

Can exact conditions improve machine-learned density functionals?

Jacob Hollingsworth¹, Li Li (李力)¹, Thomas E. Baker^{1,2}, Kieron Burke^{1,3}

¹ *Department of Physics and Astronomy, University of California, Irvine, CA 92697, USA*

² *Institut quantique & Département de physique,
Université de Sherbrooke, Québec J1K 2R1 Canada and*

³ *Department of Chemistry, University of California, Irvine, CA 92697, USA*

(Dated: Monday 12th February, 2018)

Historical methods of functional development in density functional theory have been largely guided by analytic conditions that constrain the exact functional one is trying to approximate. Recently, machine-learned functionals have been created by interpolating the results from a small number of exactly solved systems to unsolved systems that are similar in nature. For simple cases, using an exact condition, we find improvements in the learning curves of machine-learned approximations. We also find that the significance of the improvement depends on the nature of the interpolation manifold of the machine-learned functional.

1. INTRODUCTION

Density functional theory (DFT) is a popular approach to electronic structure calculations in both the material sciences and chemistry [1, 2]. By approximating various components of the energy of a quantum system with functionals of the density, DFT bypasses the need to solve the Schrödinger equation for the fully interacting problem. The utility of DFT calculations are, in turn, critically dependent upon the quality of the approximations to the energy of a system as a functional of the density. Almost all modern calculations employ the Kohn-Sham scheme [3], in which only the exchange-correlation (XC) energy need be approximated as a functional of the spin-densities.

Within the arena of density functional approximation development, there is tension between the desire to derive approximations from general principles of quantum mechanics (e.g., PBE [4]) versus fitting to known molecular and materials data (e.g., Minnesota functionals [5]). Functionals derived from general principles tend to work well for predicting most properties of most systems [4, 6, 7], where as functionals that are entirely or partially fit to empirical systems tend to perform better for properties of those specific systems or similar ones [8].

Recently, an entirely new type of approximate functional has been developed, using algorithms from the general area of machine learning [9, 10]. Machine-learned density functionals are empirical functionals that work by extrapolating the exact values of the functional on a handful of densities (training set) to predict the value of the functional on new densities (test set). We emphasize that this empiricism is very different from that previously used in DFT. In ML, the functional is approximated in an intrinsically non-local fashion, requiring many parameters (up to one million) [9], but producing approximations with chemical accuracy on the interpolation manifold, including especially stretched bonds. Thus a ML non-interacting kinetic energy remains accurate as a bond breaks [11] (unlike semilocal counterparts) while an ML XC energy can include strong correlation, even in the thermodynamic limit [10]. (Besides finding density functionals, machine learning is more com-

monly used in chemical and material science to make accurate predictions of chemical and physical properties. [12–21])

This kind of DFT development raises a natural question: are exact conditions, which are often built in to human-based functionals, useful in the design of machine-learning approximations? By manually imposing exact conditions on machine-learned density functionals, one introduces prior knowledge to the machine-learned functionals that may foreseeably enable easier training. Among the simplest, and most powerful, exact conditions in KS DFT are those produced by coordinate scaling. For instance, the exchange energy functional scales in a simple way when the density is squeezed or stretched:

$$E_x[n_\gamma] = \gamma E_x[n], \quad n_\gamma(\mathbf{r}) = \gamma^3 n(\gamma\mathbf{r}) \quad (1)$$

where γ is a positive real number. All popular exchange functionals satisfy this condition, and it determines the local density approximation up to a constant [22].

So far, ML functionals have been designed without the use of such conditions, in order to test principles and limitations of ML construction of functionals in their most primitive form. We can now phrase our question very precisely. If we can *impose* a condition like Eq. (1) on an ML functional, i.e., construct the functional so that the condition is automatically satisfied, do we get a more accurate approximation for a given amount of data? We shall see that the answer is typically yes, but whether that gain in accuracy is significant depends on the nature of the interpolation manifold.

This paper is organized as follows. Sec. 2 reviews the formalism of KS DFT and machine learning. In this section, we also discuss recent progress in applying machine learning to DFT and standard systems which have been used to test of novel ideas in DFT. In Sec. 3, we present the theory behind our approach and motivate it with an example. In Sec. 4, we explain in detail the methods used to generate reference densities needed to use our ML formalism. In Sec. 5, we compare the performance of a traditional ML functional to one that satisfies an exact scaling condition by construction. In Sec. 6, we interpret our results and offer an explanation in terms of a simple model system. In Sec. 7, we

explain how these results may improve the applicability of machine learned functionals and discuss possible extensions of this work.

2. BACKGROUND

A. DFT

The main implementation of all modern DFT calculations is the Kohn-Sham (KS) scheme [3]. To solve an N electron system in an external potential $v(\mathbf{r})$ using the KS scheme, one considers a system of N non-interacting electrons. This system is described by a set of orbitals, $\{\phi_i\}$, that may be attained by solving the Kohn-Sham equations:

$$\left(-\frac{1}{2}\nabla^2 + v_s[n](\mathbf{r})\right)\phi_i(\mathbf{r}) = \epsilon_i\phi_i(\mathbf{r}), \quad (2)$$

where $v_s[n](\mathbf{r})$ is the Kohn-Sham potential. We have written this equation in atomic units where $\hbar = m_e = e = 1$, and will continue to use these units throughout the rest of the paper. The Kohn-Sham potential is chosen so that the density of this non-interacting system,

$$n(\mathbf{r}) = 2 \sum_{i=1}^{N/2} |\phi_i(\mathbf{r})|^2, \quad (3)$$

matches that of the fully interacting system. For simplicity, we have assumed the system is spin unpolarized. This allows one to write the energy of the fully interacting system as

$$E[n] = T_s[n] + U[n] + E_{XC}[n] + \int d^3r n(\mathbf{r})v(\mathbf{r}), \quad (4)$$

where we have defined

$$T_s[n] = - \sum_{i=1}^{N/2} \int d^3r \phi_i[n](\mathbf{r})\nabla^2\phi_i[n](\mathbf{r}) \quad (5)$$

$$U[n] = \frac{1}{2} \int d^3r \int d^3r' \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \quad (6)$$

and $E_{XC}[n]$ is defined by Eq. (4). Many successful approximations to $E_{XC}[n]$ have been constructed, enabling a self-consistent solution to Eq. (2) and the functional derivative of Eq. (4) [4, 6, 7].

Orbital-free DFT is a methodology that predates KS (in fact, goes back to the Thomas-Fermi theory [23, 24] of 1927) that removes the need to solve for the eigenfunctions of the Kohn-Sham equation. To make KS DFT orbital free, one can construct an approximation to $T_s[n]$, which removes the dependence of Eq. (4) on the Kohn-Sham orbitals. Orbital-free calculations hold the promise of by-passing the bottleneck of solving the KS equations. In the special case of one electron, or two spin-unpolarized electrons, the exact non-interacting kinetic energy functional is given by the von Weisacker form:

$$T_s^{VW}[n] = \frac{1}{8} \int_{-\infty}^{\infty} dx \left(\frac{n'(x)^2}{n(x)} \right). \quad (7)$$

Modern XC functional approximations mostly fall on Jacob's ladder [25], where an increasing number of ingredients are aimed to produce higher accuracy. The most commonly used approximations are the local density approximation (LDA), which applies the exact XC functional for the uniform electron gas to all systems. LDA is a local approximation, meaning that it depends only on the density at each point. Higher up on Jacob's ladder are generalized gradient approximations (GGA) and meta GGAs, which introduce a dependence on the gradient and laplacian, respectively, of the density. Above these are hybrid functionals, which incorporate some amount of exact exchange energy.

B. Machine-learned functionals

The machine learning approximations are of a very different kind [9, 10, 26, 27]. Machine-learned functionals do not possess a convenient closed form, and typically are not intended to accurately describe general systems. Instead, they fit a function to a dataset of densities for which the exact value of the functional has already been determined through other means, known as the training set.

Kernel ridge regression (KRR) [28] is a non-linear form of regression that uses the kernel trick. For this work, we choose a gaussian kernel function. To model a functional $T_s^{ML(\sigma,\lambda)}[n]$ using kernel ridge regression, we write

$$T_s^{ML(\sigma,\lambda)}[n] = \sum_{i=1}^m \alpha_i e^{-\frac{|n-n_i|^2}{2\sigma^2}}, \quad (8)$$

where m is the number of training data and $\{n_i\}$ is the set of densities for which $T_s[n]$ is known. The inner product must be a measure of the similarity between densities, and here is defined by

$$|n - n'|^2 = \int d^3r (n(\mathbf{r}) - n'(\mathbf{r}))^2. \quad (9)$$

The $\{\alpha_i\}$ are determined so that the loss function

$$J(\boldsymbol{\alpha}) = \sum_{i=1}^m \left(T_s^{ML(\sigma,\lambda)}[n_i] - T_s[n_i] \right)^2 + \lambda \boldsymbol{\alpha}^T \mathbf{K} \boldsymbol{\alpha} \quad (10)$$

is minimized. Here, $\boldsymbol{\alpha}$ is a vector of weights defined by $\alpha_i = \alpha_i$ and \mathbf{K} is the kernel matrix defined by

$$\mathbf{K}_{ij} = e^{-\frac{|n_i - n_j|^2}{2\sigma^2}}. \quad (11)$$

The minimization of Eq. (10) may be done analytically, yielding

$$\boldsymbol{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{T}_s, \quad (12)$$

where, \mathbf{I} is the $m \times m$ identity matrix, and \mathbf{T}_s is an $m \times 1$ vector such that $T_{s_i} = T_s[n_i]$.

In KRR, λ and σ are hyperparameters of the model. Optimal choice of hyperparameters prevents overfitting and

helps the model generalize well on out-of-sample data. They are determined by cross-validation, a process where part of the training set, known as the validation set, is hidden from the model during training. In this work, we perform leave one out cross validation, meaning that, for a training set of N_T densities, we train on a set of $N_T - 1$ densities, and the remaining density forms the validation set. We pick the pair of λ and σ that produce the lowest error on the validation set. Then, we rotate the data point being used in the validation set, eventually forming a set of N_T best values for λ and σ . We then pick the median value of each to parameterize our model, and retrain the model using the full training set.

KRR functionals are entirely non-local. Thus, they can describe effects due to non-local behavior of the exact functional that are unable to be reproduced by semilocal functionals. For example, KRR approximations to $F[n]$ are able to reproduce the dissociation curve of H_2 [10], whereas approximations to $F[n]$ that use a semilocal E_{XC} approximation are unable to reproduce this result.

C. Progress with ML-DFT

Most of the work, like that reported here, involves the non-interacting kinetic energy functional. The non-interacting kinetic energy functional, defined in Eq. (5), typically requires one to solve the Kohn-Sham equation before evaluation. However, nearly every KS DFT calculation performed solves the Kohn-Sham equations and thus can serve as data to train a machine-learned kinetic energy functional, which would enable later calculations to be orbital-free.

D. 1D model systems

In this work, we will use two model 2-electron 1D systems to test the basic idea of using exact conditions to improve machine-learned density functionals.

The first may be denoted the 1D Hooke's atom. The regular (3D) Hooke's atom is two electrons in a harmonic well, but repelling each other with a Coulomb repulsion. This has analytic solutions for an infinite discrete set of force constants, and has been a popular toy model for testing DFT ideas. The same behavior is also seen in 1D, but now the interaction is replaced by a delta repulsion (as the Coulomb interaction is too singular in 1D, and the delta repulsion produces an electron-electron cusp that exactly mimics that of 3D). Thus the Hamiltonian is

$$H = -\frac{1}{2} \left(\frac{d^2}{dx_1^2} + \frac{d^2}{dx_2^2} \right) + \frac{1}{2} \omega^2 (x_1^2 + x_2^2) + \lambda \delta(x_1 - x_2). \quad (13)$$

This Hamiltonian is separable into a center of mass and relative coordinate. The first is just a harmonic oscillator, while the second can be solved numerically [29, 30]. As well

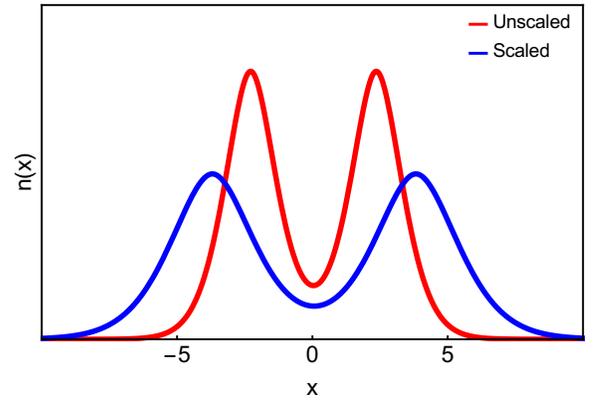


FIG. 1. (color online) A model density is shown before (red) and after scaling (blue) by a factor of $\gamma = 1/2$.

as a model for electronic structure, these systems are also important in cold-atom physics.

More recently, an entirely new 1D mimic of electronic structure has been developed, with the principal aim being to test ideas of strong correlation. These systems often consist of chains of 1D H atoms which can be accurately and efficiently solved via density-matrix renormalization group codes (DMRG) [31]. They are also designed so that standard density functional approximations, such as LDA, perform quantitatively similarly to those in 3D, yielding reasonably accurate equilibrium bond lengths and energies, but also breaking spin symmetry at the Coulson-Fisher point as a bond is stretched. For this model, we use a simple exponential for both the attraction to the nuclei and the repulsion among electrons:

$$v(x) = A \exp(-\kappa |x|), \quad (14)$$

where $A = 1.071295$, $\kappa^{-1} = 2.385345$. Just as in 3D reality, the external potential for two 'proton's separated by a distance R is given by

$$v_{ext}(x) = -v(x - R/2) - v(x + R/2), \quad (15)$$

and the interaction potential is given by

$$v_{int}(x, x') = v(x - x'), \quad (16)$$

The parameters of the exponentials have been chosen to best match the soft-Coulomb interaction potential, which is typically used in 1D calculations, as it captures certain behaviors of 3D systems [32]. We determine the densities by solving the fully interacting problem on a grid using DMRG[31].

3. THEORY

We attempt to improve machine-learning of the kinetic energy functional $T_s[n]$ of non-interacting electrons, by enforcing the exact condition:

$$T_s[n_\gamma] = \gamma^2 T_s[n], \quad (17)$$

where $n_\gamma(x)$ is defined as in Eq. (1). The effect of scaling on a density is illustrated in Fig. 1. To enforce the condition, define the square-width of a 1D density as

$$l^2[n] = \int_{-\infty}^{\infty} dx x^2 n(x). \quad (18)$$

We next define a scaled density:

$$\tilde{n}(x) = n_l(x) = l n(lx). \quad (19)$$

We can write the kinetic energy functional

$$T_s[n] = \frac{1}{l^2[n]} \tilde{T}_s[\tilde{n}]. \quad (20)$$

Thus \tilde{T}_s is a functional that only applies to densities whose square-width is 1 and any fit to that functional will by construction provide a kinetic energy functional that has the correct scaling.

To illustrate this in a trivial case, consider a single-particle in a harmonic well of mass 1 and frequency ω :

$$n_\omega(x) = \left(\frac{\omega}{\pi}\right)^{1/2} \exp(-\omega x^2), \quad (21)$$

whose exact kinetic energy is $\omega/4$. Given a variety of densities and energies for different values of ω , one could imagine learning the kinetic energy using the techniques of Sec 2. But, for each ω , $l = (2\omega)^{-1/2}$, and, independent of ω ,

$$\tilde{n}(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2) \quad (22)$$

with kinetic energy $1/8$. Insertion of these results into Eq. (20) yields the exact answer. Thus, any single data point would be sufficient to learn the entire curve.

4. CALCULATIONS

For the 1D Hooke's atom, the Hamiltonian becomes separable under the coordinate transformation $X = x_1 + x_2$, $u = \frac{x_1 - x_2}{2}$, which yields a system of equations

$$\left(-\frac{1}{4} \frac{d^2}{dX^2} + \omega^2 X^2\right) \psi(X) = E_X \psi(X) \quad (23)$$

$$\left(-\frac{d^2}{du^2} + \frac{1}{4} \omega^2 u^2 + \delta(u)\right) \varphi(u) = E_u \varphi(u) \quad (24)$$

and $\Psi(u, X) = \psi(X)\varphi(u)$, where we have set $\lambda = 1$ for all calculations [29, 30]. The first equation is solved analytically and the second equation is solved numerically. We then transform Ψ back to the coordinates x_1, x_2 . We find the density by evaluating

$$n(x) = 2 \int_{-\infty}^{\infty} dx_2 |\Psi(x, x_2)|^2 \quad (25)$$

and calculate the non-interacting kinetic energy using Eq. (7). We repeat this process for 200 values of ω linearly spaced in the interval $[.1, 5]$ to form a set of reference data. Finally, for convenience, we scale all densities so to have the same square-width:

$$\gamma_\omega = \sqrt{\frac{l^2[n]}{l^2[n_{\omega=5}]}} \quad (26)$$

so that all scaled densities satisfy $l^2[\tilde{n}] = l^2[n_{\omega=5}]$. The exact constant chosen for $l^2[\tilde{n}]$ only sets the scale of the problem. Here, this value is chosen so that the scaled and unscaled systems lie approximately on the same scale to make plotting convenient.

To create the ML functional, we first choose 50 densities from the reference data which have equally spaced ω and that span the interval to form a test set. We do not include the endpoints $\omega = .1, \omega = 5$ in this set. From the remaining densities, we randomly choose 17 densities and take the first N_T of these to serve as our training set. We then use KRR to machine-learn the non-interacting kinetic energy. We then repeat this procedure on the scaled densities using the same N_T densities as before. We then pick another random set of 17 densities from the densities that have not been included in the test set, and repeat the procedure.

For the 1D H_2 case, we perform the same procedure, but now we alter the separation between the centers, R , instead of ω . The densities were found via DMRG as described above, and the non-interacting kinetic energy was found using Eq. (7). We chose 50 test densities between $R = 3$ and $R = 8$, but do not include the end-points. We train on the same numbers of training data as before. The scale factor is chosen so that all scaled densities satisfy $l^2[\tilde{n}] = l^2[n_{R=8}]$

For both scaled and unscaled functionals, we calculate mean absolute errors in the kinetic energies of the test set for $N_T = 3..17$ (odd) to form a learning curve. We repeat this process 200 times, redrawing the 17 random densities to include in the training sets over each trial.

5. RESULTS

In Fig. 2, we plot the median learning curves for the 1D Hooke's atom with solid lines. Red indicates the standard ML approximation, while blue denotes the results with scaling included. Each of the many faint lines in the background indicate one realization of the randomly chosen training points. The exact condition has produced learning curves that are a spectacular order of magnitude better than their unscaled counterparts, for a given amount of training data!

On the other hand, Fig. 3 is far less encouraging. In this figure, we show the same result for the functionals trained on H_2 densities. The scaled and unscaled functionals perform almost identically, with at most a modest gain for imposing the exact condition. The rest of this paper is devoted to understanding why performance is so much less in this case.

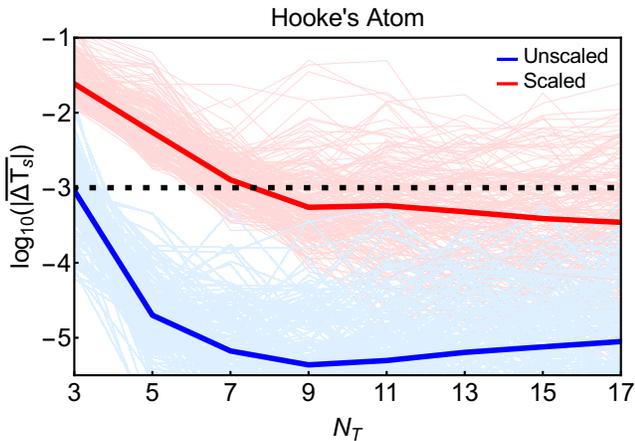


FIG. 2. (color online) The learning curves for functionals trained on scaled (blue) and unscaled (red) densities for the 1D Hooke's atom. Accuracy of 1 mH is denoted by the dashed line (black).

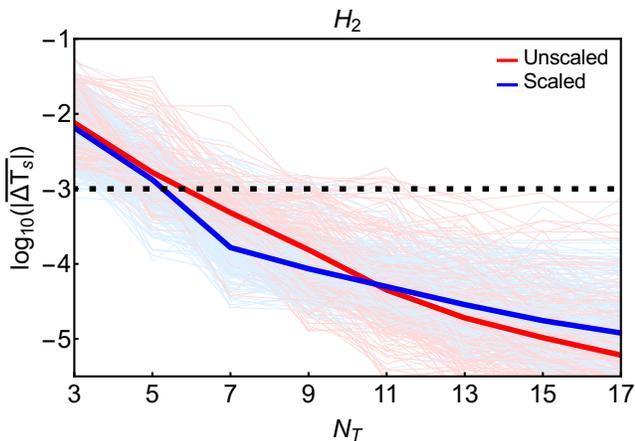


FIG. 3. (color online) Same as Fig. 2, but for H_2 densities.

Fig. 4 shows the densities of the 1D Hooke's atom in our set, both before and after scaling. It is apparent that the densities become much more similar once they have been scaled. The appearance of the double peak is a strong correlation effect, in which the two electrons are trying to avoid one another. But the decay of the scaled density at large $|x|$ is the same for all cases. To reinforce this, we plot l as a function of ω for each density, and T_s (both with and without scaling) as a function of ω . Clearly the scaled functional varies less and is easier to learn.

To quantify this, we plot in Fig. 5 the value of γ for each ω , versus $1/\sqrt{2\omega}$. The curve is close to a constant (as it would be for the non-interacting case) and only varies strongly from that as ω becomes small and the system becomes more strongly correlated. Lastly, in Fig. 6, we plot both the unscaled and scaled KS kinetic energies for the 1D Hooke's atom. If the density were not changing shape, i.e., truly non-interacting, the unscaled energy would be linear, and the scaled kinetic energy would be constant, as in our toy example.

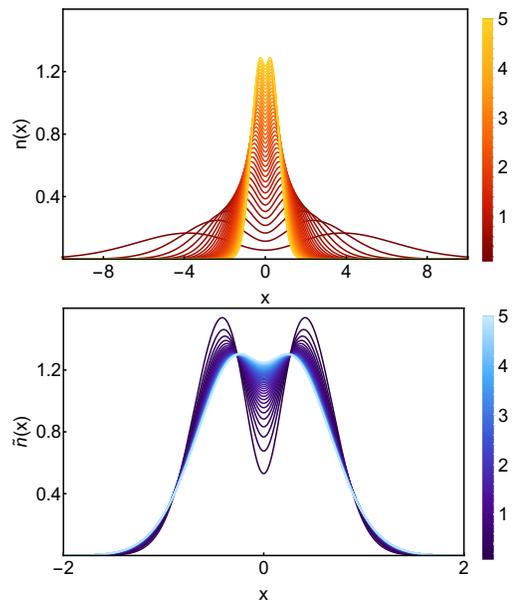


FIG. 4. (color online) The Hooke's atom densities with different ω are plotted before scaling (top) and after scaling (bottom). The color of each line indicates the value of ω . Notably, the densities appear "more similar" to each other after scaling.

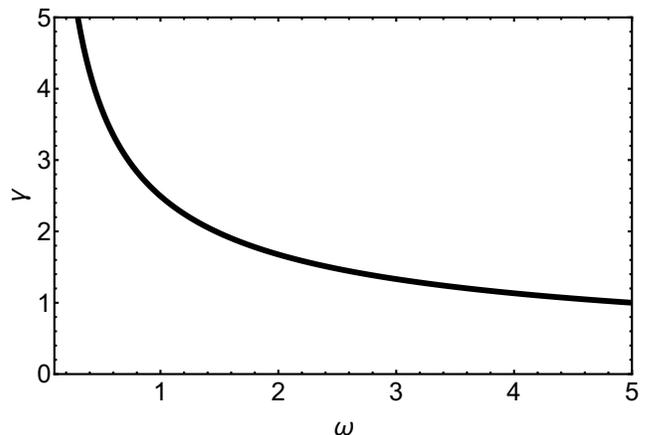


FIG. 5. We plot γ , defined in Eqn. (26), as a function of ω for the Hooke's atom.

Now let us examine the same sequence of plots for the 1D H_2 molecule as a function of R . Fig. 7 shows the densities (both unscaled and scaled) for the sequence of H_2 densities, as a function of R . Now the reverse appears true: The unscaled densities appear to change less (or no more than) the scaled densities. As R increases, the scaled densities change considerably. Moreover, even when the densities are essentially indistinguishable from the sum of atomic densities, the scaled density continues to shrink in extent around the two centers as R increases, i.e., the scaled densities continue to change with R , even when R is very large.

Of course, appearances are not definitive proof, so we

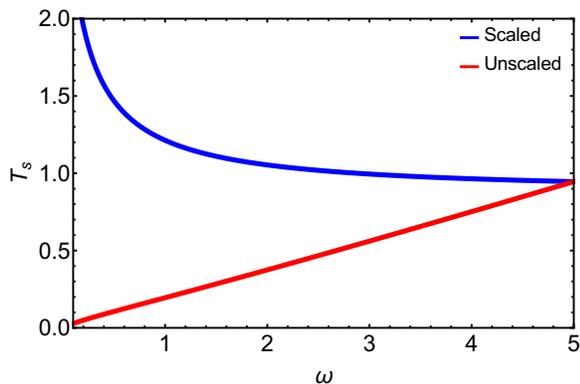


FIG. 6. (color online) The Hooke's atom non-interacting kinetic energy is plotted as a function of ω for the densities before (red) and after scaling (blue).

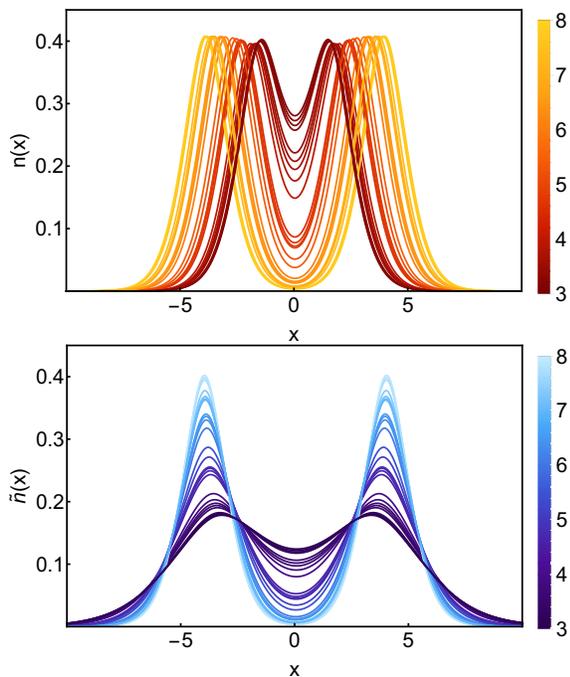


FIG. 7. (color online) The H_2 densities considered are plotted before scaling (top) and after scaling (bottom). The color each line indicates the separation, R . The densities do not appear to be made more similar by scaling.

quantify this effect. Fig. 8 shows γ for different values of R . Comparing to Fig. 5, we see that γ appears to vary linearly with R as R increases, instead of approaching a constant. When we plot the unscaled and scaled kinetic energies in Fig. 9, we see that, in sharp contrast to Fig. 8, now the scaled kinetic energy varies more over the range of densities than the unscaled one. Thus, it is not surprising that scaling is little or no help in this case.

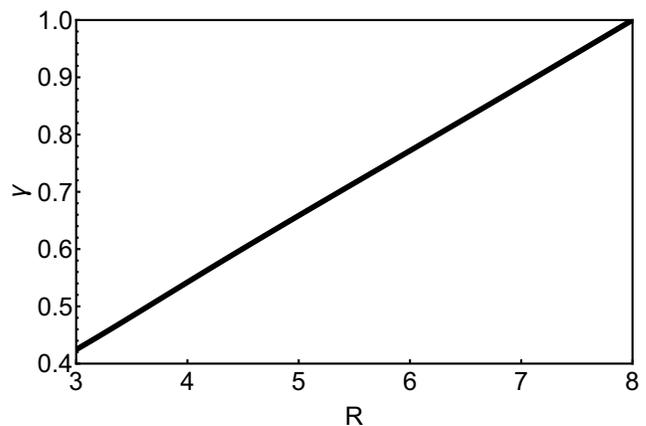


FIG. 8. We plot the scaling function, γ , as a function of R for the H_2 densities.

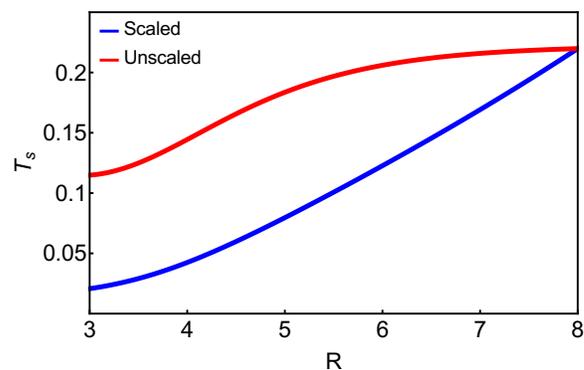


FIG. 9. (color online) The non-interacting kinetic energies as a function of the atomic separation, R , before scaling (red) and after scaling (blue) for the set of H_2 densities. Small differences in unscaled kinetic energies are exacerbated by scaling, particularly at large R .

6. DISCUSSION

We see that, for the contact interacting one-dimensional Hooke's atom described above, it is advantageous to machine learn the scaled non-interacting energy and use exact conditions to relate this to the unscaled non-interacting kinetic energy, rather than machine learn the unscaled non-interacting kinetic energy directly. Furthermore, this result does not extend to set of one-dimensional H_2 densities, where both methods of machine learning the non-interacting kinetic energy have relatively similar performance.

To understand the system-dependence of the improved performance, we analyze the densities plotted in Fig. 4 and Fig. 7. We see that, for the Hooke's atom, the densities in the reference data set are approximately scaled versions of each other. Explicitly, as ω increases, the peaks in the density shift away from the origin and become broader. Scaling the densities reverses the effect of this trend, creating new densities that may be more meaningfully compared. In contrast, for H_2 , the densities in the reference data set do not resemble scaled versions of each other. As R increases, the

peaks in the density shift away from the origin, but they do not broaden. As such, scaling these densities will make the densities more similar by aligning the peaks of the densities, but will artificially make them more dissimilar by broadening or narrowing the width of the peaks.

We can understand the poor performance of the exact condition for stretched H_2 as follows. Consider a model of two disjoint atomic densities. We model an atomic density as a gaussian:

$$n_A(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2) \quad (27)$$

and consider densities of the form

$$n(x) = n_A\left(x + \frac{R}{2}\right) + n_A\left(x - \frac{R}{2}\right), \quad (28)$$

where R is large enough so that overlap can be neglected. Solving for the square-width of this density yields

$$l[n] = \sqrt{2 + \frac{R^2}{2}}. \quad (29)$$

Then, for $R \gg 1$, Eq. (7) yields $T_s \rightarrow 1/4$, independent of R . So in principle, one could machine learn $T_s[n]$ for all densities of sufficiently large R with a single training example. On the other hand, the scaled kinetic energy does not simplify in this limit. For large R , $l[n] \approx R/\sqrt{2}$. Thus, $\tilde{T}_s[\tilde{n}] \approx R^2 T_s[n]/2 = R^2/8$ (see the large R behavior in Figs. 8 and 9). This non-trivial dependence upon R makes it at least as difficult to learn the scaled kinetic energy as it is to learn the unscaled kinetic energy.

We believe that this insight allows us to infer how our results generalize to more realistic systems. If the set of reference densities is made more similar by scaling, then we suspect that machine learning $\tilde{T}_s[\tilde{n}]$ and using the exact scaling condition will be more data efficient, as was seen in the case of the Hooke's atom. On the other hand, if the set of reference densities is not made more similar by scaling, then we suspect that either method of machine learning $T_s[n]$ and $\tilde{T}_s[\tilde{n}]$ will yield similar performance. It is worth noting that, even when the functionals produce similar errors, $\tilde{T}_s[\tilde{n}]$ may still be considered superior, as using this functional to calculate $T_s[n]$ will satisfy the exact scaling condition, Eq. (17), by construction. This is in contrast to directly machine learning the functional $T_s[n]$, which will only approximately satisfy the exact scaling condition within the training manifold and will violate this condition outside of the training manifold.

In fact, we can see the stretched case as a conflict between two exact conditions. The first is coordinate scaling, while the second is size-consistency: Once two densities stop overlapping, any functional should be equal to or tending toward the sum of its values on the two separated densities. Our imposition of the first condition made it more difficult to learn the second.

7. CONCLUSIONS

In this paper, for two simple model systems (each interacting 2-electron systems in 1D), we have imposed an exact condition (quadratic scaling with coordinates) on the machine learned Kohn-Sham kinetic energy functional. We find that typically results are improved, but much more so in the case of a harmonic trap than in the case of a bond being stretched.

We can make intelligent guesses as to what is going on, and how such findings might generalize in the real world. The Hooke's atom example should behave very similarly to the sequence of two-electron ions as a function of nuclear charge. Thus we expect imposing of the exact condition would greatly improve accuracy. But one could reasonably ask, what use is an ML kinetic energy functional for just 2-electrons in such a limited situation?

A more interesting situation would be the sequence of neutral atoms, possibly including also anions and cations. Here the densities are confined, and they must (weakly) approach the Thomas-Fermi solution for large (non-relativistic) values of Z , the nuclear charge. We suspect that, once again, the energy would be much more accurately learned with the imposition of the exact condition.

But in the simple case of stretching a diatomic bond, the changes in shape of density continue as long as the bond is stretched, and little is gained in accuracy on the interpolation manifold by imposing the condition. It suggests that, in some sense, that scaling of any density on the manifold moves one in a direction away from the manifold, yielding no additional information on the manifold itself. This would therefore appear to be the most difficult case in which to improve learning, and probably a different exact condition (such as separating to isolated atoms) would be more beneficial to the learning process. We note also that all semilocal functionals fail in this limit. Finally, we note that if we are interested in more than a simple stretch along a bond, such as vibrational and rotational motion in 3D, we can expect better performance enhancement from imposing the exact condition. This difficulty is likely related to the difficulties encountered in stretching 1D H chains in Ref. [10].

8. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1321846. KB and LL graciously acknowledge support from NSF Grant No. CHE-1240252. TEB thanks the support of the postdoctoral fellowship at Institut quantique. This work was supported by the U.S. Department of Energy, Office of Science, Basic Energy Sciences under award # DE-SC008696.

- [1] Aurora Pribram-Jones, David Gross, and Kieron Burke. Dft: A theory full of holes? *Annual Review of Physical Chemistry*, 66:283, Apr 2015. 1
- [2] A. Kiejna. Stabilized jellium –simple model for simple-metal surfaces. *Prog. Surf. Sci.*, 61:85, 1999. 1
- [3] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140(4A):A1133–A1138, Nov 1965. 1, 2
- [4] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77(18):3865–3868, Oct 1996. *ibid.* 78, 1396(E) (1997). 1, 2
- [5] Roberto Peverati and Donald G Truhlar. Quest for a universal density functional: the accuracy of density functionals across a broad spectrum of databases in chemistry and physics. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 372(2011):20120476, 2014. 1
- [6] A. D. Becke. Density-functional exchange-energy approximation with correct asymptotic behavior. *Phys. Rev. A*, 38(6):3098–3100, Sep 1988. 1, 2
- [7] Chengteh Lee, Weitao Yang, and Robert G. Parr. Development of the colle-salvetti correlation-energy formula into a functional of the electron density. *Phys. Rev. B*, 37(2):785–789, Jan 1988. 1, 2
- [8] Michael G. Medvedev, Ivan S. Bushmarinov, Jianwei Sun, John P. Perdew, and Konstantin A. Lyssenko. Density functional theory is straying from the path toward the exact functional. *Science*, 355(6320):49–52, 2017. 1
- [9] John C. Snyder, Matthias Rupp, Katja Hansen, Klaus-Robert Müller, and Kieron Burke. Finding density functionals with machine learning. *Phys. Rev. Lett.*, 108:253002, 2012. 1, 2
- [10] Li Li, Thomas E. Baker, Steven R. White, and Kieron Burke. Pure density functional for strong correlation and the thermodynamic limit from machine learning. *Phys. Rev. B*, 94:245129, Dec 2016. 1, 2, 3, 7
- [11] John C. Snyder, Matthias Rupp, Katja Hansen, Leo Blooston, Klaus-Robert Müller, and Kieron Burke. Orbital-free bond breaking via machine learning. *J. Chem. Phys.*, 139:224104, 2013. 1
- [12] Albert P. Bartók, Mike C. Payne, Risi Kondor, and Gábor Csányi. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.*, 104:136403, Apr 2010. 1
- [13] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.*, 108:058301, Jan 2012.
- [14] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98:146401, Apr 2007.
- [15] Kun Yao, John E. Herr, Seth N. Brown, and John Parkhill. Intrinsic bond energies from a bonds-in-molecules neural network. *The Journal of Physical Chemistry Letters*, 8(12):2689–2694, 2017. PMID: 28573865.
- [16] Raghunathan Ramakrishnan and Anatole von Lilienfeld. Machine learning, quantum chemistry, and chemical space. pages 225–256, 04 2017.
- [17] Matthias Rupp, Raghunathan Ramakrishnan, and O. Anatole von Lilienfeld. Machine learning for quantum mechanical properties of atoms in molecules. *The Journal of Physical Chemistry Letters*, 6(16):3309–3313, 2015.
- [18] Kristof T. Schütt, Farhad Arbabzadah, Stefan Chmiela, Klaus R. Müller, and Alexandre Tkatchenko. Quantum-chemical insights from deep tensor neural networks. *Nature Communications*, 8:13890 EP –, Jan 2017. Article.
- [19] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato. Grammar Variational Autoencoder. *ArXiv e-prints*, March 2017.
- [20] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural Message Passing for Quantum Chemistry. *ArXiv e-prints*, April 2017.
- [21] K. Yao, J. E. Herr, D. W. Toth, R. Mcintyre, and J. Parkhill. The TensorMol-0.1 Model Chemistry: a Neural Network Augmented with Long-Range Physics. *ArXiv e-prints*, November 2017. 1
- [22] Kieron Burke and Lucas O. Wagner. DFT in a nutshell. *International Journal of Quantum Chemistry*, 113(2):96–101, 2013. 1
- [23] L. H. Thomas. The calculation of atomic fields. *Math. Proc. Camb. Phil. Soc.*, 23(05):542–548, 1927. 2
- [24] E. Fermi. *Rend. Acc. Naz. Lincei*, 6, 1927. 2
- [25] J. P. Perdew and K. Schmidt. In V. E. Van Doren, K. Van Alsenoy, and P. Geerlings, editors, *Density Functional Theory and Its Applications to Materials*. American Institute of Physics, Melville, NY, 2001. 2
- [26] Felix Brockherde, Leslie Vogt, Li Li, Mark E Tuckerman, Kieron Burke, and Klaus-Robert Müller. By-passing the kohn-sham equations with machine learning. *Nature Communications*, 8, 2016. 2
- [27] Li Li, John C. Snyder, Isabelle M. Pelaschier, Jessica Huang, Uma-Naresh Niranjana, Paul Duncan, Matthias Rupp, Klaus-Robert Müller, and Kieron Burke. Understanding machine-learned density functionals. *International Journal of Quantum Chemistry*, 116(11):819–833, 2016. 2
- [28] Kevin Vu, John C. Snyder, Li Li, Matthias Rupp, Brandon F. Chen, Tarek Khelif, Klaus-Robert Müller, and Kieron Burke. Understanding kernel ridge regression: Common behaviors from simple functions to density functionals. *International Journal of Quantum Chemistry*, 115(16):1115–1128, 2015. 2
- [29] R.J. Magyar and K. Burke. A density functional approach to one-dimensional interacting fermions. *Phys. Rev. A*, 70:032508, 2004. 3, 4
- [30] T Busch and G Huyet. Low-density, one-dimensional quantum gases in a split trap. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 36(12):2553, 2003. 3, 4
- [31] Steven R. White. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.*, 69(19):2863–2866, Nov 1992. 3
- [32] Thomas E. Baker, E. Miles Stoudenmire, Lucas O. Wagner, Kieron Burke, and Steven R. White. One-dimensional mimicking of electronic structure: The case for exponentials. *Phys. Rev. B*, 91:235141, Jun 2015. 3