Development of an Exchange-Correlation Functional with Uncertainty Quantification Capabilities for Density Functional Theory

Manuel Aldegunde, James R. Kermode, Nicholas Zabaras[*]

Warwick Centre for Predictive Modelling, University of Warwick
Coventry CV4 7AL, United Kingdom

Abstract

This paper presents the development of a new exchange-correlation functional from the point of view of machine learning. Using atomization energies of solids and small molecules, we train a linear model for the exchange enhancement factor using a Bayesian approach which allows for the quantification of uncertainties in the predictions. A relevance vector machine is used to automatically select the most relevant terms of the model. We then test this model on atomization energies and also on bulk properties. The average model provides a mean absolute error of only 0.116 eV for the test points of the G2/97 set but a larger 0.314 eV for the test solids. In terms of bulk properties, the prediction for transition metals and monovalent semiconductors has a very low test error. However, as expected, predictions for types of materials not represented in the training set such as ionic solids show much larger errors.

Keywords: Bayesian, Density Functional Theory, Exchange-Correlation Functional, Uncertainty Quantification
PACS: 71.15.Mb, 71.15.Nc, 31.15.eg
2010 MSC: 62F15, 82-04

1. Introduction

In the last several decades, density-functional theory (DFT) has become the most widespread framework to study materials from a fully quantum-mechanical perspective due to the favorable trade-off between accuracy and computational cost it provides. Even though the theory in principle is exact, its application calls for the use of approximations, some because of computational reasons, such as the Born-Oppenheimer approximation, and some because the exact term is not known, as is the case with the exchange-correlation (XC) energy. Even though the accuracy of the different approximations has been tested in many fields, the error that they lead to when DFT is applied to new systems remains a concern, which limits the predictive power for new systems.

Recently, several works have been published which try to address this problem from different points of view. K. W. Jacobsen *et al.* have applied the concepts of *sloppy models* [1] to DFT [2, 3, 4, 5, 6]. These are models in which a part of the model parameters, the *sloppy parameters*, are largely unimportant for the model predictions, so that they can only be determined with large uncertainty. In this framework, they start with a model $\mathcal{M}$ which depends on a set of parameters, $\boldsymbol{\theta}$. To find the probability of a given parameter set, $\boldsymbol{\theta}$, given the model, $\mathcal{M}$, and a database of experimental data, $\mathcal{D}$, a Boltzmann distribution is assumed for a cost function $C(\boldsymbol{\theta})$, which can be, for example, a least squares cost function with [2] or without [4, 6] regularisation:

$$P(\boldsymbol{\theta} \mid \mathcal{M}, \mathcal{D}) \sim \exp(-C(\boldsymbol{\theta})/T), \tag{1}$$

where T is an "effective temperature" which determines the spread of the ensemble and therefore the error estimation. This distribution of the parameters $\boldsymbol{\theta}$ is then used to generate ensembles of XC enhancement factors that can be used

[*] Corresponding author
Email addresses: M.A.Aldegunde-Rodriguez@warwick.ac.uk (Manuel Aldegunde), J.R.Kermode@warwick.ac.uk (James R. Kermode), nzabaras@gmail.com (Nicholas Zabaras)
URL: http://www2.warwick.ac.uk/wcpm/, http://www.zabaras.com (Nicholas Zabaras)

to estimate errors in different quantities. This model was used to train a meta-generalized gradient approximation (meta-GGA) exchange-correlation functional, mBEEF, using experimental data for bulk solids (lattice constant and cohesive energies), molecules (formation energies and molecular reaction energies) and surfaces (chemisorption on solid surfaces) [4, 6].

Also recently, K. Lejaeghere *et al.* have studied errors in DFT simulations from a different perspective [7]. Instead of trying to construct a new functional, they analyse the errors for a give XC functional in terms of a linear regression model between experimental and calculated data. The least-squares estimate of the slope β is taken as a measure of systematic errors, and the error ε gives the residual error bar. The scatter in DFT results comes from the ability of the XC functional to represent some materials better than others. They carry out this study on the ground-state elemental crystals at 0 K up to Rn. In this case, the prediction of the properties for a new compound not included in the set is done by applying the correction for systematic deviation and adding the residual error, $x_{pred} = \beta x_{DFT} \pm \varepsilon$, where x_{pred} is the corrected prediction for the magnitude x, β represents the systematic deviation, x_{DFT} the magnitude obtained from the DFT simulation and ε the residual error which models the inability of the DFT model to reproduce the experiment exactly.

In this work, we present the development of a new meta-GGA exchange-correlation functional with uncertainty quantification capabilities from the point of view of machine learning extending on the work of Wellendorff *et al.* [6]. We use a Bayesian approach for the determination of the regression coefficients with a relevance vector machine. Unlike the sloppy model approach using regularised least squares in [6], the use of a relevance vector machine automatically selects the most relevant terms and drops the rest, which avoids the possibly very large number of terms in the linear model and helps avoid overfitting.

In Section 2, we present the basics of DFT and the formulation of a linear model for the exchange energy functional. Next, in Section 3, we detail the Bayesian linear regression framework used to obtain the coefficients of the model as random variables. We also discuss how from these parameters we can get a predictive distribution for the modelled function and also the use of a relevance vector machine to allow for automatic model determination. The actual training of the model using atomization energies is presented in Section 4, which describes in more detail how to set up all necessary data from DFT simulations. We also include a description of how it is possible to use some indirect measurements to extend the available data set for training as well as a description of the training set we used. Numerical results testing the proposed framework are presented in Section 5 for atomization energies of molecules and solids. Even though the training data consists of energies, we can also use the framework to propagate uncertainties to other derived quantities such as bulk properties. Section 6 describes an example of this process using an equation of state, which inserts an extra layer of uncertainty. Then, numerical examples of propagation of uncertainty are shown for two bulk properties of solids, equilibrium lattice constants and bulk moduli, and for the band gap of Si. Finally, we end up summarizing the main contributions of this work as well as the numerical results and how they compare to other available XC functionals.

2. Density Functional Theory

DFT approximates the ground state energy of a material system with charge density n [8, 9]. It does so by minimizing the energy functional $E^{DFT}[n]$ for a given system. This functional is given by [10]:

$$\begin{aligned} E^{DFT}[n] = &\int n(\mathbf{r})v(\mathbf{r})\,d\mathbf{r} + T_0[n] + U[n] + E^{xc}[n] \\ =&E^{b}[n] + E^{xc}[n] = E^{b}[n] + E^{x}[n] + E^{c}[n], \end{aligned} \tag{2}$$

where $\mathbf{r}$ is the position in real space, $v(\mathbf{r})$ is an external potential, $T_0[n]$ is the non-interacting kinetic energy, $U[n]$ the classical electron-electron repulsion energy and $E^{xc}[n]$ is the XC energy functional, which is not known exactly. $E^{b}[n]$ groups all the contributions not coming from exchange and correlation. $E^{xc}[n]$ has two components, the exchange energy $E^{x}[n]$ and the correlation energy $E^{c}[n]$ [10].

By grouping the terms of the external potential, the electron-electron interaction energy, and the exchange-correlation energy, Kohn and Sham [11] introduced a self-consistent scheme to obtain the ground state energy using an equivalent non-interacting system with an effective potential. The solution of this system can be found by solving the Schrödinger equation for non-interacting particles [9]:

$$\left[-\frac{1}{2}\nabla^2 + v_{eff}(\mathbf{r})\right]\psi_i(\mathbf{r}) = \epsilon_i\psi_i(\mathbf{r}), \tag{3}$$

where $-\frac{1}{2}\nabla^2$ is the kinetic energy operator, $\psi_i(\mathbf{r})$ are the Kohn-Sham orbitals, ϵ_i are the Kohn-Sham orbital energies and $v_{eff}(\mathbf{r})$ is an effective potential composed by the external potential, the electron-electron interaction and the XC potential [9, 10]:

$$v_{eff}(\mathbf{r}) = v(\mathbf{r}) + \int \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}\, d\mathbf{r}' + v_{xc}(\mathbf{r}). \tag{4}$$

The XC correlation potential is related to the XC energy through a functional derivative with respect to the density, $v_{xc}(\mathbf{r}) = \frac{\delta E^{xc}[n]}{\delta n(\mathbf{r})}$ [9]. The Kohn-Sham orbitals can be used to obtain the electron density of the system [9],

$$n(\mathbf{r}) = \sum_i |\psi_i(\mathbf{r})|^2. \tag{5}$$

Since the effective potential is itself a functional of the density, Eqs. (3)-(5) need to be solved self-consistently, i.e., they have to be iterated until convergence.

2.1. Exchange-Correlation energy

Even though this formulation of the problem is favorable for numerical computation, there still remains the question of approximating the XC energy and potential since, as we noted before, it is unknown in general.

In [12], J. P. Perdew *et al.* introduced a hierarchy of approximations called "Jacob's ladder". If we write the XC energy as

$$E^{xc}[n] = \int n\varepsilon^{xc}(n; \mathbf{r})\, d\mathbf{r}, \tag{6}$$

where the product $n\varepsilon^{xc}$ is an XC energy density and ε^{xc} is an XC energy per electron, we can see the ladder as increasingly complex approximations for ε^{xc}. At the lowest level, the local density approximation (LDA), ε^{xc} depends only on $n(\mathbf{r})$, the density at the position where the energy is evaluated. A second level includes as well the density gradient, $\nabla n(\mathbf{r})$, and it is called the generalized gradient approximation (GGA). Since it also includes derivative information this type of approximation is called semi-local. Two of the most widely used GGA functionals are the Perdew-Burke-Ernzerhof (PBE) functional [13] and a revision to improve results for solids which modifies two of its parameters, PBEsol [14]. The third level of the hierarchy is the meta-generalized gradient approximation (meta-GGA), which adds dependence on the Kohn-Sham orbitals. Since the Kohn-Sham orbitals are in general non-local functionals of the electron density, meta-GGA approximations are also non-local. However, since meta-GGA functionals are constructed to be local in the orbitals, which are readily available from the solution of the eigenvalue problem, they retain much of the computational efficiency of the GGA [12]. Popular functionals in this category include the Tao-Perdew-Staroverov-Scuseria (TPSS) functional [15], mBEEF [6], or the "Made Simple" (MS0) functional [16]. Higher levels of the hierarchy include ingredients such as the exact expression for the exchange energy,

$$E^{x}[n] = -\frac{1}{2}\sum_i \int \frac{\psi_i^*(\mathbf{r})\psi_i(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}\, d\mathbf{r}d\mathbf{r}'. \tag{7}$$

Hybrid functionals, which combine this exact exchange with semi-local functionals are especially popular in chemistry, and some of the most widely used are the Becke-3-Lee-Yang-Parr (B3LYP) [17], PBE0 [18] and M06 [19].

Because of the non-locality of this last family of functionals, the computations are much more demanding, so in this work we restrict ourselves to the meta-GGA approximation.

2.2. Exchange energy in meta-GGA

At the meta-GGA level of approximation, the XC functional can be written as [20]

$$E^{xc}[n] = \int n(\mathbf{r})\varepsilon^{xc}\left[n(\mathbf{r}), \nabla n(\mathbf{r}), \tau(\mathbf{r})\right]\, d\mathbf{r}, \tag{8}$$

where $\tau(\mathbf{r}) = 2\sum_i' \frac{1}{2}|\nabla\psi_i(\mathbf{r})|^2$ is the kinetic energy density. The summation $\sum_i'$ runs only over the occupied Kohn-Sham orbitals $\psi_i(\mathbf{r})$.

To specify the exchange energy contribution, it is common to introduce the *exchange enhancement factor*, $F^x(n, \nabla n, \tau)$, which contains all the contributions to nonlocality through its dependence on the gradient of the electron density ∇n and the kinetic energy density τ. Using it, the exchange energy functional can be written as [20, 21, 22]

$$E^x[n] = \int n\varepsilon^x(n, \nabla n, \tau)\, d\mathbf{r} = \int n\varepsilon^x_{UEG}(n)F^x(n, \nabla n, \tau)\, d\mathbf{r}, \tag{9}$$

where $\varepsilon^x(n)$ is the exchange energy per particle of the system and $\varepsilon^x_{UEG}(n) = -3(3\pi^2 n)^{1/3}/(4\pi)$ is the exchange energy per particle of a uniform electron gas.

Furthermore, the dependence on the density gradient and the kinetic energy density is usually transformed into the dimensionless parameters s, the reduced density gradient, and α, the dimensionless deviation from a single orbital shape [20, 23]:

$$s = \frac{|\nabla n|}{2(3\pi^2)^{1/3}n^{4/3}}; \quad \alpha = \frac{\tau - \tau^W}{\tau^{UEG}}, \tag{10}$$

where $\tau^W = |\nabla n|/8n$ and $\tau^{UEG} = \frac{3}{10}(3\pi^2)^{2/3}n^{5/3}$ are the Weizsäcker and uniform electron gas kinetic energy densities, respectively. Using these parameters, the exchange energy functional can be written as [22, 24]

$$E^x[n] = \int n\varepsilon^x_{UEG}(n)F^x(s, \alpha)\, d\mathbf{r}. \tag{11}$$

2.3. Linear model for the exchange energy

To specify a DFT approximation, we have to provide two models for the exchange and correlation functionals, $E^x[n; \boldsymbol{\xi}^x]$ and $E^c[n; \boldsymbol{\xi}^c]$, where $\boldsymbol{\xi}^x$ and $\boldsymbol{\xi}^c$ are two sets of parameters which determine the XC model [10], and can be determined either empirically, fitting them to experimental data [6], or from theoretical considerations [22]. Putting these parameters explicitly, the DFT energy functional is then

$$E^{DFT}[n; \boldsymbol{\xi}^x, \boldsymbol{\xi}^c] = E^b[n] + E^x[n; \boldsymbol{\xi}^x] + E^c[n; \boldsymbol{\xi}^c]. \tag{12}$$

Following the previous studies in [2, 4], we will focus on the exchange contribution only, taking the correlation energy term from other functionals. In particular, we will compare the use of the correlation terms from the XC functionals PBE, PBEsol, MS0 and TPSS.

To specify our exchange energy model, we will use the exchange enhancement factor, whose functional form is not known [24]. In this work, we follow [4] and represent it as a linear model in a set of basis functions $\phi_i(s, \alpha)$,

$$F^x(s, \alpha) = \sum_i \xi^x_i \phi_i(s, \alpha) = (\boldsymbol{\xi}^x)^T \boldsymbol{\phi}(s, \alpha), \tag{13}$$

where we have introduced a vector notation for the linear model coefficients $\boldsymbol{\xi}^x = \{\xi^x_i\}$ and basis functions $\boldsymbol{\phi}(s, \alpha) = \{\phi_i(s, \alpha)\}$.

Also following [4], we use a truncated Legendre polynomial expansion on rational transformations of the parameters s and α:

$$F^x(s, \alpha) = \sum_i^{M_s} \sum_j^{M_\alpha} \xi^x_{ij} P_i(t_s(s)) P_j(t_\alpha(\alpha)), \tag{14}$$

where $P_i(x)$ is the Legendre polynomial of order i on x, M_s and M_α are the orders of the expansion for s and α, respectively, and ξ^x_{ij} is the enhancement factor linear model coefficient for orders i, j. The transformations on s and α are defined as

$$t_s(s) = \frac{2s^2}{q + s^2} - 1, \tag{15}$$

and

$$t_\alpha(\alpha) = \frac{(1-\alpha^2)^3}{1+\alpha^3+\alpha^6}, \tag{16}$$

where the parameter q in Eq. (15) is $q = \kappa/\mu^{GE}$, with $\kappa = 0.804$ and $\mu^{GE} = (10/81)$. These parameters are used in the PBEsol exchange functional [14], which is the basis for this transformation. In fact, $t_s(s)$ is the PBEsol enhancement factor scaled to the interval $[-1, 1)$ for $s \geq 0$ [6]. On the other hand, $t_\alpha(\alpha)$ is the α dependence of the MS0 exchange enhancement factor [16], which is confined to the interval $(-1, 1]$ for $\alpha \geq 0$.

Using this linear model for the enhancement factor, we can write our model exchange energy as

$$\begin{aligned} E^x[n;\boldsymbol{\xi}^x] &= \int n\varepsilon^x_{UEG}(n) \sum_{i=1}^{M} \xi^x_i \phi_i(s,\alpha)\, d\mathbf{r} \\ &= \sum_{i=1}^{M} \xi^x_i \int n\varepsilon^x_{UEG}(n)\phi_i(s,\alpha)\, d\mathbf{r} = \sum_{i=0}^{M-1} \xi^x_i E^x[n;\hat{\mathbf{e}}_i] = (\boldsymbol{\xi}^x)^T \mathbf{E}^x[n;\hat{\mathbf{e}}], \end{aligned} \tag{17}$$

where $M = M_s \times M_\alpha$ and $E^x[n;\hat{\mathbf{e}}_i]$ is the exchange energy obtained using the unit vector $\hat{\mathbf{e}}_i$ as coefficient vector in Eq. (13), i.e., $\phi_i(s,\alpha)$ as the enhancement factor. $\mathbf{E}^x[n;\hat{\mathbf{e}}] = \{E^x[n;\hat{\mathbf{e}}_i]\}$ is the vector notation for the exchange energy functionals obtained this way. From now on, we will drop the superscript x in the parameters since they will be the only ones we use, $\boldsymbol{\xi} \equiv \boldsymbol{\xi}^x$.

The exchange energy can be seen as a linear model where the basis functions are given by exchange energies with $\phi_i(s,\alpha)$ as the enhancement factor.

Remark 1. In this work, we will construct a surrogate model for the exchange energy functional, $E^x[n]$, given by Eq. (17), where the coefficients are random variables whose distribution will be determined by the regression process described in Section 3. The randomness of the model will account both for model error and the limited data (epistemic uncertainty) that are used to compute the distribution of $\boldsymbol{\xi}$.

Remark 2. In the context of DFT, energy and all other quantities of interest are functionals of the electron density. In particular, the exchange-correlation energy is a functional of the electron density. However, the specification of a new system on which we want to make predictions is usually obtained as a configuration of atoms in space. Therefore, to calculate its electron density a self-consistent solution of Eqs. (3)-(5) is required.

This means that in order to make predictions our surrogate model still has the cost of a self-consistent simulation as an electron density is needed to use it. This is unlike a typical regression problem where, given an input, the surrogate model would give a prediction for the output bypassing the need to run the full model and thus having a higher computational efficiency.

Remark 3. From Eq. (17), we can see that to obtain the exchange energy for a system s with electron density n, $E^x[n;\boldsymbol{\xi}^x]$, we need to evaluate each of the exchange functional basis, $E^x[n;\hat{\mathbf{e}}_i]$, for the same value of the density, n. For the training of the system, one needs to evaluate the basis functionals for the density of the training material systems. We assume here that the change in the charge density obtained performing self-consistent calculations with different XC functionals is small, which is in general a good approximation due to the variational principle [2] and has also been shown numerically [25]. This means that, as long as the density is obtained self-consistently, the XC functional chosen to calculate it is of secondary importance in the evaluation of the basis functions in Eq. (17). Under this assumption to carry out the training of the model, we can obtain the density for every system of interest running a self-consistent simulation with a common functional. If we denote the density of the training system s as n^*_s, we can then calculate $E^x[n^*_s;\hat{\mathbf{e}}_i]$ (values of the basis functions in the training charge densities), using the PBE functional as a common functional, i.e., n^*_s will be the self-consistent density obtained using the PBE XC functional for all training systems s.

3. Bayesian Linear Regression

To use the linear model of Eq. (13) for predictions, we will use a *Bayesian linear regression* model [26, 27]. The advantage of using a Bayesian framework is that the regression coefficients will be treated as random variables instead of point quantities as with, for example, classical least squares regression. The Bayesian model will allow us to quantify the uncertainty in predicted values for unseen material systems, as long as these test systems are relevant to the materials used in the training of the model.

In what follows, we use the following notation for the probability distributions that will arise in the description of the model:

- $\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \mathbf{S})$ is a Gaussian distribution on $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance $\mathbf{S}$,
- $\mathcal{G}(x \mid \alpha, \beta)$ is a Gamma distribution on x with parameters α and β,
- $\mathcal{S}t(\mathbf{x} \mid \boldsymbol{\mu}, \Lambda, \nu)$ is a Student t-distribution on $\mathbf{x}$ with parameters $\boldsymbol{\mu}$, Λ and ν.

Formally, the procedure involves assuming that a set of experimental or numerically generated data $\mathbf{t}$ are available representing noisy observations of $E^x[n_i]$ for various densities n_i. The linear model for the underlying process generating the data is of the form:

$$E^x[n; \boldsymbol{\xi}] = \sum_{i=1}^{M} \xi_i E^x[n; \hat{\mathbf{e}}_\mathbf{i}], \tag{18}$$

where ξ_i are the M parameters of the model and $E^x[n; \hat{\mathbf{e}}_\mathbf{i}]$ are the basis functions. *The first aim of Bayesian linear regression is to compute a* probability distribution *for the parameters $\boldsymbol{\xi} = \{\xi_i\}$ of the linear model given the observed data, i.e., $p(\boldsymbol{\xi} \mid \mathbf{t})$.*

Remark 4. In this work, the model parameters $\boldsymbol{\xi}$ only appear on the exchange energy term, so that the full model for the energy is

$$E[n; \boldsymbol{\xi}] = E^b[n] + E^c[n] + \sum_{i=1}^{M} \xi_i E^x[n; \hat{\mathbf{e}}_\mathbf{i}]. \tag{19}$$

We have chosen to transform the experimental data subtracting the energy contributions $E^b[n]$ and $E^c[n]$ obtained from simulations. For every training system, we run a self-consistent simulation and subtract these energy terms from the experimentally observed energies. From now on, we will refer to these energies which are obtained using both experimental and simulated data simply as the *energy training data set.*

We will assume that the observed data $\mathbf{t}$ follow on average our model exchange functional, $\boldsymbol{\xi}^T \mathbf{E}^x[n; \hat{\mathbf{e}}]$, and have an additional error term ε. This quantity represents the model accuracy as it provides a measure of the deviation between our model and the experimental results. Therefore, for a single observation t_i,

$$t_i = \boldsymbol{\xi}^T \mathbf{E}^x[n_i; \hat{\mathbf{e}}] + \varepsilon_i, \tag{20}$$

where ε_i is an error term. Under the assumption of Gaussian error with the same precision for all data points, $\beta = 1/\nu = 1/\sigma^2$, where ν is the variance and σ the standard deviation, the probability of getting a particular value for the observation for a material system with charge density n_i will follow a Gaussian distribution with mean $\boldsymbol{\xi}^T \mathbf{E}^x[n_i; \hat{\mathbf{e}}]$:

$$t_i \sim \mathcal{N}(t \mid \boldsymbol{\xi}^T \mathbf{E}^x[n_i; \hat{\mathbf{e}}], \beta^{-1}). \tag{21}$$

The likelihood function $\mathcal{L}$ gives a measure of how likely it is to obtain the observed data $\mathbf{t}$ given the assumed model. In our case it will be

$$\mathcal{L}(\mathbf{t} \mid \mathbf{n}, \boldsymbol{\xi}, \beta) = \prod_{i=1}^{N} \mathcal{N}(t_i \mid \boldsymbol{\xi}^T \mathbf{E}^x[n_i; \hat{\mathbf{e}}], \beta^{-1}), \tag{22}$$

where N is the number of experimental data, $\mathbf{n} = \{n_i\}$ are the densities of the input system, $\boldsymbol{\xi}$ are the coefficients of the linear model and β is the noise precision.

In a Bayesian framework, we can also specify any prior knowledge about the unknown parameters, $\boldsymbol{\xi}$ and β in our case. One option to facilitate the tractability of the resulting equations is the use of conjugate priors. With this selection the posterior probability distribution of the parameters will belong to the same family as the prior probability distribution [26]. For our model this means using the following priors:

- Prior on $\boldsymbol{\xi}$: $p(\boldsymbol{\xi} \mid \beta, \mathbf{m}_0, \mathbf{S}_0) = \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_0, \beta^{-1}\mathbf{S}_0)$
- Prior on β: $p(\beta \mid a_0, b_0) = \mathcal{G}(\beta \mid a_0, b_0)$

$\mathbf{m}_0$, $\mathbf{S}_0$, a_0 and b_0 are parameters which define the distribution over the model parameters and are called hyperparameters [26]. Section 3.2 will explain how they are determined from the data.

With the chosen priors for each model parameter, the joint prior probability distribution over our model parameters becomes

$$p(\boldsymbol{\xi}, \beta) = p(\boldsymbol{\xi} \mid \beta)p(\beta) = \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_0, \beta^{-1}\mathbf{S}_0)\mathcal{G}(\beta \mid a_0, b_0). \tag{23}$$

Given the likelihood and the prior, we can obtain the posterior probability distribution of the parameters given the data using Bayes' theorem [26]:

$$p(\boldsymbol{\xi}, \beta \mid \mathbf{t}) = \frac{\mathcal{L}(\mathbf{t} \mid \mathbf{n}, \boldsymbol{\xi}, \boldsymbol{\beta})p(\boldsymbol{\xi}, \beta)}{\int \mathcal{L}(\mathbf{t} \mid \mathbf{n}, \boldsymbol{\xi}, \boldsymbol{\beta})p(\boldsymbol{\xi}, \beta)\, d\boldsymbol{\xi}\, d\beta} = \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\mathcal{G}(\beta \mid a_N, b_N). \tag{24}$$

Details on how to arrive at Eq. (24) can be found in Appendix A. The parameters of the posterior distribution are:

$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \boldsymbol{\Phi}^T\boldsymbol{\Phi}, \tag{25}$$

$$\mathbf{m}_N = \mathbf{S}_N\left[\mathbf{S}_0^{-1}\mathbf{m}_0 + \boldsymbol{\Phi}^T\mathbf{t}\right], \tag{26}$$

$$a_N = a_0 + N/2, \tag{27}$$

$$b_N = b_0 + \frac{1}{2}\left(\mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0 - \mathbf{m}_N^T\mathbf{S}_N^{-1}\mathbf{m}_N + \mathbf{t}^T\mathbf{t}\right). \tag{28}$$

Finally, we have introduced the *design matrix* $\boldsymbol{\Phi}$ defined as:

$$\boldsymbol{\Phi} = \begin{pmatrix} E^x[n_1^*; \hat{\mathbf{e}}_0] & \cdots & E^x[n_1^*; \hat{\mathbf{e}}_{M-1}] \\ E^x[n_2^*; \hat{\mathbf{e}}_0] & \cdots & E^x[n_2^*; \hat{\mathbf{e}}_{M-1}] \\ \vdots & \ddots & \vdots \\ E^x[n_N^*; \hat{\mathbf{e}}_0] & \cdots & E^x[n_N^*; \hat{\mathbf{e}}_{M-1}] \end{pmatrix} = \begin{pmatrix} \mathbf{E}^x[n_1^*; \hat{\mathbf{e}}]^T \\ \mathbf{E}^x[n_2^*; \hat{\mathbf{e}}]^T \\ \vdots \\ \mathbf{E}^x[n_N^*; \hat{\mathbf{e}}]^T \end{pmatrix}, \tag{29}$$

where n_i^*, $i = 1 \ldots N$ are the self-consistent densities using the PBE XC functional for each of the N material systems used in the training set of our linear regression problem.

Remark 5. ***Further theoretical constraints on the XC energy such as the Lieb-Oxford bound [10], which gives a theoretical upper bound for the exchange enhancement factor, can be imposed through the prior, even though the resulting posterior would require numerical methods such as Markov Chain Monte Carlo for sampling.***

3.1. Predictive distribution

Once we have a distribution for the model parameters, we can calculate the probability distribution of the output of our model, $\tilde{t}$, given an input system with density $\tilde{n}$. This is called the *predictive distribution* and it is computed by averaging the likelihood of $\tilde{t}$ given $\tilde{n}$ over all sets of the parameters defined by the posterior $p(\boldsymbol{\xi}, \beta \mid \mathbf{t})$,

$$\begin{aligned} p(\tilde{t} \mid \tilde{n}, \mathbf{t}) &= \int p(\tilde{t} \mid \tilde{n}, \boldsymbol{\xi}, \beta)p(\boldsymbol{\xi}, \beta \mid \mathbf{t})\, d\boldsymbol{\xi}\, d\beta \\ &= \int \mathcal{N}(\tilde{t} \mid \boldsymbol{\xi}^T\mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1})\mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\mathcal{G}(\beta \mid a_N, b_N)\, d\boldsymbol{\xi}\, d\beta \\ &= \mathcal{S}t(\tilde{t} \mid \mu, \lambda, \nu). \end{aligned} \tag{30}$$

Therefore, for the assumptions we used, the resulting predictive distribution is a Student t-distribution $\mathcal{S}t(\tilde{t} \mid \mu, \lambda, \nu)$ with the parameters

$$\begin{aligned}
\mu &= \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]^T \mathbf{m}_N, && (31)\\
\lambda &= \frac{a_N}{b_N}\left(1 + \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]^T \mathbf{S}_N \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]\right)^{-1}, && (32)\\
\nu &= 2a_N. && (33)
\end{aligned}$$

Details of these standard derivations can be found in Appendix B.

The mean, variance and mode of this distribution are [26]:

$$\begin{aligned}
\mathrm{E}[\tilde{t}] &= \mu; \quad \nu > 1, && (34)\\
\mathrm{cov}[\tilde{t}] &= \frac{\nu}{\nu - 2}\lambda^{-1} = \frac{b_N}{a_N - 1}\left(1 + \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]^T \mathbf{S}_N \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]\right); \quad \nu > 2, && (35)\\
\mathrm{mode}[\tilde{t}] &= \mu, && (36)
\end{aligned}$$

We can see that the variance of the prediction depends on each new data point. ***Also, from Eq.*** (35) ***we can see that*** $b_N/(a_N - 1)$ ***acts as a lower bound to the variance in the prediction for any single new structure and cannot be lowered further even in the limit of an infinite number of training data points.***

Remark 6. Equation (30) gives a predictive distribution on $\tilde{t}$ *given* the density of the system, $\tilde{n}$, by averaging over all parameters $\boldsymbol{\xi}$ and β. However, as noted in Remark 2, the density of the system has to be calculated self-consistently to be used as an input to the XC functional. For each value of $\boldsymbol{\xi}$ in the integration we would have a different self-consistent density $\tilde{n}$ and therefore the integral would be analytically intractable. One way to overcome this difficulty would be using a Monte Carlo procedure as outlined in Algorithm 1.

Algorithm 1 Monte Carlo approximation for the distribution $p(\tilde{t})$.

```
Given $p(\boldsymbol{\xi},\beta)$, Eq. (24).
for $i = 0, 1, \dots, N_{samples}$ do
    Draw sample of $\boldsymbol{\xi}$ and $\beta$ from $p(\boldsymbol{\xi},\beta)$, $\boldsymbol{\xi}^i, \beta^i$.
    Calculate the self-consistent density $\tilde{n}^i$ using $E^x[n; \boldsymbol{\xi}^i]$, Eq. (17).
    Sample $\tilde{t}^i$ from Eq. (21), using $\tilde{n}^i$ and $\beta^i$.
end for
Approximate the distribution of $\tilde{t}$, $p(\tilde{t})$, using the sampled values $\{\tilde{t}^i\}$.
```

However, this method would require as many self-consistent simulations as samples of $\tilde{t}$ are needed. Therefore, we use the approximation described in Remark 3 and consider that the electron density of the system is equal for all XC functionals obtained by sampling the random variable $\boldsymbol{\xi}$. This assumption allows the use of the predictive distribution given by Eq. (30) using one single self-consistent simulation of the electron density $\tilde{n}$.

The predictive distribution can be readily extended to the case of several predictions. In this case, we obtain a multidimensional Student t-distribution where the different predictions are correlated,

$$p(\tilde{\mathbf{t}} \mid \tilde{\mathbf{n}}, \mathbf{t}) = \int p(\tilde{\mathbf{t}} \mid \tilde{\mathbf{n}}, \xi, \beta)\, p(\xi, \beta \mid \mathbf{t})\, d\xi\, d\beta = \mathcal{S}t(\tilde{\mathbf{t}} \mid \mu, \Lambda, \nu), \tag{37}$$

where the mean and covariance are now given as

$$\begin{aligned}
\mathrm{E}[\tilde{\mathbf{t}}] &= \tilde{\boldsymbol{\Phi}} \mathbf{m}_N, && (38)\\
\mathrm{cov}[\tilde{\mathbf{t}}] &= \frac{b_N}{a_N - 1}\left(\mathbf{I} + \tilde{\boldsymbol{\Phi}} \mathbf{S}_N \tilde{\boldsymbol{\Phi}}^T\right). && (39)
\end{aligned}$$

$\tilde{\mathbf{\Phi}}$ is analogous to the design matrix, but each row has the basis functions evaluated at one of the points where the predictions are made instead of at one of the training points. This shows that the errors in energy differences are not just the addition of the errors in each calculation, but they will be smaller for positively correlated variables and larger for negatively correlated variables. We can also see that the covariance in Eq. (39) ***has two terms whose relative importance will depend on the particular training and test sets. The first term in the covariance originates from the likelihood function, and since we assumed uncorrelated model error, it is a diagonal matrix. The second term contains the covariance between model coefficients and carries all the correlations.***

3.2. Hyperparameters: Evidence approximation

In a Bayesian context, our initial beliefs on the model parameters ξ and β are encoded in the prior distributions, Eq. (23)***. The more we know about the parameters, the more informative the prior distribution can be. However, when we do not have any strong indication on the precise values, the prior will be more uninformative. For example, we may encode our belief that the parameters ξ are more likely to be close to zero with a Gaussian prior distribution centered at the origin, but we may not know exactly how close they should be to zero. In this case, we would like to leave the covariance of the prior distribution as an unknown parameter and learn its value from the data.***

The hyperparameters, $\mathbf{m}_0$***,*** $\mathbf{S}_0$***,*** a_0 ***and*** b_0 ***in our model, can be determined, for example, using the*** **evidence approximation*****, which aims to maximize the marginal likelihood of the training data. The likelihood of our data, given in Eq.*** (22)***, is proportional to the probability of having obtained the data given our model, including all of its parameters. Since we have a probability for our model parameters as a function of the hyperparameters only, we can integrate the likelihood over the model parameters and obtain a function, the marginal likelihood, which gives the probability of obtaining the data as a function of the hyperparameters only. By maximizing this function with respect to the hyperparameters, we maximize the probability that our model reproduces the training data. In our case, the marginal likelihood can be written as***

$$p(\mathbf{t} \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \int p(\mathbf{t} \mid \boldsymbol{\xi}, \beta) p(\boldsymbol{\xi}, \beta \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0)\, d\boldsymbol{\xi}\, d\beta.$$

This is equivalent to maximizing the log of the marginal likelihood (*evidence function*),

$$\mathcal{E}(\mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \log p(\mathbf{t} \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0),$$

$$\begin{aligned}\mathcal{E}(\mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \frac{1}{2} \log \frac{|\mathbf{S}_N|}{|\mathbf{S}_0|} - \frac{N}{2} \log(2\pi) + \\ \log \frac{\Gamma(a_N)}{\Gamma(a_0)} + a_0 \log(b_0) - a_N \log(b_N).\end{aligned} \tag{40}$$

Using $\mathbf{m}_0 = 0$ and $\mathbf{S}_0^{-1} = \alpha\mathbb{I}$, we only have three hyperparameters and we can easily find the maximum of the evidence function.

However, in this paper, we have chosen to use a *relevance vector machine (RVM)* [28, 26]. In this case, $\mathbf{S}_0^{-1} = \mathrm{diag}(\alpha_0, \ldots, \alpha_{M-1})$ and $\mathbf{m}_0 = 0$. This means that we have $M + 1$ hyperparameters. The process tends to produce a sparsification of the regression coefficients, i.e., some of them will become very close to zero [26] and only the most relevant will be kept. This is used for automatic relevance determination of the different basis functions. ***Therefore, unlike previous empirical approaches where the functional dependence was fixed [2, 3], using this prior provides the flexibility for automatic parameter*** **and** ***model selection, which reduces the risk of overfitting present when the number of basis functions is high.*** We update the parameters in an iterative way by looking at the maximum of the evidence function for the current posterior covariance $\mathbf{S}_N$ and mean $\mathbf{m}_N$. Details of the derivation of the equations to find the maximum are in Appendix C. The α_i are updated sequentially until all are converged using Eq. (C.6) [28, 26] and then a_0 and b_0 are updated simultaneously using a Newton method. In the numerical implementation of the pruning of the model basis functions, we consider a maximum value of $\alpha_{max} = 10^{13}$ as an approximation to the limit $\alpha \to \infty$. These two updates are repeated until convergence is achieved. This process is detailed in Algorithm 2.

Algorithm 2 Hyperparameter optimization.

```
$\mathbf{S}_0^{-1} = diag(\alpha_0, \ldots, \alpha_{M-1})$, $\mathbf{m}_0 = 0$.
Select convergence criterion for the inner and outer iterations: $\theta_{inner}$, $\theta_{outer}$
Select numerical threshold $\alpha_{max}$ to detect $\alpha \to \infty$
Initialize $\alpha_i$ from random numbers $r \in (0, 10^{10})$.
repeat
    repeat
        for all $i = 0, 1, \ldots, M-1$ do
            Update $\alpha_i$ as $\alpha_i^{new} = \frac{1}{[\mathbf{S}_N]_{ii} + \frac{a_N}{b_N}[\mathbf{m}_N]_i^2}$, Eq. (C.6).
            Update $\mathbf{S}_N$, $\mathbf{m}_N$ using Eqs. (25) and (26).
        end for
    until $\Delta\alpha_i < \theta_{inner}$ or $\alpha_i > \alpha_{max}$
    Update $a_0, b_0$ with a Newton iteration using Eqs. (C.7) and (C.8).
    Update $\mathbf{S}_N$, $\mathbf{m}_N$ using Eqs. (25) and (26).
until $\Delta\alpha, \Delta a_0, \Delta b_0 < \theta_{outer}$
```

3.3. Sparsity in the relevance vector machine

We can understand better the reason for the sparsification in the RVM if we look at the marginal likelihood as a distribution over the observed data,

$$p(\mathbf{t} \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \int p(\mathbf{t} \mid \boldsymbol{\xi}, \beta) p(\boldsymbol{\xi}, \beta \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0)\, d\boldsymbol{\xi}\, d\beta. \tag{41}$$

This integral is equivalent to that in Eq. (37) *and the result is therefore*

$$p(\mathbf{t} \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \mathcal{S}t(\mathbf{t} \mid \mu_0, \Lambda_0, \nu_0). \tag{42}$$

Since $\mathbf{m}_0 = 0$ *and* $\mathbf{S}_0^{-1} = \mathrm{diag}(\alpha_0, \ldots, \alpha_{M-1})$*, the mean and covariance are now*

$$\mathrm{E}[\mathbf{t}] = 0, \tag{43}$$

$$\mathrm{cov}[\mathbf{t}] = \frac{b_0}{a_0 - 1}\left(\mathbf{I} + \boldsymbol{\Phi}\mathrm{diag}(\alpha_i^{-1})\boldsymbol{\Phi}^T\right). \tag{44}$$

Eq. (42) *for the marginal likelihood is in the space of the training data, i.e., it gives the probability of a given observation for each of the training points given the hyperparameters. We see that the covariance of the marginal likelihood has two components, the first one is isotropic and depends only on the level of model noise (through* a_0*,* b_0*) and the second one is anisotropic and depends also on the rest of the hyperparameters* $\{\alpha_i\}$ *and the design matrix. The objective of the evidence approximation is to maximize the marginal likelihood at a given training data* $\mathbf{t}$*. Since the marginal likelihood is an unimodal distribution centered at the origin, its maximization at the training data* $\mathbf{t}$ *will happen when the covariance is aligned with them.*

As an example, we consider the case of only two training points. In this case, the covariance is

$$\mathrm{cov}[\mathbf{t}] = \frac{b_0}{a_0 - 1}\mathbf{I} + \sum_{i=0}^{M-1} \frac{b_0}{(a_0 - 1)\alpha_i} \begin{pmatrix} E^x[n_1; \hat{\mathbf{e}}_i]^2 & E^x[n_1; \hat{\mathbf{e}}_i] E^x[n_2; \hat{\mathbf{e}}_i] \\ E^x[n_2; \hat{\mathbf{e}}_i] E^x[n_1; \hat{\mathbf{e}}_i] & E^x[n_2; \hat{\mathbf{e}}_i]^2 \end{pmatrix}. \tag{45}$$

If a covariance matrix associated to a basis function is poorly aligned with the experimental observation vector $\mathbf{t}$*, then any finite* α *value will lower the value of the density at* $\mathbf{t}$*. This is illustrated in Fig. 1, which shows the marginal likelihood using a simple case with only one basis function, and two observations [28, 26]. The two training data are generated from a function* $f = \sin(x) + \varepsilon$ *with* $\varepsilon \sim \mathcal{N}(x \mid 0, 1)$ *at points* $\pi/2$ *and* π*. The single basis function of the model is* $f_b = \cos(2x)$*. In this case, any finite value of* α *will lower the marginal probability at* $\mathbf{t}$ *with respect to the case where the covariance includes only the noise. Therefore, the maximization process will prune out the contribution of the basis function by taking* α *to* ∞*. More details on this behavior can be found in [28, 26].*

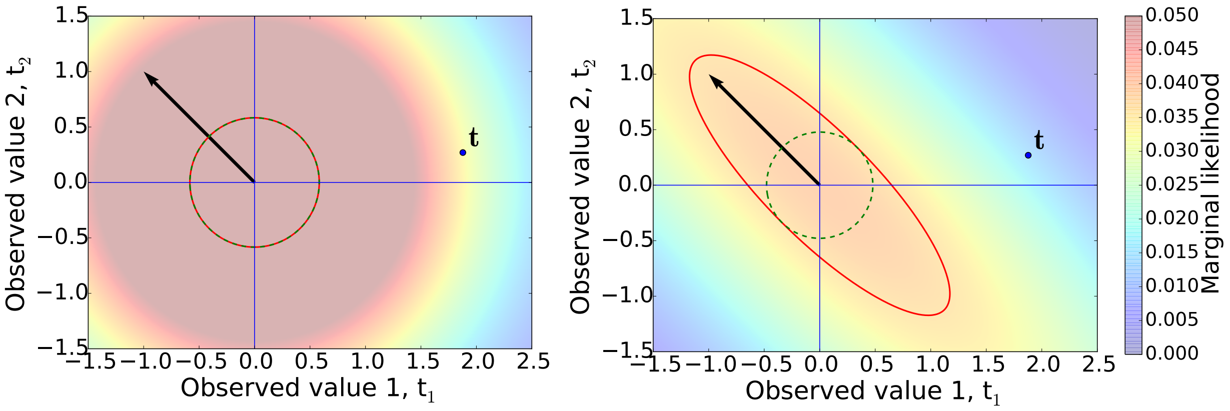


Figure 1: Marginal likelihood function, Eq. (42), for two observation (t_1, t_2). The training data **t** are shown in the picture as a black circle. We also show an equiprobability line of the marginal likelihood at a Mahalanobis distance of 1.5 from the origin using the covariance from Eq. (45) with $b_0/(a_0 - 1)$ chosen to maximize the marginal likelihood at **t**. Left: only noise is contributing to the covariance matrix ($\alpha \to \infty$). The marginal likelihood at **t** in this case is 0.033. Right: the covariance matrix includes a finite $\alpha = 0.2$. The contrition arising from the noise term is shown as a dashed green circle. The marginal likelihood at **t** for this value of alpha is reduced to 0.015. Also shown is the basis vector $(\cos(\pi), \cos(2\pi))$, which is not aligned with the data **t** for our choice of basis function.

As an example of this process, Fig. 2 shows a plot of the magnitude of the regression coefficients using $\mathbf{S}_0^{-1} = diag(\alpha_0, \ldots, \alpha_{M-1})$ for a maximum order of the model $M = M_s \times M_\alpha = 10 \times 10 = 100$. We can see that when we use the RVM most of the coefficients go to zero (within computer accuracy) showing that the corresponding basis functions have negligible relevance in the model.

4. Exchange Model Training

Since the value of the exchange energy is not a quantity that can be measured, we have to use other quantities for which experimental data are available. As the model is linear for the exchange energy, it is natural to choose the energy of different materials as the training data. One such energy available for a wide range of materials is the atomization energy (cohesive energy), which is the energy required for the total separation of all the atoms in the system.

Therefore, the atomization energy per atom of a system $M = A_{n_A} B_{n_B} \ldots$ is defined as the sum of the energies of the individual atoms minus the energy of the system divided by the number of atoms in the system:

$$E_{at} = \frac{1}{N_a}\left(\sum_I n_I E_I - E_M\right), \tag{46}$$

where $N_a = \sum n_I$ is the number of atoms in the supercell and I runs over all the species of atoms, A, B,… E_I is the energy of the isolated atom I and E_M is the energy of the system M.

Using the decomposition of the energy defined in Eq. (2), we can write the atomization energy E_{at} as:

$$\begin{aligned} E_{at} &= \frac{1}{N_a}\left(\sum_I n_I (E_I^b + E_I^x + E_I^c) - (E_M^b + E_M^x + E_M^c)\right) \\ &= E_{at}^b + E_{at}^x + E_{at}^c, \end{aligned} \tag{47}$$

where we have defined the partial atomization energies $E_{at}^\alpha = \frac{1}{N_a}\left(\sum_I n_I E_I^\alpha - E_M^\alpha\right)$, $\alpha = b, x, c$. The components E_{at}^b

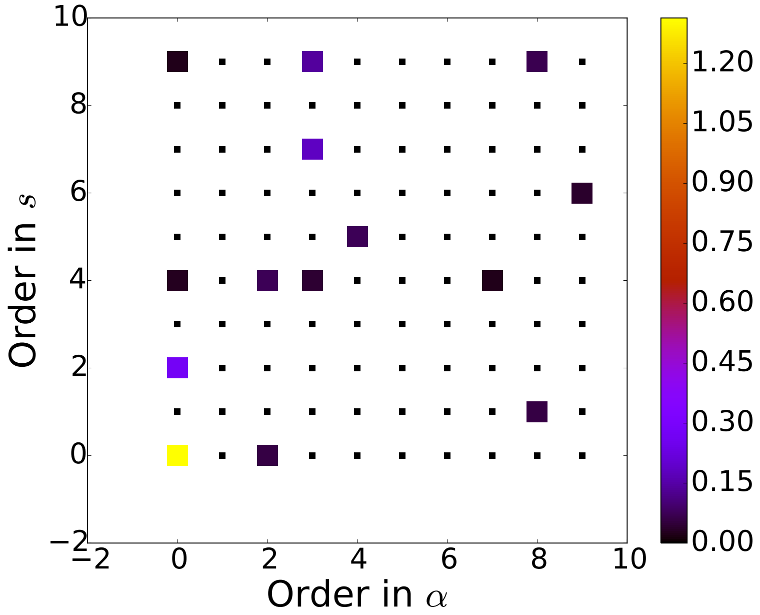


Figure 2: Magnitude of the coefficients obtained using a RVM for the determination of the hyperparameters. The parameters which go to zero have been plotted with a smaller symbol to highlight the sparsity of the model.

and E^c_{at} are fixed in our model, and, using Eq. (17), E^x_{at} can be written as

$$E^x_{at} = \boldsymbol{\xi}^T \frac{1}{N_a}\left[\sum_I n_I \mathbf{E}^x[n_i;\hat{\mathbf{e}}] - \mathbf{E}^x[n_M;\hat{\mathbf{e}}]\right], \tag{48}$$

where n_i is the electron density of the isolated atom I and n_M is the electron density of the system M.

To obtain the parameters of the posterior distribution, Eqs. (25)-(28), we need the training data vector $\mathbf{t}$ with experimental data and the design matrix $\boldsymbol{\Phi}$. We build the design matrix in Eq. (29) as

$$\boldsymbol{\Phi} = \begin{pmatrix} \frac{1}{N_a}(\sum_{I\in s_1} n_I \mathbf{E}^x[n_i;\hat{\mathbf{e}}] - \mathbf{E}^x[n_{s_1};\hat{\mathbf{e}}])^T \\ \frac{1}{N_a}(\sum_{I\in s_2} n_I \mathbf{E}^x[n_i;\hat{\mathbf{e}}] - \mathbf{E}^x[n_{s_2};\hat{\mathbf{e}}])^T \\ \vdots \\ \frac{1}{N_a}(\sum_{I\in s_N} n_I \mathbf{E}^x[n_i;\hat{\mathbf{e}}] - \mathbf{E}^x[n_{s_N};\hat{\mathbf{e}}])^T \end{pmatrix}, \tag{49}$$

where s_i are the systems in the training data set.

Remark 7. To calculate the design matrix Eq. (49), one needs to run a self-consistent simulation to obtain the electron density for (i) every system of the training data and (ii) every isolated atom which is part of any material system in the training set.

As explained in Remark 4, the observed data $\mathbf{t}$ are the experimental atomization energies minus the self-consistently calculated model part not dependent on the parameters $\boldsymbol{\xi}$, i.e.

$$\mathbf{t} = \begin{pmatrix} E^{exp}_{at}(s_1) - E^b_{at}[n_1] - E^c_{at}[n_1] \\ E^{exp}_{at}(s_2) - E^b_{at}[n_2] - E^c_{at}[n_2] \\ \vdots \\ E^{exp}_{at}(s_N) - E^b_{at}[n_2] - E^c_{at}[n_2] \end{pmatrix}. \tag{50}$$

Remark 8. To evaluate the exchange energy basis functionals for a system s, $\mathbf{E}^x[n_s; \hat{\mathbf{e}}]$, one needs its electron density, n_s. Unlike the usual situation, where we know both the experimental data at a given input point where the basis are evaluated, in our problem we do not know the electron density *a priori*. Both the energy and the electron density are obtained simultaneously from a self-consistent simulation of the system. This fact, together with the high dimensionality of the electron density field, poses problems in the application of active learning approaches for choosing the training data set.

4.1. Indirect measurements

It is possible to add other information to the training of energies indirectly to increase the size of the training set. As an example, we have included information of the experimental bulk properties for cubic materials, the equilibrium volume (V_0), equilibrium bulk modulus (B_0) and its pressure derivative (B_1). Using their experimental values within an equation of state (EOS), we can obtain information on the variation of the energy with the volume. One such EOS is the *Stabilized Jellium Equation of State* (SJEOS) [29], which has the form

$$E(V) = a + b\frac{V_0^{1/3}}{V^{1/3}} + c\frac{V_0^{2/3}}{V^{2/3}} + d\frac{V_0}{V} = \boldsymbol{\gamma}^T\boldsymbol{\phi}(V), \tag{51}$$

where V is the volume of the unit cell, $\boldsymbol{\gamma}$ is the vector of coefficients, $\boldsymbol{\gamma} = (a, bV_0^{1/3}, cV_0^{2/3}, dV_0)$, and $\boldsymbol{\phi}(V)$ the vector of basis functions, $\boldsymbol{\phi}(V) = (1, \frac{1}{V^{1/3}}, \frac{1}{V^{2/3}}, \frac{1}{V})$. The coefficients of the equation are related to the equilibrium cohesive energy (E_0), equilibrium volume, bulk modulus and first derivative of bulk modulus through the following equation [29],

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 3 & 2 & 1 & 0 \\ 18 & 10 & 4 & 0 \\ 108 & 50 & 16 & 0 \end{pmatrix} \boldsymbol{\gamma} = \begin{pmatrix} -E_0 \\ 0 \\ 9V_0B_0 \\ 27V_0B_0B_1 \end{pmatrix}. \tag{52}$$

By isotropically straining the unit cell of a solid, we can vary its volume V. Therefore, for every value of strain, we can obtain an energy difference with respect to the equilibrium value V_0, $E(V) - E(V_0)$. Adding this energy increment obtained for a set of five strains in the range $[0.95, 1.05]$ to the cohesive energy at equilibrium, we obtain five training points for each material (including the relaxed configuration) corresponding to cohesive energies of strained configurations.

Remark 9. ***Even though the errors in the energies obtained from the different sources, solids, molecules and indirect measurements will be different from each other,*** we have not considered it in the results presented in this work and for simplicity the same (unknown) noise is assumed for all energy data sets. This assumption can easily be relaxed ***using the evidence approximation.***

4.2. Training sets

To train the model, we used atomization energies of 13 cubic elemental solids from a data set of 20 cubic elemental solids (EL20) and a subset of 120 molecules from the G2/97 dataset [30]. Following the classification in Ref. [7], we have 5 solids from alkali and alkaline earth metals (K, Ca, Rb, Sr and Ba), 9 non-magnetic transition metals (V, Cu, Mo, Rh, Pd, Ag, Ta, W and Au), 5 high-coordination p block compounds (diamond, Al, Si, Ge and Sn) and 1 magnetic material (Fe). Even though in a fully Bayesian setting we would use all the available data for the model training, we have kept a small set aside for test purposes, 2 elements from each category with more than 2 elements (K, Ca, V, Cu, C and Al) and Fe. All simulations are carried out using the projector augmented-wave (PAW) method as implemented in GPAW [31, 32, 33] using plane-wave basis. An energy cut-off of 800 eV was used throughout. For solids, Brilloin zone integrations were done on a $16 \times 16 \times 16$ Monkhorst-Pack k-mesh [34]. Real-space relaxation of molecules in the G2/97 dataset was done using a maximum force of 0.05 eV/Å on each atom [4].

In the problem of training, we are looking for a XC energy functional that is able to predict the energies of new materials using the information contained in the training set. We are using an XC functional with a chosen functional form to predict energies of new data points. However, unless we know the underlying generating function for the XC energy given the density, our predictions far from the training configurations will be more

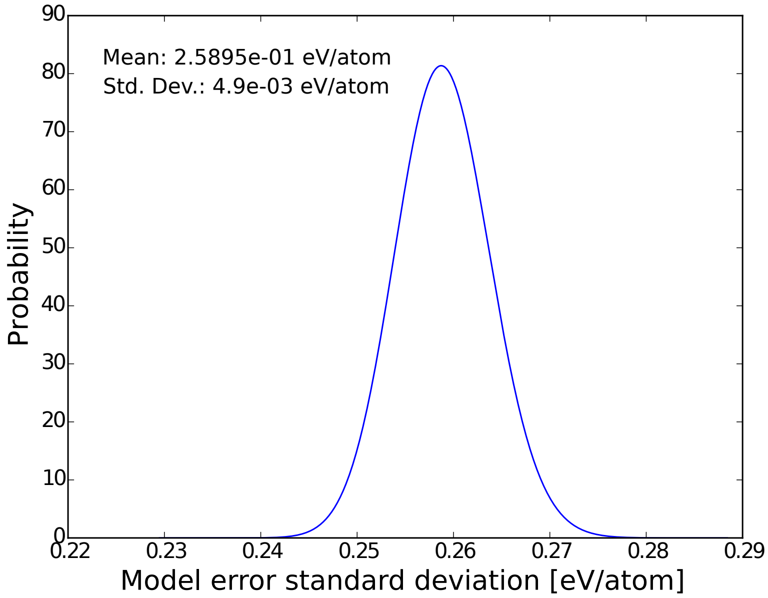


Figure 3: Distribution of the standard deviation characterizing the likelihood function and which represents the error of the model we assume for the exchange model (meta-GGA). Its mean and standard deviation are also shown.

uncertain. This means that our functional is not expected to give reliable estimations for the exchange energy in materials very different from those included in the training data set. For example, if we have not used any magnetic material in our training set, we cannot expect our model to give accurate results for a new magnetic material. An advantage of our model including uncertainty quantification is that ideally it would give a large uncertainty for those points and this information could be used as an indicator of where we cannot have confidence in the obtained result and therefore need new training points. However, as commented in Remark 8, the way in which the input density is calculated self-consistency makes this active learning approach difficult and will not be treated further in this work.

5. Prediction of Atomization Energies

The optimization of the hyperparameters was done as outlined in Algorithm 2 using convergence thresholds $\theta_{inner} = 10^{-5}\%$ and $\theta_{outer} = 10^{-4}\%$. After training the model, we obtain a posterior distribution on the linear model parameters according to Eq. (24),

$$p(\boldsymbol{\xi}, \beta \mid \mathbf{t}) = \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\mathcal{G}(\beta \mid a_N, b_N). \tag{53}$$

The distribution of $\beta = 1/\sigma^2$, the precision of the Gaussian distribution assumed in Eq. (21), gives us an estimation of the model error for the fitted quantity, atomization energies in this case, and the larger it is the smaller the discrepancy between the model and the experimental values. This error is related to the model itself and will not vanish asymptotically as we increase the data set. It is more common to report this error as the standard deviation σ instead of as the precision β. Since β is given by a Gamma distribution, $\mathcal{G}(\beta \mid a_N, b_N)$, then σ^2 follows an inverse Gamma distribution [35], $\mathcal{IG}(\sigma^2 \mid a_N, b_N)$. Therefore, σ is given by $2\sigma\mathcal{IG}(\sigma^2 \mid a_N, b_N)$ [36]. Fig. 3 shows this distribution of the standard deviation σ. The training gives us a value peaked at 0.16 eV for the model error of the cohesive energy. This value is smaller than the previously reported one of 0.31 eV in [7].

The training process also provided us with a distribution for the exchange enhancement factor linear model coefficients. Fig. 4 shows the resulting average enhancement factor, $\boldsymbol{\xi} = \mathbf{m}_N$, as a function of the reduced density gradient

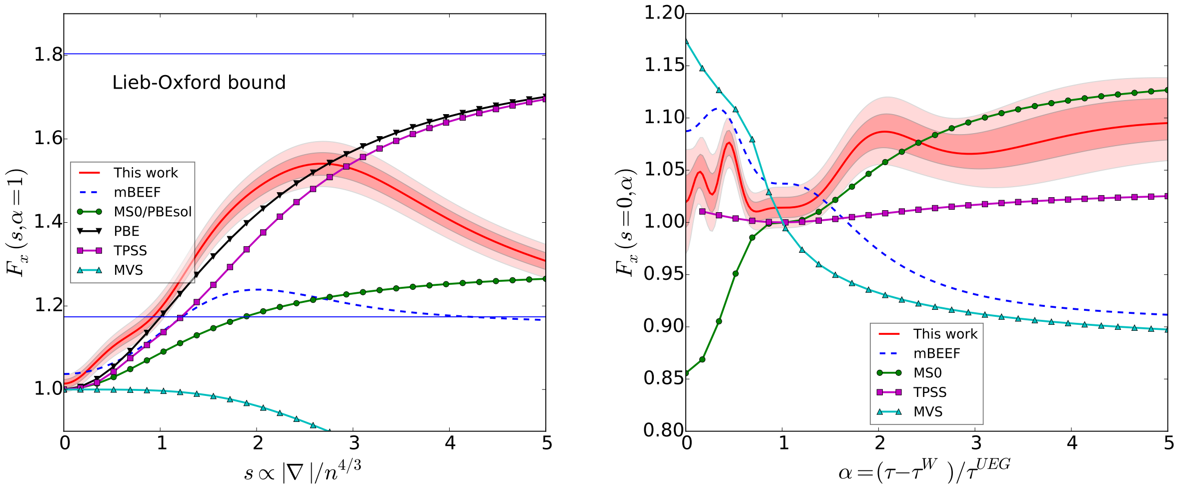


Figure 4: Exchange enhancement factors for the model developed in this work, and a series of other GGA (PBE, PBEsol) and meta-GGA (mBEEF, MS0, MVS and TPSS) functionals. The shaded regions correspond to one and two standard deviations around the average model. The left panel shows the projection on s for $\alpha = 1$ and the right panel the projection on α for $s = 0$.

s for a fixed value of $\alpha = 1$ and as a function of the reduced kinetic energy density α for $s = 1$. We also include the confidence intervals from coming from the distribution of parameters in Eq. (24), and other GGA (PBE and PBEsol) and meta-GGA (mBEEF, MS0, MVS [22] and TPSS) functionals.

Remark 10. The flat section around $\alpha = 1$ in the functional developed in this work, mBEEF and MS0 comes from the common α dependence, Eq. (16), in all of them. This property is also shared by the TPSS functional. The use of an α dependence term without zero slope at $\alpha = 1$ could allow for more flexibility in the s dependence as shown in [22] for the "Made Very Simple" (MVS) functional.

5.1. Numerical results

The training process provided us with a predictive distribution for the exchange contribution to the atomization energy of any system outside the training set. For any new system with electron density $\tilde{n}$, the predictive distribution is

$$p(\tilde{E^x} \mid \tilde{n}, \mathbf{t}) = \mathcal{S}t(\tilde{E^x} \mid \mu, \lambda, \nu), \tag{54}$$

with the parameters defined in Eqs. (31)-(33). These equations show that to obtain the parameters of the predictive distribution for the new system, we need the basis exchange energy vector $\mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]$. As we did for the training of the model, we run the simulation self-consistently using the PBE functional and construct the vector $\mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]$ using the resulting density. This is then used to calculate the parameters of the Student t-distribution of the exchange energy from Eqs. (31)-(33).

Remark 11. ***In the training of the model, we used the self-consistent PBE densities, i.e., the densities obtained solving Eqs.*** (3)-(5) ***with the PBE XC energy functional to evaluate the basis functions (Remark 3). As a posteriori check that this is a reasonable approximation, we compare for a few systems the DFT energy using our XC functional with two different densities: the self-consistent density obtained using the PBE XC functional as described above, and the self-consistent density obtained with our trained model. In the first case we run a self-consistent DFT simulation using the PBE functional and keep the resulting density,*** n_{PBE}***. This density is then fixed and used to obtain the average prediction of our model running a non self-consistent DFT simulation using Eq.*** (31) ***with*** $\tilde{n} = n_{PBE}$ ***as the exchange energy functional. We will denote this energy as*** $E_{nsc-PBE}$***. In the second case, we run a***

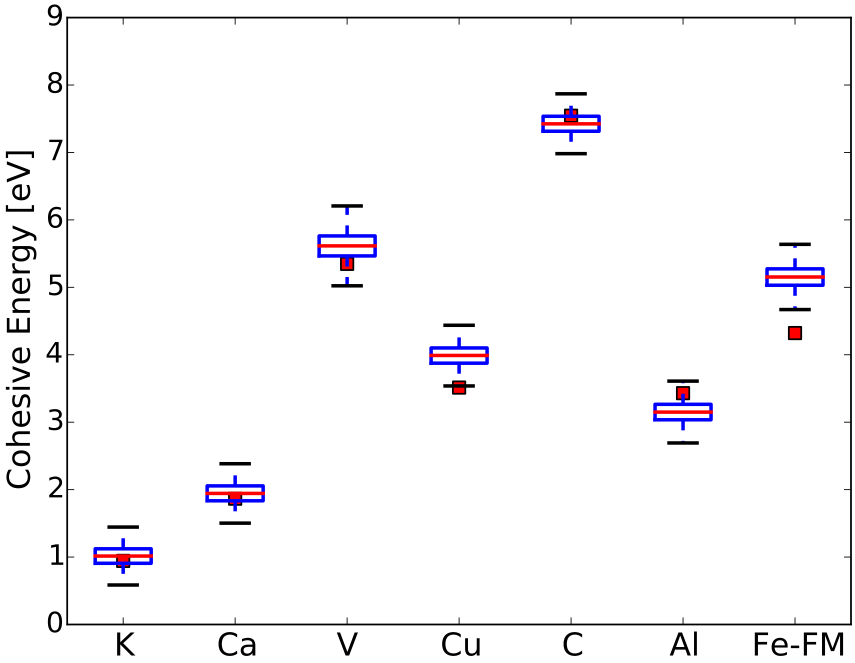


Figure 5: Box-plot of the cohesive energies of the elements in the test set. The red squares represent the experimental cohesive energies.

self-consistent DFT simulation using our average model XC energy, i.e., Eq. (18) ***with ξ given by Eq.*** (26)***. We will denote this energy as E_{sc}. We found that the absolute difference between $E_{nsc-PBE}$ and E_{sc} was below*** 1 ***meV, which is lower than the typical energy resolution in DFT applications.***

To evaluate the quality of the average predictions and compare it to other values found in the literature, we use the mean absolute error (MAE) and the mean absolute relative error (MARE). For a set of calculated data $\mathbf{x}^{calc} = \{x_i^{calc}\}$ and the corresponding set of experimental data $\mathbf{x}^{exp} = \{x_i^{exp}\}$, the MAE and MARE of the calculations are defined as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |x_i^{calc} - x_i^{exp}|, \tag{55}$$

$$MARE = \frac{1}{N}\sum_{i=1}^{N} \left|\frac{x_i^{calc} - x_i^{exp}}{x_i^{exp}}\right|. \tag{56}$$

Figure 5 shows a box-plot of the distributions of cohesive energies of the elements in the EL20 test set together with the experimental values. Except for Fe, all experimental points fall within the confidence interval given by our model. Fe is a magnetic material, which is a family not represented in our training set and therefore we can expect that the results for it will not be predictive. Also, we must bear in mind that for some magnetic elements the thermal extrapolations to 0 K used on the experimental data are no longer valid [7].

Remark 12. ***As discussed on Section 3.1, the predictions from the model are correlated. As an example of this effect, we calculate the uncertainty in the difference between the cohesive energies of K and Ca. The predicted values for both materials are*** 1.015 ± 0.165 ***and*** 1.945 ± 0.166 ***eV, respectively. The cohesive energy difference ignoring correlations, i.e., just subtracting the two random variables as obtained from Eq.*** (30)***, is*** 0.930 ± 0.234 ***eV, whereas if we include correlations, i.e., subtracting the two correlated variables as obtained from Eq.*** (37)***, it is*** 0.930 ± 0.232 ***eV. In this case, the difference is very small since the model error, which we assumed uncorrelated, dominates over the variability of the coefficients.***

XC functional	Error	G2/97-test	G2/97	EL20-test	EL20
This work	MAE	0.116	0.103	0.243	0.0975
	MARE	3.27	1.46	8.56	5.62
PBE	MAE		0.703		0.238
	MARE		5.09		6.88

Table 1: Mean absolute error (in eV) and mean absolute relative error (in %) of the predictions of atomization energies using the average model for the training sets containing molecules (G2/97) and solids (EL20).

C functional	Error	G2/97-test	G2/97	EL20-test	EL20
PBE	MAE	0.116	0.103	0.243	0.0975
	MARE	3.27	1.46	8.56	5.62
PBEsol	MAE	0.116	0.108	0.204	0.172
	MARE	2.91	1.55	6.12	4.98
vPBE	MAE	0.110	0.107	0.226	0.184
	MARE	2.72	1.41	6.45	5.17
TPSS	MAE	0.108	0.104	0.227	0.190
	MARE	2.68	1.42	6.85	5.53

Table 2: Mean absolute error (in eV) and mean absolute relative error (in %) of the predictions of atomization energies using the average model with different correlation functionals.

Table 1 summarizes the MAE and MARE of the atomization energies of the elements in the EL20 and G2/97 data sets and compares them to the ones obtained with the PBE functional. The MAE for the G2/97 data set goes down from 0.703 to 0.103 eV, which was partly expected as the PBE functional does not work particularly well with molecules [4, 22]. Furthermore, our results are better than those of BEEF-vdW (0.16 eV, GGA with van de Waals corrections), TPSS (0.28 eV) or the hybrid functionals B3LYP (0.14 eV) and PBE0 (0.21 eV) [4]. On the other hand, the performance in our solids data set shows very similar results for both functionals.

To further study the predictive capabilities of the functional, we tested it on 37 molecules from the G3-3 subset of the G3/99 data set [37]. The MAE and MARE were found to be 0.0608 eV and 0.11%, respectively. Even though it is only half of the complete G3-3 set, the MAE is less than half of the best reported in Ref. [4] for the whole data set, including LDA, GGA, meta-GGA and hybrid exchange correlation functionals.

5.2. Impact of different correlation functionals

Since the correlation part of the functional is not trained, we tried four different ones to see the impact on the results: E^c_{PBE}, E^c_{PBEsol}, E^c_{vPBE} [16, 22] and E^c_{TPSS}. Table 2 shows a comparison of the errors using the three correlations. Even though the coefficients selected by the RVM are different, as shown in Fig. 6, the error in the predictions is similar. The vPBE correlation seems to give the best results for the molecules in the test set whereas PBEsol is the best for the test set of solids, even though both of them are outperformed by the PBE correlation if training solids are also included.

6. Propagation of Uncertainty to Derived Quantities

6.1. Lattice constant and bulk modulus of cubic materials

For cubic materials, where the volume depends on only one parameter, we can easily obtain the equilibrium lattice constant and bulk modulus from a fit of computed energies at different volumes to an EOS. We use again the SJEOS as defined in Eq. (51),

$$E(V) = a + b\frac{V_0^{1/3}}{V^{1/3}} + c\frac{V_0^{2/3}}{V^{2/3}} + d\frac{V_0}{V} = \boldsymbol{\gamma}^T\boldsymbol{\phi}(V). \tag{57}$$

The regression coefficients are related to the equilibrium energy (E_0), equilibrium volume (V_0), bulk modulus (B_0) and first derivative of bulk modulus (B_1) through Eq. (52).

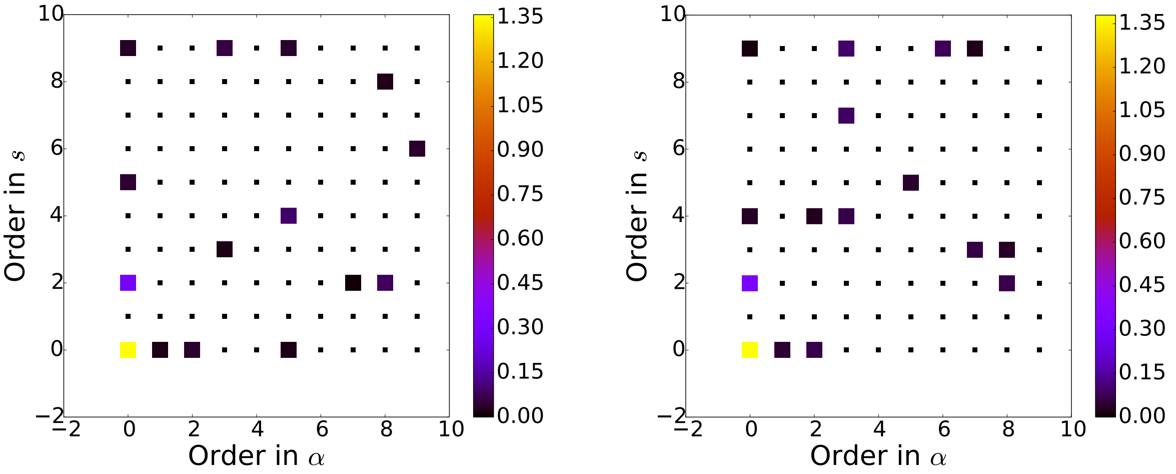


Figure 6: Magnitude of the coefficients obtained using a RVM for the determination of the hyperparameters using the PBEsol (left) and vPBE (right) correlation functionals.

To obtain their values with their associated uncertainty we proceed as follows. For each material, we run a self-consistent simulation for a set of different strains (5 points from 95% to 105% of the experimental volume) using our XC functional defined in Eq. (17) with the average values for the coefficients as defined in Eq. (26). We draw samples from the distribution in Eq. (24), and run simulations non self-consistently using the density from the self-consistent simulations. For each sample coefficients, we fit the resulting energies to the SJEOS.

Once again we use Bayesian linear regression for the process, but this time we take the error in the observed quantity (the DFT energy in this case) as given. This is related to the energy accuracy, e.g., from convergence in mesh spacing, k-points, energy cut-off, etc., and adds an extra source of uncertainty. If we assume it to be Gaussian, for each sample of XC functional coefficients the likelihood is [26]:

$$\mathcal{L}(E \mid V, \boldsymbol{\gamma}, \beta) = \prod_n \mathcal{N}(E_n \mid \boldsymbol{\gamma}^T \boldsymbol{\phi}(V_n), \delta^{-1}) = \mathcal{N}(\mathbf{E} \mid \boldsymbol{\gamma}^T \boldsymbol{\phi}, \delta^{-1}\mathbf{I}), \tag{58}$$

where δ represents the noise in the data.

Using a Gaussian as a prior for the regression coefficients, $p(\boldsymbol{\gamma}) = \mathcal{N}(\boldsymbol{\gamma} \mid \mathbf{m}_0, \mathbf{S}_0)$, the posterior for the coefficients is again a Gaussian [26],

$$p(\boldsymbol{\gamma} \mid \mathbf{E}) = \mathcal{N}(\boldsymbol{\gamma} \mid \mathbf{m}_N, \mathbf{S}_N), \tag{59}$$

where now the mean and covariance of the posterior distribution over the parameters $\boldsymbol{\gamma}$ are given by [26]:

$$\mathbf{m}_N = \mathbf{S}_N \left\{ \mathbf{S}_0^{-1} + \delta \boldsymbol{\Phi}^T \mathbf{E} \right\}, \tag{60}$$

$$\mathbf{S}_N = \mathbf{S}_0^{-1} + \delta \boldsymbol{\Phi}^T \boldsymbol{\Phi}. \tag{61}$$

Using a Monte Carlo method once more, we sample the regression coefficients and invert Eq. (52) to obtain E_0, V_0, B_0 and B_1. Keeping the values for every sample, we obtain a distribution for these quantities with combined sources of uncertainty: model inaccuracy, limited data and numerical accuracy. This procedure is illustrated in Algorithm 3:

Algorithm 3 Calculation of uncertainty for V_0 and B_0.

```
Input: system $s$ with unit cell defined by three vectors $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$.
Input: $N_1^{max}$, $N_2^{max}$, the maximum number of iterations for Monte Carlo sampling.
for 5 strains $0.95 \leq \sigma_i \leq 1.05$ do
    Strain the unit cell of the system $s$ by $\sigma_i$: $\mathbf{x}_\alpha \rightarrow \sigma_i \mathbf{x}_\alpha$, $\alpha = 1, 2, 3$.
    Self-consistent simulation of the strained system using $\boldsymbol{\xi} = \mathbf{m}_N$.
    Keep the self-consistent electron density $n_i^* = n(\sigma_i)$.
end for
$N_1 = 0$
repeat
    Sample $\boldsymbol{\xi}_{N_1}, \beta_{N_1}$ from Eq. (24).
    Non self-consistent simulation using $\boldsymbol{\xi}_{N_1}, \beta_{N_1}$ using a fixed density $n_i^*$.
    $N_2 = 0$
    repeat
        Sample $\boldsymbol{\gamma}_{N_2}$ from Eq. (59).
        Calculate $V_0$, $B_0$ inverting Eq. (52).
    until $N_2 = N_2^{max}$
    $N_1 = N_1 + 1$
until $N_1 = N_1^{max}$
Collect statistics on calculated $V_0$, $B_0$.
```

6.2. Prediction of bulk properties

To test the bulk properties we use the SL20 test set [34] since there are available data for other XC functional to compare the performance [34, 22]. It consists of 13 elemental solids (Li, Na, Ca, Sr, Ba, Al, Cu, Rh, Pd, Ag, C, Si, Ge) and binary I-VII (LiF, LiCl, NaF, NaCl), II-VI (MgO), III-V (GaAs) and IV-IV (SiC) compounds. Note that 7 of the elemental solids in the set were used in the training of the model.

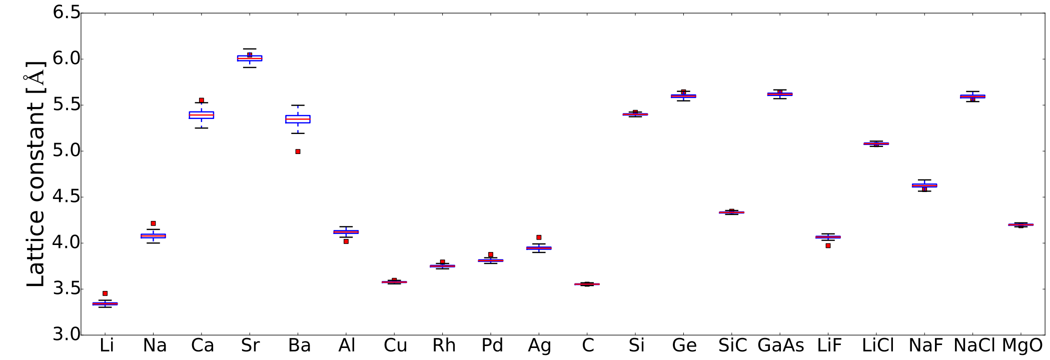
Figure 7(a) shows a box-plot of the calculated lattice constants for the elements in the set. Among the elemental solids, we see that the predictive error bars are largest for the group II elements (Ca, Sr, Ba), which together with Li have the largest absolute errors. Among the compounds, LiF has the largest error, but bearing in mind that no ionic solids were included in the training set the predictions are remarkably good, especially when poor results have been reported before with other functionals for this family of elements [38, 34].

The MAE and MARE of the predicted average lattice constants are 0.072 Å and 1.60%, respectively. The MAE is worse than other solid oriented density functionals such as PBEsol (0.036 Å) [22] and comparable to chemistry oriented semi-local functionals such as M06-L, which has a MAE of 0.071 Å, but without Ca, Sr and Ba [38, 16], which contribute importantly to the error with our functional. The mean signed error (MSE) we obtain is −0.0081 Å, which implies on average an underestimation of the lattice constants similar to the PBEsol functional but opposite to other meta-GGA functionals such as TPSS [34].

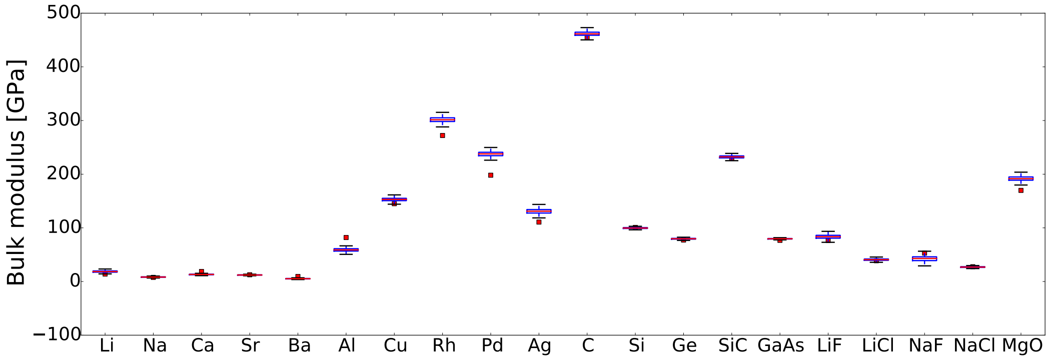
Figure 7(b) shows a box-plot of the predicted bulk modulus for the materials in the SL20 test set. The MAE and MARE of the predicted average bulk moduli are 9.71 GPa and 13.07%, respectively. These values are similar to those for other functionals such as, e.g., PBE (10.5 GPa) or TPSS (7.942 GPa) [34]. The MSE of 5.19 GPa, means that on average there is an overestimation of the bulk moduli the same as the PBE functional but opposite to the PBEsol or TPSS functionals [34].

6.2.1. Impact of the simulation convergence error

As described in Section 6, we can add an extra level of noise to the fitting to account for some level of uncertainty in the calculated energies. Even though it does not affect the average value of the calculated properties, a higher noise will increase the uncertainty and decrease the confidence in its value. Figure 8 shows the calculations of the lattice constants and bulk moduli after setting a numerical error of 10 meV, which reflects, for example, our confidence in the convergence of the simulation. There is a clear widening of the error bars, especially in the results for the elements with higher absolute errors, group I and II elements for the lattice constants and bulk moduli for the transition metals.

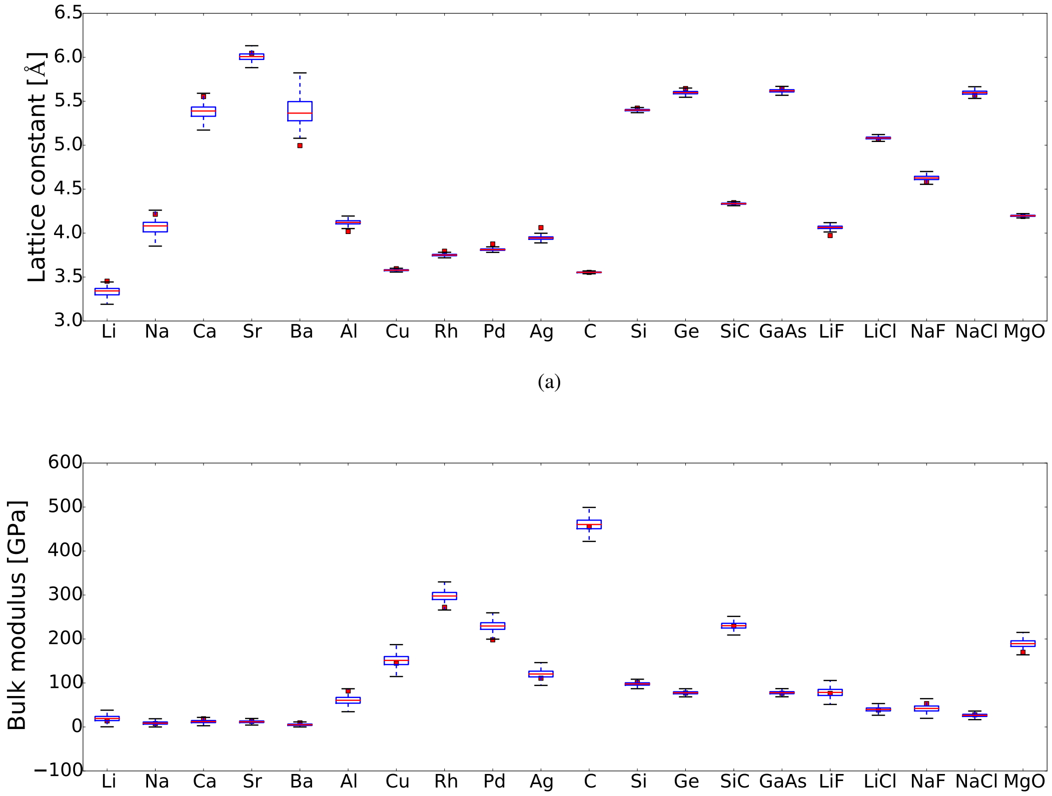


(a)



(b)

Figure 7: Box-plot of the (a) equilibrium lattice constants and (b) equilibrium bulk moduli of the elements in the SL20 test set [34]. Experimental lattice constants are corrected to static-lattice values subtracting the zero-point anharmonic expansion (ZPAE) and experimental bulk moduli are corrected for the zero-point phonon effects (ZPPE).



(a)

(b)

Figure 8: Same as Fig. 7 but with an added numerical noise of 10 meV to the regression problem to calculate the SJEOS parameters.

The widening of the error bars for the materials with high error is a proof of the ability of the proposed functional to predict the uncertainty in its predictions.

6.3. Prediction of the energy band gap of Si

As a further example of uncertainty propagation we show the calculation of the energy band gap for a semiconductor, Si. Kohn-Sham DFT cannot reproduce the band gap of materials properly as a consequence of the lack of a discontinuity in the XC energy functional with the number of electrons [39]. This can be overcome with the introduction of energy dependent XC potential [39]. An equivalent to this energy dependent XC potential can be achieved using many-body perturbation theory (MBPT), which provides a different approach to obtain the band gap. Its first order approximation, Hedin's GW approximation [40] is already a more accurate approach to tackle the band gap problem. In this framework, XC effects are included in the energy dependent *self-energy*, which is a convolution of the Green's function G and a dynamically screened Coulomb interaction W. The obtained energies correspond to *quasiparticles* (QP) describing the screened electrons and in this case the valence band maximum and conduction band minimum can be interpreted directly as the ionization potential and electron affinity in photoemission experiments and therefore can be used to calculate the experimentally measured band gap [41].

One further approximation which has been successful in calculating the band-gaps for small gap semiconductors is the G_0W_0 approximation [42, 41], which calculates the GW QP energies perturbatively on top of the one-particle Kohn-Sham (KS) orbitals and orbital energies. Therefore, in this approximations, the QP energy spectrum is directly linked to the DFT starting point. The eigenvalues obtained as perturbations to the Kohn-Sham (KS) orbital energies are

$$\varepsilon_{n\mathbf{k}}^{GW} = \varepsilon_{n\mathbf{k}}^{KS} + \left\langle \psi_{n\mathbf{k}}^{KS} \mid \Sigma - V_{xc} \mid \psi_{n\mathbf{k}}^{KS} \right\rangle, \tag{62}$$

where $\varepsilon_{n\mathbf{k}}$ and $\psi_{n\mathbf{k}}$ are the orbital energies and orbitals for band n and wave vector $\mathbf{k}$, Σ is the self-energy from the G_0W_0 approximation and V_{xc} the exchange correlation energy from DFT. All calculations are done as implemented in GPAW [43]. Since the correction depends on the KS orbitals and the XC potential, the results will depend on which XC functional is used as the initial approximation, and we will use this a measure of uncertainty for the $G_0W_0/meta-GGA$ approximation to the band structure. For each realization of our XC functional we obtain self-consistently the orbitals and their energies on a $12 \times 12 \times 12$ Monkhorst-Pack mesh and from them the G_0W_0 eigenvalues using Eq. (62). We estimate the band gap as the distance between the maximum of the highest occupied band and the minimum of the lowest unoccupied band. The variability in the band gap computed this way is shown in Fig. 9 together with the band gaps obtained directly from DFT. We can see that both approaches have a similar absolute value of the uncertainty in the band gap, even though the values obtained with the G_0W_0 approximation are, as expected, much more accurate.

7. Summary and Conclusions

We have presented a new approach based on machine learning using a Bayesian framework to obtain an exchange-correlation energy functional. In this way, the coefficients of the functional are not point estimates, but random variables, so that the resulting exchange-correlation functional is also a random variable, even though the model exchange energy basis are fixed. Imposing certain assumptions in the training process, we obtained an analytical expression for the distribution function of the model parameters. Having a random variable instead of a point estimate allows for the quantification of uncertainty in the simulation results. Uncertainties in the predictions will include limited data uncertainty, i.e., uncertainty in the training process due to the availability of limited training data, and model uncertainty, i.e., uncertainty due to the inability of the proposed model to reproduce the experimental data. Limited data uncertainty could in principle be reduced asymptotically to zero as the number of training data increases. However, our model uncertainty would not vanish in the limit of infinite basis functions as the framework we are using (meta-GGA XC functional) is intrinsically limited.

Whereas previous approaches using empirical models for the XC energy, i.e., models which fit their coefficients to experimental data, have used a fixed functional dependence, which limits the number of basis functions as they lend themselves to overfitting, we use a relevance vector machine. This means that during the training process there is an automatic model selection trough the sparsification of the model coefficients, which reduces the risk of overfitting.

The training of the linear model has been done using atomization energies to keep the linearity between the trained quantity and the exchange enhancement factor model coefficients. We have tested this model on atomization energies

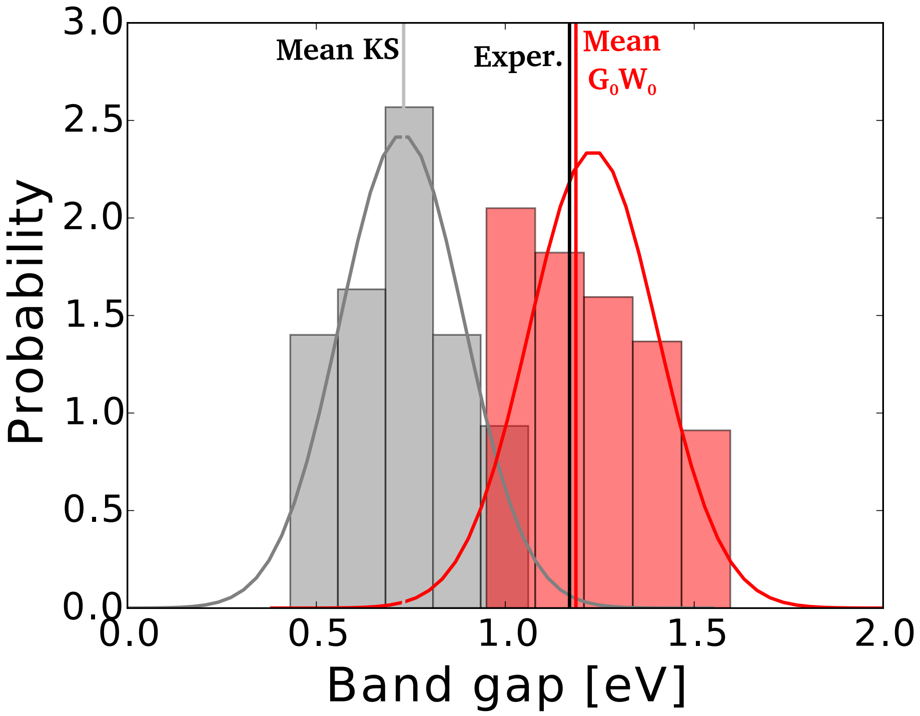


Figure 9: Histograms of the band gap of Si using KS-DFT (grey) and the G_0W_0 approximation (red) with our XC functional. Gaussian fits are also shown as a guide to the eye. The black vertical line corresponds to the experimental value, the red vertical line to the G_0W_0 band gap with the average XC functional and the grey vertical line the KS band gap with the average XC functional.

and also on bulk properties. The average model has shown very good performance for molecule atomization energies, with a mean absolute error of only 0.116 eV for the test points of the G2/97 set (28 out of the 148 points), which is better than hybrid functionals such as B3LYP or PBE0. The error for solids using the SL20 data set has been found to be larger, 0.243 eV, but comparable to the performance of the PBE functional. In terms of bulk properties, the prediction of lattice constants for transition metals and semiconductors has a very low error. However, as expected from the limitations of the method, predictions for types of materials poorly or not represented in the train set such as group I and II elements or ionic solids show larger errors and an increase in the training data will be necessary to improve the prediction capabilities for other types of materials. Finally, we also showed the propagation of uncertainty in the model coefficients to the band gap of Si. Since Kohn-Sham DFT is not appropriate for band gap calculations, we used the G_0W_0 quasi-particle approximation which allowed a good agreement between our average model and the experimental value.

Acknowledgments

This work was supported by EPSRC (Grant No. EP/L027682/1). Additional support from the Computer Science and Mathematics Division of ORNL under the DARPA EQUiPS program is acknowledged. N.Z. also thanks the Royal Society for support through a Wolfson Research Merit Award and the Technische Universität München, Institute for Advanced Study for support through a Hans Fisher Senior Fellowship, funded by the German Excellence Initiative and the European Union Seventh Framework Program under Grant Agreement No. 291763.

Appendix A. Derivation of the posterior distribution over the parameters

The starting point of the calculation is the definition of the posterior parameter distribution in Eq. (24),

$$p(\boldsymbol{\xi},\beta \mid \mathbf{t}) = \frac{\mathcal{L}(\mathbf{t} \mid \mathbf{n},\boldsymbol{\xi},\boldsymbol{\beta})p(\boldsymbol{\xi},\beta)}{\int \mathcal{L}(\mathbf{t} \mid \mathbf{n},\boldsymbol{\xi},\boldsymbol{\beta})p(\boldsymbol{\xi},\beta)\, d\boldsymbol{\xi}\, d\beta}. \tag{A.1}$$

Using Eqs. (22) and (23), the expression for the posterior is

$$p(\boldsymbol{\xi},\beta \mid \mathbf{t}) \propto \prod_{n=1}^{N} \mathcal{N}(t_n \mid \boldsymbol{\xi}^T \mathbf{E}^x[n_n;\hat{\mathbf{e}}],\beta^{-1})\mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_0,\beta^{-1}\mathbf{S}_0)\mathcal{G}(\beta \mid a_0,b_0). \tag{A.2}$$

Taking the logarithm of this equation, we obtain the following

$$\begin{aligned}\log p(\boldsymbol{\xi},\beta \mid \mathbf{t}) =& C_0 + \sum_{n=1}^{N} \log \mathcal{N}(t_n \mid \boldsymbol{\xi}^T \mathbf{E}^x[n_n;\hat{\mathbf{e}}],\beta^{-1}) \\ &+ \log \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_0,\beta^{-1}\mathbf{S}_0) + \log \mathcal{G}(\beta \mid a_0,b_0),\end{aligned} \tag{A.3}$$

where C_0 is a constant.

We can calculate these terms using the definitions of the Gaussian and Gamma distributions,

$$\mathcal{N}(\mathbf{x} \mid \mu,\mathbf{S}) = \frac{1}{(2\pi)^{D/2}|\mathbf{S}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{S}^{-1}(\mathbf{x}-\mu)\right\}, \tag{A.4}$$

$$\mathcal{G}(x \mid a,b) = \frac{b^a x^{a-1} e^{-bx}}{\Gamma(a)}, \tag{A.5}$$

where D is the dimensionality of the Gaussian distribution and $\Gamma(a)$ is the gamma function evaluated at a.

Introducing the above equations back into Eq. (A.3) leads to the following:

$$\begin{aligned}\log p(\boldsymbol{\xi},\beta \mid \mathbf{t}) =& C_0 + \frac{N}{2}(\log\beta - \log 2\pi) + \sum_{n=1}^{N} \left\{\boldsymbol{\xi}^T \mathbf{E}^x[n_n;\hat{\mathbf{e}}] - t_n\right\}^2 \\ &+ \frac{M}{2}(\log\beta - \log 2\pi) - \frac{1}{2}\log|\mathbf{S}_0| - \frac{\beta}{2}(\boldsymbol{\xi}-\mathbf{m}_0)^T \mathbf{S}_0^{-1}(\boldsymbol{\xi}-\mathbf{m}_0) \\ &+ (a_0-1)\log\beta - b_0\beta + a_0\log b_0 - \log\Gamma(a_0).\end{aligned} \tag{A.6}$$

Grouping all the terms dependent on $\boldsymbol{\xi}$,

$$\begin{aligned}\log p(\boldsymbol{\xi},\beta \mid \mathbf{t}) &= C_1(\beta, a_0, b_0) + \sum_{n=1}^{N} \left\{\boldsymbol{\xi}^T \mathbf{E}^x[n_n; \hat{\mathbf{e}}] - t_n\right\}^2 - \frac{\beta}{2}(\boldsymbol{\xi} - \mathbf{m}_0)^T \mathbf{S}_0^{-1}(\boldsymbol{\xi} - \mathbf{m}_0) \\ &= C_1(\beta, a_0, b_0) - \frac{\beta}{2}(\Phi\boldsymbol{\xi} - \mathbf{t})^T(\Phi\boldsymbol{\xi} - \mathbf{t}) - \frac{\beta}{2}(\boldsymbol{\xi} - \mathbf{m}_0)^T \mathbf{S}_0^{-1}(\boldsymbol{\xi} - \mathbf{m}_0), \end{aligned} \tag{A.7}$$

where we have introduced the design matrix Φ defined in Eq. (29), and a function $C_1(\beta, a_0, b_0)$ which does not depend on $\boldsymbol{\xi}$. We can observe that the dependence is quadratic and we can complete the square, which yields

$$\begin{aligned}\log p(\boldsymbol{\xi},\beta \mid \mathbf{t}) = &-\frac{\beta}{2}\boldsymbol{\xi}^T \left(\Phi^T\Phi + \mathbf{S}_0^{-1}\right)\boldsymbol{\xi} + \boldsymbol{\xi}^T \left(\beta\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\Phi^T\mathbf{t}\right) \\ &+ \frac{\beta}{2}\left(\mathbf{t}^T\mathbf{t} + \mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0\right) + C_1(\beta, a_0, b_0). \end{aligned} \tag{A.8}$$

The first two terms plus the term $\frac{M}{2}\log\beta$ in $C_1(\beta, a_0, b_0)$ define, up to a constant, the logarithm of a Gaussian distribution with covariance matrix

$$\beta^{-1}\mathbf{S}_N = \beta^{-1}\left(\mathbf{S}_0^{-1} + \Phi^T\Phi\right)^{-1}, \tag{A.9}$$

and mean

$$\mathbf{m}_N = \mathbf{S}_N\left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \Phi^T\mathbf{t}\right), \tag{A.10}$$

$$\begin{aligned}\log p(\boldsymbol{\xi},\beta \mid \mathbf{t}) = &-\frac{\beta}{2}(\boldsymbol{\xi} - \mathbf{m}_N)^T\mathbf{S}_N^{-1}(\boldsymbol{\xi} - \mathbf{m}_N) \\ &- \frac{\beta}{2}\left(\mathbf{t}^T\mathbf{t} + \mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0 - \mathbf{m}_N^T\mathbf{S}_N^{-1}\mathbf{m}_N\right) + C_1(\beta, a_0, b_0). \end{aligned} \tag{A.11}$$

Turning our attention to the last two terms in Eq. (A.11) minus the term $\frac{M}{2}\log\beta$ leads to the following:

$$\begin{aligned}&C_1(\beta, a_0, b_0) + \frac{\beta}{2}\left(\mathbf{t}^T\mathbf{t} + \mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0 - \mathbf{m}_N^T\mathbf{S}_N^{-1}\mathbf{m}_N\right) - \frac{M}{2}\log\beta \\ &= C_0 + \frac{N}{2}(\log\beta - \log 2\pi) - \frac{M}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{S}_0| \\ &+ \frac{\beta}{2}\left(\mathbf{t}^T\mathbf{t} + \mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0 - \mathbf{m}_N^T\mathbf{S}_N^{-1}\mathbf{m}_N\right) \\ &+ (a_0 - 1)\log\beta - b_0\beta + a_0\log b_0 - \log\Gamma(a_0) \\ &= C_2 + \left(a_0 + \frac{N}{2} - 1\right)\log\beta - \left[b_0 + \frac{1}{2}\left(\mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0 - \mathbf{m}_N^T\mathbf{S}_N^{-1}\mathbf{m}_N + \mathbf{t}^T\mathbf{t}\right)\right]\beta, \end{aligned} \tag{A.12}$$

where C_2 is a constant. The last two terms define, up to a constant, the logarithm of a Gamma distribution with parameters

$$a_N = a_0 + \frac{N}{2}, \tag{A.13}$$

$$b_N = b_0 + \frac{1}{2}\left(\mathbf{m}_0^T\mathbf{S}_0^{-1}\mathbf{m}_0 - \mathbf{m}_N^T\mathbf{S}_N^{-1}\mathbf{m}_N + \mathbf{t}^T\mathbf{t}\right). \tag{A.14}$$

Putting these two terms together, we obtain the following expression for the logarithm of the posterior distribution:

$$\log p(\boldsymbol{\xi},\beta \mid \mathbf{t}) = C_3 + \log\mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N) + \log\mathcal{G}(\beta \mid a_N, b_N). \tag{A.15}$$

Taking the exponential in Eq. (A.15) and adjusting for normalization of the posterior probability,

$$p(\boldsymbol{\xi},\beta \mid \mathbf{t}) = \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\mathcal{G}(\beta \mid a_N, b_N), \tag{A.16}$$

with $\mathbf{S}_N$, $\mathbf{m}_N$, a_N and b_N defined in Eqs. (A.9), (A.10), (A.13) and (A.14), respectively.

Appendix B. Derivation of the predictive distribution

To obtain the predictive distribution, one needs to calculate the integral

$$
\begin{aligned}
p(\tilde{t} \mid \tilde{n}, \mathbf{t}) &= \int p(\tilde{t} \mid \tilde{n}, \boldsymbol{\xi}, \beta) p(\boldsymbol{\xi}, \beta \mid \mathbf{t})\, d\boldsymbol{\xi}\, d\beta \\
&= \int \mathcal{N}(\tilde{t} \mid \boldsymbol{\xi}^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1}) \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N) \mathcal{G}(\beta \mid a_N, b_N)\, d\boldsymbol{\xi}\, d\beta \\
&= \int \mathcal{N}(\tilde{t} \mid \boldsymbol{\xi}^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1}) \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\, d\boldsymbol{\xi}\, \mathcal{G}(\beta \mid a_N, b_N)\, d\beta. \qquad \text{(B.1)}
\end{aligned}
$$

We first perform the integral on $\boldsymbol{\xi}$,

$$
\int \mathcal{N}(\tilde{t} \mid \boldsymbol{\xi}^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1}) \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\, d\boldsymbol{\xi}. \qquad \text{(B.2)}
$$

After some algebra, it can be shown [26] that the result is another Gaussian distribution on $\tilde{t}$,

$$
\mathcal{N}\left(\tilde{t} \mid \mathbf{m}_N^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1}\left(1 + \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]^T \mathbf{S}_N \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]\right)\right). \qquad \text{(B.3)}
$$

Substituting Eq. (B.3) in Eq. (B.1) leads to the following:

$$
\begin{aligned}
p(\tilde{t} \mid \tilde{x}, \mathbf{t}) &= \int \mathcal{N}(\tilde{t} \mid \boldsymbol{\xi}^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1}) \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N)\, d\boldsymbol{\xi}\, \mathcal{G}(\beta \mid a_N, b_N)\, d\beta \\
&= \int \mathcal{N}\left(\tilde{t} \mid \mathbf{m}_N^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], \beta^{-1}\left(1 + \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]^T \mathbf{S}_N \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]\right)\right) \mathcal{G}(\beta \mid a_N, b_N)\, d\beta. \qquad \text{(B.4)}
\end{aligned}
$$

The result of this integral is a Student t-distribution [26], defined as

$$
\mathcal{S}t(\tilde{t} \mid \mu, \lambda, \nu) = \frac{\Gamma(1/2 + \nu/2)}{\Gamma(\nu/2)} \frac{\lambda^{1/2}}{(\pi\nu)^{1/2}} \left[1 + \frac{\lambda(\tilde{t} - \mu)^2}{\nu}\right]^{-1/2-\nu/2}, \qquad \text{(B.5)}
$$

with

$$
\begin{aligned}
\mu &= \mathbf{m}_N^T \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}], && \text{(B.6)} \\
\lambda &= \frac{a_N}{b_N}\left(1 + \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]^T \mathbf{S}_N \mathbf{E}^x[\tilde{n}; \hat{\mathbf{e}}]\right)^{-1}, && \text{(B.7)} \\
\nu &= 2a_N. && \text{(B.8)}
\end{aligned}
$$

Appendix C. Derivation of the evidence function

The marginal likelihood $\mathcal{L}_M$ can be computed by the integration over the model parameters of the likelihood times the prior:

$$
\mathcal{L}_M(\mathbf{t} \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \int \mathcal{L}(\mathbf{t} \mid \boldsymbol{\xi}, \beta) p(\boldsymbol{\xi}, \beta \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0)\, d\boldsymbol{\xi}\, d\beta. \qquad \text{(C.1)}
$$

The expression inside the integral is proportional to the posterior distribution over the parameters $\boldsymbol{\xi}$ and β:

$$
\begin{aligned}
&\int \mathcal{L}(\mathbf{t} \mid \boldsymbol{\xi}, \beta) p(\boldsymbol{\xi}, \beta \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0)\, d\boldsymbol{\xi}\, d\beta \\
&= \int \sqrt{\frac{1}{(2\pi)^N} \frac{|\mathbf{S}_N|}{|\mathbf{S}_0|} \frac{b_0^{a_0}}{\Gamma(a_0)} \frac{\Gamma(a_N)}{b_N^{a_N}}} \mathcal{N}(\boldsymbol{\xi} \mid \mathbf{m}_N, \beta^{-1}\mathbf{S}_N) \mathcal{G}(\beta \mid a_N, b_N)\, d\boldsymbol{\xi}\, d\beta \\
&= \sqrt{\frac{1}{(2\pi)^N} \frac{|\mathbf{S}_N|}{|\mathbf{S}_0|} \frac{b_0^{a_0}}{\Gamma(a_0)} \frac{\Gamma(a_N)}{b_N^{a_N}}}. \qquad \text{(C.2)}
\end{aligned}
$$

Taking the logarithm of this function, we obtain the evidence function as shown in Eq. (40),

$$\begin{aligned}\mathcal{E}(\mathbf{m}_0, \mathbf{S}_0, a_0, b_0) &= \log \mathcal{L}_M(\mathbf{t} \mid \mathbf{m}_0, \mathbf{S}_0, a_0, b_0) \\ &= \frac{1}{2} \log \frac{|\mathbf{S}_N|}{|\mathbf{S}_0|} - \frac{N}{2} \log(2\pi) + \log \frac{\Gamma(a_N)}{\Gamma(a_0)} + a_0 \log(b_0) - a_N \log(b_N).\end{aligned} \tag{C.3}$$

To find a maximum of the evidence function, we also need the derivatives with respect to all of its parameters, $\mathbf{m}_0$, $\mathbf{S}_0$, a_0 and b_0. For the relevance vector machine, we assumed $\mathbf{S}_0^{-1} = \operatorname{diag}(\alpha_0, \ldots, \alpha_{M-1})$ and $\mathbf{m}_0 = 0$. In this case, we need derivatives with respect to α_i, a_0 and b_0. Using Jacobi's formula for the derivative of a determinant [44] and the definitions of a_N and b_N in Eqs. (A.13) and (A.14), we obtain

$$\frac{\partial}{\partial \alpha_i} \mathcal{E}(\mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \frac{1}{2} \left[\frac{1}{\alpha_i} - (\mathbf{S}_N)_{ii} - \frac{a_N}{b_N} (\mathbf{m}_N)_i^2 \right], \tag{C.4}$$

where $(\mathbf{S}_N)_{ii}$ and $(\mathbf{m}_N)_i$ are the i-th entries of the diagonal of $\mathbf{S}_N$ and $\mathbf{m}_N$, respectively. Therefore, the stationary point α_i^* in the direction α_i satisfies

$$\frac{1}{\alpha_i^*} = (\mathbf{S}_N)_{ii} + \frac{a_N}{b_N} (\mathbf{m}_N)_i^2 \,. \tag{C.5}$$

Since $\mathbf{S}_N$, $\mathbf{m}_N$ and b_N are functions of α_i^*, this equation has to be solved iteratively. One option is an iteration using the following update:

$$\frac{1}{\alpha_i^{t+1}} = \left(\mathbf{S}_N^t\right)_{ii} + \frac{a_N}{b_N^t} \left(\mathbf{m}_N^t\right)_i^2 , \tag{C.6}$$

where α_i^t is the value of α_i at iteration $t + 1$ and $\mathbf{S}_N^t$, $\mathbf{m}_N^t$ and b_N^t are evaluated at α_i^t.

The derivative with respect to a_0 is

$$\frac{\partial}{\partial a_0} \mathcal{E}(\mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \psi(a_N) - \psi(a_0) + \log \frac{b_0}{b_N}, \tag{C.7}$$

where $\psi(x)$ is the digamma function evaluated at x [45]. Finally, the derivative with respect to b_0 is

$$\frac{\partial}{\partial b_0} \mathcal{E}(\mathbf{m}_0, \mathbf{S}_0, a_0, b_0) = \frac{a_0}{b_0} - \frac{a_N}{b_N}. \tag{C.8}$$

[1] K. S. Brown, J. P. Sethna, Statistical mechanical approaches to models with many poorly known parameters, Phys. Rev. E 68 (2) (2003) 021904. doi:10.1103/PhysRevE.68.021904.

[2] J. Mortensen, K. Kaasbjerg, S. Frederiksen, J. Nørskov, J. Sethna, K. W. Jacobsen, Bayesian error estimation in density-functional theory, Phys. Rev. Lett. 95 (21) (2005) 216401. doi:10.1103/PhysRevLett.95.216401.

[3] V. Petzold, T. Bligaard, K. W. Jacobsen, Construction of new electronic density functionals with error estimation through fitting, Top. Catal. 55 (5-6) (2012) 402–417.

[4] J. Wellendorff, K. T. Lundgaard, A. Møgelhøj, V. Petzold, D. D. Landis, J. K. Nørskov, T. Bligaard, K. W. Jacobsen, Density functionals for surface science: Exchange-correlation model development with Bayesian error estimation, Phys. Rev. B 85 (23) (2012) 235149. doi:10.1103/PhysRevB.85.235149.

[5] A. J. Medford, J. Wellendorff, A. Vojvodic, F. Studt, F. Abild-Pedersen, K. W. Jacobsen, T. Bligaard, J. K. Nørskov, Assessing the reliability of calculated catalytic ammonia synthesis rates, Science 345 (6193) (2014) 197–200. doi:10.1126/science.1253486.

[6] J. Wellendorff, K. T. Lundgaard, K. W. Jacobsen, T. Bligaard, mBEEF: an accurate semi-local Bayesian error estimation density functional, J. Chem. Phys. 140 (14) (2014) 144107. doi:10.1063/1.4870397.

[7] K. Lejaeghere, V. Van Speybroeck, G. Van Oost, S. Cottenier, Error estimates for solid-state density-functional theory predictions: An overview by means of the ground-state elemental crystals, Crit. Rev. Solid State Mat. Sci. 39 (1) (2014) 1–24. doi:10.1080/10408436.2013.772503.

[8] R. G. Parr, W. Yang, Density-Functional Theory of Atoms and Molecules, Oxford University Press, 1994.

[9] R. O. Jones, O. Gunnarsson, The density functional formalism, its applications and prospects, Rev. Mod. Phys. 61 (3) (1989) 689–746. doi:10.1103/RevModPhys.61.689.

[10] M. Levy, J. P. Perdew, Density functionals for exchange and correlation energies: Exact conditions and comparison of approximations, Int. J. Quantum Chem. 49 (4) (1994) 539–548. doi:10.1002/qua.560490416.

[11] W. Kohn, L. J. Sham, Self-consistent equations including exchange and correlation effects, Phys. Rev. 140 (4A) (1965) A1133–A1138. doi:10.1103/PhysRev.140.A1133.

[12] J. P. Perdew, K. Schmidt, Jacob's ladder of density functional approximations for the exchange-correlation energy, AIP Conference Proceedings 577 (Density Functional Theory and Its Application to Materials) (2001) 1–20. doi:10.1063/1.1390175.

[13] J. P. Perdew, K. Burke, M. Ernzerhof, Generalized gradient approximation made simple, Phys. Rev. Lett. 77 (18) (1996) 3865–3868. doi:10.1103/PhysRevLett.77.3865.

[14] J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. Vydrov, G. Scuseria, L. Constantin, X. Zhou, K. Burke, Restoring the density-gradient expansion for exchange in solids and surfaces, Phys. Rev. Lett. 100 (13) (2008) 136406. doi:10.1103/PhysRevLett.100.136406.

[15] J. Tao, J. P. Perdew, V. N. Staroverov, G. Scuseria, Climbing the density functional ladder: Nonempirical meta¢generalized gradient approximation designed for molecules and solids, Phys. Rev. Lett. 91 (14) (2003) 146401. doi:10.1103/PhysRevLett.91.146401.

[16] J. Sun, B. Xiao, A. Ruzsinszky, Effect of the orbital-overlap dependence in the meta generalized gradient approximation, J. Chem. Phys. 137 (5) (2012) 051101. arXiv:arXiv:1203.2308v1, doi:10.1063/1.4742312.

[17] P. J. Stephens, F. J. Devlin, C. F. Chabalowski, M. J. Frisch, Ab initio calculation of vibrational absorption and circular dichroism spectra using density functional force fields, J. Phys. Chem. 98 (45) (1994) 11623–11627. doi:10.1021/j100096a001.

[18] C. Adamo, V. Barone, Toward reliable density functional methods without adjustable parameters: The PBE0 model, J. Chem. Phys. 110 (13) (1999) 6158–6170. doi:10.1063/1.478522.

[19] Y. Zhao, D. G. Truhlar, The M06 suite of density functionals for main group thermochemistry, thermochemical kinetics, noncovalent interactions, excited states, and transition elements: two new functionals and systematic testing of four M06-class functionals and 12 other functionals, Theor. Chem. Acc. 120 (1-3) (2008) 215–241. doi:10.1007/s00214-007-0310-x.

[20] J. P. Perdew, S. Kurth, A. Zupan, P. Blaha, Accurate density functional with correct formal properties: A step beyond the generalized gradient approximation, Phys. Rev. Lett. 82 (12) (1999) 2544–2547. doi:10.1103/PhysRevLett.82.2544.

[21] J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, C. Fiolhais, Atoms, molecules, solids, and surfaces: Applications of the generalized gradient approximation for exchange and correlation, Phys. Rev. B 46 (1992) 6671–6687. doi:10.1103/PhysRevB.46.6671.

[22] J. Sun, J. P. Perdew, A. Ruzsinszky, Semilocal density functional obeying a strongly tightened bound for exchange, Proc. Natl. Acad. Sci. U. S. A. 112 (3) (2015) 685–689. doi:10.1073/pnas.1423145112.

[23] J. Sun, B. Xiao, Y. Fang, R. Haunschild, P. Hao, A. Ruzsinszky, G. I. Csonka, G. E. Scuseria, J. P. Perdew, Density functionals that recognize covalent, metallic, and weak bonds, Phys. Rev. Lett. 111 (10) (2013) 106401. doi:10.1103/PhysRevLett.111.106401.

[24] J. P. Perdew, W. Yue, Accurate and simple density functional for the electronic exchange energy: Generalized gradient approximation, Phys. Rev. B 33 (12) (1986) 8800–8802. doi:10.1103/PhysRevB.33.8800.

[25] Y. Kanai, J. C. Grossman, Role of exchange in density-functional theory for weakly interacting systems: Quantum Monte Carlo analysis of electron density and interaction energy, Phys. Rev. A 80 (3) (2009) 032504. doi:10.1103/PhysRevA.80.032504.

[26] Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[27] D. Barber, Bayesian Reasoning and Machine Learning, Cambridge University Press, 2011. doi:10.1017/CBO9780511804779.

[28] M. E. Tipping, Sparse Bayesian learning and the relevance vector machine, J. Mach. Learn. Res. 1 (2001) 211–244.

[29] A. B. Alchagirov, J. P. Perdew, J. C. Boettger, R. C. Albers, C. Fiolhais, Energy and pressure versus volume: Equations of state motivated by the stabilized jellium model, Phys. Rev. B 63 (22) (2001) 224115. doi:10.1103/PhysRevB.63.224115.

[30] L. A. Curtiss, K. Raghavachari, P. C. Redfern, J. A. Pople, Assessment of Gaussian-2 and density functional theories for the computation of enthalpies of formation, J. Chem. Phys. 106 (3) (1997) 1063–1079. doi:10.1063/1.473182.

[31] J. Mortensen, L. Hansen, K. W. Jacobsen, Real-space grid implementation of the projector augmented wave method, Phys. Rev. B 71 (3) (2005) 035109. doi:10.1103/PhysRevB.71.035109.

[32] J. Enkovaara, C. Rostgaard, J. Mortensen, J. Chen, M. Dułak, L. Ferrighi, J. Gavnholt, C. Glinsvad, V. Haikola, H. A. Hansen, H. H. Kristoffersen, M. Kuisma, A. H. Larsen, L. Lehtovaara, M. Ljungberg, O. Lopez-Acevedo, P. G. Moses, J. Ojanen, T. Olsen, V. Petzold, N. A. Romero, J. Stausholm-Mø ller, M. Strange, G. A. Tritsaris, M. Vanin, M. Walter, B. Hammer, H. Häkkinen, G. K. H. Madsen, R. M. Nieminen, J. K. Nø rskov, M. Puska, T. T. Rantala, J. Schiø tz, K. S. Thygesen, K. W. Jacobsen, Electronic structure calculations with GPAW: a real-space implementation of the projector augmented-wave method, J. Phys.-Condes. Matter 22 (25) (2010) 253202. doi:10.1088/0953-8984/22/25/253202.

[33] S. Bahn, K. W. Jacobsen, An object-oriented scripting interface to a legacy electronic structure code, Comput. Sci. Eng. 4 (3) (2002) 56–66. doi:10.1109/5992.998641.

[34] J. Sun, M. Marsman, G. I. Csonka, A. Ruzsinszky, P. Hao, Y.-S. Kim, G. Kresse, J. P. Perdew, Self-consistent meta-generalized gradient approximation within the projector-augmented-wave method, Phys. Rev. B 84 (3) (2011) 035117. doi:10.1103/PhysRevB.84.035117.

[35] K. R. Koch, Introduction to Bayesian Statistics, Springer, 2007.

[36] J. H. Drew, D. L. Evans, A. G. Glen, L. M. Leemis, Computational Probability: Algorithms and Applications in the Mathematical Sciences, Springer, 2008.

[37] L. A. Curtiss, K. Raghavachari, P. C. Redfern, J. A. Pople, Assessment of Gaussian-3 and density functional theories for a larger experimental test set, J. Chem. Phys. 112 (17) (2000) 7374–7383. doi:10.1063/1.481336.

[38] Y. Zhao, D. G. Truhlar, Construction of a generalized gradient approximation by restoring the density-gradient expansion and enforcing a tight Lieb-Oxford bound, J. Chem. Phys. 128 (18) (2008) 184109. doi:10.1063/1.2912068.

[39] J. P. Perdew, M. Levy, Physical content of the exact Kohn-Sham orbital energies: Band gaps and derivative discontinuities, Phys. Rev. Lett. 51 (20) (1983) 1884–1887. doi:10.1103/PhysRevLett.51.1884.

[40] L. Hedin, New method for calculating the one-particle Green's function with application to the electron-gas problem, Phys. Rev. 139 (3A) (1965) A796–A823. doi:10.1103/PhysRev.139.A796.

[41] W. Chen, A. Pasquarello, Band-edge positions in GW: Effects of starting point and self-consistency, Phys. Rev. B 90 (16) (2014) 165133. doi:10.1103/PhysRevB.90.165133.

[42] M. S. Hybertsen, S. G. Louie, Electron correlation in semiconductors and insulators: Band gaps and quasiparticle energies, Phys. Rev. B 34 (8) (1986) 5390–5413. doi:10.1103/PhysRevB.34.5390.

[43] F. Hüser, T. Olsen, K. S. Thygesen, Quasiparticle GW calculations for solids, molecules, and two-dimensional materials, Phys. Rev. B 87 (23)

(2013) 235132. doi:10.1103/PhysRevB.87.235132.

[44] J. R. Magnus, H. Neudecker, Matrix Differential Calculus with Applications in Statistics and Econometrics, 3rd Edition, John Wiley & Sons, Inc., 2007.

[45] M. Abramowitz, I. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, Dover, 1972.