Supporting information

Automatic Structural Elucidation of Vacancies in Materials by Active Learning

Maicon Pierre Lourenço[1*], Lizandra Barrios Herrera[2], Jiří Hostaš[2], Patrizia Calaminici[3], Andreas M. Köster[3], Alain Tchagang[4], Dennis R. Salahub[2]

1- Departamento de Química e Física – Centro de Ciências Exatas, Naturais e da Saúde – CCENS – Universidade Federal do Espírito Santo, 29500-000, Alegre, Espírito Santo, Brasil.
2- Department of Chemistry, Department of Physics and Astronomy, CMS Centre for Molecular Simulation, IQST Institute for Quantum Science and Technology, Quantum Alberta, University of Calgary, 2500 University Drive NW, Calgary, AB, T2N 1N4, Canada.
3- Departamento de Química, CINVESTAV, Av. Instituto Politécnico Nacional 2508, AP 14-740, México D.F. 07000 México.
4- Digital Technologies Research Centre, National Research Council of Canada, 1200 Montréal Road, Ottawa, ON, K1A 0R6 Canada.

** Address correspondence to:* maiconpl01@gmail.com(MPL).

1. Input of QMLMaterial for vacancy in materials and nanoparticles

Figure S1 shows an example of the QMLMaterial input for the active learning (AL) of $CaTiO_3$. The input is organized in two blocks, where input 1 (a) controls the system and calculator, while input 2 (b) drives the settings of the machine learning.

The first line at the 'structure_file_xyz' reads the xyz file with the initial configuration of the pristine CaTiO3 ($Ca_8Ti_8O_{24}$), which in this case is called Ca8Ti8O24.opt.xyz. Then the atomic symbol of the atom to be removed is set up in 'atom_type_to_be_removed'. By default, 'atom_type_to_be_doped' is set up to '[]'. After that, we insert the index of the XYZ coordinates of the atoms that will be removed: 'atom_index_list''. The final part of input 1 calls for the calculator to be used for the local optimization calculations. Here, the DFT method in quantum espresso (QE) was employed for the energy evaluations in a PBC system. Finally, the QMLMaterial supports periodic and cluster calculations, which can be selected with the system_type flag.

The machine learning controller in input 2 starts by reading the percent of the dataset that will be used for testing (test_size flag), the cross-validation procedure to estimate the skill of the model in generating new data (sample), as well as the number of cross-validation splits (n_splits). If the flag is_max_true is active ("is_max_true = True"), then a maximization of the target property will be performed; otherwise, minimization will be done. In all of our AL applications we have scaled the total energy to be in a range between 0 to 200 since we have observed it results in better regression models. Therefore, we perform a maximization in this scaled energy space. The opt_flow_controler defines many of the parameters that drive the AL, such as the maximum number of iterations (n_iteration), the number of energy evaluations in each iteration (iteration_step), the size of the initial dataset, the type of acquisition functions, the descriptor, the size of the virtual space, and the path where the calculations will be performed. Finally, the param_grd controls the type of machine learning model and its parameters. A Gaussian process model ('gpr__kernel', 'gpr__alpha', etc.) was trained and used in the case below.

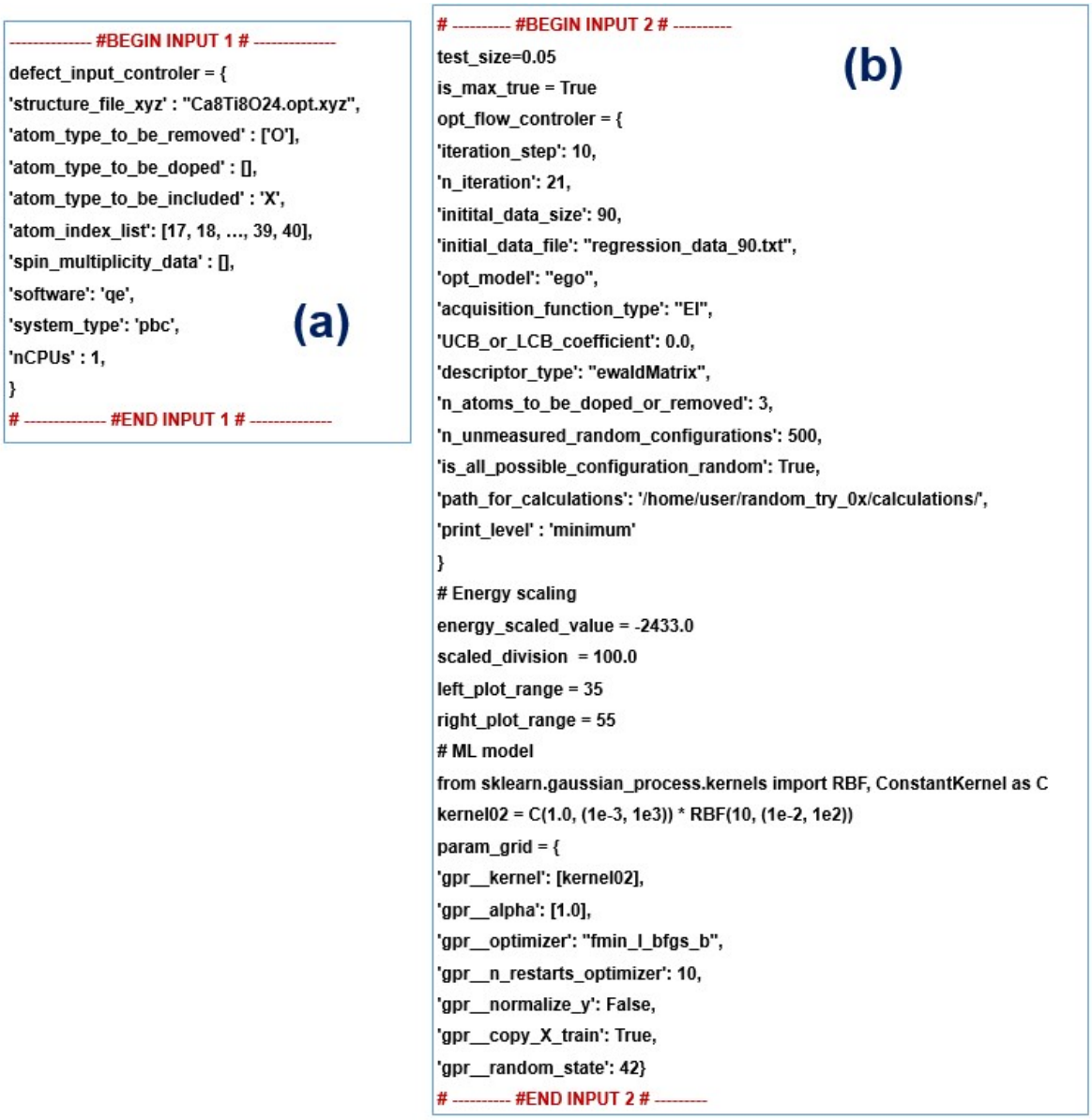

Figure S1. The mnemonic input of QMLMaterial for vacancies structural elucidation. (a) INPUT-01 deals with the molecular/solid system (cluster or periodic), the calculator (i.e. DFTB+, deMon2k or Quantum Espresso (QE)) and the symbol of the atoms that will be removed which defines the discrete search space. (b) INPUT-02 deals with the machine learning model, the active learning parameters and the sampling process (i.e., "n_unmeasured_random_configurations" that defines the non-computed (virtual) structures where the ML inference is made on).

2. Statistical Regression

The idea behind statistical regression is to obtain N *observed* properties $\mathbf{y} = (y^{(1)}, \ldots, y^{(n)})$, i = 1, …., N; to describe **y** statistically, the descriptor $\mathbf{x}^{(i)} = (x_1^{(i)}, \ldots, x_k^{(i)})$, with K variables is required. The property in our case is the total energy obtained from quantum chemistry methods: DFT, SCC-DFTB, etc. This results in a matrix **X** of dimension (N × K), called the feature matrix which is associated with the one-dimensional vector **y** (the objective function) of dimension (N). Here we define descriptor $\mathbf{x}^{(j)} = (x_1^{(j)}, \ldots, x_k^{(j)})$ in the virtual space ($^{k}_{N_{virtual}}$): j = 1, …, $^{k}_{N_{virtual}}$; where *j* stands for the *j-th* virtual (non-observed) structure.

To model the desired problem in this manner, several surrogate models, such as the Artificial Neural Network (ANN) can be used by utilizing high level libraries, such as scikit-learn[1].

After performing the regression, the statistical model is obtained and represented as:

$$\grave{y} = \hat{f}(X), \qquad \textbf{(1)}$$

where $\grave{y}$ is the vector with the predicted properties and $\hat{f}(X)$ is the statistical model (the predictor)designed from the MLP regressor.

Usually, to obtain a model without data bias, the matrices **X** and **y** – which define the initial data to obtain and test the ML models ($\mathbf{X}^{l}$,$\mathbf{y}^{l}$) – are split into two other matrices: ($\mathbf{X}^{train}$, $\mathbf{y}^{train}$), which are used to train the statistical model and ($\mathbf{X}^{test}$, $\mathbf{y}^{test}$) to validate it.

In order to obtain the average ($\mu(\mathbf{x^{(i)}})$ for the computed structures and $\mu(\mathbf{x^{(j)}})$ for the non-computed, virtual, structures) and the uncertainty (the standard deviation $\sigma(\mathbf{x^{(i)}})$ and $\sigma(\mathbf{x^{(j)}})$), the matrices ($\mathbf{X}^{train}$, $\mathbf{y}^{train}$) are partitioned K times for K-fold cross-validation (CV) or B times for non-parametric bootstrap (BS). For each partition p (in a total space of P=K or P=B) a statistical model is obtained. Hence: $\grave{y}^{p} = \hat{f}(X)$, p = 1, …, P. Then, for each descriptor in the observed data set $\mathbf{x}^{(i)}$ (or for each descriptor j in the non-computed or virtual structures space: $\mathbf{x}^{(j)}$) the average

$\mu(\mathbf{x^{(i)}})$ (or $\mu(\mathbf{x^{(j)}})$ for the virtual structures) and the standard deviation $\sigma(\mathbf{x^{(i)}})$ (or $\sigma(\mathbf{x^{(j)}})$) are obtained, as illustrated in figure S2 for the data in the observed space.

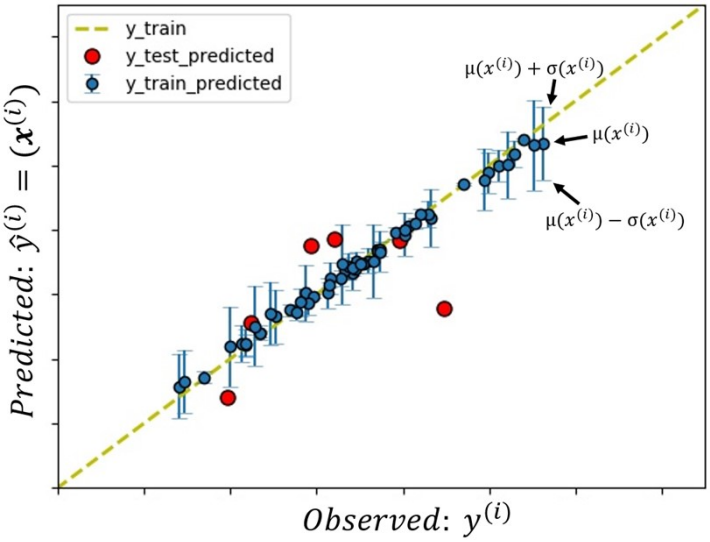

Figure S2- Plot of the observed $y^{(i)}$ and predicted $\grave{y}(x^{(i)})$ target property. The use of Gaussian process allows us to have a regression model for each data point $x^{(i)}$ represented from the mean $\mu(x^{(i)})$ and the standard deviation $\sigma(x^{(i)})$. The abscissa is the observed property and the ordinate the predicted one. The same for the unexplored space descriptors for inference and decision making: exploitation ($\mu(x^{i})$) and exploration ($\sigma(x^{(j)})$).

The mean $\mu(\mathbf{x^{(j)}})$ and the standard deviation $\sigma(\mathbf{x^{(j)}})$ for each descriptor entry in the non-observed (virtual) space (whose dimension is defined by $N^{k}_{virtual}$) will be used to obtain the acquisition function[2, 3] which is used to indicate the next candidate to be computed. The next candidate is, then, incorporated in the initial descriptor matrix: ($\mathbf{X}^{\text{train+1}}$, $\mathbf{y}^{\text{train+1}}$) and the iteration process continues one step more until the optimization of the target property.

3. Kernel functions

The GP for regression is a non-parametric Bayesian model widely employed in supervised learning[4]. It uses a prior's covariance that needs to be specified by passing a kernel object. In this work we use two customized kernel functions, obtained from common kernels implemented in scikit-learn[5] library,

$$kernel01 = DotProduct + Whitekernel, \quad \textbf{(2)}$$

and

$$kernel02 = C \cdot RBF. \quad \textbf{(3)}$$

In equation 2 the *DotProduct*, is given by,

$$k_{DotProduct}(x_i,x_j) = \sigma_0^2 + x_i \cdot x_j, \quad \textbf{(4)}$$

where σ_0^2 is a parameter that controls the inhomogeneity of the kernel. The uses of the *WhiteKernel* in 2 is to better estimate the noise level of the data. The second kernel (eq. 3) combines the constant kernel (C), and the radial-basis function kernel (RBF) given by

$$k_{RBF}(x_i,x_j) = exp\left(-\frac{d(x_i,x_j)^2}{2l^2}\right), \quad \textbf{(5)}$$

which is parameterized by the length-scale parameter (l).

4. Surface (3D) view of the acquisition functions

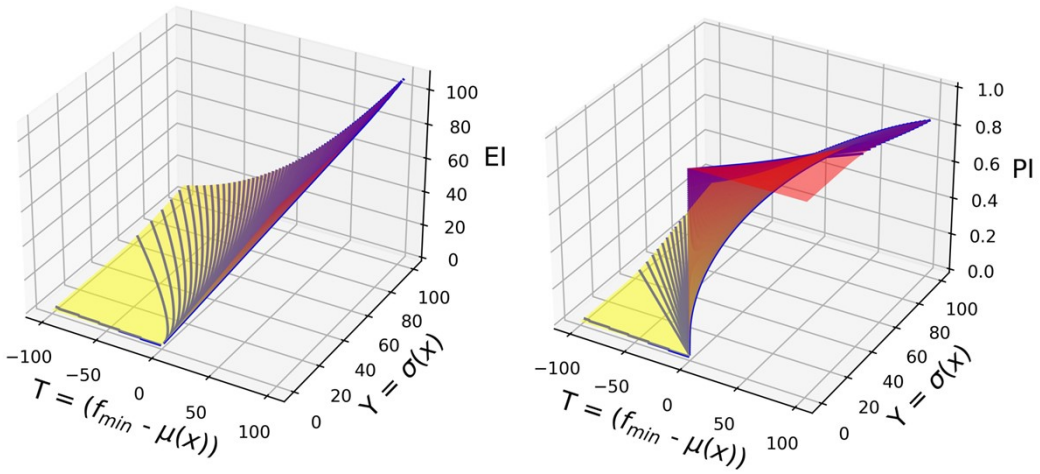

Figure S3- The surface (3D) plot of: (left) the expected improvement (EI) and (right) the probability of improvement (PI) as a function of the target prediction, T = (f_{min} - μ(X)), and the uncertainty, Y = σ(X). The minimum property observed so far is f_{min}; μ

and σ are, respectively, the mean and the standard deviation of the prediction obtained from GP for the structure configuration, represented as descriptor, X.

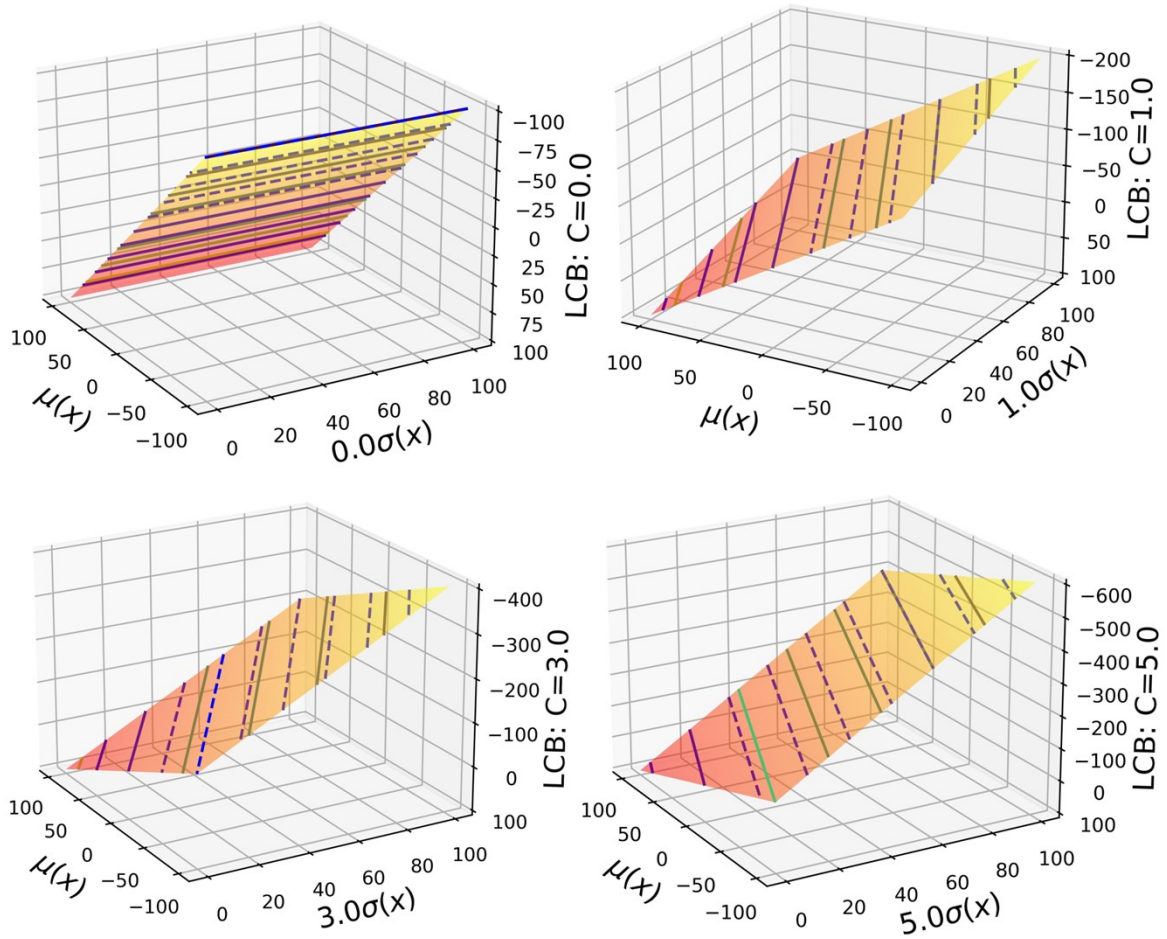

Figure S4- The surface (3D) plot of the lower confidence bound (LCB): $\mu(X) - C\sigma(X)$, with C = 0, 1, 3 and 5. Where, μ and σ are, respectively, the mean and the standard deviation of the prediction obtained from GP for the structure configuration, represented as descriptor, X.

5. PDOS of a local minimum above the putative global minimum (GM)

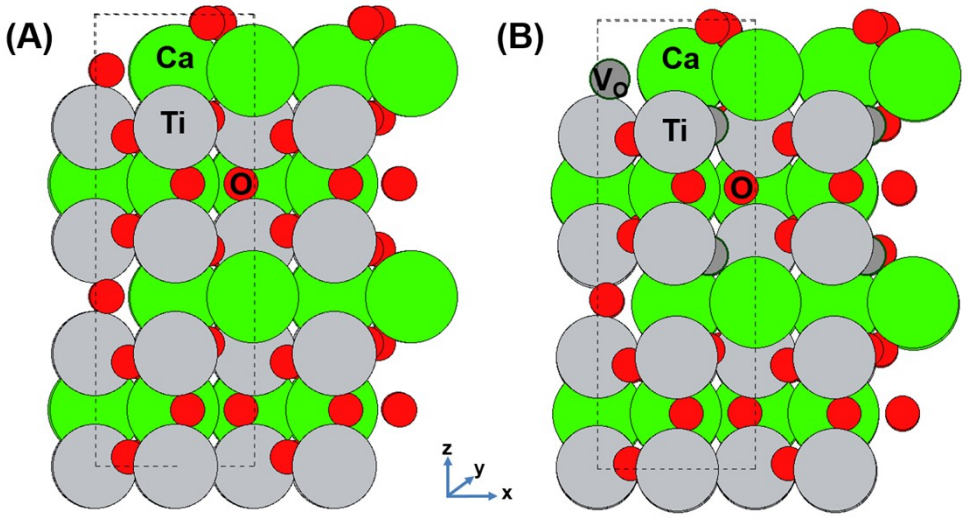

Figure S5- (A) The pristine $CaTiO_3$ and (B) some local minima $CaTiO_{2.625}$ from DFT whose three oxygen vacancy were made by hand. This structure is 0.85 eV above the putative GM found by AL (Fig. 14). Two tri coordinated and one tetra coordinated O are removed.

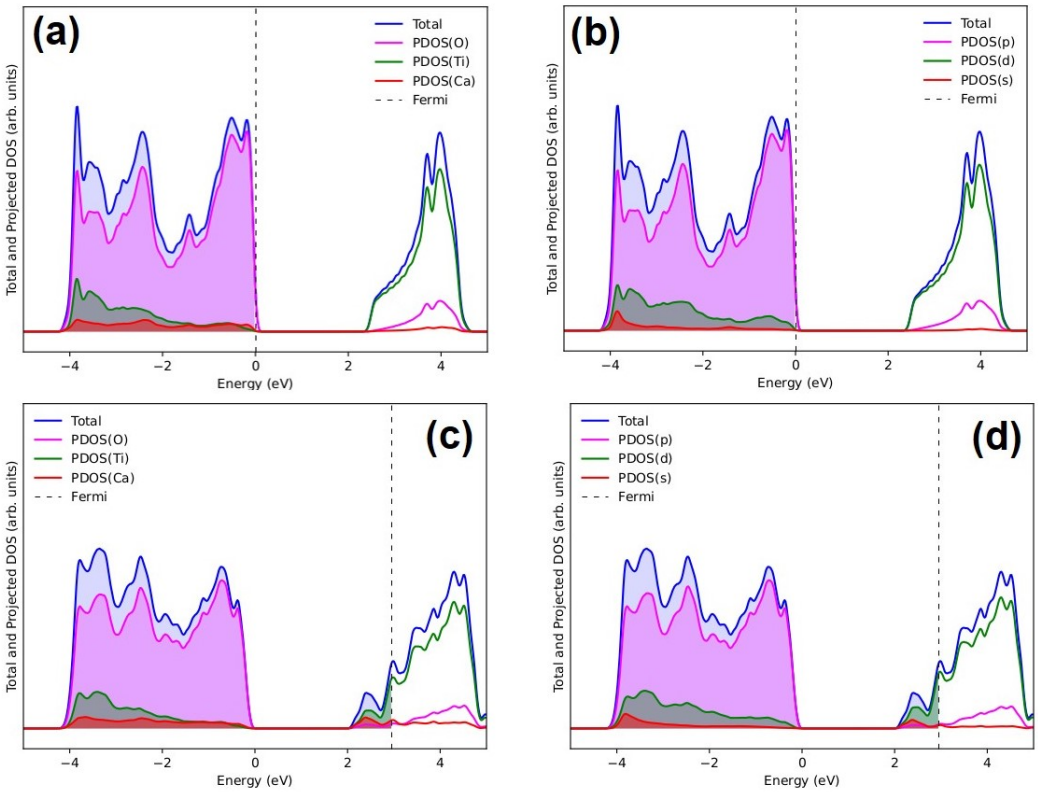

Figure S6- Projected density of states (PDOS) on atoms for the pristine $CaTiO_3$ (a and b) and for a local minima, above the global minimum, of the modified perovskite, $CaTiO_{2.625}$ (c and d). Total energy: -2433.43897078 Ry.

6. References

1. F. Pedregosa, Ga, #235, l. 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, #201 and d. Duchesnay, *J. Mach. Learn. Res.*, 2011, **12**, 2825-2830.
2. T. Lookman, P. V. Balachandran, D. Xue and R. Yuan, *npj Computational Materials*, 2019, **5**, 21.
3. D. R. Jones, M. Schonlau and W. J. Welch, *Journal of Global Optimization*, 1998, **13**, 455-492.
4. V. L. Deringer, A. P. Bartók, N. Bernstein, D. M. Wilkins, M. Ceriotti and G. Csányi, *Chemical Reviews*, 2021, **121**, 10073-10141.
5. 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, E. Duchesnay and G. Louppe, *Journal of Machine Learning Research*, 2012, **12**.