

PERSPECTIVE • OPEN ACCESS

Machine learning for the solution of the Schrödinger equation

To cite this article: Sergei Manzhos 2020 *Mach. Learn.: Sci. Technol.* **1** 013002

View the [article online](#) for updates and enhancements.

You may also like

- [Outlook for artificial intelligence and machine learning at the NLS-II](#)
Stuart I Campbell, Daniel B Allan, Andi M Barbour et al.
- [Probe microscopy is all you need](#)
Sergei V Kalinin, Rama Vasudevan, Yongtao Liu et al.
- [Building robust machine learning models for small chemical science data: the case of shear viscosity of fluids](#)
Nikhil V S Avula, Shivanand Kumar Veeram, Sudarshan Behera et al.



PERSPECTIVE

OPEN ACCESS

RECEIVED
9 December 2019REVISED
24 February 2020ACCEPTED FOR PUBLICATION
6 March 2020PUBLISHED
27 April 2020

Original content from
this work may be used
under the terms of the
[Creative Commons
Attribution 4.0 licence](#).

Any further distribution
of this work must
maintain attribution to
the author(s) and the
title of the work, journal
citation and DOI.



Machine learning for the solution of the Schrödinger equation

Sergei Manzhos

Centre Énergie Matériaux Télécommunications, Institut National de la Recherche Scientifique, 1650 Boulevard Lionel-Boulet, Varennes, QC J3X1S2, Canada

E-mail: sergei.manzhos@emt.inrs.ca**Keywords:** machine learning, Schrödinger equation, density functional theory, neural network, Gaussian process regression, kernel ridge regression, genetic algorithm

Abstract

Machine learning (ML) methods have recently been increasingly widely used in quantum chemistry. While ML methods are now accepted as high accuracy approaches to construct interatomic potentials for applications, the use of ML to solve the Schrödinger equation, either vibrational or electronic, while not new, is only now making significant headway towards applications. We survey recent uses of ML techniques to solve the Schrödinger equation, including the vibrational Schrödinger equation, the electronic Schrödinger equation and the related problems of constructing functionals for density functional theory (DFT) as well as potentials which enter semi-empirical approximations to DFT. We highlight similarities and differences and specific difficulties that ML faces in these applications and possibilities for cross-fertilization of ideas.

1. Introduction

Machine learning (ML) has been increasingly used in theoretical and computational chemistry and computational materials science. It has been particularly successful in accelerating materials research with ML-assisted alloy design, discovery of solar cell and battery materials, etc [1–6]. These uses of ML effectively create shortcuts through the field of quantum chemistry calculations. Quantum chemistry calculations themselves are ultimately based on the solution of the nuclear Schrödinger equation (SE) or the electronic SE. The nuclear SE is usually solved on a potential energy surface (of electronic energies) to obtain vibrational properties or reaction dynamics [7–9]. The electronic SE is solved directly as an ODE (ordinary differential equation) or in a roundabout way (with density functional based methods) to obtain electronic energies, band structures, and various electronic properties [10–16].

In the application to construction of interatomic potentials, machine learning has grown from being considered a curiosity with mediocre accuracy in the 1990s [17, 18] to a highly-accurate approach which is accepted as such in applied literature. This followed a series of developments since the mid-2000s from the first spectroscopically accurate molecular potential energy surfaces by Manzhos and Carrington [19] to accurate potentials suitable for large-scale solid state and nanoparticles simulations by Behler *et al* [20–22]. These developments were achieved by combining general properties of ML methods, mostly NN (neural networks) and GPR (Gaussian process regression), with modifications specific to the application, such as sum-of-product representations [23] or expansions of energy over orders of coupling [24, 25] or its representation as a sum of atomic energies dependent on the chemical environment [26, 27] as well as data selection schemes specific to the application [19, 28]. The field of machine learning for interatomic potentials has been active and diverse and will not be reviewed here.

Exploration of ML for the solution of the vibrational and electronic Schrödinger equation is also not new (as will be shown below), but as of yet ML has not established itself as a highly-accurate approach to solve the SE in applications. There are signs that it is on the verge of doing so, as promising results have recently appeared in the literature for the solution of long-outstanding genuinely difficult problems such as construction of kinetic energy and exchange-correlation functionals as well as solution of the multi-electron SE, as described below. It is likely that also in this direction, highly accurate solutions and acceptability for applications will be achieved by combining generic properties of ML methods, such as universal

approximator properties, with modifications specific to the problem of solving the SE or building functionals, including adapted architectures of ML methods and data selection schemes. It is therefore important to recognize similarities and differences in the requirements that can be addressed via ML among different applications.

In this Perspective, we survey recent uses of ML techniques to solve the Schrödinger equation, including the vibrational Schrödinger equation and the electronic problem: the electronic Schrödinger equation and the related problems of constructing exchange-correlation and kinetic energy functionals for Kohn-Sham (KS-) and orbital-free (OF-) density functional theory (DFT) as well as use of ML for semi-empirical approximations used in density functional based approaches such as DFTB (density functional tight binding) and dispersion-corrected DFT (DFT-D). We only consider the use of ML for the solution of the vibrational and electronic SE or the KS equation or the Hohenberg-Kohn (HK) equation and not methods that aim to avoid such solutions (such as those directly mapping the molecular structure to the spectrum or energy or properties without solving for the density or wavefunction [29–36]); we also do not consider uses of ML for other types of quantum mechanical modelling or other types of differential or integral equations [37–42].

We do not aim to present a comprehensive review but rather a survey allowing similarities and differences of ML uses in all these applications, as well as promising directions for future research, to transpire. This is not a review of ML methods; the key machine learning techniques that found use in quantum chemistry are well reviewed elsewhere [3–5, 43, 44], their description will not be repeated there; the reader is advised to consult the literature for their introduction, such as [45, 46] for neural networks, [47, 48] for Gaussian process regression (GPR), [49] for kernel ridge regression (KRR; note the similarity in the form of the function representation between GPR and KRR), [50] for genetic algorithms (GA) and [51] for particle swarm optimization. These methods as applied to quantum mechanics have recently been reviewed [44]. They are also briefly introduced in the literature cited below for specific areas of application of ML. We also do not review or introduce the methods to solve the vibrational Schrödinger equation or the electronic problem for the same reason; there is ample literature on these topics. Only specific issues are introduced relevant to the understanding of the ML use in those methods.

This Perspective is organized as follows: in section 2, we describe uses of ML to solve the vibrational Schrödinger equation and in section 3, the electronic Schrödinger equation (including the KS equation). These sections thus consider use of ML for ODE solution. Much of ML work for solution of the SE is done on synthetic potentials; those are considered in section 2 when the potential shape is typical of interatomic potentials and in section 3 when it is representative of the electronic SE. Section 4 is dedicated to density functional based approaches to solving the electronic problem: construction of functionals and semi-empirical potentials (pseudopotentials and dispersion corrections as well as repulsive potentials in DFTB) which are part of the electronic structure methods (as opposed to complete potential energy surfaces which are not considered here). Finally, in section 5, we highlight similarities and differences in issues and approaches among the above applications as well as in relation to the more developed applications of ML to building interatomic potentials, in the hope that this can foster some cross-fertilization of ideas.

2. Machine learning for solution of the vibrational Schrödinger equation

ML has been applied to the solution of the vibrational SE. In some works, the wavefunction was represented as values on a grid and in others, as a continuous function (such as a NN model). Zeiri *et al* [52] used a genetic algorithm to solve the SE on a one-dimensional double-well potential. GA populations consisted of numerical values of the wavefunction on an equidistant grid. The algorithm searched for the state with an energy closest to a reference energy. The lowest state was found as closest to zero reference energy, the first excited state as the closest state orthogonal to the ground state, and so on. A similar approach was used by Saha *et al* for a one-dimensional symmetric double well problem and a two-dimensional coupled harmonic oscillator considering the ground and the first excited state [53].

Makarov and Metiu [54] used genetic programming whereby a functional form of the wavefunction was constructed from primitives. The wavefunction was represented as a tree with nodes for various functions or operators and 'leaves' for parameters or variables. A GA-like algorithm was used to find the best function. They constructed in this way wavefunctions for one-dimensional single and double well potentials; the algorithm was helped by imposing asymptotic conditions [54].

Neural networks were used to directly represent bound-state wavefunctions, mostly for synthetic problems. Lagaris *et al* [55] used a NN with ridge activation functions [56] to solve the SE

$$\hat{H}\Psi(\mathbf{r}) = \left(\hat{T} + V(\mathbf{r})\right) \Psi(\mathbf{r}) = E\Psi(\mathbf{r}) \quad (1)$$

where \mathbf{r} spans the configuration space, \hat{T} is the KEO (kinetic energy operator), and V is the potential, for a one-dimensional Morse potential as well as two and three coupled anharmonic oscillators (i.e. potentials

often used to model inter-atomic interactions) for several lowest-energy states. The wavefunction was represented with a single hidden layer NN with sigmoid activation functions as

$$\Psi(\mathbf{r}) = e^{-\beta r^2} \text{NN}(\mathbf{r}) \quad (2)$$

with the exponential factors assuring correct asymptotic behavior for bound states. The NN parameters were tuned to minimize the residual at a set of points \mathbf{r}_i on equidistant grids,

$$\min \sum_i [(\hat{H} - E) \Psi(\mathbf{r}_i)]^2 \quad (3)$$

and the energy was computed from the wavefunction ($E = \langle \Psi | \hat{H} | \Psi \rangle$, $\hat{H} = \hat{T} + V$).

We note that the authors' use of a single hidden layer NN (which is a universal approximator [57, 58]) is equivalent to using a spectral representation with a tunable basis, and the use of equation (3) is equivalent to rectangular collocation (i.e. solving the SE by satisfying it at a set of collocation points rather than variationally) [59]. This has the advantage that the wavefunction expansion need not be integrable or differentiable everywhere, allowing for use of discontinuous or non-integrable functions (the common sigmoidal functions e.g. can have infinite integrals over their infinite support) which only need to be smooth at the collocation points [23, 28, 60]. Lagaris *et al* also solved the variational problem with a NN wavefunction.

Nakanishi and Sugawara [61, 62] used an NN with ridge activation functions to represent the wavefunction of a dozen lowest-energy levels of a one-dimensional SE for a harmonic oscillator, a double-well potential, and a Morse oscillator, obtaining level accuracy on the order of 0.1% with fewer than 10 neurons. A genetic algorithm was used to pre-optimize NN parameters before derivatives-based optimization. Energy was a fitting parameter. They used an amplitude-phase decomposition of the wavefunction,

$$\Psi(x) = A(x) \sin(S(x)) \quad (4)$$

where both $A(x)$ and $S(x)$ are separate single-hidden layer NNs. Asymptotics was not artificially enforced and was obtained with NN training. NNs were trained to minimize the residual on a set of about a 100 uniformly distributed random points within a range where the wavefunction has appreciable amplitude.

The above works used a separate NN for each state with all NN parameters trained together. Manzhos and Carrington [63] used linear algebra to determine linear coefficients in the NN representation of the wavefunctions, and many levels were computed at the same time from one set of nonlinear NN parameters. NNs with radial basis functions as activators (RBFNN), which are also universal approximators [64, 65], were used to ensure asymptotics instead of equation (2):

$$\Psi_k(\mathbf{r}) = \sum_{i=1}^N c_{ik} \exp \left(-b_i^2 \sum_{j=1}^d (w_{ij} - r_j)^2 \right) + h_k. \quad (5)$$

It was suggested that RBFNNs escape the curse of dimensionality [66]. The energy was a fitting parameter that was adjusted to be equal to that computed from the wavefunction. A method to solve rectangular matrix pencils by Boutry *et al* was used to solve the collocation equation [67]. In this way the SE for coupled harmonic oscillators in up to six dimensions ($d = 6$) were solved for several lowest-energy levels. Rather than using an equidistant collocation grid, it was found advantageous to use a non-uniform point distribution; the points were selected from an 'inverse potential' probability density function: points \mathbf{r} uniformly and randomly sampled in the configuration space were accepted if

$$\frac{V_{\max} - V(\mathbf{r})}{V_{\max}} > \text{rand} - \text{bias} \quad (6)$$

where *rand* is a random number uniformly distributed over [0; 1] and *bias* is used to adjust the point density at V_{\max} , the maximum used value of the potential. Only 5000 collocation points and on the order of 50 neurons were used. This distribution was found to be more efficient than a uniform grid, similar to the experience when fitting potential energy surfaces or solving the electronic Schrödinger equation [19, 28, 68, 69].

This approach was also used by the same authors to represent vibrational wavefunctions of a real molecule, H₂O [70]. It was not obvious how the accuracy achieved on synthetic potentials in prior works would translate to real molecules, and this work provided an insight. It took 6000 collocation points and 50 neurons to achieve an accuracy of about 2 cm⁻¹ for the first five vibrational levels.

Recently Teng [71] used a RBFNN together with Variational Monte Carlo (VMC) calculations to solve the SE for a two-dimensional harmonic oscillator in a linear potential and a one-dimensional particle in a box with a linear potential. Space was sampled with a very large (50 000) number of points.

Spectroscopically accurate (accuracy better than 1 cm^{-1}) ML solutions for multiple (dozens or hundreds) levels of the vibrational SE for real polyatomic molecules are still wanting. The computational power and the ML algorithms which are currently easily available suggest they are feasible. One factor that makes achieving spectroscopic accuracy with real molecules cumbersome is the use of the exact KEO. The synthetic problems cited above used simple KEOs based on second derivatives. The KEO is complicated in internal coordinates [72]. In the case of the water molecule, it was easy to apply the exact KEO analytically to the neurons, which is not the case for large molecules. We have shown, however, that this issue is easily circumvented by the application of the exact KEO numerically [73]. Sub- cm^{-1} accuracy for a four-atomic molecule ($d = 6$) can be obtained with equation (4) when using about 100 000 collocation points and about 30 000 basis functions *without* ML optimization of the basis or of the points [28, 73, 74]. It is expected that significant gains can be achieved for computational vibrational spectroscopy by basis set and point optimization with ML. This is supported by results to date with optimized bases and collocation points for vibrational [53, 74, 75] and electronic [69] (see below) problems. Multi-output NNs (as opposed to level-by-level calculations) are expected to be useful.

3. Machine learning for solution of the electronic Schrödinger equation

3.1. Wavefunction representation

The same paper of Zeiri *et al* [52] described in section 2 also applied the same GA based approach to a model electronic problem of the HF (Hartree–Fock) or DFT type (for 2 electrons) on a synthetic potential (harmonic binding potential between the electron and nucleus) as well as the helium atom. The authors achieved an accuracy on the order of 10^{-2} Ha and pointed out inefficiency of an even-spaced grid that they used. Only the ground state was considered using a single radial coordinate.

Chaudhury and Bhattacharyya [76] also represented the wavefunction as a string of values on a grid and used a GA to minimize the residual of the SE on the grid; the energy was computed from the wavefunction. One-dimensional (radial) SE for an electron in a screened Coulomb potential and an anharmonic (quartic) oscillator were solved and the ground state for a two-electron (He) atom was also computed. They also applied the GA in the framework of the perturbation theory, solving for the second- and higher-order corrections to the wavefunction, which were similarly encoded as values on a grid. The accuracy, however, was only on the order of 10^{-1} Ha for the ground state of He. The same group later used a similar approach to solve the SE for an H atom including the first excited state ($2s$) which was produced by using an orthogonality constraint as penalty terms in the fitness function [53].

Caetano *et al* [77] used a single hidden layer NNs and an amplitude-phase decomposition of the wavefunction similar to that of equation (4) (Nakanishi and Sugawara). GA was used to find near-optimal NN parameters by minimizing the SE residual on a mesh. While standard sigmoid neurons were used in the hidden layer, the output layer had Gaussian neurons, which enforced asymptotics. They compared results with energy considered as a fitting parameter with energy computed from the wavefunction; the convergence with the latter was better. For a test case of a one-dimensional double harmonic oscillator, a 0.1 mHa accuracy was obtained for the ground state. A mHa accuracy was obtained for the ground state of the H atom (one-dimensional radial SE). The KS equation was solved for He and Li atoms with a ~ 10 mHa accuracy for the total energy. They also considered the Hooke atom and the TF (Thomas-Fermi) atom.

The same paper of Lagaris *et al* [55] considered in section 2 also used their NN approach (equation (2)) to solve radial SE and Dirac equations (where separate NNs were used for the two parts of the reduced radial wavefunction) for muonic atoms and the equation for the $n + \alpha$ system. Saito [78] used a standard single hidden layer NN to represent the wavefunction of the Bose–Hubbard model.

Choo *et al* [79] and Hermann *et al* [80] used an NN representation of the wavefunction in conjunction with the variational Monte Carlo training approach to compute energies and potential curves of small molecules approaching full CI (configuration interaction) accuracy.

Machine learning has been used to select important configurations in the CI method. Coe used NNs to choose important configurations in an iterative CI scheme [81, 82]. This significantly sped up convergence and overall reduced the CPU cost of single point calculations as well as when building potential energy curves for N_2 , H_2O , and CO . Potential curves of near-full CI quality were obtained at a tenths of the cost. Single hidden layer NNs were used with (binary) occupations of the spin orbitals in the configuration of interest as inputs. The output is the predicted transformed coefficient of the configuration which is fitted to coefficients in the training set. Advantages over stochastic configuration selection were demonstrated.

It is only recently that ML, specifically deep NNs, have started to be used to solve multi-electron SE. Han *et al* [83] computed ground-state energies of Be, B, LiH, and a chain of 10 hydrogen atoms. The antisymmetric nature of the wavefunction was imposed by a product representation whose components were represented with multi-layer NNs. The ground state was obtained by minimizing the variance of the local energy. For two-electron system, a mHa accuracy was achieved. The accuracy worsened towards 10^{-1} Ha for the larger systems.

A more involved approach has been developed by Schütt *et al* [84] using a deep NN [85]. The NN inputs included atom types and position. The predictors of chemical environment of atoms and atom pairs were deduced by the first NN stage and were used to predict the energy and the Hamiltonian and overlap matrix in a basis of atomic orbitals. Ultimately, basis coefficients were predicted providing the wavefunction. A mHa accuracy was achieved for water, malondialdehyde, ethanol, and uracil.

Then works of Han *et al* and Schütt *et al* are examples of combining a generic ML method with application-specific representations to achieve interesting accuracy. This is the road that was previously taken to achieve highly accurate potential energy surfaces, and it is also proving its value with ML for SE.

3.2. Basis tuning

Schütt and VandeVondele [86] used GPR to tune a localized basis set which achieved, with a minimal basis, near-complete basis results in MD simulations of liquid water for structural observables such as pair correlation functions. The basis was built in a molecular geometry-dependent way; descriptors for GPR were sensitive to atom's chemical environment.

3.3. ML as booster to lower-level methods to approximate higher level methods

There is often a high-degree of correlation between the results of lower and higher levels of *ab initio* theory. On the basis of such correlation between Hartree–Fock and MP4 (Møller-Plesset 4th order perturbation theory) energies, Malshe *et al* [87] used a NN to predict MP4(SDQ) energies with the same basis set for configurations of vinyl bromide. The use of ML saved about 78% of the computational time that would be required for the MP4(SDQ) calculations. A single hidden layer NN with 80 to 140 neurons was used to achieve accuracies on the order of 0.03 eV.

Earlier, Silva *et al* [88] used NNs used to fit the correlation energy of atoms and diatomic molecules. They were able to recover about 90% of the correlation energy as function of atomic charges and interatomic bond lengths, with a single hidden layer NN.

McGibbon *et al* [89] trained a NN on a large set (on the order of 10^5) of CCSD(T) (coupled clusters singles and doubles with perturbative triples) data to improve MP2 (Møller-Plesset 2nd order perturbation theory) results by predicting corrections to the energies. This was shown to significantly improve binding energies of weakly bound compounds dominated by non-covalent interactions. A NN with 10 hidden layers with 50 neurons each was trained as a function of 22 features to reweight energy components of MP2 to fit CCSD(T) data.

Margraf and Reuter [90] on the other hand used MP2 to reduce the computational cost of a coupled cluster calculation. Approximate T-amplitudes estimated using MP2 were used as inputs for machine learning to predict the CCSD correlation energy using KRR.

Welborn *et al* [91] used GPR to predict MP2 and CCSD correlation energies from HF calculations, using the Fock matrix *F*, Coulomb matrix *J*, and exchange matrix *K* as descriptors. They demonstrated degrees of transferability among geometries, molecules, and elements when training the GPR model on a set of configuration of a training set of small molecules. The same group later used GPR to predict the correlation energies based on B3LYP calculations [92]. MP2 amplitudes together with other descriptors such as HF matrix elements (*F*, *J*, *K*), two-electron integrals and MP2 denominators were also used to learn CCSD amplitudes by Townsend and Vogiatzis [93] using the *k*-nearest neighbors algorithm [94].

3.4. ML for data selection

Ku *et al* [69] applied ML to optimize the collocation point set when solving the KS equation with the rectangular collocation method [68]. They used a combination of GPR and GA to reduce the number of points from about 50 000 to about 2000 while maintaining mHa accuracy. Important in such optimization is avoidance of repeated solution of the SE on trial point sets which would largely defeat the purpose. Instead of solving the KS equation on trial point sets, Ku *et al* showed that a good solution can be achieved with points which are best for ML representation of the potential.

4. Machine learning for density functional theory

Similar to the uses of ML for solving the SE, applications to DFT were often explored on simplified synthetic systems, including one-dimensional systems, which as of this writing continue to be prominent in the literature, rather than real-life molecules and solids. One reason for this is that exact ground state energies (and energy components) and densities that are used in ML can then be computed from the exact solution of the SE. In the last couple years, however, accurate machine-learned XC (exchange-correlation) functionals, kinetic energy functionals (KEF), and total energy functionals have been reported.

4.1. Exchange-correlation functionals

The field of ML for exchange-correlation functionals is in full bloom. While machine-learned functionals are still not available in end-user codes, promising results are appearing that create hope of ML functionals entering applied research soon. While many works continue to focus on synthetic, often low-dimensional systems, machine-learned functionals for real-life systems have also appeared.

Nagai *et al* [95] used a multi-layer NN to map the electron density to the Hartree-exchange-correlation potential for a test system—a one-dimensional two-particle spinless fermion model. A multi-layer NN was also used to learn the electron correlation in the Anderson-Hubbard model by Ma *et al* [96]. Custodio *et al* [97] used single- and two-hidden layer NN to learn an LSDA type functional for a one-dimensional Hubbard model. Schmidt *et al* [98] used a neural network to learn the universal exchange-correlation functional that simultaneously reproduces both the exact exchange-correlation energy and the potential, for a one-dimensional system with two strongly correlated electrons, in a non-singular potential. A multi-layer NN was used with exponential linear neurons (rather than common sigmoidal neurons), and symmetry was inbuilt into it by fixing of first layer weights. Ryczko *et al* [99] used a large deep NN to predict correlation, exchange, external, kinetic, and total energies simultaneously for model systems with harmonic oscillator and simple random external potentials. They also showed that for these systems, a deep NN is capable of predicting these directly from the external potential, effectively obviating the need for the KS scheme.

Similar to the use of ML to solve the SE, while very small errors can be achieved on such model systems (e.g. 0.03% in the ground state energy in the work of Custodio *et al*) they might not be representative of the accuracy which would be achieved with the same methods for real molecules or solids.

Zhou *et al* [100] machine-learned the exchange-correlation potential for real molecules. They showed that using a convolutional NN that learns the exchange correlation potential from the electron density based on small molecule CCSD data, one can achieve portability to larger molecules. The convolutional NN effectively generated a density-derived description inside itself. More accurate bond lengths and vibrational frequencies were obtained with the NN functional than with B3LYP for small molecules. The NN was also able to learn the weak vdW (van der Waals) interactions from CCSD data. Lei and Medford [101] fitted the exchange-correlation portion of B3LYP formation energies of small molecules to chemical accuracy. They used a two-hidden layer NN with rectified linear neurons. Space-distributed spherical harmonics were used as rotationally invariant descriptors rather than the density directly. This work used a very large data set (10 million points) and a large test set (0.55% training data and 99.45% testing data). Kolb *et al* [102] also used single hidden layer NNs to learn the exchange-correlation energies of NH₃ as functional of the density, using B3LYP and CCSD calculations as reference, with near-chemical accuracy. KKR was used by Ji and Jung [103] to learn PBE and LDA densities and exchange-correlation potentials of small molecules as a function of a representation of the molecule based on spherical averages of modified pseudopotentials at grid points. mHa accuracy was obtainable on individual molecules but worsened to dozens mHa when used on other molecules.

Nagai *et al* [104] also used a (multi-layer) NN to learn a non-local XC functional. Atomization and total energies of small molecules as good as those with popular hybrid and meta-GGA (generalized gradient approximation) functionals were obtained as well as transferability beyond the training set. The NN was trained using Monte-Carlo updates of the weights alternated with self-consistent KS-DFT calculations, rather than by back-propagation.

ML is also useful to improve on existing XC functionals. For example, Li *et al* [105] used a NN to optimize the range-separation parameter of the LC-BLYP functional. A single hidden layer NN depended on descriptors including number of atoms, number of electrons, the ZPE (zero-point energy) and the dipole moment as well as descriptors of pairwise and self-interaction. A balanced accuracy over reaction energies and barriers, ionization potentials and electronic affinities of a set of molecules was achieved.

Bayesian optimization was used to improve exchange correlation functionals. Fritz *et al* [106] used it to optimize a GGA vdW functional for calculations on water, based on a pool of 15 GGA exchange functionals. Vargas-Hernandez was able to slightly improve molecular atomization energies and bond lengths by

optimizing the free parameters of hybrid-exchange-correlation functionals with Bayesian optimization [107]. He also made the method identify the most appropriate exchange-correlation functional.

Nudejima *et al* [108] used a two-hidden layer neural network with reflected linear neurons to learn the correlation energy *density* of the CCSD(T) reference. The descriptors included the electron density, its gradient, the kinetic energy density as well as the Hartree–Fock exchange energy density and electron density based on the fractional occupation number of molecular orbitals. The model was able to achieve highly accurate fits ($R^2 > 0.999$) to the correlation energy density and could compete with several popular functionals for reaction energies between small molecules.

4.2. Kinetic energy functionals

Widespread use of OFDFT in applications beyond light metals is held back by the absence of sufficiently accurate KEFs. The high stakes of the KEF issue are well known [109, 110] and will not be repeated here. Recent years witnessed the appearance of several works using ML for KEF construction for systems where previous approximations were not good enough for use in applications (i.e. for practically all types of matter beyond light metals). Typically, the non-interacting (KS) KEF is the target, and one attempts to learn the KS KED (kinetic energy density)

$$\tau_{KS}[\rho(\mathbf{r})] = -\frac{1}{2} \sum_{i=1}^{N_{el}} \phi_i(\mathbf{r}) \Delta \phi_i(\mathbf{r}) \quad (7)$$

or the positive-definite version

$$\tau_{KS}^+[\rho(\mathbf{r})] = \frac{1}{2} \sum_{i=1}^{N_{el}} |\nabla \phi_i(\mathbf{r})|^2 \quad (8)$$

where the sum is over occupied orbitals, as a functional of a number of density-dependent variables (the density and its various derivatives and powers). Often, rather than aiming to reproduce the τ of equation (7) or (8), one fits the enhancement factor (or enhancement functional) $F = \tau^+ / \tau_{TF/vW}$, where $\tau_{TF/vW}$ is the Thomas-Fermi or von Weizsacker KED or their hybrid. Other target functions are possible such as $G = (\tau - \tau_{vW}) / \tau_{TF}$ [111].

A significant fraction of work on ML for KEFs was done on model and one-dimensional systems [112]. In their seminal works, Snyder *et al* [113, 114] used KRR to learn the energy of noninteracting ‘electrons’ in 1D as a function of the density, using model potentials—sums of three Gaussian-like wells [113] and two softened Coulombic potentials representing a 1D ‘diatomic molecule’ [114]. They obtained correct dissociation behavior of the 1D models of H_2 , He_2 , Li_2 , Be_2 , and LiH . Burke and co-workers also used KRR to learn the universal energy functional (subsuming the kinetic energy) $F[\rho] = \min_{\Psi \rightarrow \rho} \langle \Psi | \hat{T} + \hat{V}_{ee} | \Psi \rangle$, where \hat{T} is the KEO and \hat{V}_{ee} is electron repulsion, for a 1D system modeling chains of hydrogen atoms [115]. KRR worked on a representation of the density in a basis set specially constructed for this purpose. On these low-dimensional model systems, an equivalent of chemical accuracy was obtained.

Ultimately, the methods have to be validated on real-life systems, which has recently been done. The same paper of Kolb *et al* [102] mentioned above also presented an NN fit of the kinetic energy of NH_3 . They were not able to achieve chemical accuracy but were able to improve by orders of magnitude over the TF-vW model (a linear combination of the Thomas-Fermi and von Weizsacker functionals). They fitted as a function of the density and did not explore the choice of density-dependent variables, which appears to be critical to achieving chemical accuracy (*vide infra*).

Yao and Parkhill [116] used a convolutional NN to fit the enhancement factor F for a TF-vW hybrid. They explored density and scaled (dimensionless) modulus of the gradient of the density as NN inputs. Potential energy minima along bond coordinates were obtained; however, the potential energy curves were jagged and the fit showed a significantly higher test set error than the fitting set error (overfitting).

Golub and Manzhos used NNs to fit KS KED of metallic as well as (solid state and molecular) covalently bound systems [117]. They used terms of the 4th order gradient expansion as five density-dependent variables as NN inputs. A single hidden layer NN with few (about a dozen) neurons was able to closely reproduce the KS KED without overfitting. However, deep NNs were required to reproduce the KS KED simultaneously for several materials, which is a pre-requisite for a transferrable KEF. We note that limited transferability is sufficient for use in applications. Indeed, in applications of KS DFT a large number of XC functionals were proposed and are used in applications, which are advantageous for the modeling of specific classes of materials or phenomena.

Nakai group also produced accurate NN fits to KS KEDs [118]. They used electron densities and their gradients up to the third order as descriptors and fitted the enhancement factor. A three hidden layer NN

with 30 neurons per layer was used. In a subsequent work [119] Nakai group reproduced smooth potential energy curves of a number of small molecules, which is a major achievement. It was achieved with the introduction of new descriptors, namely, the distances between grid points and centers of nuclei. A five hidden layer NN with 50 neurons per layer was used.

Variables selection is an important issue in ML KEFs. While Yao and Parkhill used scaled gradients that satisfy exact conditions, Nakai and co-workers did not scale the density-dependent variables. Our own tests in the lead-up to [117] showed much worse numerics with scaling. In the work of Golub and Manzhos, as terms of formal gradient expansion were used as variables, the conditions would be satisfied in the limit of a linear fit. Non-linearity of the NN fit was mild (most of the KED variance could be captured by a linear fit); as a result, the exact conditions were, if one can say so, 'nearly satisfied'. Burke and co-workers showed on one-dimensional synthetic problems that the use of scaled variables may or may not improve machine-learned KEFs [120]; they used the KRR method. It appears that using non-scaled variables is a useful line of exploration which should not be ignored, as exact uniform density scaling does not happen in Nature.

Rather than using multiple density-dependent variables (powers and derivatives), Brockherde *et al* [121] used a representation of the density in a specially constructed basis, and this representation was fed to KRR. This is an effective way to obviate the need for higher-order powers and derivatives of the density which may be difficult to converge. They were able to machine-learn density—potential and density—energy mapping which was accurate enough to model several small molecules and to perform molecular dynamics simulations on malonaldehyde capturing the intramolecular proton transfer process. This approach was recently used to machine-learn density functionals from coupled-cluster energies based only on DFT densities to chemical accuracy, for water and ethanol [122]. The authors showed that one can reduce the computational effort by learning only the correction to a standard DFT calculation.

ML has also been used to learn the relation between the potential and the density in DFT: Wang *et al* used a RBFNN to learn such mappings along one-dimensional cuts through space and suggested that this approach can be used to deduce the underlying SE [123].

4.2.1. Data distribution.

A critical issue in machine learning of KEFs is data sampling in the space of density dependent variables. Golub and Manzhos [117] emphasized the extremely uneven KS KED data distribution. They were able to achieve accurate NN fits to the KS KED without overfitting by using a combination of partial histogram equalization and (denser) sampling on two-dimensional surfaces (in the three-dimensional Cartesian space). We also observed a much smoother distribution of the function G (than of τ) which might be useful in future ML work on KEFs. This suggests that data pre-processing may be key to achieving good ML functionals. In a subsequent work [124], Golub and Manzhos also showed that minima on the potential energy surface could be found even with KEDFs trained on single-geometry data. This suggests that it may be possible to sample the space of density-dependent variables without sampling the configuration space. This issue requires further exploration. When searching for suitable density-dependent variables, we also observed signs of multi-valuedness when using too few variables. The space of density-dependent variables should be of sufficient dimensionality for accurate KED machine learning.

4.3. ML as booster to lower-level methods to approximate higher level methods

Balabin and Lomakina [125] used NNs to estimate the DFT energy with a large basis set using lower-level energy values and molecular descriptors. They estimated energies with 6-311G(3df,3pd) and 6-311G(2df,2pd) basis sets and different functionals from energies with smaller bases and molecular descriptors (atomic composition, the gap, the polarizability, and dipole and quadrupole moments). An error of the order of $0.5 \text{ kcal mol}^{-1}$ was achieved. They found that using energies obtained with several lower-level methods is advantageous. A single hidden layer NN with several neurons was used, and it was found the NN structure did not significantly depend on the functionals or basis sets used. The NN approach was found to outperform multiple linear regression (MLR).

NNs were also used to boost the quality of optical absorption calculations. Li *et al* [126] trained a single hidden layer NN (which a GA was used to pre-train) to reduce the error of the calculated absorption energies of 150 organic molecules from 0.47 to 0.22 eV for the non-expensive TDDFT/B3LYP/6-31G(d) calculation. NN inputs included the TDDFT absorption energies, oscillator strengths and molecular descriptors including the number of double bonds, the number of aromatic rings, the number of electrons, and the HOMO-LUMO gap. The same group earlier used the approach to correct energies of formation and compared NN with MLR; similar to the work of Balabin and Lomakina, NN was found to be more powerful [127]. NNs (single hidden layer) were also used to improve heats of formation computed with various DFT and HF setups by Duan *et al* [128]. Ramakrishnan *et al* [129] used KRR to learn deviations of reference

second-order approximate coupled-cluster (CC2) singles and doubles spectra from TDDFT on a large (up to 20 000) set of small organic molecules, and were able to achieve corrected excitation energies within 0.1 eV of the reference values for the test set.

Chen *et al* [130] used multi-layer NNs to boost their energy-based fragment method which was used to compute excited states of large systems. A deep learning neural network was used for training the ML models for monomer energies and dimer and trimer interaction energies in the energy decomposition for photochemically inert surroundings of the chromophore.

Ramakrishnan *et al* [131] used kernel ridge regression to learn corrections to lower-level methods (from semi-empirical to DFT to wavefunction methods) to approximate energies with highly correlated methods. Machine-learned quantities included enthalpies, free energies, entropies, and electron correlation energies, which were learned based on a large set of organic molecules. The correction to a property was expressed as $\sum_i \alpha_i k(\mathbf{r}_b, \mathbf{R}_i)$ with $k(\mathbf{r}_b, \mathbf{R}_i) = \exp(-|\mathbf{R}_i - \mathbf{R}_b|/\sigma)$, where α_i are regression coefficients, σ is the kernel width (a hyper-parameter) and $|\mathbf{R}_i - \mathbf{R}_b|$ is a measure of similarity in the vectors of descriptors of the query molecule \mathbf{R}_b and molecules from the training set \mathbf{R}_i . For this measure, the Manhattan norm between Coulomb matrix representation [34] was used. In this way, staple methods (such as DFT/B3LYP) could be corrected to essentially G4MP2 or CCSD quality. This work also applied the approach to improving semi-empirical methods like PM7 or OM2 to DFT accuracy and for predicting correlation energy.

Lentz and Kolpak [132] used NNs to predict band gaps obtained with a hybrid functional (HSE) based on electron density obtained with the PBE functional. An RSME of the gap of about 0.2 eV was achieved for a range of solid semiconductors, which was also shown to be much better than linear regression between the PBE and HSE band gaps. They found that a single hidden layer was sufficient and outperformed more complex NN architectures.

ML has been used to improve *a posteriori* models of dispersion interactions. Gao *et al* [133] used NNs to fit dispersion corrections to DFT calculations with various functional and relatively small basis sets using CCSD(T)/CBS (complete basis set) as benchmark. Root mean square errors of the DFT calculations relative to the benchmark were improved by at least 70% with this approach. Proppe *et al* [134] used GPR to refit the errors of the PBE with D3(BJ) dispersion correction scheme using CCSD(T) as a reference, on a set of about 1250 molecular dimers.

Dick and Fernandez-Serra [135, 136] and Bogojeski *et al* [122] used NNs to learn corrections to energies (of gas phase and liquid water) obtained with a lower-level method, based on DFT densities. Rather than using directly the density as input, they developed atom-centered density-based descriptors and used Behler-type NNs (i.e. a sum of atomic contributions). Dick and Fernandez-Serra [136] used a delta-learning model (i.e. functional is built as a correction to an existing non-ML functional) to learn an XC functional as a function not only of the density but also of atomic types and positions. They also represented the density in an atom-centered basis and used a Behler-Parrinello type neural network. Accuracy of better than 10 meV vs CCSD(T) of dissociation curves of several small molecules was obtained.

4.4. Pseudopotentials

Most *ab initio* materials simulations, especially plane-wave based calculations, are facilitated by the use of pseudopotentials (PP). Highly accurate PPs used in KS-DFT are non-local. OFDFT must use local PPs, which are in general less accurate. While many PP libraries exist for KS-DFT software covering most of the periodic table, this is not the case for OFDFT. While there is general awareness of the importance of KEFs for OFDFT, the issue of PPs has been underappreciated, although good approximations to the KEF are highly unlikely in the absence of good local pseudopotentials which would permit KEF approximations to operate on smoother density. Availability of sufficiently accurate local PPs would enable large-scale simulations of many practically relevant materials as well as facilitate KEF development.

A local PP is a simple-shaped univariate function, but the calculations are very sensitive to fine details of its shape. ML based approaches to building PPs for OFDFT are very promising as they in principle are capable to find the best possible shape of the PP, especially if non-parametric approaches are used that do not impose a functional form. This is a different type of problem than in most ML applications where ML operates in multidimensional spaces. As local PPs have a well-defined asymptotic, only the non-asymptotic region around the nucleus is of interest. Legrain and Manzhos produced highly accurate PPs for Li, Na, and Mg [137] by using a parameterized functional form and tuning the parameters with a GA to reproduce structural and energetic parameters. This approach allowed to palliate in a semi-empirical way deficiencies of the OFDFT approximation (KEF), e.g. the cohesive energy could be reproduced contrary to the PP obtained with a density-inversion procedure [138]. Del Rio *et al* [139] used an evolutionary algorithm to tune Gaussian-like correction terms to the local pseudopotential to reproduce properties of liquid and solid

lithium and solid gallium with OFDFT. ML might have most success in designing non-parametric PPs but those are still wanting.

4.5. Density functional tight binding parameterization

Machine learning has been used to construct pairwise repulsive potentials of the self-consistent charge (SCC-) DFTB method [140–142]. These potentials are critical for the accuracy of the method and also offer opportunities to tune the DFTB setup to perform well for systems which were not included in the DFTB parameterization [143, 144]. Knaup *et al* [145] used a genetic algorithm to produce an accurate pair potential for N-H, whereby GA was used to tune the nodes of the spline representation of the potential. Panosetti *et al* [146] used GPR to build DFTB repulsive potentials for the modeling of Li intercalation in graphite. They achieved an accuracy similar to that of DFT for inter-layer distances and Li diffusion barriers. In Kranz *et al* [147] machine-learned repulsive potentials were produced for pairs of O, N, C, H, and F atoms by training potential parameters on energies and forces of equilibrium and non-equilibrium structures of 2100 molecules and testing on about 130 000 organic molecules. The authors started from an existing DFTB parameterization and tuned polynomial corrections to the repulsive potential. The polynomial parameters were tuned as a function of bond descriptors to account for chemical environment, to reproduce atomization energies and forces for a set of equilibrium and perturbed molecular geometries. While the fit was a traditional least-squares fit, bond clustering done with the mean-shift algorithm [148, 149] is a ML approach.

Witek and co-workers used particle swarm optimization (PSO) to tune a complete set of DFTB parameters including atom parameters (orbital energies, orbital and density confining parameters, and Hubbard parameters and their derivatives) and atom pair parameters (the number and the location of the spline knots defining, and constraints on, the repulsive potential) [150]. PSO was used due to the large resulting dimensionality of the parameter space. The parameters were fitted to reproduce structures, formation energies, bandstructures, and vibrational properties of target systems which included different phases of silicon and small molecules. They achieved accuracy for these quantities typical of high-quality DFT simulations. The electronic part of the parameterization done by Panosetti *et al* [146] also used PSO.

ML has been used for improvement of DFTB performance achieved by learning the difference in energy between DFTB and higher level methods [151]. A recent example is a work by Irle and co-workers using NN to learn a correction to DFTB (vs. DFT) for a large number of geometries of glycine [152]. A Behler-type NN modeling atom chemical environment was used [21, 45, 153, 154]. The resulting quality of the potential energy surface was similar to that of a highly accurate DFT approach.

ML can be used to parameterize semi-empirical Hamiltonians beyond the two-center ansatz of the DFTB approximation. Hegde and Bowen [155] used kernel ridge regression to build transferable prediction of DFT Hamiltonians for different environments of a Cu atom in an fcc crystal structure with near-DFT accuracy, and Dral *et al* [156] used KRR to re-optimize the parameters of the semi-empirical OM method.

5. Discussion and conclusion

Machine learning has been used to solve the Schrödinger equation or to improve the quality of lower-level solutions. When solving the Schrödinger equation, machine learning has been used to represent the wavefunction (e.g. with a neural network) in a flexible way which can be viewed (especially in the case of a single hidden layer NN) as an expansion over a tunable basis or as grid points directly entering ML-based optimization (e.g. with a genetic algorithm). ML has also been used to tune parameters entering the solution with a given method (e.g. to reweight MP2 energy components) or to fit a correction to a lower-level method. Representing the wavefunction as values on a grid is enticing as it allows a truly non-parametric representation (no functional form is imposed); however, this approach, mostly tested on simple low-dimensional problems faces in-principle difficulties in high-dimensional spaces encountered in both the electronic and vibrational problems. On the other hand, direct (generally non-uniform) grid representation may have potential for practically important low-dimensional problems such as the Kohn-Sham equation.

The use of Gaussian process regression for wavefunction representation remains largely unexplored. GPR has the potential to combine the advantages of the truly non-parametric grid representation with low-density grids. GPR is a hyper-parametric representation: while there exists an analytic expression for the function between the grid points, the functional form is only indirectly imposed via the choice of kernel type and parameters.

Several of the works considered above applied the same ML techniques to electronic and vibrational problems [52, 55]. There are similarities and differences. Both problems are linear ODEs with potentials that have wells that support bound states. In the vibrational case, however, the potential minimum is smooth while in the electronic problem it is singular (unless pseudopotentials are used). The potential in the electronic case is slower approaching the long-range asymptotic (e.g. as r^{-1}) than interatomic potentials. The

density of lowest-energy states is much more even in the vibrational problems. There is nevertheless a degree of portability of ideas between the two fields.

While ML has been used directly for wavefunction representation in both electronic and vibrational problems, ideas such as ML-based corrections or terms reweighting (of a lower-accuracy method) have been more explored in the electronic world; they are expected to be useful also for vibrational problems. As of today, spectroscopically accurate (accuracy better than 1 cm^{-1}) ML solutions for multiple (dozens or hundreds) levels of the vibrational SE for real polyatomic molecules are still wanting, and these ideas can help achieve those. Variational approaches easily achieve errors of $\ll 1\text{ cm}^{-1}$ for four- or five-atomic molecules [157]. These, however, do not take into account the error introduced when building the potential energy surface, which can be substantial. In our experience, we found that it was difficult to achieve a spectrum error much smaller than 1 cm^{-1} with ML (NN representation of the wavefunction), but ML allows solving the SE directly from *ab initio* data. It is expected that with sufficient data as accurate solutions of the SE as permitted by the dataset can be obtained.

It is only recently that milli-Hartree accuracy was achieved when solving the electronic Schrödinger equation for real life systems with ML [84]. This was achieved by combining a ML approach (NN) with a traditional basis set rather than by a straight-up application of a ML method to represent the wavefunction. This is a viable way forward to applications. Nevertheless, direct ML wavefunction representations with chemical accuracy are worth pursuing. While variational approaches often achieve μHa accuracy, their steep scaling with the number of degrees of freedom is notorious. For electronic as well as for vibrational problems, this scaling is in particular due to the use of direct-product type bases. ML allows representing the solution in a non-direct product form and effectively in an optimized basis (e.g. with NNs) and in this sense obviates exponential scaling of the cost. Variational approaches require large quadrature grids to achieve high accuracy. These also can be obviated by the use of ML-assisted representations. For example, the use of a single hidden layer NN (either ridge or radial basis function) to represent the wavefunction is equivalent to a traditional basis expansion but with a non-direct-product, tunable basis. This allows achieving quality solutions with very small bases (compared to direct product bases). In turn, the smallness of the basis allows using fewer quadrature or collocation points [59, 69]. This not only reduces CPU and memory cost of the calculation but also the cost of obtaining the underlying potential data, which could be quite expensive, e.g. when solving the nuclear Schrödinger equation for solid state, interfacial and nanoparticle systems where sampling the potential with *ab initio* calculations is costly. Rectangular collocation with radial basis functions already showed its advantages over variational approaches that use direct product bases in terms of required numbers of basis functions and collocation points even without ML optimization. ML optimization of the basis and points promises significant gains.

With the works of Kranz *et al* [137] and Chou *et al* [150] which presented parameters for a wide selection of systems, ML firmly established itself as a powerful tool for accurate parameterization of density functional tight binding. This bodes well for wider use of this method enabling large-scale and therefore more realistic simulations. A particular class of systems which are amenable to DFTB modeling and for which accurate parameterizations are still absent are organic batteries [158]. The results with ML based DFTB parameterization to date suggest that ML could be efficient to achieve accurate DFTB modeling of organic battery materials. This is especially important for modeling of polymeric materials (requiring relatively large scale models to account for amorphous character) and of realistic charge-discharge dynamics [159].

There are still a few works using ML to tune basis sets. Basis tuning is a powerful approach to diminish the cost of both electronic structure and vibrational calculations. This has been well known and used to build bases with methods other than ML [61, 160]. ML-based basis set optimization promises significant pickup and should therefore be further pursued.

ML looks extremely promising in the development of orbital-free DFT. As of now, ML has been successfully used to learn density—kinetic energy, density—energy, and potential—energy mappings as well as the mapping between the potential and density. Two methods have been widely used: kernel ridge regression and NN. The NNs have the advantage of being able to work with large datasets (millions of data), while kernel based methods are in principle somewhat more accurate (with optimal kernels and hyper-parameters) but are costlier to fit and to recall. A somewhat underappreciated area is the development of local pseudopotentials for OFDFT. Here recommended future directions include machine-learned non-parametric small-core PPs. The shape of the PP is extremely sensitive and it is difficult to construct a correct functional form, which is why non-parametric approaches are desirable, while targeting semi-core PPs might alleviate the issue of intrinsically lower accuracy of local PPs.

As the kinetic energy is a significant part of the total energy, orbital-free functionals represent a stringent test of ML which it appears to be passing. The exchange-correlation problem is more forgiving, and ML has now been used to make exchange-correlation functionals beating some of the widely used functionals and approaching the accuracy of wavefunction-based methods. In both the kinetic energy functional problem

and the XC problem, machine-learning the correction to a lower-level method (dabbed Δ -learning) has been successful and can be recommended.

A significant advantage of ML is that the parameters of the wavefunctions or orbitals need not be determined from linear algebra the way basis coefficients usually are. ML is therefore well suited to solve non-linear equations [40, 77, 161, 162]. Specifically in the context of *ab initio* calculations, this has the potential to solve the Kohn-Sham equation directly without self-consistent field cycling. This direction is little explored and also deserves further studies.

Data selection is a critical issue. While in one- and low-dimensional cases, uniform sampling is feasible and can provide high accuracy, in multiple dimensions, adapted sampling distributions are necessary, e.g. versions of the inverted potential distributions when solving the Schrödinger equation or partial histogram equalization when sampling the space of density-dependent variables in DFT. There are indications that a proper sampling scheme is critical for the success of machine learning of KEFs for OFDFT [117]. When sampling in the multidimensional space of features is sparse, techniques suitable for sparse sampling which have been shown to be effective in the field of multidimensional potential energy surfaces, such as representation with sub-dimensional component functions or dimensionality reduction [24, 25, 163], could be useful. When data are abundant and sample well all relevant regions of space, single-hidden layer NNs are sufficient and are often advantageous over multi-layer NNs, as was noted by several authors cited above; this was also our experience when fitting potential energy surfaces. It is with sparse or very unevenly distributed data that multi-layer (so-called ‘deep’) neural networks are advantageous [117].

Another important issue is feature/variables selection. The use of both scaled (i.e. satisfying the exact conditions) and unscaled densities and derivatives (gradient, Laplacian and higher order derivatives) present viable lines of exploration. An alternative which may be more robust is working with a basis expansion of the density. The proposition that it may be possible to sample the space of density-dependent variables without sampling the configuration space deserves further exploration.

ML has its lot of disadvantages. Machine learning methods typically fail when recalled outside of the learning domain (extrapolation). This can be palliated by adopting representations which enforce e.g. boundary conditions (such as wavefunction decay, as described above) or via coordinate transformations [157]. The ML methods considered above are ignorant of symmetry. This can be addressed by explicit or implicit symmetrization [46, 98, 136, 157]. ML typically requires relatively costly optimization of parameters (NN) or hyper-parameters (GPR, KRR) or extensive evolutionary searches (GA-inspired schemes). Nevertheless, with the growth of computer power, these issues are becoming easier to resolve than the exponential scaling of traditional approaches. This indicates an important place of ML techniques in the computational chemistry toolbox going forward.

Acknowledgments

We thank Tucker Carrington, Emily Carter, Chuck Witt, Kieron Burke, Hiromi Nakai and Pavlo Golub for inspiration and discussions.

Data availability statement

Data sharing is not applicable to this article as no new data were created or analysed in this study.

ORCID iD

Sergei Manzhos  <https://orcid.org/0000-0001-8172-7903>

References

- [1] Jain A *et al* 2013 *APL Mater.* **1** 011002
- [2] De Guire E *et al* 2019 *J. Am. Cer. Soc.* **102** 6385–406
- [3] Ramprasad R, Batra R, Pailania G, Mannodi-Kanakithodi A and Kim C 2017 *NPG Comput. Mater.* **3** 54
- [4] Butler K T, Davies D W, Cartwright H, Isayev O and Walsh A 2018 *Nature* **559** 547–59
- [5] Schleder G R, Padilha A C M, Acosta C M, Costa M and Fazzio A 2019 *J. Phys. Mater.* **2** 032001
- [6] Shandiz M A and Gauvin R 2016 *Comp. Mater. Sci.* **117** 270–8
- [7] Bowman J M, Carrington T and Meyer H D 2009 *Mol. Phys.* **106** 2145–82
- [8] Beck M H, Jackle A, Worth G A and Meyer H D 2000 *Phys. Rep.* **324** 1–105
- [9] Carrington T 1998 *Encyclopedia of Computational Chemistry*, ed P V R Schleyer (New York: Wiley)
- [10] Martinez J P 2017 *Ann. Phys.* **529** 1600328
- [11] Møller C and Plesset M S 1934 *Phys. Rev.* **46** 618–22
- [12] Head-Gordon M, Pople J A and Frisch M J 1988 *Chem. Phys. Lett.* **153** 503–6

- [13] Bartlett R J and Purvis G D III 1978 *Int. J. Quantum Chem.* **14** 561–81
- [14] Bartlett R J and Musiał M 2007 *Rev. Mod. Phys.* **79** 291–352
- [15] Hohenberg P and Kohn W 1964 *Phys. Rev.* **136** B864–71
- [16] Kohn W and Sham L J 1965 *Phys. Rev.* **140** A1133–8
- [17] Zupan J and Gasteiger J 1991 *Anal. Chim. Acta* **248** 1–30
- [18] Sumpter B G, Getino C and Noid D W 1994 *Annu. Rev. Phys. Chem.* **45** 439–81
- [19] Manzhos S, Wang X G, Dawes R and Carrington T 2006 *J. Phys. Chem. A* **110** 5295–304
- [20] Arthrit N, Hiller B and Behler J 2012 *Phys. Status Solidi* **250** 1191–203
- [21] Behler J and Parrinello M 2007 *Phys. Rev. Lett.* **98** 146401
- [22] Behler J 2017 *Angew Chem. Int. Ed.* **56** 12828
- [23] Manzhos S and Carrington T 2006 *J. Chem. Phys.* **125** 194105
- [24] Manzhos S and Carrington T 2007 *J. Chem. Phys.* **127** 014103
- [25] Manzhos S and Carrington T 2007 *J. Chem. Phys.* **125** 084109
- [26] Morawietz T, Sharma V and Behler J 2012 *J. Chem. Phys.* **136** 064103
- [27] Jovan Jose K V, Arthrit N and Behler J 2012 *J. Chem. Phys.* **136** 194111
- [28] Kamath A, Vargas-Hernández R A, Krems R V, Carrington T and Manzhos S 2018 *J. Chem. Phys.* **148** 241702
- [29] Brewster M E, Huang M-J, Harget A and Bodor N 1992 *Tetrahedron* **8** 3463–72
- [30] de Almeida M B, Braga A P, Braga J P, Belchiorb J C and Yareda G F G 2000 *Phys. Chem. Chem. Phys.* **2** 103–7
- [31] Hu L, Wang X, Wong L and Chen G 2003 *J. Chem. Phys.* **119** 11501–7
- [32] Mills K, Spanner M and Tamblyn I 2017 *Phys. Rev. A* **96** 042113
- [33] Pavlov A V, Serdyuk J A and Ustinov A B 2019 *J. Phys. Conf. Ser.* **1236** 012050
- [34] Rupp M, Tkatchenko A, Muller K-A and von Lilienfeld O A 2012 *Phys. Rev. Lett.* **108** 058301
- [35] Chandrasekaran A, Kamal D, Batra R, Kim C, Chen L and Ramprasad R 2019 *NPJ Comput. Mater.* **5** 22
- [36] Montavon G *et al* 2013 *New J. Phys.* **15** 095003
- [37] Han J, Jentzen A and Weinan E 2018 *Proc. Natl Acad. Sci. USA* **115** 8505–10
- [38] Arsenault L-F, Lopez-Bezanilla A, von Lilienfeld O A and Millis A J 2014 *Phys. Rev. B* **90** 155136
- [39] Carleo G and Troyer M 2017 *Science* **355** 602–6
- [40] Raissi M and Karniadakis G E 2018 *J. Comput. Phys.* **357** 125–41
- [41] Grisafi A, Fabrizio A, Meyer B, Wilkins D M, Corminboeuf C and Ceriotti M 2019 *ACS Cent. Sci.* **5** 57–64
- [42] Fabrizio A, Grisafi A, Meyer B, Ceriotti M and Corminboeuf C 2019 *Chem. Sci.* **10** 9424
- [43] Jia Z-A, Yi B, Zhai R, Wu Y-C, Guo G-C and Guo G-P 2019 *Adv. Quantum Technol.* **2** 1800077
- [44] Rupp M 2015 *Int. J. Quantum Chem.* **115** 1058–73
- [45] Behler J 2015 *Int. J. Quantum Chem.* **115** 1032–50
- [46] Manzhos S, Dawes R and Carrington T 2015 *Int. J. Quantum Chem.* **115** 1012–20
- [47] Rasmussen C E and Williams C K I 2006 *Gaussian Processes for Machine Learning* (Cambridge, MA: MIT Press)
- [48] Sacks J, Schiller S B and Welch W J 1989 *Technometrics* **31** 41
- [49] Vu K, Snyder J C, Li L, Rupp M, Chen B F, Khelif T, Muller K-R and Burke K 2015 *Int. J. Quantum Chem.* **115** 1115–28
- [50] Mitchell M 1998 *An Introduction to Genetic Algorithms* (Cambridge, MA: MIT Press)
- [51] Eberhart R and Kennedy J 1995 *Proc. of the Sixth Int. Symp. on Micro Machine and Human Science* (Piscataway, NJ: IEEE) pp 39–43
- [52] Zeiri Y, Fattal E and Kosloff R 1995 *J. Chem. Phys.* **102** 1859–62
- [53] Saha R, Chaudhury P and Bhattacharyya S P 2001 *Phys. Lett. A* **291** 397–406
- [54] Makarov D E and Metiu H 2000 *J. Phys. Chem. A* **104** 8540–5
- [55] Lagaris I E, Likas A and Fotiadis D I 1997 *Comput. Phys. Commun.* **104** 1–14
- [56] Scarselli F and Tsoi A C 1998 *Neural Netw.* **11** 15–37
- [57] Hornik K, Stinchcombe M and White H 1989 *Neural Netw.* **2** 359–66
- [58] Hornik K 1991 *Neural Netw.* **4** 251–7
- [59] Manzhos S, Yamashita K and Carrington T 2011 *Chem. Phys. Lett.* **511** 434–9
- [60] Gorban A N 1998 *Appl. Math. Lett.* **11** 45
- [61] Nakanishi H and Sugawara M 2000 *Chem. Phys. Lett.* **327** 429–38
- [62] Sugawara M 2001 *Comput. Phys. Commun.* **140** 366–80
- [63] Manzhos S and Carrington T 2009 *Can. J. Chem.* **87** 864–71
- [64] Liao Y, Fang S-C and Nuttle H L W 2003 *Neural Netw.* **16** 1019
- [65] Wu W, Nan D, Long J and Ma Y 2008 *Neural Netw.* **21** 1462
- [66] Mulero-Martinez J I 2007 *Neurocomputing* **70** 1439
- [67] Boutry G, Elad M, Golub G H and Milanfar P 2005 *SIAM J. Matrix Anal. Appl.* **27** 582
- [68] Manzhos S and Carrington T 2018 *J. Chem. Phys.* **149** 204105
- [69] Ku J, Kamath A, Carrington T and Manzhos S 2019 *J. Phys. Chem. A* **123** 10631–42
- [70] Manzhos S, Yamashita K and Carrington T 2009 *Chem. Phys. Lett.* **474** 217–21
- [71] Teng P 2018 *Phys. Rev. E* **98** 033305
- [72] Meyer H 2002 *Annu. Rev. Phys. Chem.* **53** 141–72
- [73] Manzhos S and Carrington T 2016 *J. Chem. Phys.* **145** 224110
- [74] Manzhos S, Wang X and Carrington T 2018 *Chem. Phys.* **509** 139–44
- [75] Chan M, Manzhos S, Carrington T and Yamashita K 2012 *J. Chem. Theory Comput.* **8** 2053–61
- [76] Chaudhury P and Bhattacharyya S P 1998 *Chem. Phys. Lett.* **296** 51–60
- [77] Caetano C, Reis J L, Amorim J, Ruv Lemes M and Dal Pino A 2011 *Int. J. Quantum Chem.* **111** 2732–40
- [78] Saito H 2017 *J. Phys. Soc. Jpn.* **86** 093001
- [79] Choo K, Mezzacapo A and Carleo G 2019 [arXiv:1909.12852v1](https://arxiv.org/abs/1909.12852v1)
- [80] Hermann J, Schätzle Z and Noé F 2019 [arXiv:1909.08423v2](https://arxiv.org/abs/1909.08423v2)
- [81] Coe J P 2018 *J. Chem. Theory Comput.* **14** 5739–49
- [82] Coe J P 2019 *J. Chem. Theory Comput.* **15** 6179–89
- [83] Han J, Zhang L and Weinan E 2019 *J. Comput. Phys.* **399** 108929
- [84] Schütt K T, Gastegger M, Tkatchenko A, Müller K-R and Maurer R J 2019 *Nature. Commun.* **10** 1–10
- [85] Schütt K T, Sauceda H E, Kindermans P-J, Tkatchenko A and Muller K-R 2018 *J. Chem. Phys.* **148** 241722

- [86] Schütt O and VandeVondele J 2018 *J. Chem. Theory Comput.* **14** 4168–75
- [87] Malshe M, Pukrittayakamee A, Raff L M, Hagan M, Bukkapatnam S and Komanduri R 2009 *J. Chem. Phys.* **131** 124127
- [88] Silva G M, Acioli P H and Pedroza A C 1997 *J. Comput. Chem.* **18** 1407–14
- [89] McGibbon R T, Taube A G, Donchev A G, Siva K, Hernandez F, Hargus C, Law K-H, Klepeis J L and Shaw D E 2017 *J. Chem. Phys.* **147** 161725
- [90] Margraf J T and Reuter K 2018 *J. Phys. Chem. A* **122** 6343–8
- [91] Welborn M, Cheng L and Miller T F III 2018 *J. Chem. Theory Comput.* **14** 4772–9
- [92] Cheng L, Welborn M, Christensen A S and Miller T F III 2019 *J. Chem. Phys.* **150** 131103
- [93] Townsend J and Vogiatzis K D 2019 *J. Phys. Chem. Lett.* **10** 4129–35
- [94] Friedman J H, Bentely J and Finkel R A 1977 *ACM Trans. Math. Softw.* **3** 209–26
- [95] Nagai R, Akashi R, Sasaki S and Tsuneyuki S 2018 *J. Chem. Phys.* **148** 241737
- [96] Ma J, Zhang P, Tan Y, Ghosh A W and Chern G W 2019 *Phys. Rev. B* **99** 085118
- [97] Custodio C A, Filletti E R and França V V 2019 *Sci. Rep.* **9** 1886
- [98] Schmidt J, Benavides-Riveros C L and Marques M A L 2019 *J. Phys. Chem. Lett.* **10** 6425–31
- [99] Ryczko K, Strubbe D A and Tamblyn I 2019 *Phys. Rev. A* **100** 022512
- [100] Zhou Y, Wu J, Chen S and Chen G H 2019 *J. Phys. Chem. Lett.* **10** 7264–9
- [101] Lei X and Medford A J 2019 *Phys. Rev. Mater.* **3** 063801
- [102] Kolb B, Lentz L C and Kolpak A M 2017 *Sci. Rep.* **7** 1192
- [103] Ji H and Jung Y 2018 *J. Chem. Phys.* **148** 241742
- [104] Nagai R, Akashi R and Sugino O 2019 [arXiv:1903.00238v5](https://arxiv.org/abs/1903.00238v5)
- [105] Liu Q, Wang J C, Du P L, Hu L H, Zheng X and Chen G H 2017 *J. Phys. Chem. A* **121** 7273–81
- [106] Fritz M, Fernández-Serra M and Soler J M 2016 *J. Chem. Phys.* **144** 224101
- [107] Vargas-Hernandez R 2019 [arXiv:1903.10678v3](https://arxiv.org/abs/1903.10678v3)
- [108] Nuddejima T, Ikabata Y, Seino J, Yoshikawa T and Nakai H 2019 *J. Chem. Phys.* **151** 024104
- [109] Witt W C, Del Rio B G, Dieterich J M and Carter E A 2018 *J. Mater. Res.* **33** 777–95
- [110] Karasiev V V and Trickey S B 2012 *Comput. Phys. Commun.* **183** 2519–27
- [111] Xia J and Carter E A 2015 *Phys. Rev. B* **91** 045124
- [112] Li L, Snyder J C, Pelaschier I M, Huang J, Niranjan U-N, Duncan P, Rupp M, Muller K-A and Burke K 2016 *Int. J. Quantum Chem.* **116** 819–33
- [113] Snyder J C, Rupp M, Hansen K, Muller K-R and Burke K 2012 *Phys. Rev. Lett.* **108** 253002
- [114] Snyder J C, Rupp M, Hansen K, Blooston L, Müller K-R and Burke K 2013 *J. Chem. Phys.* **139** 224104
- [115] Li L, Baker T E, White S R and Burke K 2016 *Phys. Rev. B* **94** 245129
- [116] Yao K and Parkhill J 2016 *J. Chem. Theory Comput.* **12** 1139–47
- [117] Golub P and Manzhos S 2019 *Phys. Chem. Chem. Phys.* **21** 378–95
- [118] Seino J, Kageyama R, Fujinami M, Ikabata Y and Nakai H 2018 *J. Chem. Phys.* **148** 241705
- [119] Seino J, Kageyama R, Fujinami M, Ikabata Y and Nakai H 2019 *Chem. Phys. Lett.* **734** 136732
- [120] Hollingsworth J, Li L, Baker T E and Burke K 2018 *J. Chem. Phys.* **148** 241743
- [121] Brockherde F, Vogt L, Li L, Tuckerman M E, Burke K and Müller K-R 2019 *Nature. Commun.* **8** 872
- [122] Bogojewski M, Vogt-Maranto L, Tuckerman M E, Mueller K-R and Burke K 2019 ChemRxiv. Preprint (<http://doi.org/10.26434/chemrxiv.8079917.v1>)
- [123] Wang C, Zhai H and You Y-Z 2019 *Sci. Bull.* **64** 1228–33
- [124] Golub P and Manzhos S 2019 *Comput. Phys. Commun.* in revision
- [125] Balabin R M and Lomakina E I 2009 *J. Chem. Phys.* **131** 074104
- [126] Li H, Shi L L, Zang M, Su Z, Wang X, Hu L and Chen G 2007 *J. Chem. Phys.* **126** 144101
- [127] Wang X, Wong L, Hu L, Chan C, Su Z and Chen G 2004 *J. Phys. Chem. A* **108** 8514–22
- [128] Duan X-M, Li Z-H, Song G-L, Wang W-N, Chen G-H and Fan K-N 2005 *Chem. Phys. Lett.* **410** 125–30
- [129] Ramakrishnan R, Hartmann M, Tapavicza E and von Lilienfeld O A 2015 *J. Chem. Phys.* **143** 084111
- [130] Chen W-K, Fang W-H and Cui G 2019 *J. Phys. Chem. Lett.* **10** 7836–41
- [131] Ramakrishnan R, Dral P O, Rupp M and von Lilienfeld O A 2015 *J. Chem. Theory Comput.* **11** 2087–96
- [132] Lentz L C and Kolpak A M 2020 *J. Phys.: Condens. Matter* **32** 155901
- [133] Gao T, Li H, Li W, Li L, Fang C, Li H, Hu L, Lu Y and Su Z-M 2016 *J. Cheminf.* **8** 24
- [134] Proppe J, Gugler S and Reiher M 2019 *J. Chem. Theory Comput.* **15** 6046–60
- [135] Dick S and Fernandez-Serra M 2019 *J. Chem. Phys.* **151** 144102
- [136] Dick S and Fernandez-Serra M 2019 ChemRxiv. Preprint (<http://doi.org/10.26434/chemrxiv.9947312.v2>)
- [137] Legrain F and Manzhos F 2015 *Chem. Phys. Lett.* **622** 99–103
- [138] Huang C and Carter E A 2008 *Phys. Chem. Chem. Phys.* **10** 7109–20
- [139] Del Rio B, Dieterich J M and Carter E A 2017 *J. Chem. Theory Comput.* **13** 3684–95
- [140] Elstner M, Porezag D, Jungnickel G, Elsner J, Haugk M, Frauenheim T, Suhai S and Seifert G 1998 *Phys. Rev. B* **58** 7260–8
- [141] Yang Y, Yu H, York D, Cui Q and Elstner M 2007 *J. Phys. Chem. A* **111** 10861–73
- [142] Gaus M, Cui Q and Elstner M 2011 *J. Chem. Theory Comput.* **7** 931–48
- [143] Gaus M, Chou C-P, Witek H and Elstner M 2009 *J. Phys. Chem. A* **113** 11866–81
- [144] Lourenço M P, da Silva M C, Oliveira A F, Campos Quintão M and Duarte H A 2016 *Theor. Chem. Acc.* **135** 250
- [145] Knaup J M, Hourahine B and Frauenheim T 2007 *J. Phys. Chem. A* **111** 5637–41
- [146] Panosetti C, Annies S B, Grosu C, Seidlmayer S and Scheure C 2019 [arXiv:1904.13351](https://arxiv.org/abs/1904.13351)
- [147] Kranz J, Kubillus M, Ramakrishnan R, von Lilienfeld O A and Elstner M 2018 *J. Chem. Theory Comput.* **14** 2341–52
- [148] Fukunaga K and Hostetler L 1975 *IEEE Trans. Inf. Theory* **21** 32–40
- [149] Cheng Y 1995 *IEEE Trans. Pattern Anal.* **17** 790–9
- [150] Chou C-P, Nishimura Y, Fan C-C, Mazur G, Irle S and Witek H A 2016 *J. Chem. Theory Comput.* **12** 53–64
- [151] Li H, Collins C, Tanha M, Gordon G J and Yaron D J 2018 *J. Chem. Theory Comput.* **14** 5764–76
- [152] Zhu J, Vuong V Q, Sumpter B G and Irle S 2019 *MRS Commun.* **9** 867–73
- [153] Behler J 2011 *J. Chem. Phys.* **134** 074106
- [154] Behler J 2014 *J. Phys.: Condens. Matter* **26** 183001
- [155] Hegde G and Bowen R C 2017 *Sci. Rep.* **7** 42669

- [156] Dral P, von Lilienfeld O A and Thiel W 2015 *J. Chem. Theory Comput.* **11** 2120–5
- [157] Majumder M, Hegger S E, Dawes R, Manzhos S, Wang X-G, Carrington T, Li J and Guo H 2015 *Mol. Phys.* **113** 1823–33
- [158] Manzhos S 2019 *Curr. Opin. Green Sustain. Chem.* **17** 8–14
- [159] Chen Y, Lueder Y, Ng M F, Sullivan M and Manzhos S 2018 *Phys. Chem. Chem. Phys.* **20** 232–7
- [160] Nakata A, Bowler D R and Miyazaki T 2015 *Phys. Chem. Chem. Phys.* **17** 31427–33
- [161] Monterola C and Saloma C 2001 *Opt. Express* **9** 72–84
- [162] Monterola C and Saloma C 2003 *Opt. Commun.* **222** 331–9
- [163] Manzhos S, Yamashita K and Carrington T 2009 *Comput. Phys. Commun.* **180** 2002–12