Structural descriptors for machine learning in materials science

Bruno Focassio[1, *]

[1] *Nanoscience and Advanced Materials graduate program, Federal University of ABC (UFABC), 09210-580 Santo André , São Paulo, Brazil*

(Dated: August 26, 2021)

CONTENTS

I. INTRODUCTION

Applying machine learning (ML) methods to atomistic systems such as molecules and periodic systems is now a highly active area of research with new applications and increasingly new developments [1]. The applications include predicting the thermodynamic stability of solids [2], formation energy of molecules [3, 4], high-throughput search of new compounds [5], creation of force-fields with quantum-mechanical accuracy [6–11] and several others [1, 12].

The goal to use ML methods in atomistic systems is to obtain system properties as fast, easy and accurately as state-of-the-art quantum mechanical methods. One of the approaches to reach this goal is to directly predict the target properties given an atomistic structure, dubbed the structure-property relation, represented in the workflow of Fig. 1. *Ab initio* calculations are performed attaining information from the nuclei positions and atomic charges using some approximation of the Schrodinger equation. The most used methods is Density Functional Thoery []. However, such simple representation is usually not adequate for input to machine learning algorithms since, among other characteristics, this input is not rotational and translational invariant as physically required [3, 12]. Using this as input would require a model to learn rotational and translation invariance, which would be complex and would require a massively large dataset [12]. To circumvent such problem, the idea is to construct a representation, also called descriptor, which encodes the information of the atomistic structure, but in a suitable input for the learning model. Training the learning model with a dataset of such descriptors for different systems allows to predict the target property. With appropriate training, validation and testing, the model is then applied to predict the properties of unknown data, as accurate as *ab initio* methods at a fraction of the computational cost.

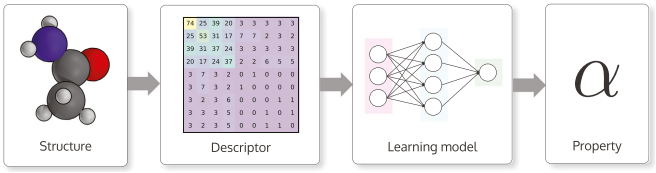

FIG. 1. Workflow of creating machine learning models for prediction of materials/molecules properties using atomistic structures. Adapted from ref. [13].

A good descriptor can be constructed by following a set of requirements that follow physical intuition and guarantees their transferability to different systems [3, 4, 12]. These requirements are: (i) translational and rotational invariance; (ii) invariance with respect to changes in the atomic indices, as this is a purely computational construct, should not alter the predicted properties; (iv) unique, as there is a single way to construct a descriptor from the atomic structure by its definition; (v) it should be continuous, since small changes in the atomic structure should result in small changes in the descriptor; (vi) it should be compact, the descriptor needs to contain sufficient information of the system for performing the prediction with a minimum number of features; (vii) it should be computationally cheap, i.e., it should be simple and fast to obtain. In the following section we will describe a selected list of descriptors that are often used in the literature. The list is far from complete and there is a vast collection of other descriptors available [1].

* b.focassio@ufabc.edu.br

II. DESCRIPTORS

A. Coulomb matrix

The Coulomb matrix (CM) [3] descriptor encodes in a simple form the same information that enters a quantum mechanical Hamiltonian for a electronic structure calculation, i.e., the atoms' coordinates and charges, $\{\mathbf{R}_i\}$ and $\{Z_i\}$, respectively. The mathematical formulation is similar to the Coulomb potential for any pair o atoms in the molecule, resulting in the Coulomb matrix describing the whole molecule:

$$M_{ij}^{\text{Coulomb}} = \begin{cases} 0.5 Z_i^{2.4} & \text{for } i = j \\ \dfrac{Z_i Z_j}{R_{ij}} & \text{for } i \neq j \end{cases} \tag{1}$$

where the off-diagonal terms correspond to the "interaction" between any two atoms as the usual Coulomb repulsion. The diagonal terms is obtained by fitting the atomic energies to the nuclear charge obtaining both the prefactor and exponent to the nuclear charge [3, 14].

Although simple, the Coulomb matrix does not handle periodic boundary conditions which is important when studying materials systems. This is addressed in the following by the Ewald sum matrix and sine matrix.

B. Ewald sum matrix

Considering periodic systems, the Coulomb potential can be summed over neighbors and all lattice sites within the infinite periodic crystal, as in the sum

$$\phi_{ij} = \sum_{\mathbf{n}} \frac{Z_i Z_j}{|\mathbf{R}_i - \mathbf{R}_j| + \mathbf{n}} \tag{2}$$

with $\mathbf{n}$ is the lattice vector $\mathbf{n} = h\mathbf{a} + k\mathbf{b} + l\mathbf{c}$, with the lattice spanned by the $\mathbf{a}$,$\mathbf{b}$, and $\mathbf{c}$ vectors.

However, this infinite sum is known to be difficult to converge, which is circumvented by the Ewald summation technique coupled with a background potential to treat non-charge neutral system [4, 12]. The sum in Eq. (2) is divided in contributions, two rapidly converging sums and a constant term which deals with the background potential and self-interaction with the periodic images:

$$M_{ij}^{\text{Ewald}} = \begin{cases} \phi_{ij}^{\text{real}} + \phi_{ij}^{\text{recip}} + (\phi_{ij}^{\text{self}} + \phi_{ij}^{\text{bg}}) & \text{for } i = j \\ 2(\phi_{ij}^{\text{real}} + \phi_{ij}^{\text{recip}} + \phi_{ij}^{\text{bg}}) & \text{for } i \neq j \end{cases} \tag{3}$$

The terms in Eq. (3) are given by

$$\phi_{ij}^{\text{real}} = \frac{1}{2} Z_i Z_j \sum_{\mathbf{n}'} \frac{\text{erfc}(\alpha |\mathbf{R}_\text{i} - \mathbf{R}_\text{j} + \mathbf{n}|)}{|\mathbf{R}_i - \mathbf{R}_j + \mathbf{n}|} \tag{4}$$

where $\mathbf{n}'$ denotes that when $\mathbf{n} = 0$ the pairs $i = j$ are not taken into account.

$$\phi_{ij}^{\text{recip}} = \frac{2\pi}{V} Z_i Z_j \sum_{\mathbf{G}} \frac{e^{-|\mathbf{G}|^2/(2\alpha)^2}}{|\mathbf{G}|^2} \cos\left(\mathbf{G} \cdot (\mathbf{R}_i - \mathbf{R}_j)\right) \tag{5}$$

where $\mathbf{G}$ is a reciprocal lattice vector.

$$\phi_{ij}^{\text{self}} = \begin{cases} -\dfrac{\alpha}{\sqrt{\pi}} Z_i^2 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \tag{6}$$

$$\phi_{ij}^{\text{bg}} = -\frac{\pi}{2V\alpha^2} Z_i Z_j^2 \qquad \forall\, i, j \tag{7}$$

where V is the unit cell volume and α is a screening parameter chosen to be $\alpha = \left(N\pi^3/V^2\right)^{1/6}$[12, 15], with N the number of atoms in the unit cell.

To computationally deal with the sums in Eq. (4) and (5) a cutoff is chosen, $n \leqslant n_{cut}$ and $G \leqslant G_{cut}$, where $n_{cut} = \sqrt{-\ln A}/\alpha$ and $G_{cut} = 2\alpha\sqrt{-\ln A}$, with A defined by the used in order to control the accuracy of the sums [4, 12, 15].

C. Sine matrix

Although physically meaningful, the Ewald sum matrix is computationally demanding to evaluate for large systems. The sine matrix given by Eq. (8) is a trade-off between capturing the correct interatomic Coulomb potential for periodic systems and computational cost.

$$M_{ij}^{\text{sine}} = \begin{cases} 0.5 Z_i^{2.4} & \text{for } i = j \\ \phi_{ij} & \text{for } i \neq j \end{cases} \tag{8}$$

where ϕ_{ij} is given by

$$\phi_{ij} = \frac{Z_i Z_j}{|\mathbf{B} \cdot \sum_{k=\{x,y,z\}} \hat{\mathbf{e}}_k \sin^2\left(\pi \mathbf{B}^{-1} \cdot (\mathbf{R}_i - \mathbf{R}_j)\right)|} \tag{9}$$

with $\mathbf{B}$ a matrix formed by the lattice vectors, $\mathbf{B} = \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}^T$.

The three matrix descriptors, Coulomb, Ewald sum and sine matrices are not invariant under atomic index permutations for the same chemical species, as is desirable for ML descriptors. Different approaches exist to achieve permutation invariance in these matrices [3, 4, 14]: (i) the matrix can be diagonalized and represented by its eigenvalues, which are invariant under rows or columns changes; (ii) The rows or columns can be sorted using a norm, e.g. Euclidean norm; (iii) different matrices can be created by small variations in the atom indexing, these can be randomly selected and used to

augment the dataset. With this augmented dataset, the learning algorithm becomes more robust against changes in the indexing. Also, these representations can be used to treat systems with a different number of atoms by zero padding the matrices given the maximum number of atoms in systems that form the dataset.

Figures 2 and 3 demonstrate these matrix representations for a methanol molecule and carbon in the diamond face-centered cubic (FCC) lattice. In Fig. 2 the Coulomb matrix for methanol demonstrates the zero padding in order to account for comparison of this representation with the representation of molecules with larger number of atoms. In this example, the first row corresponds to the carbon atom interacting with the others, the second row corresponds to the oxygen and the hydrogen atoms are encoded in the remaining rows. Figure 3 compares the Coulomb, Ewald and sine matrix for a carbon unit cell in the diamond FCC lattice. As expected, the Coulomb matrix does not capture periodic boundary conditions since the interactions with the periodic images is not given. Both Ewald and the sine matrix correctly captures the periodic boundary conditions. Moreover, one observes similar trends in both Ewald and sine matrix, however the Ewald matrix captures the self-interaction energy while the sine matrix, as well as the Coulomb matrix, approximates the potential energy for the free atom.

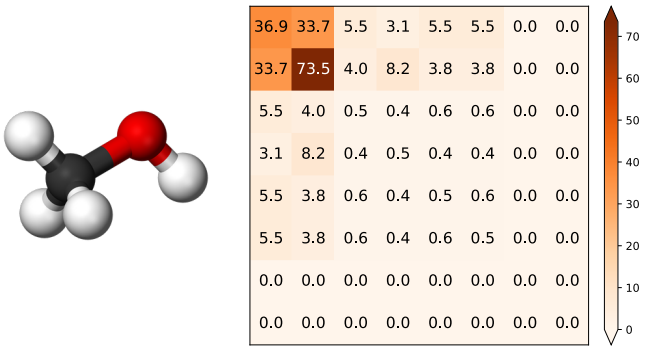

FIG. 2. (left) Methanol molecule and its Coulomb matrix (right). Adapted from ref. [13].

D. Many-body tensor representation

The many-body tensor representation (MBTR)[16] encodes chemical and structural information by breaking down the representation into k-body terms which are then grouped according to the chemical elements. MBTR naturally deals with periodic boundary conditions and finite structures. First, the representation defines a geometry function g_k that transforms a set of k atoms into a scalar feature. For $k = 1$ the geometry function encodes the atomic numbers $g_1(Z_l) : Z_l$, for $k = 2$ the distance $g_2(\mathbf{R}_l, \mathbf{R}_m) : |\mathbf{R}_l - \mathbf{R}_m|$ or inverse distance $g_2(\mathbf{R}_l, \mathbf{R}_m) : 1/|\mathbf{R}_l - \mathbf{R}_m|$ between two atoms, and for $k = 3$ the angle $g_3(\mathbf{R}_l, \mathbf{R}_m, \mathbf{R}_n) : \angle(\mathbf{R}_l - \mathbf{R}_m, \mathbf{R}_n - \mathbf{R}_m)$ or cosine of the angle $g_3(\mathbf{R}_l, \mathbf{R}_m, \mathbf{R}_n) : \cos\angle(\mathbf{R}_l - \mathbf{R}_m, \mathbf{R}_n - \mathbf{R}_m)$ between three atoms. These scalar values are broadened by a Gaussian kernel density D_k

$$D_1^l(x) = \frac{1}{\sigma_1\sqrt{2\pi}} e^{-(x-g_1(Z_l))^2/2\sigma_1^2} \tag{10}$$

$$D_2^{l,m}(x) = \frac{1}{\sigma_2\sqrt{2\pi}} e^{-(x-g_2(\mathbf{R}_l,\mathbf{R}_m))^2/2\sigma_2^2} \tag{11}$$

$$D_3^{l,m,n}(x) = \frac{1}{\sigma_3\sqrt{2\pi}} e^{-(x-g_3(\mathbf{R}_l,\mathbf{R}_m,\mathbf{R}_3))^2/2\sigma_3^2} \tag{12}$$

where σ_k is the standard deviation of the Gaussian kernel, and x runs over a range of values predefined by the user such that it covers all g_k values. These distributions are then summed for each combination of chemical species,

$$\mathrm{MBTR}_1^{Z_1}(x) = \sum_l^{|Z_1|} w_1^l D_1^l(x) \tag{13}$$

$$\mathrm{MBTR}_2^{Z_1,Z_2}(x) = \sum_l^{|Z_1|}\sum_m^{|Z_2|} w_2^{l,m} D_2^{l,m}(x) \tag{14}$$

$$\mathrm{MBTR}_3^{Z_1,Z_2,Z_3}(x) = \sum_l^{|Z_1|}\sum_m^{|Z_2|}\sum_n^{|Z_3|} w_3^{l,m,n} D_3^{l,m,n}(x) \tag{15}$$

where the sums for l, m, and n runs over all atoms with atomic number Z_1, Z_2, and Z_3, respectively. w_k is the weighting function used to control the importance of different terms, giving more importance to atoms that are close together. Besides, for periodic systems, weighting functions must be used in order to converge the sums in Eq. (13)–(15). For $k = 1$, there is usually no weighting $w_1^l = 1$. For $k = 2, 3$ one can use exponential weighting

$$w_2^{l,m} = e^{-s_k|\mathbf{R}_l - \mathbf{R}_m|} \tag{16}$$

$$w_3^{l,m} = e^{-s_k(|\mathbf{R}_l - \mathbf{R}_m| + |\mathbf{R}_m - \mathbf{R}_n| + |\mathbf{R}_l - \mathbf{R}_n|)} \tag{17}$$

where the parameter s_k is used to tune the cutoff distance, ignoring contributions less than a cutoff weight $w_k^{\min}$, $w_k < w_k^{\min}$.

In order to deal with periodic systems, the periodic images of atoms in neighboring cells are taken into account by simple repeating the unit cell in all directions. To avoid double counting atoms, one of the index l, m, and n must be in the original cell. Different cell sizes representing the same material yields different representations, this is corrected by normalizing the representation by the number of atoms or the Euclidean norm.

Constructing a MBTR requires one to set a number of system-dependent parameters. One chooses which k-body terms to consider and how they are accounted for. At each k term, one sets the broadening of the Gaussian kernel which controls the sensibility to systems changes.

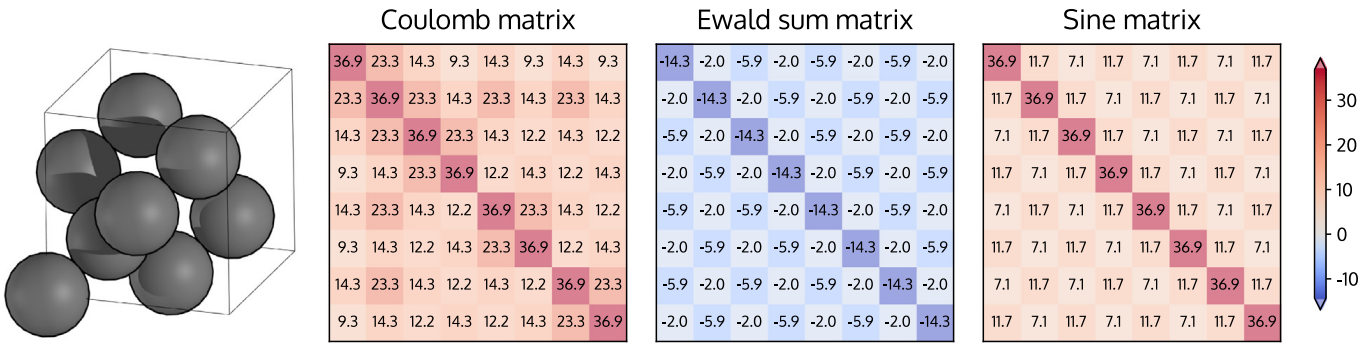

FIG. 3. Coulomb matrix, Ewald sum matrix and sine matrix for a periodic silicon structure obtained using the conventional cell. The geometry is shown on the left, and the representations are depicted from left to right. Adapted from ref. [12].

Too small values results in sharp, delta-like distributions, and too large values makes the MBTR broad and insensible to small changes in the structure.

Figure 4 shows the MBTR obtained for a water molecule. Fig. 4 shows the distributions for $k = 1, 2, 3$. For $k = 1$ the obtained spectra shows a higher intensity for hydrogen than for oxygen due to the higher number of hydrogen atoms. The higher atomic number for oxygen makes the peak shift to larger values. For $k = 2$, with the inverse distance, the H–H bond is shown before the two O–H bonds. In $k = 3$, we obtain first the two H–H–O bond angle and then the larger H–O–H bond angle. The final feature vector is obtained by appending the spectra in sequence, which requires controlling the contribution of each term to avoid unbalancing the contribution between different terms in the machine learning process, this is corrected by normalizing each contribution to unit norm before appending.

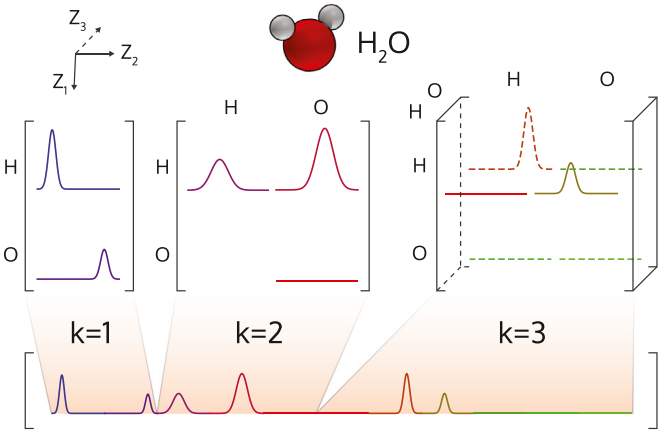

FIG. 4. Schematic representation of the MBTR for a water molecule. The spectra inside the matrices depict the MBTR_k for $k = 1, 2, 3$ allowing the visualization of each contribution. Adapted from ref. [12].

E. Smooth Overlap of Atomic Positions

The smooth overlap of atomic positions (SOAP)[17] expands the local environment into Gaussian atomic densities with spherical harmonics and radial basis functions.There is a number of ways to combine the SOAP output into a representation of the entire atomic structure. First, one obtain the atomic density ρ^Z for each species using centered Gaussians on each atom [18],

$$\rho^Z(\mathbf{r}) = \sum_i^{|Z|} \exp\left(-\frac{|\mathbf{r} - \mathbf{R}_i|^2}{2\sigma^2}\right) \tag{18}$$

where the summation runs over all atoms with atomic number Z, building different atomic densities for each chemical species.

Centering the atomic density in Eq. (18) at the point of interest ($\mathbf{r} = 0$), $\rho^Z(\mathbf{r})$ is expanded in a set of orthonormal radial basis functions g_n and spherical harmonics Y_{lm} as,

$$\rho^Z(\mathbf{r}) = \sum_{nlm} c^Z_{nlm} g_n(r) Y_{lm}(\theta, \phi) \tag{19}$$

Using Eq. (18) and (19) one can obtain the coefficients through,

$$c^Z_{nlm} = \iiint_{\mathbb{R}^3} \mathrm{d}V\, g_n(r) Y_{lm}(\theta, \phi) \rho^Z(\mathbf{r}) \tag{20}$$

When expanding the atomic density one may choose real or complex spherical harmonics [17]. The first is computationally preferable to expand the real atomic density $\rho^Z(\mathbf{r})$ [12]. The real spherical harmonics Y are defined as,

$$Y_{lm}(\theta, \phi) = \begin{cases} \sqrt{2}(-1)^m \mathrm{Im}[Y_l^{|m|}(\theta, \phi)] & \text{for } m < 0 \\ Y_l^0 & \text{for } m = 0 \\ \sqrt{2}(-1)^m \mathrm{Re}[Y_l^m(\theta, \phi)] & \text{for } m > 0 \end{cases} \tag{21}$$

where Y_l^m corresponds to the complex orthonormalized spherical harmonics defined as

$$Y_l^m(\theta,\phi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) e^{im\phi} \tag{22}$$

with $P_l^m(\cos\theta)$ the Legendre polynomials.

The output given by this representation is the partial power spectra [18–20] vector $\mathbf{p}$ where each contribution is given by

$$p_{nn'l}^{Z_1,Z_2} = \pi\sqrt{\frac{8}{2l+1}} \sum_m c_{nlm}^{Z_1\,*} c_{n'lm}^{Z_1} \tag{23}$$

The vector $\mathbf{p}$ is constructed by appending the partial contributions for all unique combinations of atomic number Z_1,Z_2, all unique pairs of radial basis functions n,n' up to $n_{\max}$ and all angular degrees l up to $l_{\max}$. The parameters $n_{\max}$ and $l_{\max}$ need to be set by the user. Using larger $n_{\max}$ and $l_{\max}$ makes SOAP more accurate, but rapidly increases the number of generated features.

One is free to consider different types of radial basis functions g_n. A computationally convenient choice is given by Gaussian type orbitals (GTOs) since they allow for analytical integration to compute the coefficients c_{nlm}^Z in Eq. (20) [12]. This basis is defined as

$$g_{nl}(r) = \sum_{n'}^{n_{\max}} \beta_{nn'l} r^l e^{-\alpha_{nl} r^2} \tag{24}$$

where the weights $\beta_{nn'l}$ are used to orthonormalize the radial basis functions, and are calculated by the Löwdin orthogonalization

$$\boldsymbol{\beta} = \mathbf{S}^{-1/2} \tag{25}$$

$$S_{nn'} = \langle \phi_{nl} | \phi_{n'l} \rangle = \int_0^\infty \mathrm{d}r r^2 r^l e^{-\alpha_{nl} r^2} r^l e^{-\alpha_{n'l} r^2} \tag{26}$$

where $\boldsymbol{\beta}$ is a matrix composed by the weights $\beta_{nn'l}$ and $\mathbf{S}$ the basis overlap matrix. The decay parameters α_n are chosen for the non-normalized orbitals ($r^l e^{-\alpha_{nl} r^2}$) decays to a threshold value at a cutoff distance r_{cut}. The parameter r_{cut} controls the maximum reach of the basis.

Another reasonable choice for the basis functions is to expand the atomic density into polynomials, as in ref. [17]. These are defined as

$$g_n(r) = \sum_{n'}^{n_{\max}} \beta_{nn'} (r - r_{\text{cut}})^{n+2} \tag{27}$$

Figure 5 compares the two radial basis functions. The GTOs shape change with l and the polynomial basis is invariant under l changes. GTOs decays approximately to zero up to a threshold value at the cutoff radius r_{cut}, in contrast with the polynomial functions that by definition decay exactly to zero at the cutoff radius.

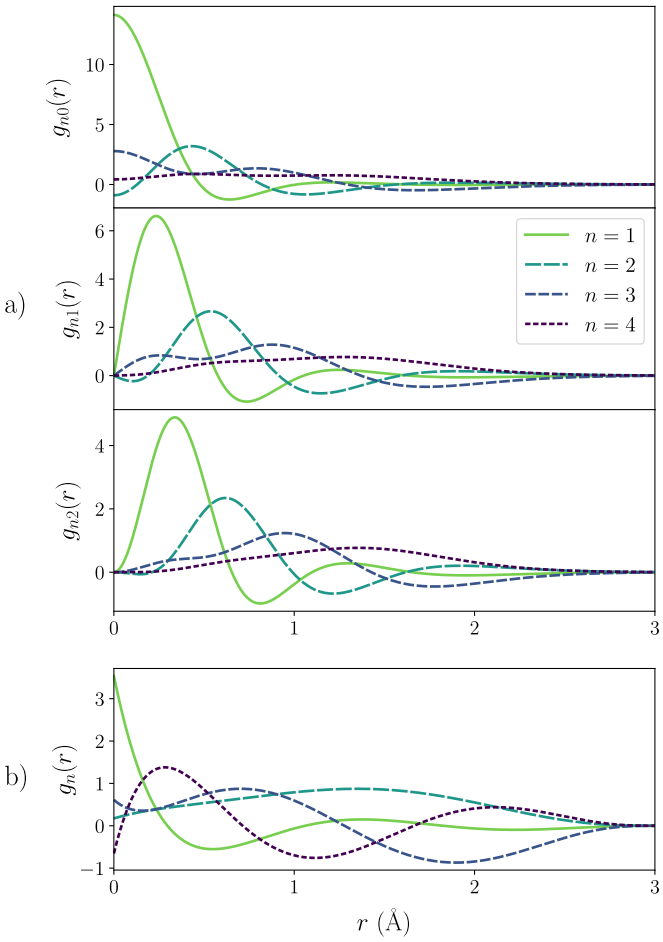

FIG. 5. Comparison of the orthonormalized radial basis function for (a) Gaussian type orbitals (GTOs) and (b) Polynomial radial basis functions. Basis functions obtained with $r_{\text{cut}} = 3$ and n up to $n_{\max} = 4$. For the GTOs we represent $l = 0, 1, 2$. Adapted from ref. [12].

Moreover, SOAP is used to obtain a representation of local environments and cannot be directly used as input for predictions related to the entire structure. For instance, the SOAP descriptor can be averaged over the local sites, this can be performed in two different ways: (i) first averaging over sites then averaging over the magnetic quantum number, Eq. (28); (ii) averaging over the power spectra of different sites, Eq. (29). The first equivalent to obtaining the coefficients for different sites, averaging over sites, then obtaining the power spectra with the averaged coefficient.

$$p_{nn'l}^{Z_1,Z_2} \sim \sum_m \left(\frac{1}{n}\sum_i c_{nlm}^{i,Z_1}\right)^* \left(\frac{1}{n}\sum_i c_{n'lm}^{i,Z_2}\right) \quad (28)$$

$$p_{nn'l}^{Z_1,Z_2} \sim \frac{1}{n}\sum_i \sum_m (c_{nlm}^{i,Z_1})^* (c_{n'lm}^{i,Z_2}) \quad (29)$$

These representations are often used to compute distances (similarities) within the dataset. The usual procedure for matrix representations is to compute the distance between each matrix element or between its eigenvalues [3]. Using the MBTR or SOAP, the resulting vector for each structure can be used in a similar manner to construct the euclidean distance. Besides, since SOAP represents the local environment, instead of using the average feature vector to compute similarities, it is usual to define a SOAP kernel to measure the similarity between two structures,

$$K^{\text{SOAP}}(\mathbf{p},\mathbf{p}') = \left(\frac{\mathbf{p}\cdot\mathbf{p}'}{\sqrt{\mathbf{p}\cdot\mathbf{p}\quad\mathbf{p}'\cdot\mathbf{p}'}}\right)^{\xi} \quad (30)$$

also, several other custom kernels can be used to combine the information of multiple sites [19].

III. APPLICATIONS

In this section we briefly describe some applications of the representations above in materials science and machine learning problems. The representation is one of the ingredients to build a machine learning model or perform data analysis, extracting patterns within the data, facilitating visualization, extracting information and gaining prediction capability.

Rupp *et al.* [3] and Faber *et al.* [4] proposed the matrix representations and used these alongside a machine learning model to predict the properties of molecules. The authors tested the representations at the QM7 database, and targeted the prediction of formation energies. The QM7 database contains molecules with up to 23 atoms and at maximum 7 heavy atoms (C, O, N, and S), totaling 7165 molecules whose properties where computed at the DFT level [3, 21]. Using a large dataset with at maximum 7 atoms such as C, O, N, S [3] reached the minimum MAE of 0.3 eV/atom with the Coulomb matrix, while ref. [4] reached MAEs around 0.07 eV/atom, which is comparable to the DFT error for calculating formation energies[22]. However, when applying these representations to periodic systems of of the Materials Project [23], the MAE increase by an order of magnitude, 0.49 eV/atom for the Ewald sum matrix and 0.37 eV/atom for the sine matrix [4]. Although these representations are able to deal with periodic boundary conditions, the chemical space spanned by the QM7 dataset is more restrict than that of the Materials Project [23]. This difference in performance evidences a key aspect of applying machine learning to molecular systems and materials science related to the representability of the chemical space. With a broader chemical space, it is expected that the performance decreases since the model needs a much larger set of examples to train.

A robust implementation of these and more descriptors is described in ref. [12]. Himanen *et al.* [12] readily demonstrated the use of the above representations to predict the formation energy of inorganic crystals. They gather the data from the Open Quantum Materials Database (OQMD) [24]. They apply a screening procedure to obtain a dataset with a maximum of 10 atoms per unit cell and a maximum of 6 different chemical species, resulting in 222 215 entries of structures, each with a calculated formation energy. In order to understand the performance of each descriptor they obtain a learning curve, i.e., the performance of the model with respect to the training data volume. The selected algorithm is a kernel ridge regression (KRR) as implemented in sci-kit learn [25]. As they are performing a learning/prediction task using periodic structures, the Coulomb matrix is only included as a baseline descriptor. For the SOAP descriptor, the authors take the average power spectra over the sites and use GTO basis functions. The hyper-parameters for the learning algorithm and selected parameters of the descriptors are optimized with grid search. Figure6 shows the learning curves.

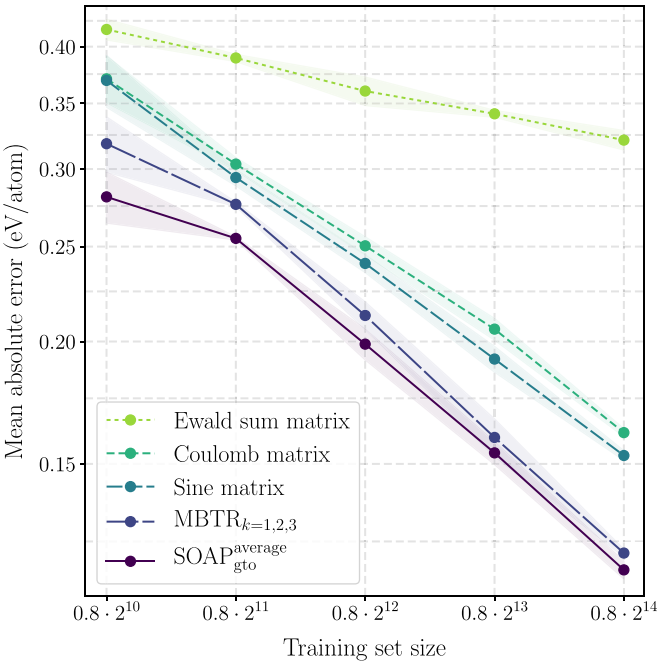

FIG. 6. Mean absolute error (MAE) for formation energies using kernel ridge regression comparing several descriptors. MAE in function of the training set size using a screened portion of the OQMD database. Adapted from ref. [12].

Using SOAP the smallest mean absolute error (MAE) obtained is 0.117 eV/atom with the largest training set. Although comparable to the standard deviation of the

training set, the value is comparable with other works in the literature [5, 6] and can be further decreased by altering the average type, increasing the functions used, and using learning algorithms capable of directly combining local environments information [12]. The Ewald sum matrix and sine matrix present a much larger MAE, as expected from other studies in the literature. This is mainly due to the incorporated information in these descriptors. The potential energy of the free atoms is shown to be important, which results in a poor performance for the Ewald sum matrix when compared to both sine and Coulomb matrix. Comparing the performance of the MBTR$_k$ parts shows that the distance information ($k = 2$) greatly contribute to the performance, while the others show a lower contribution. However the performance is greatly boosted by including the three terms k = 1,2, and 3. The results also shows that SOAP averaging over sites results in a slightly smaller MAE than MBTR, although it is expected that MBTR will perform better than SOAP for larger training sets.

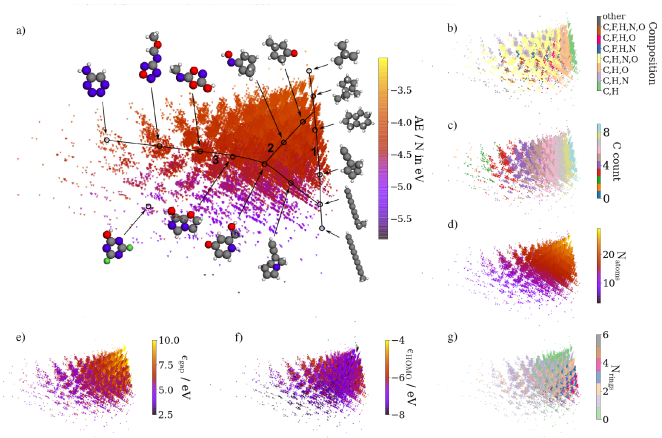

FIG. 7. Kernel PCA maps of the QM9 database using a SOAP representation. The subpanels are color-coded according to structural descriptors (b, c, d, g) and quantum mechanical properties (a, e, f). Adapted from ref. [18].

Combining structural representations with unsupervised machine learning algorithms results in a powerful methodology to analyze and extract information on large databases [18, 19]. References [18, 19] provides several examples of dimensionality reduction methods combined with SOAP representation. Figure 7 illustrates one of such results. Cheng *et al.* [18] represented the molecules in the QM9 database using SOAP and performed a kernel principal component analysis to obtain a 2D map for the database. The QM9 dataset contains 133 885 organic molecules with up to nine heavy atoms (C,O,N, and F). The principal component analysis (PCA) is a dimensionality reduction technique that finds first few eigenvectors of $\mathbf{X}^T\mathbf{X}$, where $\mathbf{X}$ is the data matrix, with feature vectors forming the columns and the different data instances forming the rows. By representing the eigenvectors (principal components) with the largest eigenvalues of this matrix one represents the linear combination of the features that maximizes the variance of the data [1]. Choosing the first two principal components one obtain a two-dimensional representation as in Fig. 7. Using a kernel with PCA, one is able to construct a non-linear combination of features, resulting in a custom measure of distance within the dataset [18], while in regular linear PCA the distance measured in the hyper-dimensional space is approximately kept in the lower dimensional space. The resulting map, Fig. 7 is useful for navigating in this large dataset and is able to translate complex and useful information into a simple two-dimensional space. Figure 7a demonstrates that pure hydrocarbons forms a separate cluster and lie along the "path" 1, walking through the vertical axis in path 1 transitions from linear to cyclic and to simple molecules. In the map, walking from right to left adds other heteroatoms to the molecules. Color-coding this map with other properties gives insight into other patterns presented in the data. For instance, in Fig. 7b from left to right the chemical diversity within the molecule decreases, and in Fig. 7c we confirm that the x-axis encodes information regarding the number of carbon atoms in the molecules, and in Fig. 7d we observe that molecules with higher number of atoms group in the upper part of the map. Correlating these information with Fig. 7e indicates that unsaturated compounds tend to have lower gaps [18].

A. Practical example: predicting properties of molecules

In this practical example we applied the Coulomb matrix, SOAP and MBTR to predict the HOMO-LUMO gap of small organic molecules. The code to reproduce these results are available at https://github.com/bfocassio/descriptors_ml

With our problem established, we need data to train the model. We use a subset of the GDB-9 database [26] containing up to 7 heavy elements (C, N, O, and F), totaling 3982 molecules. For each molecule in the dataset, there is an atomistic structure and a large set of properties computed at the B3LYP level. In order to represent the data for the learning algorithm, we choose investigate three different representations, namely, the Coulomb matrix, SOAP and MBTR. We obtain the representations using the implementations provided by ref. [12]. Using the Coulomb matirx, we sort the rows according to their Euclidean norm. For SOAP we GTO basis as radial basis functions, with 0.4 as standard deviation, $r_{\text{cut}} = 6$, $n_{\text{max}} = 8$, and $l_{\text{max}} = 8$. To obtain the SOAP representation for the whole structure, we average over the sites before summing the magnetic quantum number. Using MBTR, we choose the atomic number, inverse of distance and cosine of the angle as the one, two and three-body terms, the standard deviations for each of these terms is 0.6, 0.2 and 0.0005, respectively. Each k-term is normalized according to their Euclidean norm and the resulting

feature vector is normalized.

After representing the data we choose the learning algorithm as the kernel ridge regression which is a standard test platform for these molecular properties predictions [3, 4]. For the CM we use the Laplacian kernel, whereas for the SOAP and MBTR, we use the Gaussian kernel [16]. Both the regularization parameter and kernel width were roughly optimized. The train and test set are split using the 80%/20% ratio.

Figure 8 shows the parity plots obtained for each representation, evaluated on the test set. The Coulomb matrix results in a slightly larger MAE than the other representations. Moreover, MBTR show an improvement over SOAP. Even though the representations yield different performances, there is no superior representation. For instance, the MBTR is more computationally expensive than the other representations.

To investigate how the different representations affect the model performance with the size of the training set, Fig. 9 plots the learning curves for the three models. As expected the MAE decreases as the training set increases. The trend observed in Fig. 8 is maintained, MBTR results in the smallest MAE for all the training set sizes, and it is expected to decrease even more if the data volume is increased [12]. The difference between the Coulomb matrix and SOAP is decreased for larger training sets, however, this is artificial since our dataset contains simple systems that are non-periodic.

IV. SUMMARY

We presented a short introduction to atomistic descriptors. With increasing complexity, we briefly described the derivation of several descriptors. Generating descriptors for atomistic modeling is essential to apply statistical and machine learning analysis to the data. They are used to capture and translate all atomistic physical information to the algorithm. With the increasing power of computational tools available, it is essential to extract information and, otherwise inaccessible, patterns within the generated data. Choosing the descriptors for every problem requires testing, and there is no free lunch as there is no superior descriptor. The Coulomb matrix is a simple descriptor; however, it does not handle periodic systems and requires sorting/transformations for encoding invariance with atomic indices. However, both the sine and Ewald sum matrix can encode periodic systems; nevertheless, they are more computationally demanding. The SOAP descriptors have shown to be decisive in various applications [1], from the prediction of properties to force-fields, and its performance can be easily increased by adding more basis functions, paying the computational cost. The SOAP descriptor encodes local environments and requires more complex techniques to represent the entire structure. The MBTR naturally encodes the whole atomistic structure and usually yields good results. However, it generates large feature vectors that increase the computational cost of the learning task.

[1] G. R. Schleder, A. C. M. Padilha, C. M. Acosta, M. Costa, and A. Fazzio, J. Phys. Mater. **2**, 032001 (2019).

[2] G. R. Schleder, C. M. Acosta, and A. Fazzio, ACS Appl. Mater. Interfaces **12**, 20149 (2020).

[3] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, Phys. Rev. Lett. **108**, 058301 (2012), arXiv:1109.2618.

[4] F. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, Int. J. Quantum Chem. **115**, 1094 (2015), arXiv:1503.07406.

[5] L. Ward, R. Liu, A. Krishna, V. I. Hegde, A. Agrawal, A. Choudhary, and C. Wolverton, Phys. Rev. B **96**, 024104 (2017).

[6] K. Choudhary, B. DeCost, and F. Tavazza, Phys. Rev. Mater. **2**, 083801 (2018), arXiv:1805.07325.

[7] V. Botu, R. Batra, J. Chapman, and R. Ramprasad, J. Phys. Chem. C **121**, 511 (2017), arXiv:1610.02098.

[8] G. Sivaraman, A. N. Krishnamoorthy, M. Baur, C. Holm, M. Stan, G. Csányi, C. Benmore, and Á. Vázquez-Mayagoitia, npj Comput. Mater. **6**, 1 (2020).

[9] T. D. Huan, R. Batra, J. Chapman, S. Krishnan, L. Chen, and R. Ramprasad, npj Comput. Mater. **3**, 1 (2017).

[10] V. Botu and R. Ramprasad, Phys. Rev. B - Condens. Matter Mater. Phys. **92**, 1 (2015), arXiv:1505.02701.

[11] J. S. Smith, O. Isayev, and A. E. Roitberg, Chem. Sci. **8**, 3192 (2017), arXiv:1610.08935.

[12] L. Himanen, M. O. Jäger, E. V. Morooka, F. Federici Canova, Y. S. Ranawat, D. Z. Gao, P. Rinke, and A. S. Foster, Comput. Phys. Commun. **247**, 106949 (2020), arXiv:1904.08875.

[13] L. Himanen, M. O. Jäger, E. V. Morooka, F. Federici Canova, Y. S. Ranawat, D. Z. Gao, P. Rinke, and A. S. Foster, Dscribe 0.4.0 (2020), accessed on nov. 2020, `https://singroup.github.io/dscribe`.

[14] R. Ramakrishnan, M. Hartmann, E. Tapavicza, and O. A. von Lilienfeld, J. Chem. Phys. **143**, 084111 (2015), arXiv:1504.01966.

[15] R. A. Jackson and C. R. A. Catlow, Mol. Simul. **1**, 207 (1988).

[16] H. Huo and M. Rupp, arXiv (2017), arXiv:1704.06439.

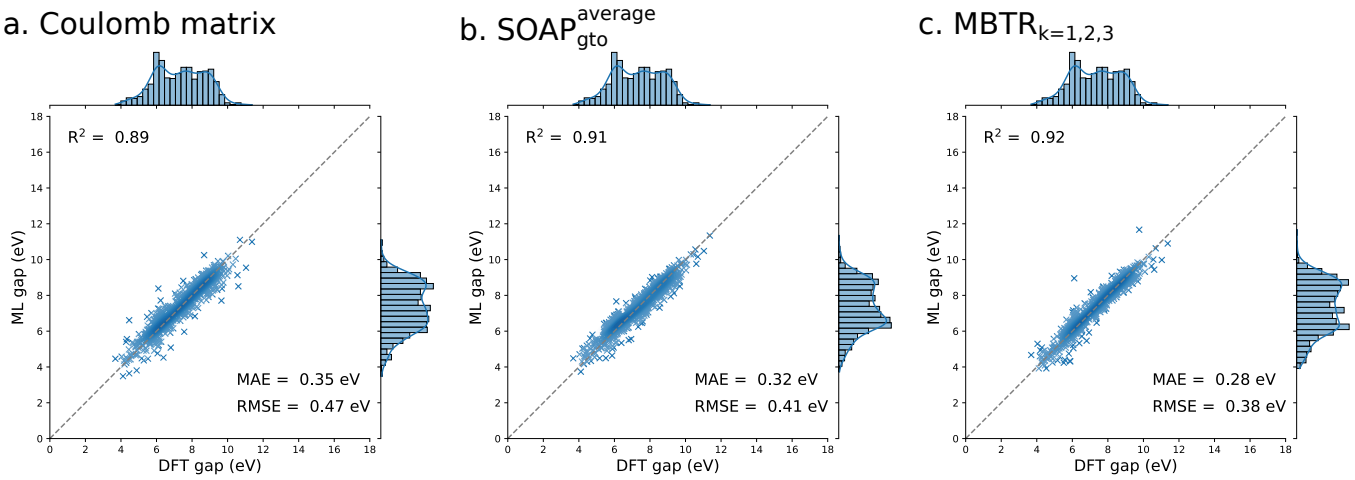

FIG. 8. Correlation scatter plot evaluated for the test set using a kernel ridge regression model trained with the three representations, (a) Coulomb matrix, (b) SOAP and (c) MBTR on a subset of the GDB-9 database.

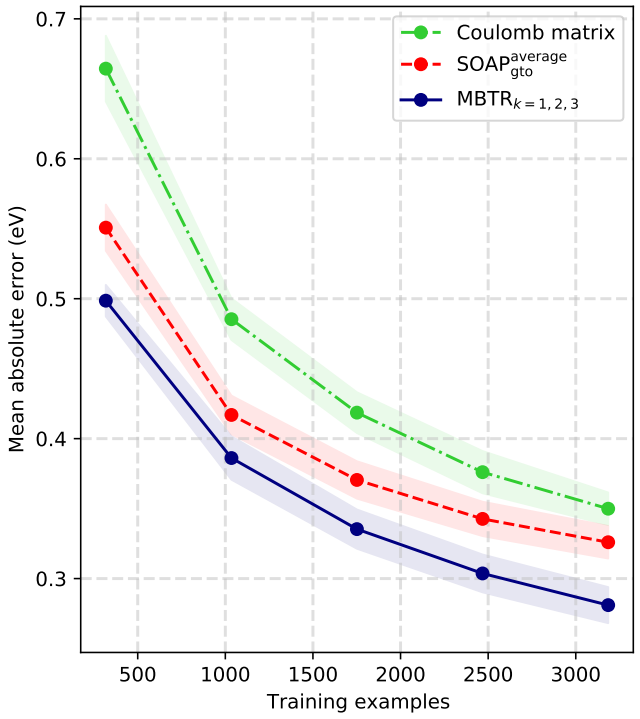

FIG. 9. Learning curves obtaining with 5-fold cross validation for the kernel ridge regression model trained with the three representations, Coulomb matrix, SOAP and MBTR.

[17] A. P. Bartók, R. Kondor, and G. Csányi, Phys. Rev. B **87**, 184115 (2013), arXiv:1209.3140.

[18] B. Cheng, R.-R. Griffiths, S. Wengert, C. Kunkel, T. Stenczel, B. Zhu, V. L. Deringer, N. Bernstein, J. T. Margraf, K. Reuter, and G. Csanyi, Acc. Chem. Res. **53**, 1981 (2020).

[19] S. De, A. P. Bartók, G. Csányi, and M. Ceriotti, Phys. Chem. Chem. Phys. **18**, 13754 (2016), arXiv:1601.04077.

[20] A. P. Bartók, S. De, C. Poelking, N. Bernstein, J. R. Kermode, G. Csányi, and M. Ceriotti, Sci. Adv. **3**, 10.1126/sciadv.1701816 (2017), arXiv:1706.00179.

[21] L. C. Blum and J.-L. Reymond, J. Am. Chem. Soc. **131**, 8732 (2009).

[22] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996).

[23] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, APL Mater. **1**, 011002 (2013).

[24] J. E. Saal, S. Kirklin, M. Aykol, B. Meredig, and C. Wolverton, JOM **65**, 1501 (2013).

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, J. Mach. Learn. Res. **12**, 2825 (2011).

[26] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, Sci. Data **1**, 140022 (2014).