sqrt{\frac{1}{|b|}\sum_{i\in b}|y_i - \bar{y}(\mathbf{x}_i)|^2}$$

where $|b|$ labels the number of data points in bin b.

If the estimated variance $\sigma^2(\mathbf{x}_i)$ follows a functional form such as Eq. (1), the free parameter α^2 must be adjusted, for example, following the procedure of Eq. (26)—to align the expected variance with the observed MSE. Notably, in log-log space, varying α^2 results in a rigid shift of the entire plot. Thus, even for uncalibrated UQ estimates, a linear correlation between the expected and observed distributions of (squared) residuals should still be apparent. If there is a poor linear correlation, this suggests that the Gaussian-like UQ framework defined by Eq. (1) may be inadequate, as can occur in NN models with only few neurons per layer, and/or that a *local* calibration $\alpha^2(\mathbf{x})$ may be necessary.

Along these lines it is worth mentioning the insightful work by Pernot, Pernot which involves stratifying the evaluation of a given calibration score, such as Eq. (26), based on the predicted uncertainty: the dataset is split into *subsets* where the predicted uncertainty falls within certain ranges (e.g., low, medium, or high uncertainty predictions); calibration metrics are then calculated independently for each subset. This allows for a more detailed analysis of how well the model is calibrated across different levels of predicted uncertainty. By performing different types of stratification/binning of the dataset, Pernot shows that it is also possible to distinguish between *consistency* (the conditional calibration with respect to prediction uncertainty) and *adaptivity* (the conditional calibration with respect to input features), and that good consistency and good adaptivity are rather distinct and complementary calibration targets.

2.4.3 Miscalibration area

In the literature,[46] (mis)calibration is often discussed in terms of the similarity between the expected and observed cumulative distribution functions (CDFs) of residuals. A model is considered calibrated when the miscalibration area between these CDFs is small, indicating consistency between the predicted and actual uncertainties. The sign of the miscalibration area can further reveal whether the model's estimated uncertainties are under- or over-confident, providing additional diagnostic insight.

Nonetheless, methods based on the miscalibration area can be challenging to interpret. In fact, they provide an indirect assessment of UQ quality, since, by comparing CDFs, they inherently compare higher moments of the distributions. For example, two distributions may share similar second moments–quantities typically interpreted as uncertainty—but diverge significantly in their CDFs due to deviations from Gaussianity, such as skewness or heavy tails. This conflation of uncertainty with other aspects of distribution's shape highlights a key limitation of such approaches.

For this reason, we recommend prioritizing the calibration strategies discussed in Secs. 2.4.1 and 2.4.2, which more clearly align with the intended interpretation of uncertainty.

2.5 Conformal prediction

Most of the UQ strategies described so far employ input-dependent uncertainties whose evaluation are largely independent of the values of the targets in the dataset (see point 3. in Sec. 2.1). A complementary and rather opposite idea is based on conformal predictions (CPs), which—for regression tasks—provide a way to construct *confidence intervals* for a continuous target prediction $\tilde{y}_\star$, such that the intervals contain the true value with a predefined probability (e.g., 95%). Notably, CPs offers a distribution-free approach to uncertainty quantification. The idea of CPs stems from seminal concepts developed by Ronald Fisher in the 1930s,[47] and then applied to the context of ML by Vladimir Vovk and collaborators in the 1990s (for a pedagogical review, see e.g. Shafer and Vovk). In a nutshell, the CP procedure reads as follows:

- train a regression model to give predictions $\tilde{y}_i$
- use a separate calibration dataset $\mathscr{C}$

View Article Online
DOI: 10.1039/D5DD00102A

- compute the *nonconformity score* $s_{i\in\mathscr{C}}$ (typically the absolute error, $s_i = |y_i - \tilde{y}_i|$)
- determine the $(1-\alpha)$-quantile $q_{1-\alpha}$ of the sorted scores. For instance, if $\alpha = 0.05$, then 95% of the scores $s_{i\in\mathscr{C}}$ shall lie below the value $q_{0.95}$
- for a new input $\mathbf{x}_\star$ construct the prediction interval as $[\tilde{y}_\star - q_{1-\alpha}, \tilde{y}_\star + q_{1-\alpha}]$.

The prediction interval contains the true y with at least $1-\alpha$ confidence. The key assumption in CP is that the new (test set's) error distribution is representative of both the training and calibration sets error distribution. This allows $1-\alpha$, derived from the calibration data $\mathscr{C}$, to apply universally across all inputs $\mathbf{x}$. As a result, $q_{1-\alpha}$ acts as a *global* threshold for constructing prediction intervals, a measure of the model's "typical error" that encapsulates how large the prediction errors tend to be (up to the $(1-\alpha)$-quantile), independent of any specific input $\mathbf{x}$. Nonetheless, if the new set of inputs is significantly out of distribution, meaning it differs substantially from the training and calibration data, the assumptions underlying CP may no longer hold, and the prediction intervals obtained from CP might lose their validity.

In the context of atomistic modeling of materials, CPs have been recently used, e.g., in Hu *et al.* and Zaverkin *et al.*, for the UQ of ML interatomic potentials predictions.

2.6 Are all uncertainty estimates the same?

Standardized benchmarks and evaluation protocols for ML model accuracies in atomistic modeling have been established only recently. Shortly after, limitations of these benchmarks where also identified, and this continues to be an area of ongoing research. Similarly, a consensus on UQ methods reliability is still lacking, since no unique set benchmarks for uncertainty estimation has been developed yet.

Tran *et al.* compared the accuracy and uncertainty of multiple machine learning approaches for predicting the adsorption energy of small molecules on metals. The most effective approach combines a convolutional neural network (CNN) for feature extraction with a Gaussian process regressor (GPR) for making predictions. This hybrid model not only provided accurate adsorption energy estimates but also delivers reliable uncertainty quantification.

Tan *et al.* evaluated ensembling-based uncertainty quantification methods against single-model strategies, including mean-variance estimation, deep evidential regression, and Gaussian mixture models. Results, using datasets spanning in-domain interpolation (rMD17) to out-of-domain extrapolation (bulk silica glass), showed that no single method consistently outperforms the others. Ensembles excel at generalization and robustness, while MVE performs well in-domain, and GMM is better suited for out-of-domain tasks. The authors concluded that, overall, single-model approaches remain less reliable than ensembles for UQ in NNIPs.

Kahle and Zipoli reported that NN potentials ensembles may result overconfident, underestimating the uncertainty of the model. Further, they require to be calibrated for each system and architecture. This was verified across predictions for energy and forces in an atomic dimer, an aluminum surface slab, bulk liquid water, and a benzene molecule in vacuum. Bayesian NN potentials, obtained by sampling the posterior distribution of the model parameters using Monte Carlo techniques, were proposed as an alternative solution towards better uncertainty estimates.

Further, the integration of UQ methods with existing machine learning architecture is often streamlined for one specific approach only ([53,54]) and it is rarely the case that one single workflow allows for the adoption of a diverse set of ML UQ methods.

2.7 Size extensivity of uncertainty estimates

Another important consideration in ML for atomistic modeling is the size extensivity of properties targeted by the ML models, and how that propagates to the uncertainty estimates. Take ML interatomic potentials as an example, where the models are trained to predict total energies of chemical systems as a sum of atomic contributions. It is unclear how the uncertainties of the systems grow with their size. One could rationalize two extrema: one is a perfectly crystalline system with all-equivalent atomic environments, leading to maximal correlation between local predictions and hence uncertainties would strictly grow as N, the number of atoms. The other would be a dilute gas of atoms or molecules with no correlation, leading to the growth of uncertainty in quadrature, i.e., scaling with $\sqrt{N}$. Real chemical systems are expected to have components that can be distinguished with both scaling behaviors.

Kellner and Ceriotti[37] have investigated the size extensivity of uncertainty estimates for bulk water systems using their shallow ensembling method for a deep NN model. In their analysis, they decomposed the uncertainties on differently sized bulk water systems into "bias" and "residual" terms. The bias term, computed by taking the absolute difference between the mean predicted and reference energies for a given system size, was found to scale roughly with N. The remaining residual term, which would largely capture the random distortions of the water molecules, was then found to correlated with $\sqrt{N}$. Given the significant contributions from both terms, their experiments showcase the non-trivial trends in the size extensivity of ML model uncertainties for real material systems, which exposes the limitations of approaches where extensivity is ignored or a naive scaling law is assumed. Size extensivity of uncertainty estimates presented, in the context of gradient features—has also been discussed Zaverkin2024.

2.8 Uncertainty propagation

Besides being an alternative strategy with respect to the direct application of the formulae of Sec. 2.1, the use of ensembles is particularly useful in several physical applications that require the propagation of uncertainty to derived quantities that are function f of the regressor's output, $z(\mathbf{x}) = f(y(\mathbf{x}))$. In fact, only in few simple cases uncertainty can be propagated analytically from the UQ formulae presented in Sec. 2.1 in the form of Eq. (1), as it is

View Article Online
DOI: 10.1039/D5DD00102A

the case, for instance, whenever the linear approximation

$$\sigma_z(\mathbf{x}_\star) \approx \left.\frac{\mathrm{d}f}{\mathrm{d}y}\right|_{\mathbf{x}_\star} \sigma_\star \tag{29}$$

is sufficiently accurate. In the general scenario one can easily resort to *explicit sampling* of the models' distribution, and generate an ensemble of M models from which uncertainty can be propagated as in:

$$\begin{aligned} z^{(m)}(\mathbf{x}) &= f(y^m(\mathbf{x})) \\ \bar{z}(\mathbf{x}) &= \frac{1}{M}\sum_{m=1}^{M} z^{(m)}(\mathbf{x}) \\ \sigma_z^2(\mathbf{x}) &= \frac{1}{M-1}\sum_{m=1}^{M}[z^{(m)}(\mathbf{x}) - \bar{z}(\mathbf{x})]^2 \end{aligned} \tag{30}$$

Explicit sampling can also be useful to account for the correlated nature of the errors made by ML models. Excessively conservative UQ estimates are made, e.g., when computing differences of observables–such as the relative energy $E(\mathbf{x}_1) - E(\mathbf{x}_2)$ for nearby configurations $\mathbf{x}_1$ and $\mathbf{x}_2$–if independent errors are assumed across configurations. In reality, ML models often produce highly correlated predictions in such scenarios, meaning that while absolute uncertainties may be significant, the uncertainty on differences (which are often more physically relevant) can be much smaller. Sampling $\Delta E^{(m)} \equiv E^{(m)}(\mathbf{x}_1) - E^{(m)}(\mathbf{x}_2)$ and then estimating the mean value $\overline{\Delta E} \equiv \frac{1}{M}\sum_{m=1}^{M}\Delta E^{(m)}$, and the uncertainty as $\frac{1}{M-1}\sum_{m=1}^{M}[\Delta E^{(m)} - \overline{\Delta E}]^2$ is a viable option offered by explicit sampling, which naturally incorporates correlations between ML estimates that are function of multiple configurations.

In ML-driven atomistic simulations, UQ is also needed to single out the uncertainty ascribable to ML models from the statistical one due to a poor sampling (i.e. too short trajectories): in fact it would be pointless to run very long MD simulations if the uncertainty due to ML cannot be lowered below a given threshold.

A more subtle problem arises in the realm of MLIPs, whenever one aims at propagating uncertainty to thermostatic observables (e.g., the radial distribution function, the mean energy of a system, etc.) where ML uncertainty on the energy of a given structure enters the Boltzmann weight of thermodynamic averages, or – equivalently under the erdogic hypothesis – affects the sampling of the phase space via molecular dynamics simulations. In Imbalzano *et al.*, the availability of model-dependent predictions was leveraged to propagate uncertainty to thermostatic observables while running a single trajectory with the mean MLIP of the ensemble, by applying simple reweighing strategies.

For instance, consider the case of training a committee of M ML interatomic potentials to learn the ML potential energy surface $V(\mathbf{r})$, where $\mathbf{r}$ indicates the set of positions of all the atoms of a system. The potential energy of the i-th model is labeled by $V^{(i)}(\mathbf{r})$, and the mean potential energy of the committee by $\overline{V}(\mathbf{r})$, as in Eq. (17). Then, for a given observable $a(\mathbf{r})$ of the atomic positions, its canonical average, computed by sampling the configurational space according to the Boltzmann factor $\exp[-\beta V^{(i)}(\mathbf{r})]$, where $\beta = (k_\mathrm{B}T)^{-1}$, can be equivalently expressed in terms of a canonical average using the Boltzmann factor $\exp[-\beta\overline{V}(\mathbf{r})]$ associated to the mean potential of the committee as:

$$\langle a\rangle_{V^{(i)}} = \frac{\int \mathrm{d}\mathbf{r}\, w^{(i)}(\mathbf{r})\, a(\mathbf{r})\, e^{-\beta\overline{V}(\mathbf{r})}}{\int \mathrm{d}\mathbf{r}\, w^{(i)}(\mathbf{r})\, e^{-\beta\overline{V}(\mathbf{r})}} \tag{31}$$

where $w^{(i)} \equiv \exp[-\beta(V^{(i)}(\mathbf{r}) - \overline{V}(\mathbf{r}))]$. Therefore, by performing a single experiment to sample the configuration space (e.g., via Monte Carlo or molecular dynamics) using the mean potential of the committee, one can post-process the result to obtain the set $\langle a\rangle_{V^{(i)}}$, $i = 1,\ldots,M$, whose standard deviation across the committee quantifies the uncertainty on the thermodynamic average $\langle a\rangle$. Statistically more stable approximations also exist, based on a cumulant expansion, to overcome sampling efficiency issues stemming from direct application of Eq. (31).[11,37,55]

Looking ahead, fundamental questions remain. E.g., from a physical perspective: *what are the key ingredients towards the definition of a rigorous theory for uncertainty propagation for time-dependent thermodynamic observables, such as correlation functions?* Such a question is relevant for spectra and transport coefficients, obtained from ML-driven molecular dynamics simulations,[56–65] since its answer would make it possible to quantify the uncertainty on these dynamical observables in an efficient way, bypassing time-consuming, brute-force approaches that require running several trajectories.

2.9 Model Misspecification

In the UQ approaches of the previous sections, we have tacitly assumed that the regression problem is specified, i.e., that, except for i.i.d. noise, the ground truth can be in principle captured by the model form for some value of the weights. Model misspecification occurs when the assumed form of the model does not match the true data-generating process. The implication on UQ is rather important: as the number of samples $N_\mathrm{train} \to \infty$, the posterior over parameters $p(\mathbf{w} \mid \mathscr{D})$ becomes sharply peaked around a point estimate (e.g., MAP or MLE). This is fine *if the model is correctly specified*, i.e., the true data-generating distribution lies within the model class. But if the model is misspecified, then the parameters may converge to values that are optimal only within the *incorrect* model class—and the posterior uncertainty on weights may shrink, even though predictive uncertainty *should not*. Many fundamental theorems in statistical modeling and Bayesian inference actually assume that the problem is well-specified, and need to be modified to account for misspecification.[66]

Misspecification is a form of systematic model error, distinct from overfitting or random noise. In the context of atomistic simulations, modern ML architectures based on NN have in general enough capacity to fit any function to training data without overfitting; nonetheless, misspecification may still arise due to omitted physics, inadequate data coverage, or improper model assumptions. While it is universally true that *"all models are wrong*,[67] this limitation becomes especially critical in the context of misspecification, which can lead to unreliable or misleading UQ. For instance, if the model cannot represent long-range interactions that are present in the underlying physics—as in the

View Article Online
DOI: 10.1039/D5DD00102A

binding curves of molecular dimers[68]—all members of an ensemble may still agree on an incorrect prediction, such as vanishing forces beyond the model's short-range cutoff. This leads to artificially low uncertainty estimates. Such issues cannot be resolved even with calibration strategies like Eq. (26), since no global calibration can simultaneously account for both the regions where the model performs well (short distances) and those where it systematically fails (long distances). Another case where misspecification occurs concerns wrong functional forms as, e.g., in ML interatomic potentials trying to model a sharp repulsive wall with a smooth function.

3 Uncertainty and robustness

Having reviewed a wide range of UQ methods for ML models, we now extend our discussion to the "robustness" of the models and their predictions. By robustness, we refer to the ability of a model to maintain good accuracy and precision under various types of perturbations, noise, and adversarial conditions in the provided input. Approaching the robustness of ML models requires the knowledge of when and where the ML model fails or stops being applicable, *even in the absence of target values* unlike the case of UQ. Through such an understanding and quantification of ML model robustness, one can propose efficient methods for rational dataset construction/augmentation and active learning for ML training.

In the context of atomistic modeling, robustness is important for novel materials discovery, where models are often used to predict the properties of new phases or materials that lie outside the existing dataset. The concept of robustness can also be straightforwardly extended to the predictions of "local" (e.g., atom-centered) or "component-wise" (e.g., range-separated) quantities of the chemical systems, which do not correspond to physically observable targets. This is especially relevant for ML models constructed to make global predictions on the system as the sum of local and component-wise predictions on distinguishable parts and their associated features. This is indeed a common practice in ML for atomistic modeling.

3.1 A geometrical perspective on in- and out- of distribution

Intuitively, a prediction is likely to be accurate and precise if it takes place in the region corresponding to the distribution of training points. Towards the definition of robust prediction, it is then relevant to explore the definition of in- and out-of distribution. A first perspective to this end concerns a geometric framework and a convex hull construction. The convex hull of a set of training samples $\mathscr{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in the feature space is defined as the smallest convex set that contains all points in $\mathscr{X}$. Mathematically, the convex hull is expressed as:

$$\mathrm{Conv}(\mathscr{X}) = \left\{ \mathbf{x} \;\middle|\; \mathbf{x} = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i,\ \alpha_i \geq 0,\ \sum_{i=1}^{n} \alpha_i = 1 \right\},$$

where α_i are convex coefficients ensuring that any point $\mathbf{x}$ inside the hull is a weighted combination of the training samples $\mathbf{x}_i$.

This construction provides a method to distinguish in-distribution samples, which lie within $\mathrm{Conv}(\mathscr{X})$, from out-of-distribution samples, which fall outside the convex hull. Extrapolation, in this context, refers to the model's attempt to make predictions for such out-of-distribution points by extending patterns learned from the training data, often resulting in increased uncertainty and reduced accuracy (Figure 4 left panels).

The convex hull evaluation faces significant challenges in high-dimensional spaces. The computational cost of constructing and evaluating convex hulls increases with dimensionality, making this approach impractical for large-scale, high-dimensional machine learning tasks. Even more importantly, the number of points required to approximate the convex hull grows exponentially with the dimensionality of the feature space, a problem commonly referred to as the "curse of dimensionality." Consequently, the convex hull becomes increasingly sparse in high dimensions, causing most points in the space to be classified as out-of-distribution.[69] This observation also holds for the case of atomistic machine learning models based on local atomic environment representation.[70]

Importantly, while the intrinsic dimensionality of high-dimensional representation may be still low, low-dimensional projections (e.g., D=2 or D=3) for visualization or analysis, can introduce artifacts that misrepresent the true relationships in the data, such as incorrectly classifying in-distribution samples as out-of-distribution due to oversimplified boundaries. This is also relevant in machine learning for atomistic modeling, where the information high-dimensional representation can be reduced to a small but not too small amount of principal components.[70]

3.2 A statistical perspective on in- and out- of distribution

An alternative to the convex hull for defining in- and out-of-distribution samples is to use a density-based method. Here one evaluates the likelihood of a sample belonging to the training distribution by estimating the sampling density in the feature space (Figure 4 right panels)

In an adaptive k-nearest-neighbor (k-NN) density estimation procedure proposed by Zeni *et al.*[70], each test point $\mathbf{x}^*$ is temporarily inserted into the training set so that its k^* nearest neighbors among the training samples can be identified. This process makes it possible to compute the local density at $\mathbf{x}^*$ via:

$$\rho(\mathbf{x}^*) = \frac{k^* - 1}{M V^*}, \tag{32}$$

where M is the total number of training examples, and V^* is the volume corresponding to the k^* neighbors . The number k^* is selected in an adaptive manner for each test point to optimize the precision of the resulting density estimate. Moreover, the volume V^* is determined by the hypersphere of dimension d, where d represents the intrinsic dimensionality of the training set as computed using the TwoNN estimator[71,72].

Through this methodology, the resulting metric reflects the degree to which an unseen atomic environment lies in a well-sampled portion of the representation space. Importantly, it also correlates with the errors observed in machine-learned regression potentials. Furthermore, the same authors report a strong consis-

View Article Online
DOI: 10.1039/D5DD00102A

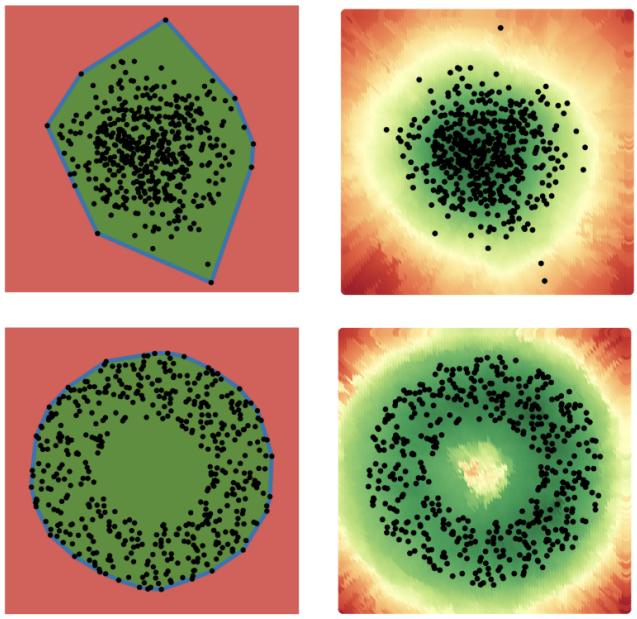

Fig. 4 The four panels illustrate two example distribution of points in two dimensions. In the left panels, in- and out- of distribution regions are categorized according to a convex-hull geometric construction (left). In the right panels in- and out- of distribution are defined according to a density-based criterion. For the left panels, the green region is defined as in-distribution, the red region is out-of-distribution. For the right panels, color from green to red highlight areas moving from in- to out-of distribution. Figure courtesy of Claudio Zeni.

tency between this density-based measure, model-specific error estimator – namely the predictive uncertainty from a committee of models trained on different subsamples of a larger training set – and model-free estimators – namely Hausdorff Distance between the prediction point and the training set. This result is also consistent with reports contrasting other model-specific and model-free approaches based on distances (e.g. latent space distances[73]).

In related work, Schultz *et al.* utilized kernel density estimation in feature space to evaluate whether new test data points fall within the same domain as the training samples. Their approach illustrates that chemical groups traditionally considered unrelated exhibit pronounced divergence according to this similarity metric. Moreover, they show that higher dissimilarity correlates with inferior predictive performance (manifested as larger residuals) and less reliable uncertainty estimates. They additionally propose automated methods to define thresholds for acceptable dissimilarity, enabling practitioners to distinguish between in-domain predictions and those lying outside the model's scope of applicability.

We similarly consider the work of Zhu *et al.* in the context of statistical methods to define extrapolation and interpolation and in- and out- of distribution. Given a specific training set, a feature vector for each point is derived from the latent space representation of a NequIP[76] model. Next, a Gaussian mixture model (GMM) is fitted on this distribution. A negative log-likelihood can be then obtained by evaluating the fitted GMM on the feature vector associated to any test point. Higher negative log-likelihood were observed for points resulting in higher predictions uncertainty.

To conclude, we note that, while statistical estimates of in- and out- of distribution are of interest because of their efficiency and effectiveness, questions remains. The magnitude of these metrics depends on the chosen representation,[70] and its precise correlation with the mean absolute error (MAE) is contingent upon both the system and the model employed.

3.3 Transferability

Transferability, in the context of machine learning for atomistic modeling, is often defined as the ability of a ML model to maintain its accuracy when applied to structures sampled under conditions different from those in the training dataset. However, the definition of these "different conditions" has remained somewhat weak, and can be summarized as follows (also illustrated in 5):

- *Phase Transferability* refers to the ability of an ML potential trained on certain phases of a material to accurately predict properties of other ones (e.g. different polymorphs or phases), assuming both are sampled at similar temperatures.[77–79]
- *Temperature Transferability* concerns the accuracy of the ML potential when, e.g., trained on structure sampled at high temperatures and tested on structures at lower temperatures, or viceversa.[78,80–82]
- *Compositional Transferability* refers to the ability of a ML model when providing predictions for systems with unseen compositions with accuracy comparable to known stoichiometries. This can refer both to predictions for unseen stoichiometries, or for unseen elements (alchemical learning).[83–85]

The standard according to which a model is deemed transferable across different conditions remains rather flexible too. Criteria used to relate transferability and model accuracy so far have included:

- The error incurred by the model in the test domain is *comparable* with the one observed for the training domain.
- The error in the test domain is *sizably larger* from the one in the training domain, but remains *acceptable* for practical purposes (e.g., energy errors below 10 meV/atom, force errors below 100 meV/Å).
- Simulations remain *stable over long timescales*, showing no significant energy drift or sampling of unphysical configurations.

The lack of a rigorous and standardized definition of transferability challenges our attempt to unify conclusions drawn from studies concerned in assessing ML model transferability in the context of atomistic modeling. Heuristic observations on transferability report that:

View Article Online
DOI: 10.1039/D5DD00102A

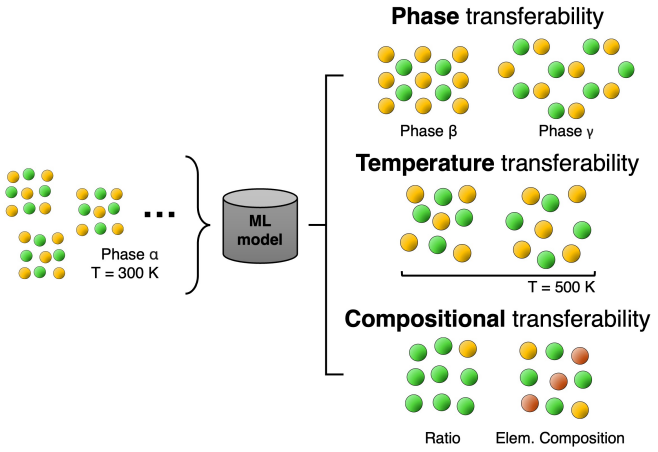

Fig. 5 Illustration of possible transferability tests. A model is trained on initial database (left) consisting of structures in phase α sampled at $T = 300$ K. Its transferability may be tested for the case of different phases (β and γ in the illustration), temperatures (e.g., $T = 500$ K), or compositions (i.e. different stoichiometries or elemental compositions).

- *Phase Transferability*: There is often a trade-off between accuracy and generality when applying ML potentials across different phases. This trade-off is generally acceptable for many practical applications.[39,80]
- *Temperature Transferability*: Models trained on high-temperature data tend to generalize well to lower-temperature conditions.[78,81]
- *Compositional Transferability*: Interpolation within the compositional space is generally feasible, but extrapolation to entirely new stoichiometries or elements (e.g., alchemical learning) poses significant challenges, unless tailored schemes are adopted.[83,84]

The relationship between a model's complexity and its transferability has also been a subject of discussion. In principle, more flexible models are more susceptible to overfitting, which can reduce transferability. Empirically, this tendency has been observed in ML methods based on feed-forward neural networks.[77] Importantly, modern high-order graph convolutions and/or physics-inspired (e.g., symmetry conserving) architectures have not exhibited this limitation, suggesting that increased complexity does not necessarily compromise transferability, at least within the data- and parameter-sizes considered in those applications.[86]

3.4 Quantifying Robustness

To meaningfully interpret the robustness of a prediction in itself, study its dependence on the datatset composition, and compare the robustness of a prediction on one input with another, the necessity to perform such an assessment in a quantitative manner arises.

To address this problem, recently, Chong and coworkers have introduced the concept of *"prediction rigidities"* (PRs),[14,15,87] which are metrics that quantify the robustness of ML model predictions. Derivation fo the PRs begin from considering the response of ML models to perturbations in their predictions, via their loss, by adopting the Lagrangian formalism. A modified loss function can be defined as shown below:

$$\hat{\mathscr{L}}(\mathbf{w}) = \mathscr{L}(\mathbf{w}) + \lambda \left[\varepsilon_\star - (\boldsymbol{\phi}_\star^o)^\mathsf{T}(\mathbf{w} - \mathbf{w}_o)\right] \tag{33}$$

where $\boldsymbol{\phi}_i^o \equiv \frac{\partial \bar{y}_\star}{\partial \mathbf{w}}\Big|_{\mathbf{w}_o}$, and $\varepsilon_\star$ is the perturbation of the model prediction for the input of interest denoted by $\star$. It is then possible to perform constrained minimization of this new loss, leading to the following expression that solely depends on $\varepsilon_\star$:

$$\hat{\mathscr{L}}_o(\varepsilon_\star) = \mathscr{L}_o + \frac{1}{2}\left.\frac{\partial^2 \hat{\mathscr{L}}_o}{\partial \varepsilon_\star^2}\right|_{\varepsilon_\star=0} \varepsilon_\star^2 \tag{34}$$

Here, the "curvature" at which the model responds to the perturbation in the prediction is given by $\partial^2 \hat{\mathscr{L}}_o / \partial \varepsilon_\star^2$, which can be further derived as follows:

$$\left.\frac{\partial^2 \hat{\mathscr{L}}_o}{\partial \varepsilon_\star^2}\right|_{\varepsilon_\star=0} \equiv \frac{1}{(\boldsymbol{\phi}_\star^o)^\mathsf{T}(\mathbf{H}_o)^{-1}\boldsymbol{\phi}_\star^o} \tag{35}$$

One can recognize the crucial connection between the $\mathbf{H}_o$ appearing in this expression and the $\mathbf{H}_o$ defined in Eq. (9), as well as the orignal Eq. (1) defined for the Mahalanobis distance. Note that $\mathbf{H}_o$ is often approximated as the sum of the outer products of *structural* features over the training set, which would be the sum or mean of atomic features of the given structure and an indirect approach to consider the "groupings" of local environments that are present as structures in the training set.

A few important remarks should be made here:

- absence of any calibration parameters in the PRs hint that these are purely dataset and representation-dependent parameters, and hence distinct from being a UQ metric;
- dependence on the dataset and model training details is determined by $\mathbf{H}_o$, which can adopt a Gauss-Newton approximation scheme and be computed in a similar manner as Eq. (11), and remains constant for a given model;
- there is freedom in how $\star$ is defined — it is possible to compute the PRs for any data point as long as it is definable within the input parameter space, furthermore, one can also target specific local predictions or component-wise predictions of the model, resulting in *local* PR (LPR) or *component-wise* (CPR) that quantitatively assess the robustness of intermediate model predictions that do not have corresponding physical observables.

The robustness metrics introduced thus far are solely dependent on the dataset distribution (i.e. structural diversity of material systems and their local environments) and remain detached from the distribution of the target metric. One must be mindful of the repercussions, which is that complexity of the target quantity landscape is ignored: if the target landscape is smooth, learning may require fewer data points to achieve the target accuracy; if it is rough, more data points would be needed to resolve the complex landscape and achieve desirable accuracy.

The quality of data and representation may be similarly relevant. For example, as discussed by Aldeghi et al.[88] and van

View Article Online
DOI: 10.1039/D5DD00102A

Tilborg et al.[89], *"activity cliffs"*—instances where structurally similar pairs of molecules that exhibit large differences in the targets—negatively affect the model performance. By learning a representation, e.g., through contrastive learning, that correctly separates such structures the learning problem is simplified. Also, a modified Shapley analysis was also proposed for analysing and interpreting the impact of a datapoint in the training set on model prediction outcomes[90,91].

3.5 Dataset improvement and active learning

In atomistic modeling, tasks such as identifying global minima in complex energy landscapes and estimating statistical observables from molecular dynamics sampling require efficient exploration of vast and high-dimensional spaces. A recurring question then emerges: *What (additional) data should one select to gather, to build a "better" model (i.e., capable of more reliable/robust predictions)?* The problem of optimal data selection is in fact crucial in two main scenarios:

1. Generation of new targets is relatively expensive and/or time consuming. In such a case one may like to know *in advance* for which new data point inputs $\mathbf{x}$ compute the target y (i.e. assign a label);

2. There is a vast pool of data and one aims to select a subset of data points. This is critical for, e.g., the construction of the representative set of sparse kernel models, although it is common in modern deep learning training strategies to use all the data at disposal.

A first example of workflows iteratively improving the accuracy of an atomistic model was the "learn on the fly" hybrid scheme proposed by Csányi *et al.* Here, fitted potentials (based upon an analytical formulation[92] or machine learning[93]) are refined using a predictor-corrector algorithm and quantum calculations to ensure reliable simulations of complex molecular dynamics.

Since the units of the uncertainty naturally allow for the definition of interpretable thresholds and tolerance criteria, uncertainty can be naturally adopted as the metric to identify configurations where model predictions are too uncertain, for which additional information is necessary to steer the model towards more robust predictions.

Numerous active learning schemes (Figure 6 for interatomic potentials based upon uncertainty thresholds have been proposed in the last years, encompassing a variety of materials and chemistry, from heterogeneous catalysis[94,95] to energy materials,[96] from reactions in solutions[97,98] to molecular crystals[99]. More recently, biasing the sampling towards configurations corresponding to highly uncertain prediction was brought forward as a strategy to ensure the collection of varied training set, and the training of a model presenting an uncertainty always below a specific threshold across a (large) region of interest in the configurational space.[50,100–103]

In the next subsections, we show that several "active learning" approaches to sequentially select new, optimal data points can be framed in the context of the maximum gain of information, as first discussed by MacKay in the context of the Bayesian interpretation

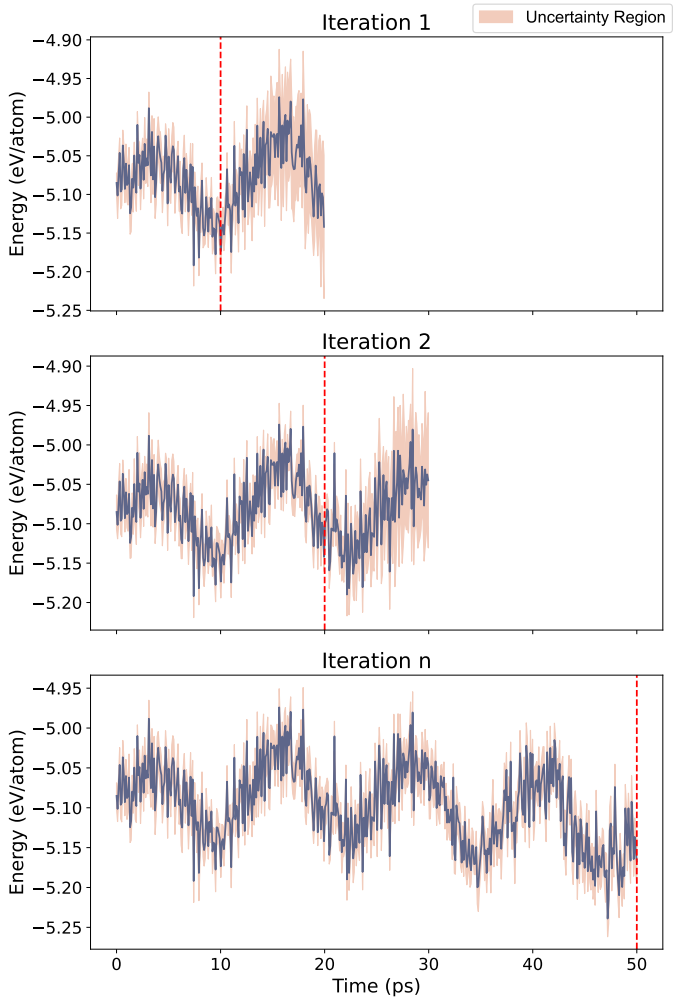

Fig. 6 Example of active learning for energy evolution over time for different iterations. The orange region indicates uncertainty. In Iteration 1 and Iteration 2, uncertainty increases significantly after the red dashed line leading to termination of the simulation. The last iteration (Iteration n) represents the case where the previous active learning iteration result in a stable, accurate, and precise simulation.

of learning. We also show that apparently model-free approaches do effectively identify new points where uncertainties would be the largest.

3.5.1 Maximizing information gain

Consider a dataset $\mathscr{D}$ of N_{train} points with feature matrix $\mathbf{F}$. From the information theory standpoint, we can define the Shannon entropy

$$S \equiv -\int \mathrm{d}\mathbf{w}\, p(\mathbf{w}|\mathscr{D}) \log[p(\mathbf{w}|\mathscr{D})] \tag{36}$$

where $p(\mathbf{w}|\mathscr{D})$ is a probability measure of the parameters, the weights $\mathbf{w}$, given the model architecture and dataset $\mathscr{D}$. In the Bayesian interpretation, $p(\mathbf{w}|\mathscr{D})$ is the posterior distribution of the weights, which assumes the form of a multivariate Gaussian distribution, Eq. (9), whenever the model is linear or a Laplace approximation around the MAP optimal weights is performed,

View Article Online
DOI: 10.1039/D5DD00102A

leading to

$$S = -\frac{1}{2}\log(\det \mathbf{H}_o) + \text{constants}. \tag{37}$$

Here, as in Sec. 2, $\mathbf{H}_o$ is the Hessian matrix of the loss function at optimum. As previously discussed, the generalized Gauss-Newton approximation implies:

$$\mathbf{H}_o \approx \mathbf{F}^\top \mathbf{F} \tag{38}$$

Let us now take a new point of features $\mathbf{f}_\star$ and add it to the dataset. The new feature matrix $\mathbf{F}_{\text{new}}$ is obtained by concatenating the row vector $\mathbf{f}_\star^\top$ to $\mathbf{F}$. The new Hessian becomes

$$\mathbf{H}_{o,\text{new}} \approx \mathbf{F}^\top \mathbf{F} + \mathbf{f}_\star \mathbf{f}_\star^\top \tag{39}$$

and the new Shannon entropy is

$$\begin{aligned} S_{\text{new}} &= -\frac{1}{2}\log(\det \mathbf{H}_{o,\text{new}}) + \text{constants} \\ &\approx -\frac{1}{2}\log(\det(\mathbf{F}^\top \mathbf{F} + \mathbf{f}_\star \mathbf{f}_\star^\top)) + \text{constants} \end{aligned} \tag{40}$$

One can use the *matrix determinant lemma*[104] to express:

$$\det(\mathbf{F}^\top \mathbf{F} + \mathbf{f}_\star \mathbf{f}_\star^\top) = [1 + \mathbf{f}_\star^\top (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{f}_\star] \det(\mathbf{F}^\top \mathbf{F}) \tag{41}$$

The total information gain from adding $\star$ to the dataset, ΔI, is

$$\Delta I \equiv -(S_{\text{new}} - S) = \frac{1}{2}\log[1 + \mathbf{f}_\star^\top (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{f}_\star] \tag{42}$$

which is maximized when the (scaled) variance $\mathbf{f}_\star^\top (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{f}_\star$ is largest: therefore, to obtain maximal information gain (MIG), a next point should be chosen where the uncertainty, Eq. (1), is currently largest. The MIG criterion has been recently used by Kästner's group for active learning in atomistic simulations, see Zaverkin *et al.*, which also focus on active selection of batches of new data points,[25] rather than the incremental, one-at-a-time approach. See the seminal works on multiple point selections by Fedorov and Luttrell, where analytic expressions for the expected information gain from *a set* of measurements are discussed, and batch selection strategies are explored in detail, showing how optimal placements shift with the number of samples, signal-to-noise ratio, and prior constraints.

The MIG criterion also motivates, from an information theory perspective, the addition of structures characterized by environments with lowest *local* prediction rigidity as active learning criterion, as in Chong *et al.*. We remark that:

- the MIG criterion is *independent of the specific target*, which need not be computed in advance to perform the active data selection.
- if we consider the initial dataset $\mathscr{D}$ as fixed, then maximizing the information gain implies looking for $\star$ to satisfy:

$$\max_\star \det(\mathbf{F}_{\text{new}}^\top \mathbf{F}_{\text{new}}) \tag{43}$$

  which is called D-optimality criterion in optimal design theory. $\mathbf{F}_{\text{new}}^\top \mathbf{F}_{\text{new}}$ is sometimes called, in this context, the *Fisher information matrix* of the new dataset. We review the use of D-optimality for active learning in atomistic simulations in Sec. 3.5.2.
- In this approach, the noise on data is taken the same for all the data (in fact, a single calibration constant α^2 was used in Sec. 2). Generalization to the case of sample-dependent noise is nonetheless straightforward.

While the selection of a single datum based on maximum expected information gain is a well-established approach in active learning, practical applications often require evaluating how much information a single new datum contributes with respect to a set of target points (e.g., a test set or region of interest). MacKay—see section 4.1 of Ref.[9]—can be credited for generalizing data-point selection by considering information gain across a set of input points, representing a region of interest. However, a naïve use of the joint information gain of the interpolated values can lead to suboptimal results, as it may favor inducing correlations among outputs rather than minimizing their individual uncertainties. A more appropriate strategy that MacKay suggests is to maximize the mean marginal information gain (MMIG) across these points (which we label by $u = 1, \ldots, V$), independently, i.e., in formulae–for noiseless observations:

$$\text{MMIG} \equiv -\frac{1}{2}\sum_{u=1}^{V} p_u \log\left[1 - \frac{[\mathbf{f}_\star^\top (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{f}_u]^2}{[\mathbf{f}_u^\top (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{f}_u][\mathbf{f}_\star^\top (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{f}_\star]}\right] \tag{44}$$

where p_u is the probability that we are asked to predict y_u, and acts as a tunable weight. This approach provides a principled basis for an acquisition strategy with the overall goal of improving model performance across the domain of interest. Alternative, yet similar, results are obtained via Q-optimality,[105] which is an optimal design criterion that aims to minimize the mean squared error of predicted outputs at specific points of interest. In Bayesian or active learning contexts, it is formulated as selecting data points that reduce the average predictive variance for a set of points (or a region), i.e. that minimizes $\sum_u p_u [\mathbf{f}_u^\top (\mathbf{F}_{\text{new}}^\top \mathbf{F}_{\text{new}})^{-1} \mathbf{f}_u]$ when one (or more) data points are actively selected.

3.5.2 D-optimality

Lysogorskiy *et al.* show that the maximum deviation within an ensemble of models—in our notation $\max_i |\bar{y}^{(i)} - \bar{y}|$, where y is the total potential energy or a force component of a structure— fully correlates with the so-called D-optimality criterion, which is then used for active learning strategy. In the latter, one seeks to find 1) an optimal (sub)set of data samples and 2) to quantify how much a new sample $\star$ is represented. Specifically, if we collect all the dataset features in the "tall" $N_{\text{train}} \times N_f$ matrix $\mathbf{F}$, step 1) looks at a subsampling N_f data points, i.e. selecting N_f out of N_{train} rows to obtain a new, $N_f \times N_f$ matrix $\tilde{\mathbf{F}}$, such that

$$\det(\tilde{\mathbf{F}}^\top \tilde{\mathbf{F}}) \tag{45}$$

is maximal. Then, in step 2), a new sample $\star$ of features $\mathbf{f}_\star$ is selected from a pool of new data (e.g. structures generated via MD trajectory) so that

$$|\mathbf{f}_\star^\top \tilde{\mathbf{F}}^{-1}| = \sqrt{\mathbf{f}_\star^\top (\tilde{\mathbf{F}}^\top \tilde{\mathbf{F}})^{-1} \mathbf{f}_\star} \tag{46}$$

View Article Online
DOI: 10.1039/D5DD00102A

is maximal (or larger than a given threshold $\gamma_{\text{th}} \geq 1$).

This is the same criterion found the previous section from the theory of maximal information gain, i.e. the quest for the sample $\star$ with largest variance, Eq. (1). Nevertheless, this time, the adopted metric is $\tilde{\mathbf{G}} = (\tilde{\mathbf{F}}^\top \tilde{\mathbf{F}})^{-1}$, obtained with the D-optimally subsampled dataset. Notice that

$$\begin{aligned} &\det(\tilde{\mathbf{F}}^\top \tilde{\mathbf{F}}) = \det(\tilde{\mathbf{F}})^2 \\ &\leq \sum_S \det(\mathbf{F}_S)^2 = \det(\mathbf{F}^\top \mathbf{F}) = \lambda_1^2 \lambda_2^2 \cdots \lambda_{N_f}^2 \end{aligned} \tag{47}$$

where λ_i are the singular values of $\mathbf{F}$, S runs over all the $\binom{N_{\text{train}}}{N_f}$ combinations in which one can select N_f rows out of N_{train} to build the $N_f \times N_f$ matrix $\mathbf{F}_S$. The second line follows from the Binet-Cauchy formula. A threshold on γ must be set to determine whether a given point is in- or out-of-distribution during the active learning cycle. The original paper by Podryabinkin and Shapeev suggests that a threshold of $\gamma \leq 1$ corresponds to prediction in in-distribution regime, and $\gamma \gg 1$ would correspond to strong out-of-distributions regimes. D-optimal-based active learning for atomistic simulations and machine learning interatomic potentials construction has been implemented and extensively used first and foremost by the communities developing moment tensor potentials and atomic cluster expansion potentials [107,109–112]. A generalization of the D-optimality criterion to the case of nonlinear dependence of the model upon its parameters has been proposed by Gubaev *et al.*. Just like the maximum-information-gain criterion, D-optimality proves to be largely more efficient than both random and CUR- (and FPS-) based selection [107].

3.5.3 Empirical forms of dataset entropy

Another approach that is used in the atomistic modeling community is based on the empirical estimate of the entropy of a distribution of dataset features. In this context, this set is usually taken as the set of features of *atomic environments* [114,115]. Karabin and Perez propose the following estimator for the entropy of a distribution of features *in a given configuration* A (e.g., a given structure in a simulation cell) of N_A atomic environments:

$$S_{\text{KP}}(A) = \frac{1}{N_A} \sum_{i=1}^{N_A} \log\left(N_A \min_j |\mathbf{x}_{A_i} - \mathbf{x}_{A_j}| \right) \tag{48}$$

where $|\mathbf{x}_{A_i} - \mathbf{x}_{A_j}|$ is the Euclidean distance between the atomic descriptors (features) of atoms i and j. This configuration-dependent entropy S_{KP} is then used for active dataset construction: the training set is incrementally built by adding independent local minima of the "effective (free) energy":

$$V(A) = E_{\text{repulsive}}(A) - K S_{\text{KP}}(A) \tag{49}$$

where $E_{\text{repulsive}}$ is a short-range repulsive term penalizing very small distances between atoms, and K is an entropy scaling coefficient which controls the relative importance of the two contributions. The minima are found via a simple annealing procedure at a given (fixed) cell volume. Notice that one could also construct a *global* S_{KP} by considering *all* the atomic environment features present in the dataset. [§]

Nonetheless, S_{KP} diverges to $-\infty$ whenever the features associated to environments i and j in Eq. (48) coincide. Schwalbe-Koda *et al.* solve this issue by defining a dataset entropy of a set of N_{env} environments:

$$S_{\text{SK}} = -\frac{1}{N_{\text{env}}} \sum_{i=1}^{N_{\text{env}}} \log\left[\frac{1}{N_{\text{env}}} \sum_{j=1}^{N_{\text{env}}} k(\mathbf{x}_i, \mathbf{x}_j) \right] \tag{50}$$

where $k(\mathbf{x}_i, \mathbf{x}_j)$ is some kernel function expressing the similarity between environments i and j. This formulation is also used to define a "differential entropy":

$$\delta S_{\text{SK}}(\mathbf{x}_\star) = -\log\left[\sum_{i=1}^{N_{\text{env}}} k(\mathbf{x}_i, \mathbf{x}_\star) \right] \tag{51}$$

which is then used for active learning and as a "model-free uncertainty estimator" of a new input for a given dataset [115].

We highlight that even when the dataset is built with targets that are structural properties, rather than local atomic properties, no "grouping" of local environments into structures is taken into account in these data-entropy based schemes, as it is instead done in the construction of the metric tensor in Mahalanobis distance (see Eq. (34)). It is further relevant to note that any kernel can be written (Mercer's theorem) as

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sum_a \lambda_a \varphi_a(\mathbf{x}_i) \varphi_a(\mathbf{x}_j) = \boldsymbol{\phi}^\top(\mathbf{x}_i)\, \boldsymbol{\phi}(\mathbf{x}_j) \tag{52}$$

where $\lambda_a \geq 0$ and φ_a are the eigenvalues and eigenfunctions of the kernel with respect to a measure μ:

$$\int k(\mathbf{x}, \mathbf{x}') \varphi_a(\mathbf{x}') \mathrm{d}\mu(\mathbf{x}') = \lambda_a \varphi_a(\mathbf{x}) \tag{53}$$

and the (possibly infinite) components of $\boldsymbol{\phi}(\mathbf{x})$ are $\phi_a(\mathbf{x}) = \sqrt{\lambda_a}\varphi_a(\mathbf{x})$, for every possible $\mathbf{x}$. In such a case,

$$\sum_{i=1}^{N_{\text{env}}} k(\mathbf{x}_i, \mathbf{x}_\star) = N_{\text{env}} \overline{\boldsymbol{\phi}}^\top \boldsymbol{\phi}(\mathbf{x}_\star) \tag{54}$$

where $\overline{\boldsymbol{\phi}} = \frac{1}{N_{\text{env}}} \sum_{i=1}^{N_{\text{env}}} \boldsymbol{\phi}(\mathbf{x}_i)$ is the array of mean (latent) features over the dataset, i.e. the coordinates of the center of the dataset latent features. If we replace this into Eq. (51) we obtain, since the logarithm is a monotonic function, that the maximum differential entropy is given when the argument of the logarithm is minimal, i.e.

$$\max_\star \left[\delta S_{\text{SK}}(\mathbf{x}_\star)\right] \Leftrightarrow \min_\star \left[\overline{\boldsymbol{\phi}}^\top \boldsymbol{\phi}(\mathbf{x}_\star) \right] \tag{55}$$

Unfortunately, in contrast to the forms of active learning of the previous subsections, based on Eq. (1), here the complexity of the dataset is effectively "averaged out" by considering $\overline{\boldsymbol{\phi}}$ in the latent feature space. Notice that if the features in the latent feature space are centered (see Appendix C for a discussion on whether

[§] Karabin and Perez report that *"Extensions to global entropy-maximization over the whole training set (in contrast to the local configuration-by-configuration optimization presented here) are in development and will be reported in an upcoming publication."* Montes de Oca Zapiain *et al.* still adopts the local approach.

View Article Online
DOI: 10.1039/D5DD00102A

latent feature centering is legitimate or not), $\overline{\boldsymbol{\phi}}$ vanishes, and one incurs into the additional problem that the argument of $\min_\star$ vanishes for any $\mathbf{x}_\star$.

A different perspective on assessing the proximity of two distribution (e.g., training points features and test point features) was proposed by Zeni *et al.*. Their study considered 2-body machine learning potentials and the Kullback-Leibler divergence between distributions of interatomic distances in the training and test sets to rationalize prediction errors in machine learning potentials. The Kullback-Leibler divergence is an asymmetry statistical measure to quantify the information loss when a probability distribution $q(\xi)$ associated to a dataset $\mathscr{Q}$ over some sample space Ξ is used to approximate another probability distribution $p(\xi)$ associated to a dataset $\mathscr{P}$ on the same sample space:

$$D_{\mathrm{KL}}(\mathscr{P}\|\mathscr{Q}) = \sum_{\xi\in\Xi} p(\xi)\log\frac{p(\xi)}{q(\xi)} \tag{56}$$

A positive correlation between KL divergence and the mean absolute error of 2-body kernels was observed, highlighting the importance of including training data that captures interatomic distances relevant to the test set. The KL divergence was thus proposed as a measure to assess how well structural features in the training dataset align with those in the test dataset and interpret model errors in a case study concerning machine learning potentials for Ni nanoclusters[80]. Extending the assessment to 3-body machine learning potentials and the KL divergence of bond-angle distribution functions also resulted in the observation of positive correlation between the two quantities. We notice an hysteretical behavior exists in this metric, whereby the net cross-entropy change in first adding a point to the dataset and then removing it is nonzero (see Appendix B).

3.6 Bayesian Optimization

Bayesian optimization is a method designed to find the optimal value of a function efficiently. It uses a probabilistic model to predict the behavior of the function across the input space, guiding the search toward regions where the model is either uncertain or expects to find better results[117–120](Figure 7).

While similar to active learning in its iterative approach and reliance on uncertainty to guide decisions, Bayesian optimization differs in its goal. Active learning focuses on generally improving a model's predictions. Bayesian optimization aims to optimize an objective function directly. Instead of stochastically sampling or evaluating all possibilities, Bayesian optimization identifies the next point to sample by balancing two goals: exploring unknown regions (where the function behavior is uncertain) and exploiting promising areas (where the function is predicted to perform well). Once the function is evaluated at the chosen point, the new information is used to update the model, and the process repeats until an optimal solution is found[121].

Building upon these premises, Bayesian optimization acquisition deciding where to evaluate the objective function next and uncertainty estimates are at the heart of this process: e.g., Improvement[122] uses uncertainty to identify areas where the potential for improvement is highest. Probability of Improvement[121] factors in uncertainty to assess the likelihood of finding better outcome. The Upper Confidence Bound[123] takes a more explicit approach, blending the model's predictions with a weighted measure of uncertainty.

Bayesian optimization has emerged as a powerful tool in atomistic modeling, offering efficient strategies for navigating complex energy landscapes and exploring vast configuration spaces. By leveraging probabilistic models, it enables the optimization of potential energy surfaces for intricate atomistic systems, guiding the search toward minima or other critical points with minimal computational cost. Challenges in Bayesian Optimization may arise if it is performed in a too highly dimensional space or if the property landscape is rough (that is to say, properties change rapidly with respect to a small change in the feature space, akin to the discussion presented in Sec 3.4).

Successful applications of BO in atomistic modeling have been showcased for molecular[124,125], crystalline[126–129] and disordered systems[130], as well as complex interfaces[131]. Beyond identifying stable configurations, Bayesian optimization facilitates the exploration of structures to achieve specific target properties or locate configurations along the Pareto front in multi-objective predictions. This capability makes it highly valuable for material where balancing multiple competing properties is often required. Applications include (but are not limited to) metallic glasses mechanical properties[132], multi-principal[133] or high-entropy alloys[134,135] and their catalytic properties[136–138], or electrolytes properties for energy storage applications[139].

4 Uncertainty and emergent atomistic machine learning approaches

Data-efficient methods offer a compelling pathway to achieve highly accurate prediction, while significantly reducing the data and resource requirements. Notable emergent approaches in this area include Universal and Foundation Models, their fine-tuning, as well as Delta- and Multi-fidelity Learning.

4.1 Foundation models

Universal and foundation models (e.g.,[86,140,141]) are designed to capture broad physical relationships by training on diverse datasets. The accuracy of these models hinges on the large number of diverse training data; if critical subdomains are underrepresented, predictions in those regions may falter nevertheless.[142]

The benchmark of foundation models so far took place against established metrics, informing on errors in thermodynamic stability.[143] Community benchmarks and validation practices further did rarely account neither for uncertainties nor for their propagation. An attempt to introduce performance assessment against an observable (thermal conductivity) drawn from molecular dynamics has been recently introduced.[144] Similarly, UQ has been introduced in foundation models,[141] adopting the last-layer approximation described in Eq. (14), with almost no additional computational load with respect to inference of raw prediction. We consider these key steps towards the definition of probing and informative benchmarks and validation practices.

Fine-tuning and transfer learning build upon the knowledge

View Article Online
DOI: 10.1039/D5DD00102A

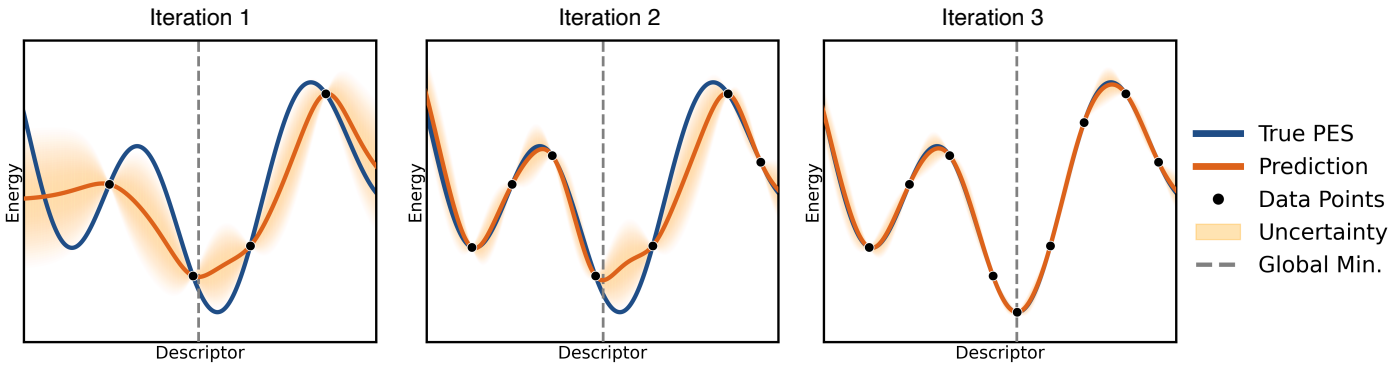

Fig. 7 Illustrative example of Bayesian optimization to sampling the minimum of a one-dimensional energy landscapes.

encoded in pre-trained models, such as universal and foundation models, adapting them to specific tasks or systems through targeted retraining with smaller datasets. This approach enhances data efficiency, and often robustness, as the base model serves as a strong starting point. Their achievement span across diverse areas of machine learning, also including atomistic modeling.[145–149] Nevertheless, challenges arise when the pre-trained model's domain significantly differs from the target domain. Question thus arise in reference to the effect of different strategies on the reliability and robustness of the uncertainty estimators.

4.2 Multi-level approaches

Delta-learning[150] predicts corrections of a simpler and (relatively) inexpensive model - such as a classical forcefield, a semi-empirical force-field, or a low quality DFT level) - achieving high accuracy with minimal training data by concentrating on residual discrepancies. The quality of the baseline model and the representativeness of the training data are nevertheless critical to Delta-learning model accuracy and precision.

Multi-fidelity learning[151] integrates information from datasets of varying accuracy and cost, effectively linking low-fidelity data to high-fidelity outputs. By synthesizing information from multiple sources, this approach enhances robustness while reducing the dependence on high-cost data. However, inconsistencies between fidelities and the challenge of accurately propagating uncertainties associated from models considering multiple fidelity levels demand careful consideration.

Open questions remain on how uncertainty estimate is affected by the use of these data-efficient models. These include - but are not limited to - a reflection on whether the simultaneous learning of multiple level of theory advantageous in terms of both data efficiency and robustness, also in relation to their effect on prediction uncertainty.

5 Beyond atomistic modeling

Many of the theories and arguments described in this perspective have broader relevance that extends beyond atomistic simulations. While a detailed discussion of error sources and uncertainties in the design, synthesis, characterization, and understanding of materials and processes lies beyond the scope of this work, we emphasize that estimating and propagating uncertainty is critical across all stages of the materials development cycle. Uncertainties arise, for instance, in the reproducibility of synthetic protocols—often influenced by hidden variables—, in the interpretation of spectroscopy and microscopy signals, in the construction and use of structure–property relationships, and in the iterative optimization of processes to achieve target performance metrics. These considerations highlight several key domains where uncertainty quantification deserves focused attention:

- Models trained to predict or guide synthesis strategies may be significantly affected by noise and bias in experimental data, making robust uncertainty estimates essential for actionable predictions.[152])
- Machine learning tools developed to accelerate materials characterization must account for ambiguities in signal assignment and model interpretability.[153]
- Surrogate models used to establish structure–property relationships or optimize structures towards target properties are often based on regression over high-dimensional descriptors. These models thus benefit from UQ strategies already employed in atomistic modeling, such as ensemble methods or Bayesian approximations.[154]
- Multiscale modeling frameworks require principled approaches for propagating uncertainty across scales — from atomistic to continuum — where even well-calibrated models at a lower scale may induce unpredictable errors at a higher one.[155]

6 Conclusions

In this perspective, we have examined the integration of machine learning and uncertainty quantification (UQ) in atomistic modeling, with a focus on methods to estimate uncertainties. We discussed state-of-the-art approaches, including Bayesian frameworks and ensemble techniques, and explored their applications in improving prediction reliability, guiding data acquisition through active learning and Bayesian optimization, and assessing the influence of uncertainties on equilibrium observables estimates. We also explored the influence of dataset composition and construction strategies on model accuracy, uncertainty, transferability, and robustness. We finally considered emergent data-

View Article Online
DOI: 10.1039/D5DD00102A

efficient approaches and highlighted emergent questions concerning prediction uncertainty estimate when leveraging these methods.

Taken together, our work underscores the role of rigorous UQ frameworks for guiding data-driven modeling and the value of thoughtful dataset construction in enhancing the transparency and robustness of ML-based atomistic modeling. As new techniques—especially those geared toward data-efficient learning—continue to mature, careful validation and thorough uncertainty assessments become even more critical to maintain trust in model predictions. We hope this perspective stimulates further development and integration of UQ protocols into atomistic modeling efforts.

Finally, we emphasize that the challenges and strategies for managing uncertainty in atomistic modeling echo a broader scientific discourse extending beyond this specific domain. Across materials science, physics, and chemistry, there is a renewed drive to establish clear standards for assessing information and uncertainty, from the reproducibility of synthetic protocols—whether in organic or materials synthesis—to the quality of data gleaned from real- and inverse-space characterization methods. The same principles underpin efforts to gauge the reliability of outputs from generative AI for materials discovery, large language models, automated image and spectrum analyses, and multi-modal approaches alike. Similar to Tycho Brahe's endeavor, these collective efforts will contribute to robust, transparent, and reproducible scientific findings.

Author contributions

K.R and F.G. initial conceptualization. All authors contributed to the writing and reviewing of the manuscript.

Conflicts of interest

There are no conflicts to declare.

Data availability

No primary research results, software or code have been included and no new data were generated or analysed as part of this review.

7 Acknowledgments

We thank Alfredo Fiorentino, Matthias Kellner, Davide Tisi, and Claudio Zeni for fruitful discussions and critical comments to an early version of the manuscript. We similarly thank the anonymous reviewer for their comments. This article is based upon work from COST Action CA22154 - Data-driven Applications towards the Engineering of functional Materials: an Open Network (DAEMON) supported by COST (European Cooperation in Science and Technology). F.G. acknowledges financial support from UNIMORE through the FAR 2024 project "Revolutionizing All-Solid-State Sodium Batteries with Advanced Computational Tools and Mixed Glass Former Effects," PI: Alfonso Pedone, CUP E93C24001990005, and from the EMPEROR Project, CUP E93C24001040001, sponsored by the National Quantum Science and Technology Institute (Spoke 5), Grant No. PE00000023, funded by the European Union – NextGeneration EU. S.C. acknowledges the support by the Swiss National Science Foundation (Project 200020_214879). S.B. thanks the European Union Horizon 2020 research and innovation program under grant agreement no. 857470 and from the European Regional Development Fund via the Foundation for Polish Science International Research Agenda PLUS program grant No. MAB PLUS/2018/8. S.B. acknowledges the Foundation for Polish Science for the FIRST TEAM FENG.02.02-IP.05-0177/23 project.

A Nyström approximation

In the Nyström approximations one considers a regression problem, Eq. (5), with N_f equal to the number of sparse points, M, and where

$$[\boldsymbol{\phi}(\mathbf{x})]^\top = \mathbf{k}(\mathbf{x},\mathbf{X}_s)^\top \mathbf{U}_s \boldsymbol{\Lambda}_s^{-1/2}. \tag{57}$$

In this formula, $\mathbf{k}(\mathbf{x},\mathbf{X}_s)$ is the vector of the kernels between the input point $\mathbf{x}$ and each of the points in the sparse set, that are collected in the matrix $\mathbf{X}_s \in \mathbb{R}^{M\times D}$, while $\mathbf{U}_s \in \mathbb{R}^{M\times M}$ is the matrix of the eigenvectors of the sparse set kernel matrix, $\mathbf{K}_s \equiv \mathbf{K}(\mathbf{X}_s,\mathbf{X}_s)$, that has as entries the kernel between two points of the sparse set:

$$\mathbf{K}_s = \mathbf{U}_s \boldsymbol{\Lambda}_s \mathbf{U}_s^\top. \tag{58}$$

The diagonal matrix $\boldsymbol{\Lambda}$ collects the eigenvalues, ordered to correspond to $\mathbf{U}_s$. The kernel matrix of the training set is then approximated as (Nyström formula):

$$\begin{aligned}\mathbf{K}(\mathbf{X},\mathbf{X}) &\approx \mathbf{K}(\mathbf{X},\mathbf{X}_s)\,[\mathbf{K}_s]^{-1}\,\mathbf{K}(\mathbf{X},\mathbf{X}_s)^\top \\ &= \mathbf{K}(\mathbf{X},\mathbf{X}_s)\mathbf{U}_s\boldsymbol{\Lambda}_s^{-1/2}\boldsymbol{\Lambda}_s^{-1/2}\mathbf{U}_s^\top\mathbf{K}(\mathbf{X},\mathbf{X}_s)^\top \\ &= \boldsymbol{\Phi}\boldsymbol{\Phi}^\top\end{aligned} \tag{59}$$

Here, $\mathbf{X} \in \mathbb{R}^{N_{\text{train}}\times D}$ is the training set matrix, and $\mathbf{K}(\mathbf{X},\mathbf{X}_s) \in \mathbb{R}^{N_{\text{train}}\times M}$ is the kernel matrix between the training set points and the sparse set points. After centering the ■ matrix, the variance on the prediction for input $\star$ is readily obtained as Eq. (7).

B Hysteresis of cross-entropy gain/loss

Consider an initial dataset $\mathscr{Q}$ and a probability density q associated to it, then add a data point to obtain the distribution p associated with the new dataset $\mathscr{P}$. The Kullback-Leibler (KL) divergence is given by:

$$D_{\text{KL}}(\mathscr{P}\|\mathscr{Q}) = \sum_{\xi\in\Xi} p(\xi)\log\left(\frac{p(\xi)}{q(\xi)}\right) \tag{60}$$

Then, starting from the dataset $\mathscr{P}$, remove a data point to return to $\mathscr{Q}$. The KL divergence in this case is:

$$D_{\text{KL}}(\mathscr{Q}\|\mathscr{P}) = \sum_{\xi\in\Xi} q(\xi)\log\left(\frac{q(\xi)}{p(\xi)}\right) \tag{61}$$

In general, $D_{\text{KL}}(\mathscr{Q}\|\mathscr{P}) \neq -D_{\text{KL}}(\mathscr{P}\|\mathscr{Q})$. This results in a form of information hysteresis in the cycle $\mathscr{Q}\to\mathscr{P}\to\mathscr{Q}$, if we associate the KL divergence with the concept of information gain or loss. Notation was kept loose on purpose: Information hysteresis would exist irrespective of whether p and q represent (posterior) probability distribution of the weights,[9] as in Sec. 3.5.1, or the

View Article Online
DOI: 10.1039/D5DD00102A

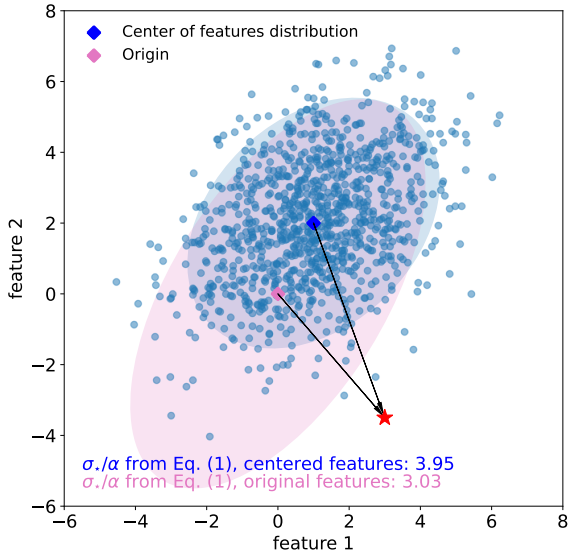

Fig. 8 Effect of application of Eq. (1), where $\mathbf{f}_\star$ and $\mathbf{F}$ are centered or not with respect to the center of the dataset's features distribution, for a toy model with two features. The (unitless) Mahalanobis distance is obtained by expressing the uncertainty from Eq. (1) in units of calibration parameter α.

probability distribution of dataset features, as in Secs. 3.5.3.

C Why (not) center (pseudo)features?

In the discussion above, where the uncertainty of a prediction was interpreted as a Mahalanobis distance, we assumed that the distribution of the (pseudo)features was centered at zero. In linear regression, centering the features ensures that the intercept has a meaningful interpretation, such as representing the mean response when all predictors are at their mean values. In kernel methods, however, the focus shifts to pairwise similarities encoded in the kernel matrix, which implicitly maps the data into a latent space of pseudofeatures.

Centering the kernel—either directly or by centering the pseudofeatures as in the Nyström approximation—adjusts the distribution of data representation in latent space, thereby affecting the variance of predictions, which may reflect both the global mean effect and deviations from this mean. Centering also isolates variability purely due to deviations, aligning the variance estimate more closely with the concept used in linear models, where centering is standard.

However, pseudofeatures centering has a direct impact on variance estimates, especially for bias-less models. For instance, consider a NN model where the output $y_\star = \mathbf{f}_\star^\top \mathbf{w}^L$ is forced to vanish for vanishing features no matter the values assumed by the last-layer weights: the uncertainty on the prediction, $\sigma_\star(\mathbf{f}_\star = 0)$, is always zero by construction—see Eq. (1). In general, the variance for predictions near (far from) the origin of the latent space will be small (large). Nonetheless, this is a characteristic of the model: one could in fact argue that re-centering may introduce spurious *a posteriori* effects that clash with how the model ultimately represents (or learns) data in latent space. By centering the input features one effectively removes the global mean effect, ensuring that the predictions reflect deviations based solely on the relative relationships between data points. Yet, it is not evident that performing centering on pseudofeatures (that are the way the kernel represents data, or are learned by the model in NN architectures) should be encouraged.

Notes and references

1 J. M. Vinter Hansen, *Astronomical Society of the Pacific Leaflets, Vol. 4, No. 166, p. 121*, 1942, **4**, 121.
2 J. Dai, S. Adhikari and M. Wen, *Reviews in Chemical Engineering*, 2024.
3 M. Kulichenko, B. Nebgen, N. Lubbers, J. S. Smith, K. Barros, A. E. A. Allen, A. Habib, E. Shinkle, N. Fedik, Y. W. Li, R. A. Messerly and S. Tretiak, *Chemical Reviews*, 2024, **124**, 13681–13714.
4 P. C. Mahalanobis, *Proceedings of the National Institute of Science of India*, 1936, **2**, 49–55.
5 C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*, MIT press Cambridge, MA, 2006.
6 M. Tipping, *Advances in neural information processing systems*, 2000, **13**, 633.
7 D. J. MacKay, *Neural computation*, 1992, **4**, 415–447.
8 D. J. MacKay, *Neural computation*, 1992, **4**, 448–472.
9 D. J. MacKay, *Neural computation*, 1992, **4**, 590–604.
10 D. A. Cohn, *Neural networks*, 1996, **9**, 1071–1083.
11 G. Imbalzano, Y. Zhuang, V. Kapil, K. Rossi, E. A. Engel, F. Grasselli and M. Ceriotti, *The Journal of Chemical Physics*, 2021, **154**, 074102.
12 E. Daxberger, A. Kristiadi, A. Immer, R. Eschenhagen, M. Bauer and P. Hennig, *Advances in Neural Information Processing Systems*, 2021, **34**, 20089–20103.
13 V. Zaverkin and J. Kästner, *Machine Learning: Science and Technology*, 2021, **2**, 035009.
14 S. Chong, F. Grasselli, C. Ben Mahmoud, J. D. Morrow, V. L. Deringer and M. Ceriotti, *J. Chem. Theory Comput.*, 2023, **19**, 8020–8031.
15 F. Bigi, S. Chong, M. Ceriotti and F. Grasselli, *Machine Learning: Science and Technology*, 2024, **5**, 045018.
16 J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington and J. Sohl-Dickstein, *arXiv preprint arXiv:1711.00165*, 2017.
17 A. Jacot, F. Gabriel and C. Hongler, *Advances in neural information processing systems*, 2018, **31**, 8571.
18 J. Lee, L. Xiao, S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein and J. Pennington, *Advances in neural information processing systems*, 2019, **32**, 8572.
19 C. Liu, L. Zhu and M. Belkin, *Advances in Neural Information Processing Systems*, 2020, **33**, 15954–15964.
20 M. Lázaro-Gredilla and A. R. Figueiras-Vidal, *IEEE transactions on neural networks*, 2010, **21**, 1345–1351.
21 J. Wenger, F. Dangel and A. Kristiadi, *arXiv preprint arXiv:2310.00137*, 2023.
22 J. Zeng, D. Zhang, D. Lu, P. Mo, Z. Li, Y. Chen, M. Rynik, L. Huang, Z. Li, S. Shi *et al.*, *The Journal of Chemical Physics*,

View Article Online
DOI: 10.1039/D5DD00102A

2023, **159**, 054801.
23 J. Behler and M. Parrinello, *Phys. Rev. Lett.*, 2007, **98**, 146401.
24 V. Zaverkin, D. Holzmüller, I. Steinwart and J. Kästner, *Digital Discovery*, 2022, **1**, 605–620.
25 D. Holzmüller, V. Zaverkin, J. Kästner and I. Steinwart, *Journal of Machine Learning Research*, 2023, **24**, 1–81.
26 P. Izmailov, S. Vikram, M. D. Hoffman and A. G. G. Wilson, International conference on machine learning, 2021, pp. 4629–4640.
27 R. M. Neal, *See arXiv preprint arXiv:1206.1901, also published in Handbook of Markov Chain Monte Carlo*, 2011.
28 S. Thaler, G. Doehner and J. Zavadlav, *Journal of Chemical Theory and Computation*, 2023, **19**, 4520–4532.
29 T. Rensmeyer, B. Craig, D. Kramer and O. Niggemann, *Digital Discovery*, 2024, **3**, 2356–2366.
30 J. Harrison, J. Willes and J. Snoek, *arXiv preprint arXiv:2404.11599*, 2024.
31 Y. Gal and Z. Ghahramani, *arXiv preprint arXiv:1506.02142*, 2015.
32 D. A. Nix and A. S. Weigend, Proceedings of 1994 ieee international conference on neural networks (ICNN'94), 1994, pp. 55–60.
33 J. Busk, M. N. Schmidt, O. Winther, T. Vegge and P. B. Jørgensen, *Physical Chemistry Chemical Physics*, 2023, **25**, 25828–25837.
34 E. Heid, C. J. McGill, F. H. Vermeire and W. H. Green, *Journal of Chemical Information and Modeling*, 2023, **63**, 4012–4029.
35 B. Lakshminarayanan, A. Pritzel and C. Blundell, *Advances in neural information processing systems*, 2017, **30**, 6402.
36 J. Carrete, H. Montes-Campos, R. Wanzenböck, E. Heid and G. K. Madsen, *The Journal of Chemical Physics*, 2023, **158**, 204801.
37 M. Kellner and M. Ceriotti, *Machine Learning: Science and Technology*, 2024, **5**, 035006.
38 S. J. Nowlan and G. E. Hinton, in *Evaluation of Adaptive Mixtures of Competing Experts*, 1991.
39 C. Zeni, A. Anelli, A. Glielmo, S. de Gironcoli and K. Rossi, *Digital Discovery*, 2024, **3**, 113–121.
40 B. M. Wood, M. Dzamba, X. Fu, M. Gao, M. Shuaibi, L. Barroso-Luque, K. Abdelmaqsoud, V. Gharakhanyan, J. R. Kitchin, D. S. Levine, K. Michel, A. Sriram, T. Cohen, A. Das, A. Rizvi, S. J. Sahoo, Z. W. Ulissi and C. L. Zitnick, 2025.
41 F. Musil, M. J. Willatt, M. A. Langovoy and M. Ceriotti, *Journal of chemical theory and computation*, 2019, **15**, 906–915.
42 D. Levi, L. Gispan, N. Giladi and E. Fetaya, *Sensors*, 2022, **22**, 5540.
43 P. Pernot, *Journal of Chemical Physics*, 2022, **156**, 114109.
44 P. Pernot, *The Journal of Chemical Physics*, 2022, **157**, 144103.
45 P. Pernot, *APL Machine Learning*, 2023, **1**, 046121.
46 K. Tran, W. Neiswanger, J. Yoon, Q. Zhang, E. Xing and Z. W. Ulissi, *Machine Learning: Science and Technology*, 2020, **1**, 025006.
47 R. A. Fisher, *Annals of eugenics*, 1935, **6**, 391–398.
48 G. Shafer and V. Vovk, *Journal of Machine Learning Research*, 2008, **9**, 371.
49 Y. Hu, J. Musielewicz, Z. W. Ulissi and A. J. Medford, *Machine Learning: Science and Technology*, 2022, **3**, 045028.
50 V. Zaverkin, D. Holzmüller, H. Christiansen, F. Errica, F. Alesiani, M. Takamoto, M. Niepert and J. Kästner, *npj Comput. Mater.*, 2024, **10**, 83.
51 A. R. Tan, S. Urata, S. Goldman, J. C. Dietschreit and R. Gómez-Bombarelli, *npj Computational Materials*, 2023, **9**, 225.
52 L. Kahle and F. Zipoli, *Phys. Rev. E*, 2022, **105**, 015311.
53 Y. Litman, V. Kapil, Y. M. Y. Feldman, D. Tisi, T. Begušić, K. Fidanyan, G. Fraux, J. Higer, M. Kellner, T. E. Li, E. S. Pós, E. Stocco, G. Trenins, B. Hirshberg, M. Rossi and M. Ceriotti, *The Journal of Chemical Physics*, 2024, **161**, 062504.
54 A. Moriarty, K. Morita, K. T. Butler and A. Walsh, *Journal of Open Source Software*, 2022, **7**, 3700.
55 M. Ceriotti and D. E. Manolopoulos, *Physical Review Letters*, 2012, **109**, 100604.
56 G. C. Sosso, D. Donadio, S. Caravati, J. Behler and M. Bernasconi, *Physical Review B*, 2012, **86**, 104301.
57 C. Verdi, F. Karsai, P. Liu, R. Jinnouchi and G. Kresse, *npj Computational Materials*, 2021, **7**, 156.
58 D. Tisi, L. Zhang, R. Bertossa, H. Wang, R. Car and S. Baroni, *Physical Review B*, 2021, **104**, 224202.
59 P. Pegolo, S. Baroni and F. Grasselli, *npj Computational Materials*, 2022, **8**, 24.
60 C. Malosso, L. Zhang, R. Car, S. Baroni and D. Tisi, *npj Computational Materials*, 2022, **8**, 139.
61 M. F. Langer, F. Knoop, C. Carbogno, M. Scheffler and M. Rupp, *Physical Review B*, 2023, **108**, L100302.
62 D. Tisi, F. Grasselli, L. Gigli and M. Ceriotti, *Phys. Rev. Mater.*, 2024, **8**, 065403.
63 P. Pegolo and F. Grasselli, *Frontiers in Materials*, 2024, **11**, 1369034.
64 P. Pegolo, E. Drigo, F. Grasselli and S. Baroni, *The Journal of Chemical Physics*, 2025, **162**, 064111.
65 E. Drigo, S. Baroni and P. Pegolo, *Journal of Chemical Theory and Computation*, 2024, **20**, 6152–6159.
66 B. J. Kleijn and A. W. Van der Vaart, *Electron. J. Statist.*, 2012, **6**, 354.
67 G. E. Box, *Journal of the American Statistical Association*, 1976, **71**, 791–799.
68 A. Grisafi and M. Ceriotti, *The Journal of chemical physics*, 2019, **151**, 204105.
69 R. Balestriero, J. Pesenti and Y. LeCun, *Learning in High Dimension Always Amounts to Extrapolation*, 2021, `https://arxiv.org/abs/2110.09485`.
70 C. Zeni, A. Anelli, A. Glielmo and K. Rossi, *Phys. Rev. B*, 2022, **105**, 165141.
71 E. Facco, M. D'Errico, A. Rodriguez and A. Laio, *Scientific Reports*, 2017, **7**, 12140.

View Article Online
DOI: 10.1039/D5DD00102A

72 A. Glielmo, I. Macocco, D. Doimo, M. Carli, C. Zeni, R. Wild, M. d'Errico, A. Rodriguez and A. Laio, *Patterns*, 2022, **3**, 100589.

73 J. P. Janet, C. Duan, T. Yang, A. Nandy and H. J. Kulik, *Chemical science*, 2019, **10**, 7913–7922.

74 L. E. Schultz, Y. Wang, R. Jacobs and D. Morgan, *Determining Domain of Machine Learning Models using Kernel Density Estimates: Applications in Materials Property Prediction*, 2024, https://arxiv.org/abs/2406.05143.

75 A. Zhu, S. Batzner, A. Musaelian and B. Kozinsky, *J. Chem. Phys.*, 2023, **158**, 164111.

76 S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt and B. Kozinsky, *Nature Communications*, 2022, **13**, 2453.

77 A. K. A. Kandy, K. Rossi, A. Raulin-Foissac, G. Laurens and J. Lam, *Physical Review B*, 2023, **107**, 174106.

78 F. G. Mattioli, F. Sciortino and J. Russo, *Journal of Physical Chemistry B*, 2023, **127**, 3894.

79 G. P. Wellawatte, G. M. Hocky and A. D. White, *Journal of Chemical Physics*, 2023, **159**, 085103.

80 C. Zeni, K. Rossi, A. Glielmo, Ádám Fekete, N. Gaston, F. Baletto and A. D. Vita, *Journal of Chemical Physics*, 2018, **148**, 241739.

81 C. J. Owen, S. B. Torrisi, Y. Xie, S. Batzner, K. Bystrom, J. Coulter, A. Musaelian, L. Sun and B. Kozinsky, *npj Computational Materials*, 2024, **10**, 92.

82 S. Stocker, J. Gasteiger, F. Becker, S. Günnemann and J. T. Margraf, *Machine Learning: Science and Technology*, 2022, **3**, 045010.

83 G. Imbalzano and M. Ceriotti, *Physical Review Materials*, 2021, **5**, 063804.

84 N. Lopanitsyna, G. Fraux, M. A. Springer, S. De and M. Ceriotti, *Physical Review Materials*, 2023, **7**, 045802.

85 J. Nam, J. Peng and R. Gómez-Bombarelli, *Interpolation and differentiation of alchemical degrees of freedom in machine learning interatomic potentials*, 2024, https://arxiv.org/abs/2404.10746.

86 I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, F. Berger, N. Bernstein, A. Bhowmik, S. M. Blau, V. Cărare, J. P. Darby, S. De, F. D. Pia, V. L. Deringer, R. Elijošius, Z. El-Machachi, F. Falcioni, E. Fako, A. C. Ferrari, A. Genreith-Schriever, J. George, R. E. A. Goodall, C. P. Grey, P. Grigorev, S. Han, W. Handley, H. H. Heenen, K. Hermansson, C. Holm, J. Jaafar, S. Hofmann, K. S. Jakob, H. Jung, V. Kapil, A. D. Kaplan, N. Karimitari, J. R. Kermode, N. Kroupa, J. Kullgren, M. C. Kuner, D. Kuryla, G. Liepuoniute, J. T. Margraf, I.-B. Magdău, A. Michaelides, J. H. Moore, A. A. Naik, S. P. Niblett, S. W. Norwood, N. O'Neill, C. Ortner, K. A. Persson, K. Reuter, A. S. Rosen, L. L. Schaaf, C. Schran, B. X. Shi, E. Sivonxay, T. K. Stenczel, V. Svahn, C. Sutton, T. D. Swinburne, J. Tilly, C. van der Oord, E. Varga-Umbrich, T. Vegge, M. Vondrák, Y. Wang, W. C. Witt, F. Zills and G. Csányi, *A foundation model for atomistic materials chemistry*, 2024, https://arxiv.org/abs/2401.00096.

87 S. Chong, F. Bigi, F. Grasselli, P. Loche, M. Kellner and M. Ceriotti, *Faraday Discuss.*, 2025.

88 M. Aldeghi, D. E. Graff, N. Frey, J. A. Morrone, E. O. Pyzer-Knapp, K. E. Jordan and C. W. Coley, *Journal of Chemical Information and Modeling*, 2022, **62**, 4660–4671.

89 D. van Tilborg, A. Alenicheva and F. Grisoni, *Journal of Chemical Information and Modeling*, 2022, **62**, 5938–5951.

90 T. Liu and A. Barnard, Proceedings of Machine Learning Research, 2023.

91 A. S. Barnard and B. L. Fox, *Chemistry of Materials*, 2023, **35**, 8840–8856.

92 G. Csányi, T. Albaret, M. C. Payne and A. De Vita, *Physical Review Letters*, 2004, **93**, 175503.

93 Z. Li, J. R. Kermode and A. De Vita, *Phys. Rev. Lett.*, 2015, **114**, 096405.

94 J. Vandermause, Y. Xie, J. S. Lim, C. J. Owen and B. Kozinsky, *Nature Communications*, 2022, **13**, 5183.

95 K. Tran and Z. W. Ulissi, *Nature Catalysis*, 2018, **1**, 696.

96 A. Marcolongo, T. Binninger, F. Zipoli and T. Laino, *ChemSystemsChem*, 2020, **2**, e1900031.

97 K. Rossi, V. Jurásková, R. Wischert, L. Garel, C. Corminbœuf and M. Ceriotti, *Journal of Chemical Theory and Computation*, 2020, **16**, 5139.

98 H. Zhang, V. Juraskova and F. Duarte, *Nature Communications*, 2024, **15**, 6114.

99 A. Anelli, H. Dietrich, P. Ectors, F. Stowasser, T. Bereau, M. Neumann and J. van den Ende, *CrystEngComm*, 2024, **26**, 5845–5849.

100 C. Schran, K. Brezina and O. Marsalek, *Journal of Chemical Physics*, 2020, **153**, 104105.

101 C. van der Oord, M. Sachs, D. P. Kovács, C. Ortner and G. Csányi, *npj Comput. Mater.*, 2023, **9**, 168.

102 M. Kulichenko, K. Barros, N. Lubbers, Y. W. Li, R. Messerly, S. Tretiak, J. S. Smith and B. Nebgen, *Nature Computational Science*, 2023, **3**, 230.

103 D. Schwalbe-Koda, A. R. Tan and R. Gómez-Bombarelli, *Nature Communications*, 2021, **12**, 5104.

104 R. Vrabel, *International Journal of Pure and Applied Mathematics*, 2016, **111**, 643–646.

105 V. V. Fedorov, *Theory of optimal experiments*, Academic Press, New York, 1972.

106 S. Luttrell, *Inverse Problems*, 1985, **1**, 199.

107 Y. Lysogorskiy, A. Bochkarev, M. Mrovec and R. Drautz, *Phys. Rev. Mater.*, 2023, **7**, 043801.

108 E. V. Podryabinkin and A. V. Shapeev, *Computational Materials Science*, 2017, **140**, 171–180.

109 I. S. Novikov, Y. V. Suleimanov and A. V. Shapeev, *Phys. Chem. Chem. Phys.*, 2018, **20**, 29503–29512.

110 E. V. Podryabinkin, E. V. Tikhonov, A. V. Shapeev and A. R. Oganov, *Phys. Rev. B*, 2019, **99**, 064114.

111 E. V. Podryabinkin, A. G. Kvashnin, M. Asgarpour, I. I. Maslenikov, D. A. Ovsyannikov, P. B. Sorokin, M. Y. Popov

View Article Online
DOI: 10.1039/D5DD00102A

and A. V. Shapeev, *J. Chem. Theory Comput.*, 2022, **18**, 1109–1121.

112 F. N. Jalolov, E. V. Podryabinkin, A. R. Oganov, A. V. Shapeev and A. G. Kvashnin, *Adv. Theory Simul.*, 2024, **7**, 2301171.

113 K. Gubaev, E. V. Podryabinkin and A. V. Shapeev, *The Journal of chemical physics*, 2018, **148**, 241727.

114 M. Karabin and D. Perez, *The Journal of Chemical Physics*, 2020, **153**, 094110.

115 D. Schwalbe-Koda, S. Hamel, B. Sadigh, F. Zhou and V. Lordi, *arXiv preprint arXiv:2404.12367*, 2024.

116 D. Montes de Oca Zapiain, M. A. Wood, N. Lubbers, C. Z. Pereyra, A. P. Thompson and D. Perez, *npj Computational Materials*, 2022, **8**, 189.

117 J. Mockus, W. Eddy and G. Reklaitis, *Bayesian Heuristic approach to discrete and global optimization: Algorithms, visualization, software, and applications*, Springer Science & Business Media, 2013, vol. 17.

118 B. Shahriari, K. Swersky, Z. Wang, R. P. Adams and N. De Freitas, *Proceedings of the IEEE*, 2015, **104**, 148–175.

119 R. Ramprasad, R. Batra, G. Pilania, A. Mannodi-Kanakkithodi and C. Kim, *npj Computational Materials*, 2017, **3**, 54.

120 Y. Zhang, D. W. Apley and W. Chen, *Scientific reports*, 2020, **10**, 4924.

121 J. Snoek, H. Larochelle and R. P. Adams, *Advances in neural information processing systems*, 2012, **25**, 2951.

122 D. R. Jones, M. Schonlau and W. J. Welch, *Journal of Global optimization*, 1998, **13**, 455–492.

123 N. Srinivas, A. Krause, S. M. Kakade and M. Seeger, *arXiv preprint arXiv:0912.3995*, 2009.

124 M. Todorović, M. U. Gutmann, J. Corander and P. Rinke, *Npj computational materials*, 2019, **5**, 35.

125 B. J. Shields, J. Stevens, J. Li, M. Parasram, F. Damani, J. I. M. Alvarado, J. M. Janey, R. P. Adams and A. G. Doyle, *Nature*, 2021, **590**, 89–96.

126 A. Tran, J. Tranchida, T. Wildey and A. P. Thompson, *The Journal of Chemical Physics*, 2020, **153**, 074705.

127 T. Yamashita, N. Sato, H. Kino, T. Miyake, K. Tsuda and T. Oguchi, *Physical Review Materials*, 2018, **2**, 013803.

128 Y. Zuo, M. Qin, C. Chen, W. Ye, X. Li, J. Luo and S. P. Ong, *Materials Today*, 2021, **51**, 126–135.

129 M. Ghorbani, M. Boley, P. Nakashima and N. Birbilis, *Scientific Reports*, 2024, **14**, 8299.

130 R. Shintaku, T. Tamura, S. Nogami, M. Karasuyama and T. Hirose, *Physical Chemistry Chemical Physics*, 2024, **26**, 27561–27566.

131 S. Ju, T. Shiga, L. Feng, Z. Hou, K. Tsuda and J. Shiomi, *Physical Review X*, 2017, **7**, 021024.

132 T. Mäkinen, A. D. Parmar, S. Bonfanti and M. J. Alava, *arXiv preprint arXiv:2411.05529*, 2024.

133 D. Khatamsaz, B. Vela, P. Singh, D. D. Johnson, D. Allaire and R. Arróyave, *npj Computational Materials*, 2023, **9**, 49.

134 V. Torsti, T. Mäkinen, S. Bonfanti, J. Koivisto and M. J. Alava, *APL Machine Learning*, 2024, **2**, 016119.

135 D. Kurunczi-Papp and L. Laurson, *Modelling and Simulation in Materials Science and Engineering*, 2024, **32**, 085013.

136 J. K. Pedersen, C. M. Clausen, O. A. Krysiak, B. Xiao, T. A. Batchelor, T. Löffler, V. A. Mints, L. Banko, M. Arenz, A. Savan *et al.*, *Angewandte Chemie*, 2021, **133**, 24346–24354.

137 J. Ohyama, Y. Tsuchimura, H. Yoshida, M. Machida, S. Nishimura and K. Takahashi, *The Journal of Physical Chemistry C*, 2022, **126**, 19660–19666.

138 J. P. Janet, S. Ramesh, C. Duan and H. J. Kulik, *ACS Central Science*, 2020, **6**, 513.

139 R. Jalem, K. Kanamori, I. Takeuchi, M. Nakayama, H. Yamasaki and T. Saito, *Scientific Reports*, 2018, **8**, 5845.

140 B. Deng, P. Zhong, K. Jun, J. Riebesell, K. Han, C. J. Bartel and G. Ceder, *Nature Machine Intelligence*, 2023, **5**, 1031–1041.

141 A. Mazitov, F. Bigi, M. Kellner, P. Pegolo, D. Tisi, G. Fraux, S. Pozdnyakov, P. Loche and M. Ceriotti, *arXiv preprint arXiv:2503.14118*, 2025.

142 S. P. Niblett, P. Kourtis, I.-B. Magdău, C. P. Grey and G. Csányi, *Transferability of datasets between Machine-Learning Interaction Potentials*, 2024, `https://arxiv.org/abs/2409.05590`.

143 J. Riebesell, R. E. A. Goodall, P. Benner, Y. Chiang, B. Deng, G. Ceder, M. Asta, A. A. Lee, A. Jain and K. A. Persson, *Matbench Discovery – A framework to evaluate machine learning crystal stability predictions*, 2024, `https://arxiv.org/abs/2308.14920`.

144 B. Póta, P. Ahlawat, G. Csányi and M. Simoncelli, *arXiv preprint arXiv:2408.00755*, 2024.

145 V. Zaverkin, D. Holzmüller, L. Bonfirraro and J. Kästner, *Physical Chemistry Chemical Physics*, 2023, **25**, 5383–5396.

146 D. Buterez, J. P. Janet, S. J. Kiddle, D. Oglic and P. Lió, *Nature Communications*, 2024, **15**, 1517.

147 H. Kaur, F. D. Pia, I. Batatia, X. R. Advincula, B. X. Shi, J. Lan, G. Csányi, A. Michaelides and V. Kapil, *Faraday Discussions*, 2025, **256**, 120–138.

148 S. J. Pan and Q. Yang, *IEEE Transactions on Knowledge and Data Engineering*, 2010, **22**, 1345–1359.

149 A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly and N. Houlsby, *ECCV*, 2020.

150 R. Ramakrishnan, P. O. Dral, M. Rupp and O. A. von Lilienfeld, *Journal of Chemical Theory and Computation*, 2015, **11**, 2087–2096.

151 A. E. A. Allen, N. Lubbers, S. Matin, J. Smith, R. Messerly, S. Tretiak and K. Barros, *npj Computational Materials*, 2024, **10**, Article 154.

152 O. Voznyy, L. Levina, J. Z. Fan, M. Askerka, A. Jain, M.-J. Choi, O. Ouellette, P. Todorović, L. K. Sagar and E. H. Sargent, *ACS Nano*, 2019, **13**, 11122–11128.

153 J. Leeman, J. S. Yuhan Liu, S. B. Lee, P. Bhatt, L. M. Schoop and R. G. Palgrave, *PRX Energy*, 2024, **3**, 011002.

154 Q. Liang, A. E. Gongora, Z. Ren, A. Tiihonen, Z. Liu, S. Sun, J. R. Deneault, D. Bash, F. Mekki-Berrada, S. A. Khan, K. Hippalgaonkar, B. Maruyama, K. A. Brown, J. F. III and

View Article Online
DOI: 10.1039/D5DD00102A

T. Buonassisi, *npj Computational Materials*, 2021, **7**, 188.

155 M. M. Billah, M. Elleithy, W. Khan, S. Yildiz, Z. E. Eger, S. Liu, M. Long and P. Acar, *Advanced Engineering Materials*, 2024, **26**, 2401299.

View Article Online
DOI: 10.1039/D5DD00102A

Assistant Professor, Dr. Kevin Rossi,
Materials Science and Engineering Dept.,
& Climate Safety and Security Center,
Technische Universiteit Delft,
Mekelweg 2, 2628 CD
Delft, Netherlands

15.04.2024

Dear Editorial team,

We are pleased to submit our manuscript titled "Uncertainty in the Era of Machine Learning for Atomistic Modeling" for consideration in Digital Discovery (following preliminary discussions with Alexander Whiteside, Assistant Editor, Digital Discovery).

As part of the data availability statement, I confirm that no primary research results, software or code have been included and no new data were generated or analysed as part of this review.

Sincerely Yours,
Kevin Rossi