The Journal of Chemical Physics

RESEARCH ARTICLE | JULY 17 2023

An exploration of machine learning models for the determination of reaction coordinates associated with conformational transitions

Special Collection: Machine Learning Hits Molecular Simulations

Nawavi Naleem ; Charlles R. A. Abreu ; Krzysztof Warmuz ; Muchen Tong ; Serdal Kirmizialtin ; Mark E. Tuckerman

J. Chem. Phys. 159, 034102 (2023)
https://doi.org/10.1063/5.0147597
CHORUS

AIP Publishing

The Journal of Chemical Physics ARTICLE pubs.aip.org/aip/jcp

An exploration of machine learning models for the determination of reaction coordinates associated with conformational transitions

Cite as: J. Chem. Phys. 159, 034102 (2023); doi: 10.1063/5.0147597
Submitted: 23 February 2023 • Accepted: 23 June 2023 •
Published Online: 17 July 2023

Nawavi Naleem,[1] Charlles R. A. Abreu,[2] Krzysztof Warmuz,[3] Muchen Tong,[4] Serdal Kirmizialtin,[1,4,8,a)] and Mark E. Tuckerman[4,5,6,7,b)]

AFFILIATIONS

[1] Chemistry Program, Science Division, New York University, Abu Dhabi, UAE
[2] Chemical Engineering Department, Escola de Química, Universidade Federal do Rio de Janeiro, 21941-909 Rio de Janeiro, RJ, Brazil
[3] Computer Science Program, Science Division, New York University, Abu Dhabi, UAE
[4] Department of Chemistry, New York University (NYU), New York, New York 10003, USA
[5] Courant Institute of Mathematical Sciences, New York University, New York, New York 10012, USA
[6] NYU-ECNU Center for Computational Chemistry at NYU Shanghai, 3663 Zhongshan Rd. North, Shanghai 200062, China
[7] Simons Center for Computational Physical Chemistry at New York University, New York, New York 10003, USA
[8] Center for Smart Engineering Materials, New York University, Abu Dhabi, UAE

Note: This paper is part of the JCP Special Topic on Machine Learning Hits Molecular Simulations.
[a)] **Electronic mail:** serdal@nyu.edu
[b)] **Author to whom correspondence should be addressed:** mark.tuckerman@nyu.edu

ABSTRACT

Determining collective variables (CVs) for conformational transitions is crucial to understanding their dynamics and targeting them in enhanced sampling simulations. Often, CVs are proposed based on intuition or prior knowledge of a system. However, the problem of systematically determining a proper reaction coordinate (RC) for a specific process in terms of a set of putative CVs can be achieved using committor analysis (CA). Identifying essential degrees of freedom that govern such transitions using CA remains elusive because of the high dimensionality of the conformational space. Various schemes exist to leverage the power of machine learning (ML) to extract an RC from CA. Here, we extend these studies and compare the ability of 17 different ML schemes to identify accurate RCs associated with conformational transitions. We tested these methods on an alanine dipeptide in vacuum and on a sarcosine dipeptoid in an implicit solvent. Our comparison revealed that the light gradient boosting machine method outperforms other methods. In order to extract key features from the models, we employed Shapley Additive exPlanations analysis and compared its interpretation with the "feature importance" approach. For the alanine dipeptide, our methodology identifies ϕ and θ dihedrals as essential degrees of freedom in the $C7_{ax}$ to $C7_{eq}$ transition. For the sarcosine dipeptoid system, the dihedrals ψ and ω are the most important for the *cis* α_D to *trans* α_D transition. We further argue that analysis of the full dynamical pathway, and not just endpoint states, is essential for identifying key degrees of freedom governing transitions.

I. INTRODUCTION

Molecular dynamics simulations allow for changes in chemical or physical processes to be monitored with high spatial and temporal resolution. At the heart of such molecular changes lie conformational transitions. Understanding the essential degrees of freedom that govern a conformational transition is crucial to elucidate the kinetics and thermodynamics of the molecule of interest.

The problem of finding the essential coordinates that drive the process during a transition is highly challenging due to the large number of degrees of freedom involved. The first step in finding the reaction coordinate is the identification of a set of collective variables (CVs), which are molecular features that clearly distinguish

conformations well separated by energetic barriers. Trajectories connecting different regions of the configurational space can serve to identify a subset of CVs that contribute to the transition pathway.

Machine learning algorithms are powerful tools to identify reaction coordinates from a given set of collective variables.[1–5] Earlier studies on determining reaction coordinates from a set of collective variables focus on fundamental chemical processes.[6–13] In addition to these studies, different machine learning methods, such as neural networks, regression, and dimensionality reduction techniques, have been used to suggest reaction coordinates.[14–26]

Committor analysis provides a natural reaction coordinate to investigate the transition paths of structural transitions from molecular simulations. The transition path sampling (TPS) approach is used successfully to study chemical processes and protein folding.[27–30] For a phase space region well separated from metastable states A, B, the committor value of $P_B(x)$ reports the probability that trajectories initiated at x with velocities sampled from the Maxwell–Boltzmann distribution reach state B before reaching state A. The phase space hypersurface with $P_B(x) = 0.5$ is of interest to chemistry, as it defines the dividing surface that separates the states for the conformational transition understudy.

Although the committor analysis provides the committor distribution generically as a reaction coordinate, it does not directly provide insights into the essential degrees of freedom that govern the change in the committor values and make up the key components of the reaction coordinate. To find essential coordinates, various committor-based methods have been proposed that do not employ transition path sampling (TPS).[31–33] A more comprehensive review of the previous work on this topic can be found in the reviews in Refs. 34 and 35.

The seminal work of Dellago *et al.*[6] utilizing transition path sampling (TPS) inspired many follow-up studies to investigate and interpret the reaction coordinates. Introduction of the likelihood maximization method based on TPS by Peters and Trout[8] provided a practical approach to investigating reaction coordinates and identifying and ranking important features of conformational space that contribute to the transition pathways. The study of Rogal *et al.* employed the maximum likelihood approach to committor analysis, which proved to be successful in reducing the complexity of the high-dimensional systems and, at the same time, accurately describing the dynamics of molecular transitions.[9] The work of Ma and Dinner utilized a genetic algorithm together with a neural network and proposed reaction coordinates incorporating conformational changes and solvent degrees of freedom in explicit water simulation.[14] The likelihood maximization approach has been used to study many complex chemical processes, such as nucleation problems,[36–41] ion pair association,[42] chemical reactions in solution with quantum mechanical/molecular mechanical models,[43–46] ion incorporation at kink sites during crystal growth,[47] and protein folding.[48,49] Later extensions of this method, including inertial likelihood maximization, provided additional improvements,[50,51] notably effective for inertial barrier crossings, such as those in chemical reactions. The forward flux sampling is also another approach, explored to uncover the reaction coordinate.[52]

Instead of maximizing the likelihood, Mori *et al.* proposed minimizing a cross-entropy cost function,[19,53,54] which, consistent with earlier work,[9,14] also provided an accurate description of the conformational dynamics of the alanine dipeptide in vacuum.

Despite the success of recent studies in determining essential coordinates for transition path analysis, it remains to be seen whether current approaches can be extended to more complex molecular systems. In addition, the increasingly important role played by machine learning techniques in molecular simulation has fostered a diverse collection of regression models that could be used instead of the cross-entropy approach. It remains to be seen whether different machine learning (ML) methods give rise to a similar interpretation of the dynamics. A comparison of multiple regression methods to determine reaction coordinates has the potential to provide a more robust method for reaction coordinates from transition paths.

Following the approach of Refs. 19, 53, and 54, in this work, we introduce a computational framework that identifies essential coordinates from transition path sampling. Figure 1 shows the basic workflow of this study. As illustrated, the first step is to use enhanced sampling methods to obtain the metastable states of the molecule of interest. Methods such as umbrella methods,[55] metadynamics,[56] adiabatic free energy dynamics,[57–60] and adaptive biasing potential methods[61] could be used in this stage. In this study, we employed Unified Free Energy Dynamics (UFED),[62] which allows for the exploration of the high-dimensional free energy surface associated with conformational transitions. Next, we extract conformations at the dividing surface of the metastable states to sample committor probabilities. We trained our models using 17 machine learning methods with a given set of collective variables in order to predict the committor value. We evaluated the performance of each method with a set of rigorous measures.

We find that decision tree-based approaches outperform regression methods in describing the dynamics of the conformational transitions. The light gradient boost machine (LGBM), in particular, gives rise to the highest accuracy among the decision tree methods. Although many ML methods performed well in the alanine dipeptide system, the conformational transition of the disarcosine peptoid served as a more challenging benchmark system for assessing the performance of the ML models. Because the dimensionality of the conformational landscape of the sarcosine dipeptoid is larger than that of the alanine dipeptide, the cross-entropy method used in earlier studies failed to describe the dynamics. In order to identify the key features of a selected conformational transition in this system, we utilized SHapley Additive exPlanations (SHAP) analysis as an alternative method to reveal features of importance for targeted dynamical transitions. SHAP analysis provides a new way to rank and visualize essential degrees of freedom and their interactions with each other. Systematic strategies for incorporating explainability into the optimization of reaction coordinates are discussed in Ref. 35. Numerous other studies have emphasized the significance of interpreting reaction coordinates.[30,34,63] We have found that SHAP analysis, as illustrated in Fig. 1, offers a more comprehensive interpretation of the reaction coordinates. This is because it assigns importance to features and provides an overall understanding of their effect on the outcome through the inclusion of sign information.

Using our computational framework, ML models trained by transition path sampling of the conformational states $C7_{ax}$ and $C7_{eq}$ of the alanine dipeptide in vacuum automatically select the dihedral

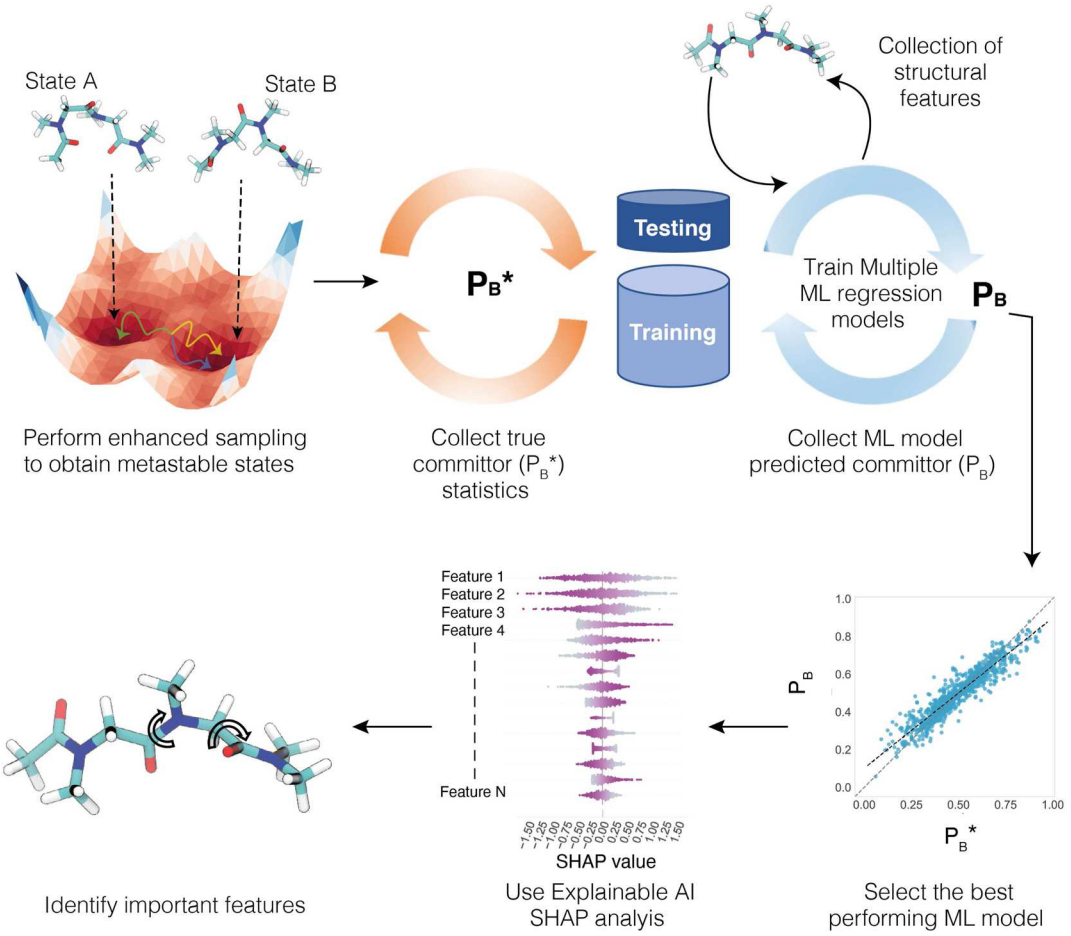

FIG. 1. Steps in the workflow: First, enhanced sampling simulations were performed on the molecule of interest to identify stable conformational states. Then, conformations were extracted from this trajectory, and for each conformation, a set of committor simulations were performed to calculate the true committor values (P_B^*). This provides a committor distribution, and also, from the same conformations, a set of collective variables (CVs) are obtained as molecular input features. Then, multiple regression models were trained to estimate the committor values (P_B), and the best model was selected. For this selected model, explainable artificial intelligence (AI), specifically SHAP analysis in the present study, is used to determine the important CVs from the full set of CVs.

of angles ψ and ϕ among 45 angles. Furthermore, consistent with previous work,[7,19,54] our approach highlights the importance of the angle θ as an essential coordinate to study the transition of this system. In the case of disarcosine in an implicit solvent, we focus on the transition between the states *cis* α_D and *trans* α_D. The LGBM and decision tree approaches identified the ψ and ω angles as the essential degrees of freedom.

This paper is organized as follows. In Sec. II, we describe the theoretical and methodological elements of our approach, including the UFED enhanced sampling algorithm, the modeling of the committor or reaction coordinate in terms of CVs, and the various machine learning methods investigated. In Sec. III, we provide computational details. In Sec. IV, results for the gas-phase alanine peptide and disarcosine in an implicit solvent are presented. Conclusions are given in Sec. V.

II. THEORY

In the following, we summarize the theoretical approaches used in our study.

A. UFED method for enhanced sampling

Exploring the conformational space of molecules is essential to identify the stable regions that dominate the conformational ensemble. An enhanced sampling method that overcomes kinetic barriers allows one to navigate the energy surface effectively. One of the major obstacles in this process is the selection of the CVs on which to apply biasing forces. A reasonable practical strategy is to target a large and possibly redundant set of CVs and bias all of them. This step requires an enhanced sampling method capable of biasing more than three CVs, for which adiabatic techniques,[57–59,62] such as

Unified Free Energy Dynamics (UFED),[62] constitute a viable choice to explore the free energy surface of conformational transitions in higher dimensions. Consequently, we chose to employ the UFED method in this study.

The UFED method incorporates aspects of various enhanced sampling methods, such as the temperature accelerated molecular dynamics[58] or driven adiabatic free energy dynamics,[59] metadynamics,[56] and the use of a biasing force as in adaptive bias force.[64] The UFED method is formulated in an extended phase-space, wherein a set of n dynamical variables $(s_1, \ldots, s_n \equiv s)$ are harmonically coupled to the CVs and are propagated using a set of corresponding fictitious momenta $\pi_1, \ldots, \pi_n$ and mass-like parameters $\mu_1, \ldots, \mu_n$. The aim of UFED is to generate the free energy surface $A(s)$ using the following equations of motion:

$$\begin{aligned} m_i \ddot{\mathbf{r}}_i &= -\frac{\partial U}{\partial \mathbf{r}_i} + \sum_{\alpha=1}^{n} \kappa_\alpha (s_\alpha - q_\alpha(\mathbf{r})) \frac{\partial q_\alpha}{\partial \mathbf{r}_i} + \mathrm{Bath}(T), \\ \mu_\alpha \ddot{s}_\alpha &= -\kappa_\alpha (s_\alpha - q_\alpha(\mathbf{r})) - \frac{\partial U_{\mathrm{bias}}}{\partial s_\alpha} + \mathrm{Bath}(T_s), \end{aligned} \tag{1}$$

where $\mathbf{r}_1, \ldots, \mathbf{r}_N \equiv \mathbf{r}$ are the physical atomic coordinates, m_i are the associated masses, and κ_α are the harmonic coupling constants. The particle equation is coupled to a thermostat at physical temperature T [Bath(T)], and the equation for s_α is coupled to a thermostat at temperature $T_s >> T$ [Bath(T_s)]. In the adiabatic limit, Eq. (1) generates the mean force

$$\begin{aligned} F_\alpha(s) &= -\frac{\partial A}{\partial s_\alpha} = -k_B T_s \frac{\partial}{\partial s_\alpha} \ln P_\kappa(s) \\ &= \langle \kappa_\alpha (q_\alpha(r) - s_\alpha) \rangle_s. \end{aligned} \tag{2}$$

Here, $P_\kappa(s)$ is the high-dimensional histogram. Sampled values of $F_\alpha(s)$ can be used to fit $A(s)$ using a model such as a basis-set expansion[62] or a neural network.[65] The bias potential $U_{\mathrm{bias}}(s)$ in the UFED approach takes the form of a metadynamics-like bias in the extended phase space,

$$U_{\mathrm{bias}}(s,t) = h \sum_{l, t_l < t} e^{-\|s - s(t_l)\|^2 / 2\sigma^2}. \tag{3}$$

The gradient of $U_{\mathrm{bias}}(s,t)$ is added to the equations of motion as indicated in Eq. (1), $\|\cdot\|$ is the L^2 norm, and h and σ are the height and width of the added Gaussians. Moreover, a sparse binning scheme can be employed, where the mean force is accumulated only in populated regions, enabling a robust generation of high-dimensional free-energy surfaces. Further details of the UFED approach can be found in Ref. 62.

B. Modeling committor values from CVs

Once the states are identified, we sample transitions. Transition paths connecting basins serve to find the committor values. Here, the true values are denoted with an asterisk $(^*)$ for the committor, p_B^*, and the reaction coordinate, $r(q)^*$. The corresponding numerical estimates from the ML models are denoted without asterisk for the committor, p_B, and the reaction coordinate, $r(q)$.

In order to train machine learning (ML) models, we adopted two strategies. In the first strategy (method 1), we trained the model based on the values collected directly from true committor values. Here, an implicit mapping from reaction coordinates to committor is implied. In this method, a linear combination of features (q_m), here the complete set of CVs, and the corresponding coefficients (α_m) serve to predict the committor value via

$$P_B = \sum_{m=1}^{M} \alpha_m q_m + \alpha_0, \tag{4}$$

where M is the number of features and α_0 is the bias term that serves to prevent overfitting. The optimized coefficients, α_m, rank the features based on their importance.

In the second approach, following Refs. 8, 50, 53, and 54, we represent the reaction coordinate as a linear combination of features and train our models based on the reaction coordinate value. In this method (method 2), we first compute the reaction coordinate $[r(q)]$, as shown in the following equation:

$$r(q) = \sum_{m=1}^{M} \alpha_m q_m + \alpha_0. \tag{5}$$

Then, as expressed in the following equation, we pass $r(q)$ through an explicit sigmoidal function to obtain the corresponding committor value:

$$P_B = \frac{1}{1 + e^{-r(q)}}. \tag{6}$$

In both methods, some of the machine learning models we studied do not contain explicit coefficients (α_m), and for these cases, combining the feature values (q_m) alone leads to a prediction of the respective committor or reaction coordinate. Therefore, for models based on decision trees, the reduction of variance of the committor or the reaction coordinate approach was employed, as explained later in the text.

C. Machine learning models

The ML models explored in this study can be classified into two major categories. The first category includes standard regression methods. Linear, ridge, lasso, elastic net, and cross-entropy models are included in this group. Regression methods aim to optimize the coefficients of descriptors with the help of a minimization function. The difference between regression methods lies mainly in the residuals used in each method. As an example, the linear regression model obtains optimized coefficients (α_i) by minimizing the residual sum of squares between the predicted values (P_B) and true (P_B^*) values. In linear regression, we minimize the residual sum of squares between the target and the predicted values expressed with

$$\min_{\alpha} \|P_B - P_B^*\|_2^2. \tag{7}$$

Alternatively, the ridge model employs an additional complexity parameter (γ) to avoid overfitting, resulting in the following expression:

$$\min_{\alpha} \|P_B - P_B^*\|_2^2 + \gamma \|\alpha_i\|_2^2. \tag{8}$$

On the other hand, the elastic net model incorporates L_1- and L_2-norm regularization terms in the minimization function with an additional parameter ρ to obtain the optimized coefficients, i.e.,

$$\min_{\alpha} \frac{1}{2\, n_{samples}} \|P_B - P_B^*\|_2^2 + \rho\, \gamma \|\alpha_i\|_1 + \frac{\gamma\,(1-\rho)}{2} \|\alpha_i\|_2^2. \quad (9)$$

The second category of ML methods tested uses ensemble methods based on decision trees. Generally, in decision trees, categorical variables use the "information gain." For continuous variables such as the committor value, we aim at reducing the variance

$$\sigma(P_B^*) = \frac{\sum (P_B^* - \overline{P_B^*})^2}{n}, \quad (10)$$

which is achieved by splitting the nodes, where P_B^* is the target feature value and $\overline{P_B^*}$ is the mean of feature P_B^* for the number of data points n.

For a continuous input feature, such as a dihedral angle, the values are sorted in ascending order, and each unique dihedral value is used to split the committor data. Then, for each child node, we calculate the variance [Eq. (10)]. For each split, we calculate the weighted average variance of the child nodes. We select the split that produces the child node with the lowest weighted average variance. These steps are repeated until all data points are separated.

These models use the concept of ensemble methods, where an ensemble of weak learners is trained to improve the model prediction in contrast to a single strong learner. Weak learners are decision trees, and their outputs are combined to deliver better outcomes. The ensemble models can be categorized into two types: bagging and boosting. Boosting methods, the primary focus of this study, adopt the strategy of sequentially adding weak learners to the model and filtering out the observations that a learner captures correctly at every step. Next, they develop new weak learners to handle the remaining misclassified observations. The final prediction is the average prediction of the many learners parallel sampling the same dataset.

The light gradient boosting machine model, which is our prime focus in this study, uses a gradient boosting type of algorithm.[67] The main steps of the gradient boosting algorithm are given in Table I. Here, we first compute the predictions of the "base model" $F_0(x)$, which is set to the mean value. This is shown in the first line of Table I, L is the least-squares loss function, y_i is the target value (P_B^*), and γ_m are the predicted values (P_B). This results in the base model $F_0(x)$ being the mean value of the target variable y_i. M is the number of iterations in the for loop, representing the M decision trees, and N is the number of data points. The next step, as shown in line 3, is to calculate the pseudo-residuals $(\tilde{y}_i)$ of existing $m-1$ models. This is the difference between the target and the predicted one, which is denoted by the negative direction gradient through a loss function. We then use the pseudo residuals as the target values and perform the mth training step. Here, the mth prediction model is $h(x;\ a_m)$ and $\{a_m\}$ are its parameters. Each decision tree computes a different multiplier (γ_m). The new ensemble model $F_m(x)$ is computed by updating the previous ensemble model $F_{m-1}(x)$ by linear superposition. The hyperparameter p_r is the learning rate, which is the regularization term in the range of 0–1. The superior performance of the light gradient boosting machine model has been reported recently.[68–70]

TABLE I. The gradient boosting algorithm adopted from Ref. 66. The variable definitions and algorithm steps are explained in detail in the text.

1	$F_0(x) = \arg\min_{\gamma} \sum_{i=1}^{N} L(y_i - \gamma)$
2	***for*** $m = 1$ to M ***do***
3	$\tilde{y}_i = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}, \quad i = 1, N$
4	Generate mth prediction model $h(x;\ a_m)$;
5	$a_m = arg\ min_{a,\zeta} \sum_{i=1}^{N} [\tilde{y}_i - \zeta\ h(x_i; a_m)]^2$
6	$\gamma_m = arg\ min_{\gamma} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \gamma\ h(x; a_m))$
7	$F_m(x) = F_{m-1}(x) + p_r \gamma_m h(x;\ a_m)$
8	***end for***

D. Cross-entropy minimization method in regression

The maximum likelihood approach in Ref. 8 uses a linear combination of features together with a hyperbolic tangent functional form $P_B = (1 + \tanh(r(q)^k))/2$ to calculate the predicted committor (P_B) value (referred to as the MXLK-t model). It uses the following equation to derive the likelihood maximization:

$$L(\alpha) = \prod_{q^k \to B} P_B(r(q^k)) \prod_{q^k \to A} (1 - P_B(r(q^k))). \quad (11)$$

$(q^k \to B)$ denotes the number of trajectories reached at state B out of a total number of shooting moves performed for each snapshot in the committor analysis. Similarly, $(q^k \to A)$ denotes the number of trajectories reached at state A. $r(q^k)$ is the set of collective variables at shooting points.

A modified version of the maximum likelihood approach is utilized by Jung *et al.* in Ref. 15 that uses a neural network to predict the reaction coordinate. Here, $[r(q)]$ is passed through a sigmoidal functional form [similar to Eq. (6)] to obtain the predicted committor (P_B) value (referred to here as the MXLK-s model). The following equation is used in parameterization:

$$l_{MT_2} = \sum_{n=1}^{N} [x_B^n\ \ln(1 + e^{-r(q,n)}) + x_A^n\ \ln(1 + e^{r(q,n)})], \quad (12)$$

where x_A^n and x_B^n are the numbers of trajectories out of the total number of committor shoots performed for a given conformation and $r(q,n)$ is the trained reaction coordinate with the nth conformation. As an extension of the maximum likelihood method, the use of the cross-entropy minimization (CREM) method to model committors has gained popularity recently.[19,53,54]

Maximum likelihood is reformulated to minimize the cross-entropy by Mori *et al.* in Ref. 54 that uses a linear combination of features together with a hyperbolic tangent function $P_B = (1 + \tanh(r_q))/2$ to acquire the predicted committor (P_B) value (referred to as the CREM-t model),

$$l_{MT_1} = -\sum_{n=1}^{N} P_{B,(n)}^*\ \ln\ P_{B,(n)}\ -\ \sum_{n=1}^{N} (1 - P_{B,(n)}^*)\ \ln\ (1 - P_{B,(n)}), \quad (13)$$

which guarantees that the true committor value P_B^* and the calculated committor value P_B are similar. The number n is the conformation index, and N is the total number of conformations.

Furthermore, we investigated the cross-entropy minimization approach employed by Mori *et al.* with Eq. (13) together with a sigmoidal functional form [used Eq. (6) instead of tanh] to obtain the predicted committor value (P_B) (referred to as the CREM-s model).

In addition, for MXLK-s, CREM-s, and CREM-t methods similar to Ref. 54, we adopt a L2-norm regularization to avoid overfitting,

$$\hat{l}_{MT_x(L2\ \text{norm})} = l_{MT_x} + \frac{\lambda}{2}\sum_{m=1}^{M} \|\alpha_m\|^2, \tag{14}$$

where the λ hyperparameter adjusts the strength of the trained coefficients α_m excluding the bias (α_0). M is the total number of features.

Since both methods employ a sigmoidal functional form and a L2-norm regularization, as method 1, we chose CREM-s and, as method 2, we selected MXLK-s in the comparison.

E. SHAP analysis to derive the importance of features

Once the training is complete, it is desirable to identify the important CVs that contribute to the conformational transition. The conventional approach for deriving this information is to use optimized coefficients ranked according to their amplitude or "feature importance" if the model is based on decision trees. Different ML models use different types of algorithms in the training process. This makes it difficult to compare various types of trained ML models.

Alternatively, interpretable ML algorithms, such as SHAP[71,72] (SHapley Additive exPlanations), can be utilized to quantify the importance of the features and interpret the transition mechanism. The Shapley values (ϕ_i) were first introduced by Shapley[71] and later incorporated as an interpretable ML algorithm in SHAP by Lundberg and Lee.[72]

The Shapley value of a feature is its contribution to the predicted value, weighted and summed over all possible feature value combinations. The Shapley value ϕ_i is computed using the following equation:

$$\phi_i = \sum_{S\subseteq F\setminus\{i\}} \frac{|S|\,!\,(|F|-|S|-1)\,!}{|F|\,!}\,[f_{S\cup\{i\}}(x_{S\cup\{i\}}) - f_S(x_S)]. \tag{15}$$

As explained in detail in Ref. 72, S is a subset chosen from the set F containing all the features. For a given feature $\{i\}$, a model $f_{S\cup\{i\}}$ is trained with that feature present, and another model f_S is trained without the feature. Then, the predictions of the two models are compared for the current input $f_{S\cup\{i\}}(x_{S\cup\{i\}}) - f_S(x_S)$, where x_S denotes the values of the input features in the subset S. In this way, the Shapley score value (ϕ_i) corresponds to the average of the marginal contribution across all features and feature subsets of the dataset.

Extension of the Shapley values to SHAP inherits the concepts of game theory to explain model predictions. Here, SHAP starts with some base value for prediction based on prior knowledge and attempts to add features of the dataset one by one and to understand the impact of the added feature on the final prediction. SHAP specifies the explanations as

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z_i'. \tag{16}$$

The explanation model prediction $[g(z')]$ is computed with Eq. (16), where ϕ_0 is the base value, that is, the output of the model if all input features are disregarded. Most of the time, it is the average of the values of the predicted feature. ϕ_i is the Shapley value, where $z_i' \in \{0,1\}^M$ is the simplified input feature vector and M is the number of input features.

There are several flavors of SHAP interpretation algorithms available depending on the ML model. For LGBM, we used TreeExplainer, and for CREM, we used LinearExplainer. However, there are also model-agnostic SHAP Explainer types, such as KernelExplainer. Note that the SHAP values should only be used to interpret the model. It cannot be used to evaluate the quality of the trained model.

Features with large absolute Shapley values are important. In order to obtain a global importance measure, we compute the average of the absolute Shapley values per feature across the dataset as in Eq. (17), where I_j is the global importance rank of the feature, n is the number of data points, and $|\phi_j|$ is the absolute Shapley value for the jth feature. Then, the descending order of I_j gives the global feature ranking,

$$I_j = \frac{1}{n}\sum_{i=1}^{n} |\phi_j|. \tag{17}$$

In addition to the global ranking of dominant features for the entire projection, we use SHAP calculations to determine a local feature importance hierarchy. We also utilized SHAP calculations to determine the feature interactions or feature correlations within a chosen feature set.

III. COMPUTATIONAL DETAILS

A. MD simulation setup and conformational sampling

In order to investigate various ML models, we selected two benchmark systems: (i) the alanine dipeptide in vacuum and (ii) the disarcosine peptoid in implicit water. Interatomic interactions are represented using AMBER99SB-ILDN.[73] We used the improved generalized Born solvent model for implicit solvent model calculations (GBn2).[74] The simulation box sizes were set to 3.6 × 3.6 × 3.6 nm^3 for the alanine dipeptide and 6.0 × 5.4 × 5.6 nm^3 for the disarcosine peptoid. The lengths of all bonds involving hydrogen atoms were constrained using the SHAKE[75] algorithm. The equations of motion were solved using a time step of one femtosecond and the geodesic Langevin integrator[76,77] with the friction coefficient set at 10.0 ps^{-1}. The sampling was performed in the NVT ensemble, with a target temperature of 300 K for alanine dipeptide and 500 K for disarcosine. All simulations were performed using the OpenMM[78] MD engine.

We used the unified free energy dynamics (UFED)[62] approach to enhance conformational sampling. The dihedral angles ϕ_3 and ψ_2 of the alanine dipeptide served as slow variables. For the

disarcosine system, we used the five dihedral angles of the backbone $[\phi_2, \phi(-1)_2, \psi_1, \psi(-1)_1, \omega_1]$ to enhance conformational sampling. UFED calculations were performed using the UFEDMM Python package[77] implemented in OpenMM. The masses of the extended phase-space variables were set at 30 D/(nm rad^2), the force constants were set to 1000 kJ/(mol rad^2), and the temperature of the extended variables was set to 1500 K. Simulations of length 300 ns were used to sample the conformations. We saved the coordinates every 1 ps interval for data analysis. We projected the sampled data onto free energy surfaces of two dimensions for each system in order to locate the metastable states and the transition state regions. Note that this is a low-dimensional projection of the five-dimensional surface generated for the disarcosine peptoid.

B. Committor analysis between stable conformational states

For a reactive transition path, the committor value P_B is defined as the probability that a trajectory initiated at the given conformation with randomized velocities drawn from a Maxwell Boltzmann distribution will arrive in state B before arriving in state A. The starting conformations in state A have $P_B = 0$, and the conformations in state B have $P_B = 1$.

The committor values in this study were calculated by shooting 100 times from each snapshot sampled in the vicinity of the transition state region using the path sampling algorithm[8] and monitoring whether each trajectory arrives in state B before state A. Each simulation was seeded with random velocities and was terminated as soon as the trajectory entered one of the two states. A preliminary grid search was conducted on the free energy surface to find a location close to the transition state where the committor values deliver a range between 0 and 1.

The OpenPathSampling[79–81] Python package "Committor-Simulation" functionality was used to perform committor simulations. For both the alanine dipeptide and the disarcosine peptoid, ≈6000 snapshots from the vicinity of the transition state region were randomly extracted from the UFED trajectories to generate an initial pool for the committor analysis. Moreover, for each of the shoots, the corresponding initial dihedral values were recorded together with the committor values. Consequently, each type of molecule involves more than 600 000 trajectories. Then, from the snapshot pool, a normal distribution of ≈3500 data points was randomly extracted, where the P_B mean is centered around ≈0.5, as shown in Figs. 2(c) and 9(d), to train the ML models. Shapiro and Wilk[82] values for the extracted normal distributions were calculated to ensure that the p-value is greater than 0.05. Learning rate plots, as shown in the Sec. IV, were used to ensure that the size of the distribution used was adequate to obtain the convergence of the trained ML models.

C. Machine learning (ML) analysis

The true committor values (P_B^*) together with the corresponding dihedral features (45 for alanine dipeptide and 66 dihedrals from disarcosine) comprised the dataset. First, each dihedral feature is converted to sine and cosine angles to account for the

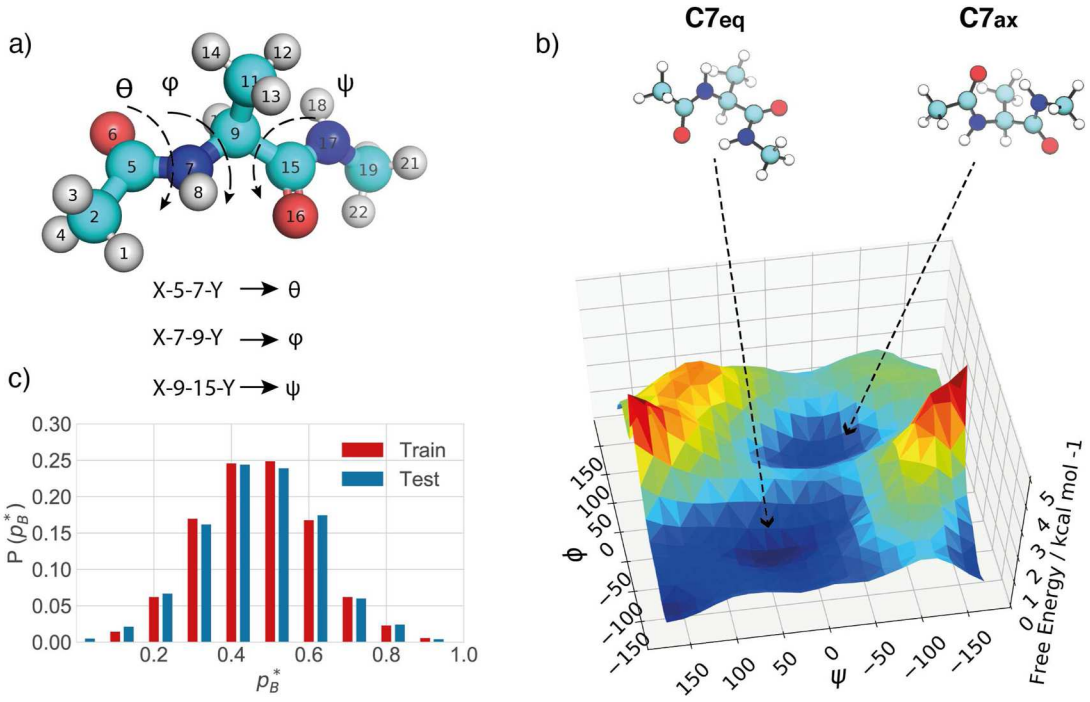

FIG. 2. (a) The alanine dipeptide molecule with atom indices and dihedral angles as abbreviated in Table S1. (b) The conformations at $C7_{eq}$, $C7_{ax}$ states and the free energy surface of the alanine dipeptide in vacuum projected to ϕ_3 and ψ_2 dihedral angles. (c) The true committor probability distribution for the trained and test data.

periodicity of the feature. The whole dataset is then normalized with the z-score method, which scales the feature values to range from −2 to 2. In addition, if the two features have perfect colinearity, then one of them was randomly removed from the dataset. The dataset was then divided into 80:20 ratios for the training and testing datasets. In every training of the ML model, a 10-fold cross-validation was used. Other than the maximum likelihood and CREM models, for which we generated our own code, we employed standard libraries, such as Scikit-learn,[83,84] Yellowbrick,[85] SHAP,[86–88] pandas,[89] PyCaret,[90] matplotlib,[91] seaborn,[92] numpy,[93] and scipy[94] in this process. We used the visual molecular dynamics (VMD)[95] software for molecular visualization.

Here, 17 different types of regression ML models [Light Gradient Boosting Machine (LGBM),[67] Extra Trees Regressor (EXTR),[96] Random Forest Regressor (RAFR),[97] Gradient Boosting Regressor (GRBR),[98] Decision Tree Regressor (DECT), Bayesian Ridge (BAYR),[99] Orthogonal Matching Pursuit (ORMP),[100] AdaBoost Regressor (ADAB),[101] Ridge Regression (RIDR), Linear Regression (LINR), Huber Regressor (HUBR), K Nearest Neighbors Regressor (KNNR), Passive Aggressive Regressor (PAGR), Lasso Regression (LASR),[102] Elastic Net (ELAN), Lasso Least Angle Regression (LLAR), and Cross-Entropy Minimization (CREM)] were trained. For method 1, we trained the models on the true committor values (P_B^*), and for method 2, we used the reaction coordinate $[r(q)^*]$ obtained by inversing the sigmoidal functional form with the true committor value of Eq. (6).

Furthermore, for each ML model in this study, six different types of typical regression assessing metric values [MAE (Mean Absolute Error), MSE (Mean Squared Error), RMSE (Root Mean Squared Error), R^2 (coefficient of determination to assess whether the regression model fits the true data; this ranges from 0 to 1 and the higher the values the better the fit), RMSLE (Root Mean Squared Log Error), and MAPE (Mean Absolute Percentage Error)] as the average of the ten-fold cross-validation scores were calculated for the training dataset. The models were then arranged according to the R^2 values to choose the best-performing model. However, it should be noted that when comparing methods 1 and 2, in method 1, the true committor (P_B^*) value ranges from 0 to 1, while in method 2, the true reaction coordinate $[r(q)^*]$ ranges from −5 to 5. Therefore, for comparison between the two methods, in Tables S IV and S V presented for method 2, we have included scores calculated for the predicted reaction coordinates $[r(q)]$ subsequently converted to the committor value (P_B) using Eq. (6). To avoid the singularity in Eq. (6), we set $P_B^* = 0.0001$ when $P_B^* = 0$ and $P_B^* = 0.9999$ when $P_B^* = 1$. Moreover, for feature ranking, the SHAP values were calculated with the Python package Shap (https://github.com/slundberg/shap) implemented by Lundberg and Lee[72] using the final trained ML model together with the test dataset.

IV. RESULTS AND DISCUSSION

A. Alanine dipeptide in vacuum

To evaluate the performance of the machine learning (ML) models, we used the alanine dipeptide in vacuum as our first test system. We generated the conformational free-energy surface using the Unified Free Energy Dynamics (UFED) approach, with the two dihedral angles (ϕ, ψ) as our coarse variables. In Figs. 2(a) and 2(b), we show the model system and the resulting free-energy surface. We identified the metastable states ($C7_{eq}$, $C7_{ax}$) from the free-energy surface and the range of ϕ and ψ values that define these states (Table II), and we identified the dividing surface that separates the two basins by visually inspecting the free-energy surface (FES). We then sampled transitions from the dividing surface to either of the two states to obtain the committor values. It is worth noting that this approach does not necessitate an exact determination of the location of the transition state. For the construction of the committor (method 1) or reaction coordinate (method 2), we used all 45 dihedral angles as features. The atom indices used to define these dihedrals are shown in Fig. 2(a), and Table S1 provides a detailed list of these dihedrals. Figure 2(c) shows the distribution of committor values (P_B^*).

The committor distributions represented in Fig. 2(c) were based on the training data for the machine learning (ML) models described in Sec. III. The generated committor data were then divided into two groups: training and test data. We used a ten-fold cross-validation method to assess the metrics of each model. As different metrics measure different aspects of accuracy, we report and rank the models based on these metrics, summarized in Tables SIII and SIV of the supplementary material.

The results of the performance of the models are also visually summarized in Fig. 3. This time we focused only on the root mean square error (RMSE) and the cross correlation score (R^2). Surprisingly, most ML models show that $R^2 > 0.6$ and RMSE values are below 0.15. Lasso Least Angle Regression (LLAR), Lasso Regression (LASR), and Elastic Net (ELAN) exhibit relatively poor performance, while other models accurately predict the committor values of the alanine dipeptide constructed from the full set of dihedral angles. The performance of the models shows a weak dependence on the choice of the mathematical representation of the reaction coordinate (method 1 or method 2).

A notable observation is that the LGBM model provides the highest R^2 score and lowest RMSE for both the training and test datasets using both methods. Therefore, we selected it as our optimal method of choice. It is worth noting that it is possible to further fine-tune the hyperparameters of other ML models to improve their accuracy. However, for the sake of simplicity, we focused on comparing the LGBM model to the CREM model, which has been used in earlier studies.[19,53,54]

To assess the convergence of our results, we first compare the learning rate of the two ML models. We computed the model prediction score as a function of training instances. We examined the training and cross-validation scores separately. In comparison, the results are displayed in Fig. 4 for LGBM and CREM. The learning rates of MXLK are shown in Fig. S13(b). Regarding learning rates, CREM shows faster convergence; however, its accuracy is less than

TABLE II. Alanine dipeptide metastable state definitions and the transition state (TS) region used to extract snapshots for the committor analysis.

States	ϕ_3 range	ψ_2 range
$C7_{eq}$	$-130 \geq and \leq -30$	$0 \geq and \leq 180$
$C7_{ax}$	$30 \geq and \leq 130$	$-180 \geq and \leq 0$
Snapshots extracted	$-30 \geq and \leq 20$	$-80 \geq and \leq -30$

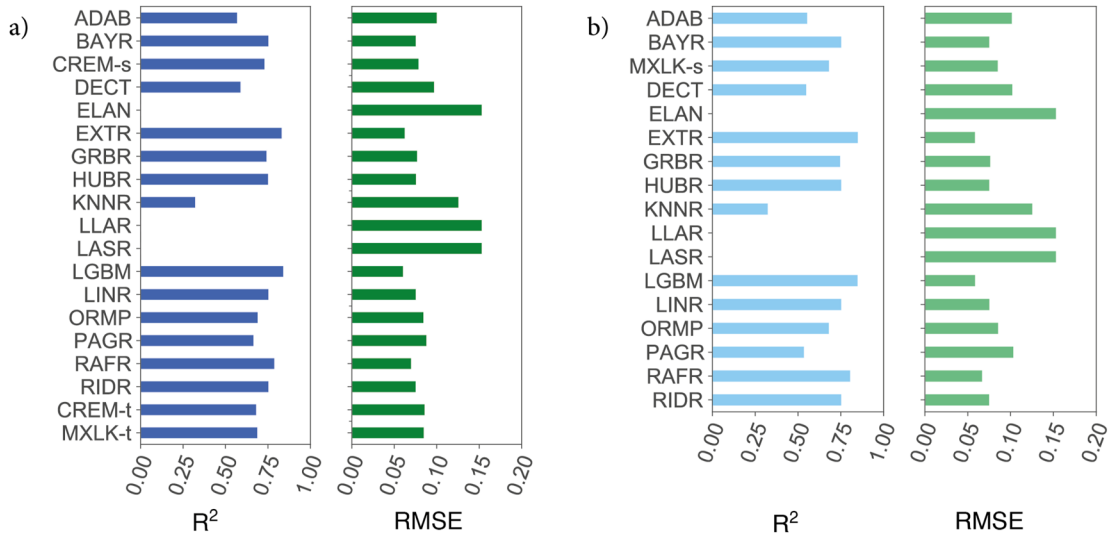

FIG. 3. The alanine dipeptide in vacuum, RMSE, and R^2 values for different ML models trained (a) with method 1 and (b) with method 2. The method 2 output $r(q)$ is converted to P_B values as demonstrated in Table SIV. Training is done with the AdaBoost Regressor (ADAB), Bayesian Ridge (BAYR), Cross-Entropy Minimization (CREM), Decision Tree Regressor (DECT), Elastic Net (ELAN), Extra Trees Regressor (EXTR), Gradient Boosting Regressor (GRBR), Huber Regressor (HUBR), K Neighbors Regressor (KNNR), Lasso Least Angle Regression (LLAR), Lasso Regression (LASR), Light Gradient Boosting Machine (LGBM), Linear Regression (LINR), Orthogonal Matching Pursuit (ORMP), Passive Aggressive Regressor (PAGR), Random Forest Regressor (RAFR), and Ridge Regression (RIDR). MXLK-t follows the likelihood maximization approach with the tanh functional form reported by Peters and Trout;[8] MXLK-s employs the likelihood maximization with a sigmoidal function [similar to Eq. (6)] used by Jung *et al.*[15] CREM-t uses the tanh functional form reported by Mori *et al.*,[54] and CREM-s is using the cross-entropy minimization with sigmoidal function [Eq. (6)]. In (a) and (b), the method 1 and method 2 models are aligned for comparison. Bars with zero R^2 values are not displayed.

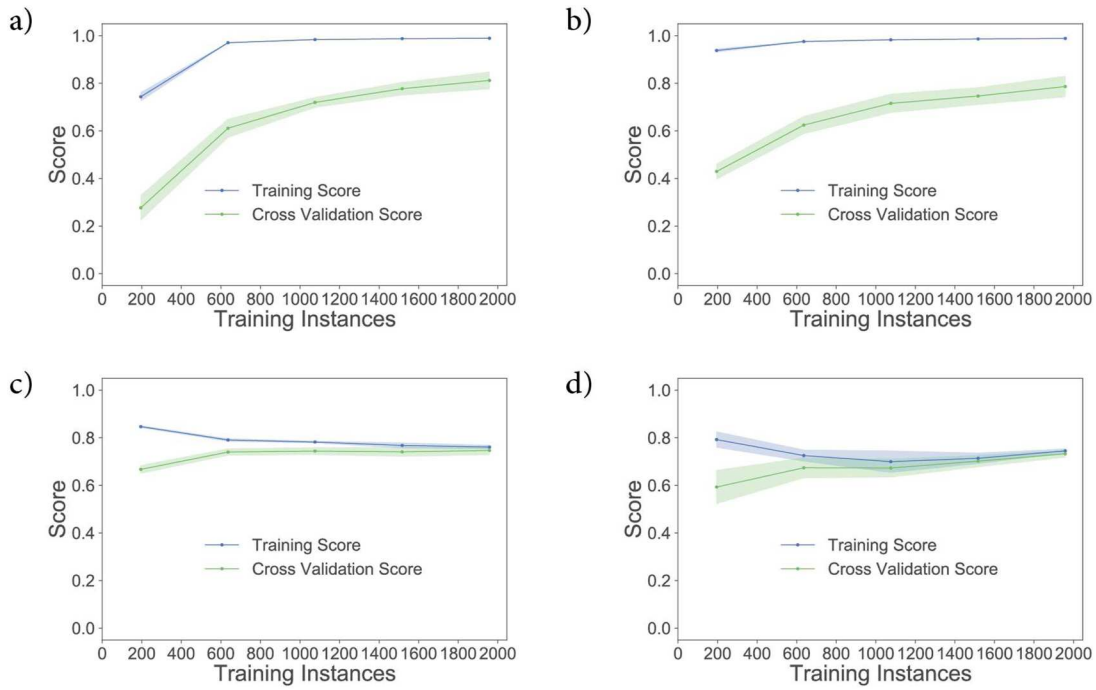

FIG. 4. Alanine dipeptide in vacuum learning rate plots from the two selected ML models: (a) LGBM method 1, (b) LGBM method 2, (c) CREM method 1 (CREM-s), and (d) CREM method 2 (MXLK-s).

that of the LGBM. The learning rate curves for LGBM and CREM converge about 600 data points for training scores in both ML models. While CREM training was completed after 600 data points for the cross-validation score, LGBM continued to improve as more data were added to the pool.

To provide a more detailed comparison of the accuracies offered by these two selected machine learning models, we present correlation plots of the test datasets. These plots assess the accuracy of the methodology by showing the diagonal relation between the actual data (x axis) and its predicted value (y axis). We compare the LGBM model using two mathematical representations [Figs. 5(a) and 5(b)]. We also show the MXLK comparison in the supplementary material [Fig. S15(a)]. We observe a similar performance for method 1 and method 2. The major difference is observed when we compared the two machine learning models (LGBM and CREM) [Figs. 5(a) and 5(b) vs Figs. 5(c) and 5(d)]. Similar to the differences in RMSE and R^2 scores, LGBM shows a slightly better performance than CREM in providing higher cross correlations and better diagonal fit of the data. Note that the difference is minor between the ML models.

After the training, the coefficients extracted from the model rank the features based on their importance. For example, features with high weight α_m in CREM suggest that this feature is essential to determining the committor value and likely plays a role in the conformational transition. To examine the feature importance analysis of each method, we compared the features selected by the LGBM and CREM models. Figure 6 shows the ranking of the dominant features based on each method and the ML model.

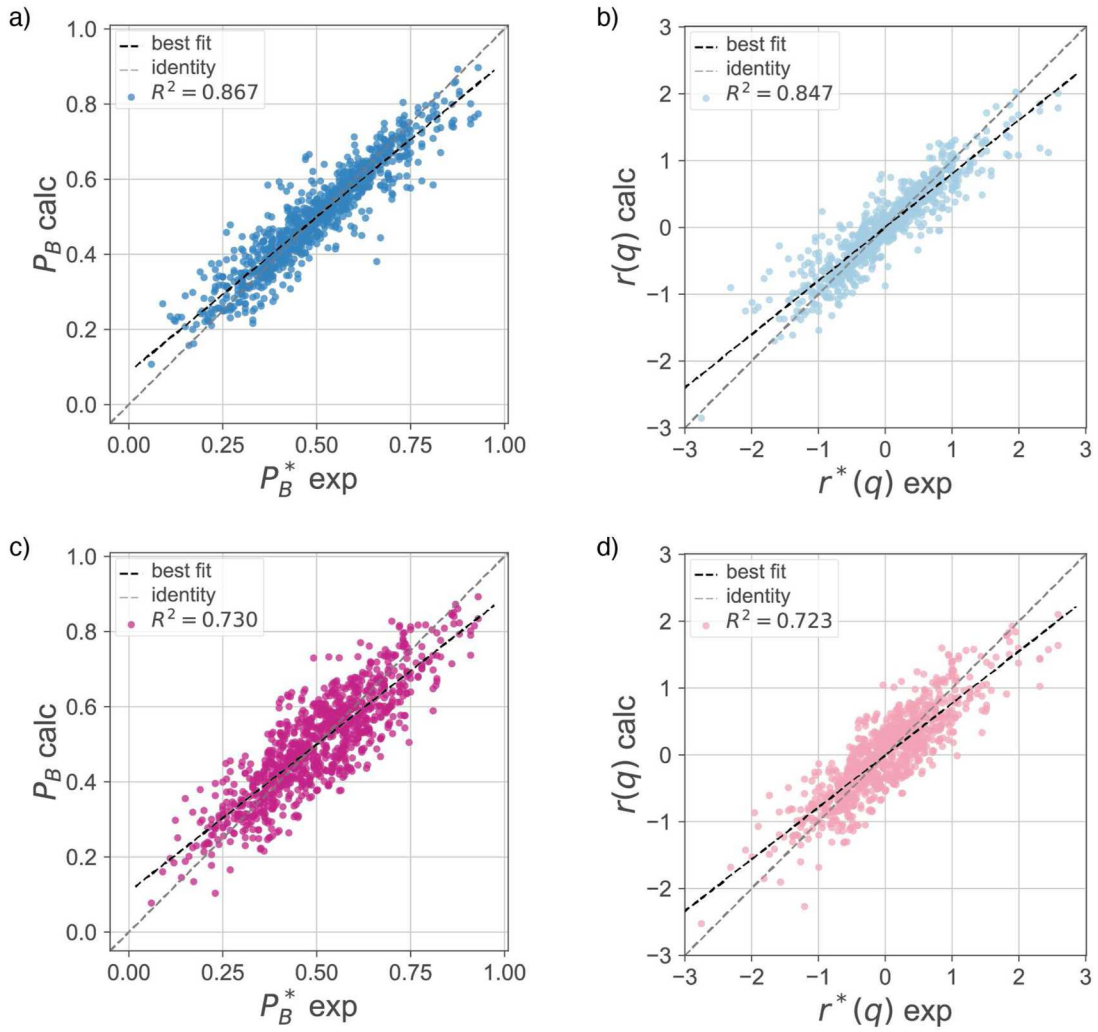

FIG. 5. Alanine dipeptide in vacuum correlation plots for the test data from the two ML models: (a) LGBM method 1, (b) LGBM method 2, (c) CREM method 1 (CREM-s), and (d) CREM method 2 (MXLK-s). P_B^* exp denotes the true committor value, while P_B calc denotes the model prediction.

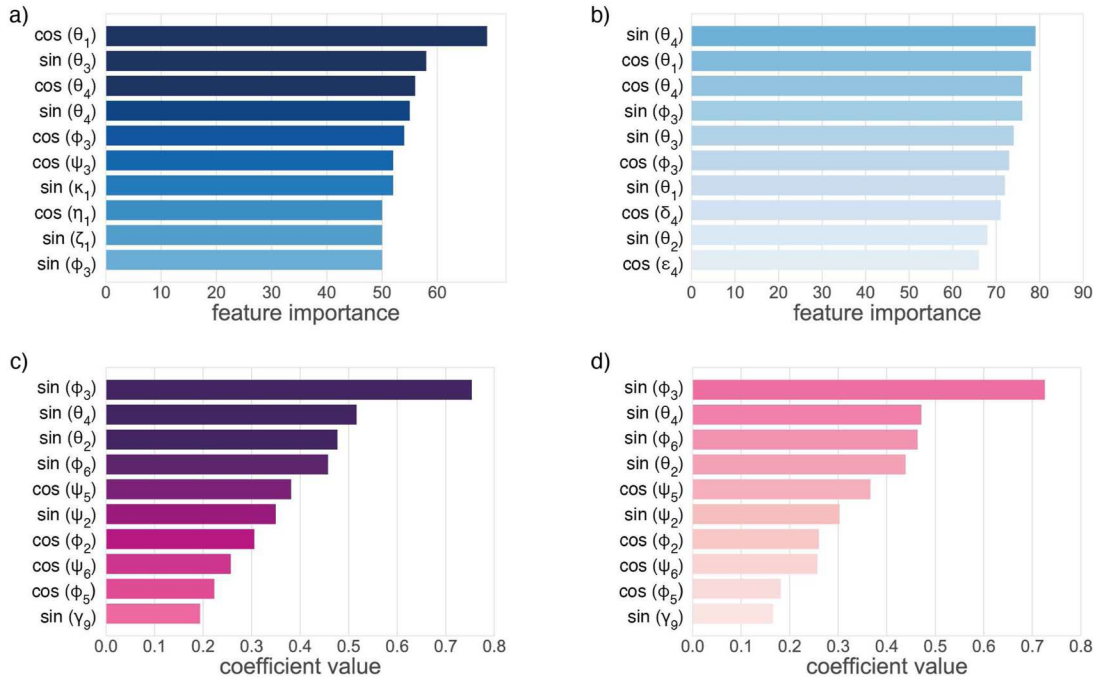

FIG. 6. Alanine dipeptide in vacuum feature ranking plots for the two ML models: (a) LGBM feature importance using method 1, (b) LGBM feature importance using method 2, (c) optimized coefficient values for CREM method 1 (CREM-s), and (d) optimized coefficient values for CREM method 2 (MXLK-s). Dihedral abbreviations are displayed in Fig. 2(a) and in Table SI.

Feature importance analysis allows for providing a ranking between the features. Interestingly, in both ML models and methods, the three angles θ, ϕ, and ψ were ranked high for the transition of alanine dipeptides C_{ax} to C_{eq} in vacuum. The importance of ϕ and ψ for alanine dipeptide is well established. Studies have also reported the importance of the angle θ, especially when the alanine dipeptide is in vacuum.[7,14,19,54,103] In our study, minor differences were observed between the two methods in ML models. Based on the LGBM model, the highest-ranked features are θ_1, θ_3, θ_4, and ϕ_3 dihedrals. In contrast, the CREM model identified ϕ_3, θ_4, θ_2, and ϕ_6 as the most important features.

Another approach to uncovering details of transition pathways is by using SHapely Additive exPlanations (SHAP) analysis. SHAP analysis provides a more in-depth analysis of the essential features and offers insights into how each feature contributes to the observed committor value. Details of the SHAP methodology can be found in Sec. II and in Refs. 71 and 72. In this study, we focus on how this algorithm, coupled with the ML model, provides a detailed mechanistic understanding of the dynamics of the conformational transitions in the case of alanine dipeptides.

Figure 7 displays the SHAP global feature ranking in a SHAP summary plot for the ML models. A SHAP summary plot displays the features ranked from most significant to least significant (y axis), similar to the analysis of the importance of the feature in Fig. 6. The x axis of the summary plot reports the distribution of conformations projected on the SHAP value for each important feature (Fig. 7). In addition, the color bar in the summary plot establishes a relationship between the feature and the outcome, here in our case, the committor value. The distribution of data points centered around a zero SHAP value suggests that an increase/decrease of the feature does not impact the predicted committor value. A distribution centered on a *positive* SHAP value suggests that the increase in the feature leads to an *increase* in the predicted committor value. Similarly, the distribution of the SHAP value localized in the *negative* region implies a negative correlation.

We will now examine the insights derived from the two models using SHAP analysis and compare our results with the analysis of feature importance in Fig. 6. The SHAP analysis of the LGBM model identifies ϕ and θ dihedrals as essential, in agreement with previous reports.[7,14,54,103] Note that the exact dihedrals identified differ from the "feature importance" analysis displayed in Figs. 6(a) and 6(b), but the importance of θ and ϕ is captured in both approaches. For the LGBM model, method 1 implies that dihedrals ϕ_2, ϕ_3, θ_4, and θ_2 are essential to describe the dynamics. For method 2, the model suggests the same dihedrals with a slight change in ranking. As the color bars indicate, higher $\sin\phi_2$ feature values negatively correlate with the value of the committor. In contrast, higher $\sin\phi_3$ feature values positively correlate with the value of the committor for the alanine dipeptide in a vacuum.

In Figs. 7(c) and 7(d), we present the SHAP analysis for the CREM model. Similar to LGBM, the model selects five to six dihedrals to explain the committor value. In agreement with LGBM, the CREM model identifies ϕ_3, θ_4, θ_2, and ϕ_6 as important degrees of

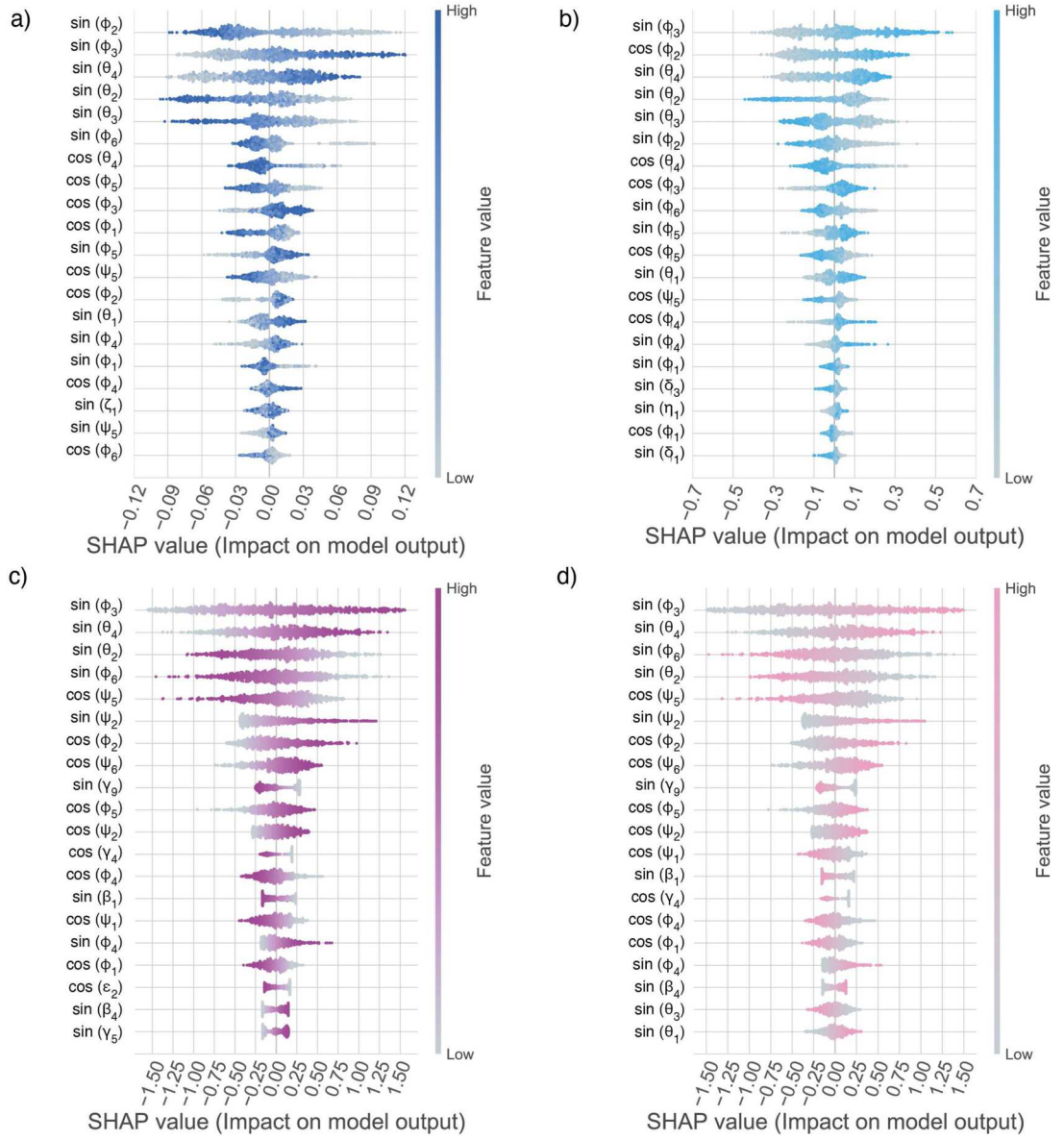

FIG. 7. Alanine dipeptide in vacuum SHAP feature ranking plots for the two ML models: (a) LGBM method 1, (b) LGBM method 2, (c) CREM method 1 (CREM-s), and (d) CREM method 2 (MXLK-s). Dihedral abbreviations are displayed in Fig. 2(a) and in Table SI. The *y* axis ranks the features from the most important (top) to the least (bottom).

freedom. Furthermore, the CREM model also suggests that ψ_5 and ψ_2 are a secondary set of important features [Figs. 7(c) and 7(d)]. Strikingly, the feature importance displayed in Fig. 6 and the SHAP analysis of the CREM model produce the same rankings for the top set. The SHAP analysis and optimized CREM coefficients share the same ranking in the alanine dipeptide. The top-ranked features of method 1 and method 2 are the same, with minor variations in the order observed. These results are similar to the work of Kikutsuji *et al.*[19] using a different version of SHAP analysis.

SHAP values computed based on Eq. (15) allow for the assessment of the importance of each feature, which is then used to rank them. In addition to the global feature importance computed based on the entire range of the committor, the decision plots monitor the convergence of local feature rankings to various committor values. This way, we monitor how our ML model improves by adding new features to describe specific committor values, namely, P_B = 0, 0.25, 0.50, 0.75, and 1 (Figs. S9–S12). Based on this analysis, we conclude that including five to six features is sufficient to achieve

high accuracy in predicting the committor values based on dihedral angles.

The differences in the feature ranking approaches give rise to which method is more accurate in describing the dynamics of alanine dipeptides. To address this, we trained the models using feature importance ranking (Fig. 6) or SHAP feature ranking (Fig. 7) by adding features one by one. We computed the RMSE values for the top N input features of the LGBM and CREM models. The results are shown in Fig. 8.

We found that the SHAP ranking generally resulted in lower RMSE values than the feature importance ranking for LGBM. Both method 1 and method 2 displayed similar accuracy for alanine dipeptides. Unlike the SHAP rankings, the feature importance ranking resulted in higher RMSE values for the same number of features. Interestingly, the two representations (method 1 and method 2) also show variations.

LGBM with SHAP analysis resulted in a highly accurate description of the dynamics, even with the first four dihedrals. The CREM model, however, shows a high RMSE value suggestive of poor prediction with the first four features selected. All versions of CREM give rise to the same dependence on several input features, and accuracy remains lower than the LGBM-SHAP combination.

Based on LGBM coupled with SHAP, we identify four dihedrals in alanine dipeptide that capture the dynamics of $C7_{eq}$ to $C7_{ax}$ in vacuum. Our automated methodology reduces the conformation space from 45 dihedrals to four. The selected angles ϕ and θ give rise to high precision in describing the dynamics. Our approach suggests that the dihedral ψ, often used to monitor the transition, is of secondary importance in describing the dynamics of the alanine dipeptide in vacuum.

The feature importance analysis helps to interpret transition pathways robustly by reducing degrees of freedom. With all plots in Fig. 7, it is clear that only a handful of the 45 dihedral angles (90 features) play a crucial role. The consensus of the selected features by the ML models confirms the general usefulness of ML in identifying reaction coordinates of conformational transitions.

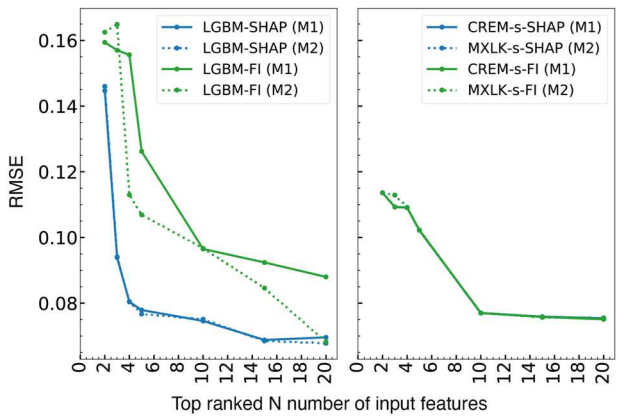

FIG. 8. RMSE score for the N number of top-ranked features trained instead of the full set of features for the two ML models for alanine dipeptides in vacuum using method 1 (M1) and method 2 (M2). The method 2 $r(q)$ values are converted to P_B values before calculating the respective RMSE values. The green color is for the feature importance (FI) ranking in Fig. 6, and the blue color is for the SHAP ranking in Fig. 7. The solid lines are for method 1, and the dashed lines are for method 2.

The SHAP feature ranking calculations treat all input features as independent, but correlations may exist between them. To identify these correlations, we used interaction plots. An interaction plot visualizes the SHAP values projected on two features simultaneously, allowing for a more detailed examination of relationships between features that may show variations between ML models. Our analysis used a decision tree-based correlation analysis for LGBM and a linear SHAP interaction plot for CREM. From the top-ranked global feature rankings (Figs. S1–S4), we selected the four highest correlating input features. On the x axis of the plots, we display the top-ranked global feature value, while the color bars indicate the most correlated feature selected from the pool. For instance, Fig. S1(b) indicates that for LGBM method 1, $\sin\phi_2$ is the top-ranked global feature highly correlated with $\cos\theta_4$, $\sin\theta_4$, $\sin\theta_2$, and $\cos\theta_2$. Dark color bars correspond to higher values of $\sin\theta_4$ on the x axis and negative SHAP values, implying that these two dihedral combinations lower the predicted committor value.

B. Disarcosine peptoid in implicit water

As our second example, we studied the conformational transition of a peptoid system. Peptoids are a class of peptidomimetic oligomers composed of N-substituted glycine units. Despite their inability to form hydrogen-bond networks, they adopt stable three-dimensional structures not accessible by standard peptides. Peptoids exhibit notable characteristics, such as the ability to introduce diverse side-chain functionalities and resistance to hydrolytic degradation by proteases. As a result, they have become potential candidates for biomedical applications with superior biocompatibility and potent biological activities.[104–110]

We focused on the conformational transition of the disarcosine peptide [Fig. 9(a)]. Due to its relatively complex structure and the use of an implicit water model in our study, this system posed a greater challenge than the alanine dipeptide. Disarcosine has 66 dihedral angles, listed in Table SII. Following the convention,[105,107] the conformational state is characterized by middle backbone angles, designated here as $\phi(-1)_2$, $\psi(-1)_1$, ω_1, ϕ_2, and ψ_1.

In order to sample the conformational space of disarcosine, we used UFED simulations, in which all middle backbone dihedrals (five CVs) were targeted for enhanced sampling. The free energy surface was then projected onto the dihedral angle pairs ϕ_2 and ψ_1 and ψ_1 and ω_1 [Figs. 9(b)–9(d)]. Previous studies[105–107] reported that the *cis* α_D and *trans* α_D states are essential regions of the conformational space. Consistent with these studies, we observe distinct minima corresponding to those states [Figs. 9(b) and 9(c)]. Selecting conformations at the dividing surface between the two basins (Table III), we shot unbiased trajectories to sample transition paths. In Fig. 9(d), we show the computed committor distributions constructed for the training and test datasets.

We applied the ML approaches described earlier to predict the committor values based on the dihedrals input. Tables SV and SVI show the mean of the ten-fold cross-validation results obtained from the ML models trained on the dataset. Similarly to Sec. IV A, we evaluated the performance of two mathematical representations, referred to as method 1 and method 2. We present the performance of each approach based on R^2 and RMSD in Fig. 10.

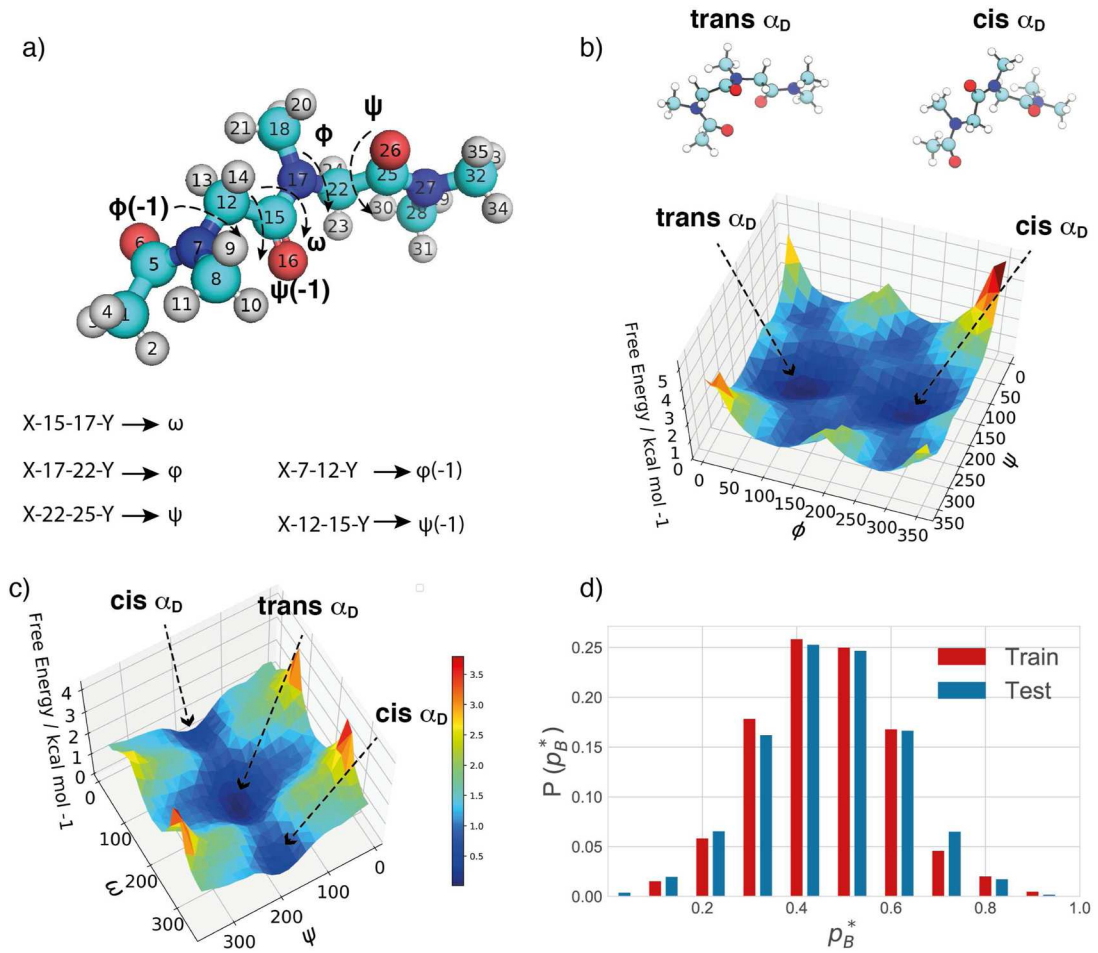

FIG. 9. (a) The disarcosine molecule with atom indices and dihedral angle abbreviations (Table SII). The two stable conformers *cis* α_D, *trans* α_D states and the UFED free energy map projected onto the angles (b) ϕ_2 and ψ_1 and (c) ω_1 and ψ_1. (d) The true committor probability distributions for test and train datasets.

TABLE III. The disarcosine conformational states and the range used to define the transition state (TS) for committor analysis.

States	ϕ_2 range	ψ_1 range	ω_1 range
cis α_D	$200 \geq and \leq 350$	$150 \geq and \leq 250$	$(0 \geq and \leq 50)$ or $(300 \geq and \leq 360)$
trans α_D	$50 \geq and \leq 150$	$150 \geq and \leq 250$	$125 \geq and \leq 225$
Snapshots extracted	$150 \geq and \leq 200$	$150 \geq and \leq 250$	$250 \geq and \leq 300$

Due to the added complexity of disarcosine, we observed an overall reduction in the ML models' performance. The ML models that performed well on the alanine dipeptide system showed a similar trend with disarcosine, suggesting that the performance rankings are independent of the specific molecule under study. The LGBM remained the best-ranked among the ML models (see Tables SV and SVI) with an R^2 score of 0.75 and an RMSE score of 0.07. The correlation plots remained high with the computed and predicted data and stayed diagonal for methods 1 and 2 (see Fig. 12). One striking observation we made is that, while the training scores converge with about 600 data points, the training scores continued to increase with added data, suggesting that further improvements could be possible for LGBM.

In contrast, the CREM model's R^2 score fell to around 0.4, with an RMSE greater than 0.1 for both methods, suggesting a poor description of the committor values. Learning curves show no improvement after 600 training instances [Figs. 11(c) and 11(d)]. The cross correlation plots also fail to hold their

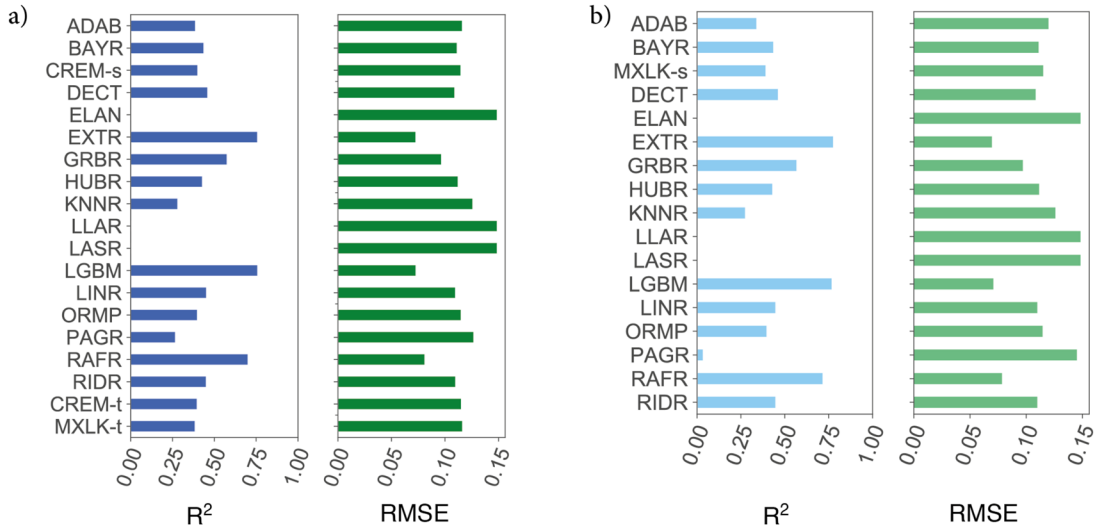

FIG. 10. Disarcosine in implicit water, RMSE, and R^2 values for different ML models trained (a) with method 1 and (b) with method 2. The method 2 output $r(q)$ converted to P_B values as demonstrated in Table SVI. We benchmarked the ML methods described in Fig. 3. Results with zero R^2 values are not shown.

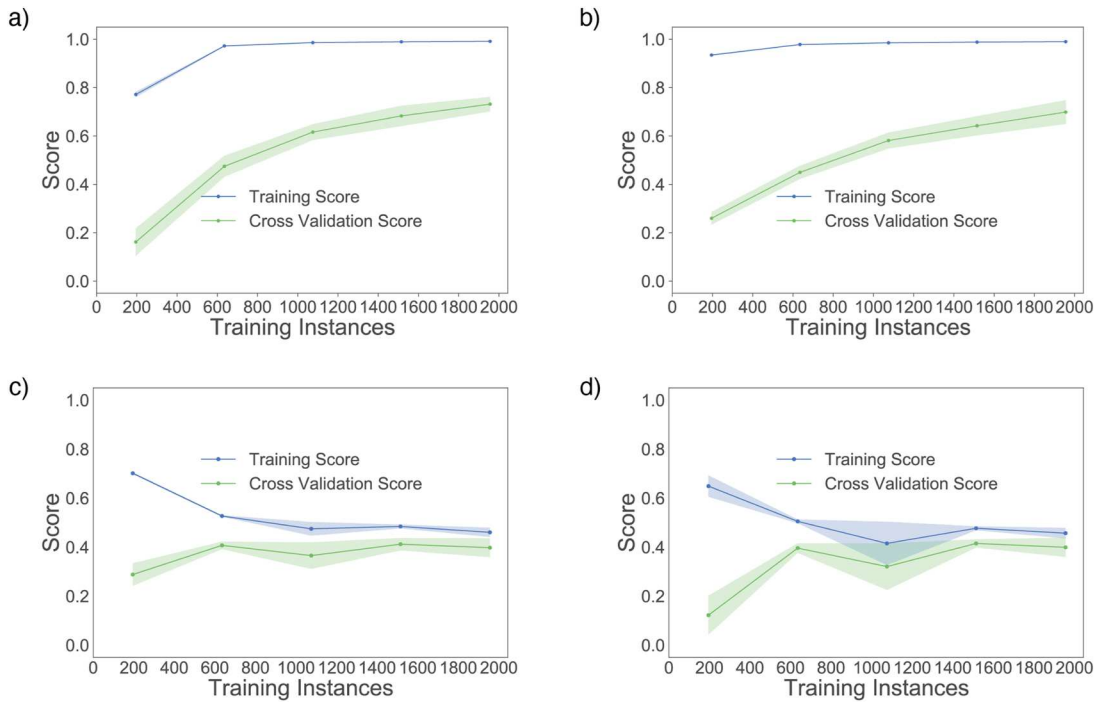

FIG. 11. Disarcosine in implicit water learning rate plots from the two ML models: (a) LGBM method 1 and (b) LGBM method 2. (c) CREM method 1 (CREM-s) and (d) CREM method 2 (MXLK-s).

diagonal shape, and correlation scores in CREM [Figs. 12(c) and 12(d)] remain low in disarcosine, unlike in the case of alanine dipeptide. The learning rates of MXLK also show the same trend [Fig. S14(b)].

As the CREM model did not perform well, we focused on LGBM. We investigated the important features leading to the conformational transition of disarcosine between the two states. We looked at features selected by the LGBM feature extraction method

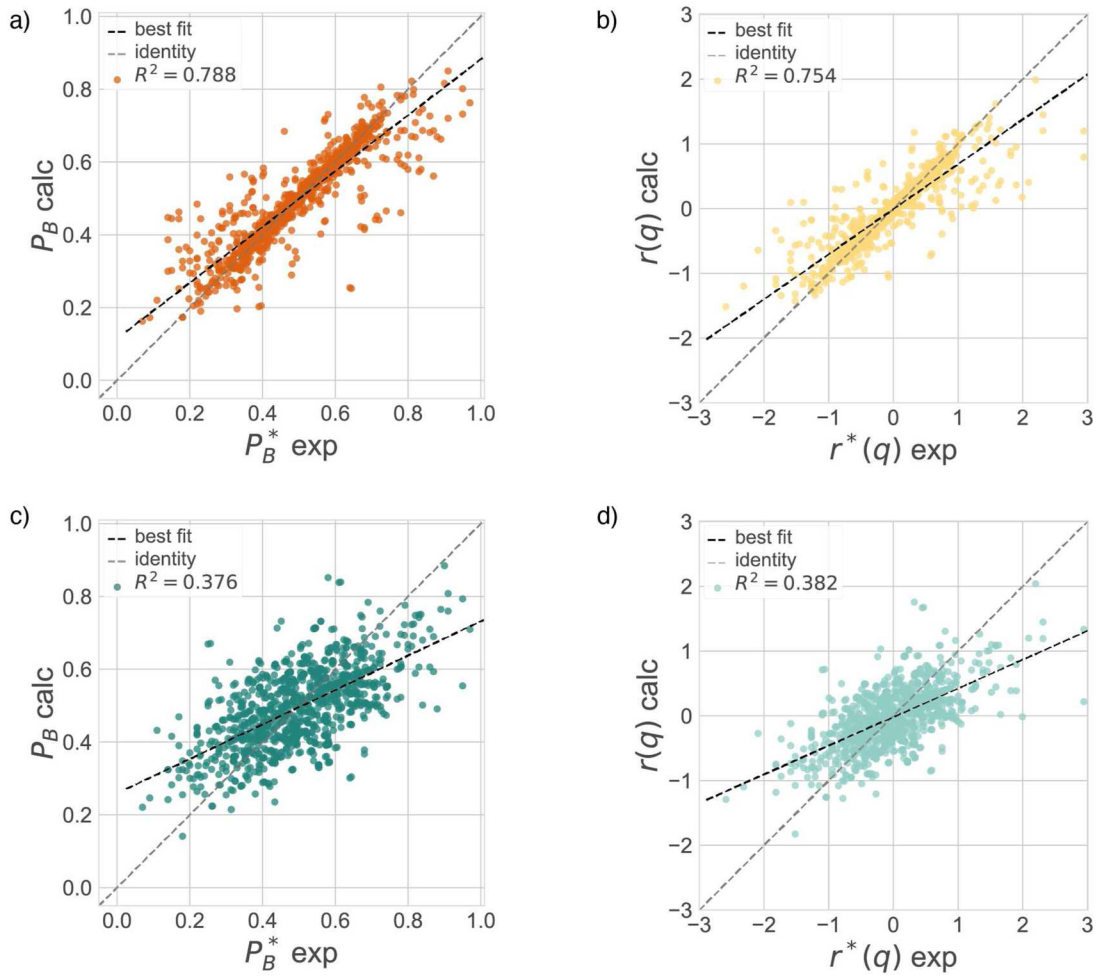

FIG. 12. Disarcosine in implicit water correlation plots for the test data from the two ML models: (a) LGBM method 1 and (b) LGBM method 2. (c) CREM method 1 (CREM-s) and (d) CREM method 2 (MXLK-s). P_B^* exp denotes the true committor value, while P_B calc denotes the model prediction.

(LGBM-F) and LGBM coupled with the SHAP feature extraction method (LGBM-SHAP). Results are summarized for comparison in Figs. 13 and 14.

Previous studies highlight the importance of ψ_1, ω_1, and ϕ_2 dihedral angles.[105–107] In Fig. 13, we show what the ML algorithm with method 1 identifies $\sin\omega_3$ as the most important feature, followed by $\cos\phi_2$ and $\cos\omega_3$. On the other hand, method 2 shows fairly different dihedrals in the top five rankings. The angles identified by method 1 were picked but with lower feature importance scores. The discrepancies observed between the two mathematical representations suggest problems in the LGBM-feature ranking combination as the complexity of molecular structure increases.

The LGBM coupled with SHAP analysis, on the other hand, identifies $\sin\psi_2$, $\cos\omega_3$, $\cos\omega_1$, and $\cos\omega_4$ as the five essential degrees of freedom for both method 1 and method 2. A slight variation in the ranking order is the only difference between the two representations, suggesting that LGBM-SHAP is a more consistent model to provide essential features.

Unlike the discrepancy in feature rankings between mathematical representations, LGBM coupled with SHAP global ranking offers a more consistent picture that weakly depends on the choice (method 1, 2). Then, we turn our attention to the SHAP interpretation. We first check the important features locally. SHAP feature ranking analyses were performed for values of $P_B \approx 0$, 0.25, 0.5, 0.75, 1 from the test dataset, as shown in the decision plots for each method (Figs. S13–S16). One striking difference is that the number of CVs needed to accurately represent the committor values becomes higher than that of the alanine dipeptide, likely due to the added complexity. Instead of four to five dihedrals that were sufficient to describe committor values with set accuracy in the case of alanine dipeptides, SHAP analysis suggests about ten features to describe the transition in the case of disarcosine.

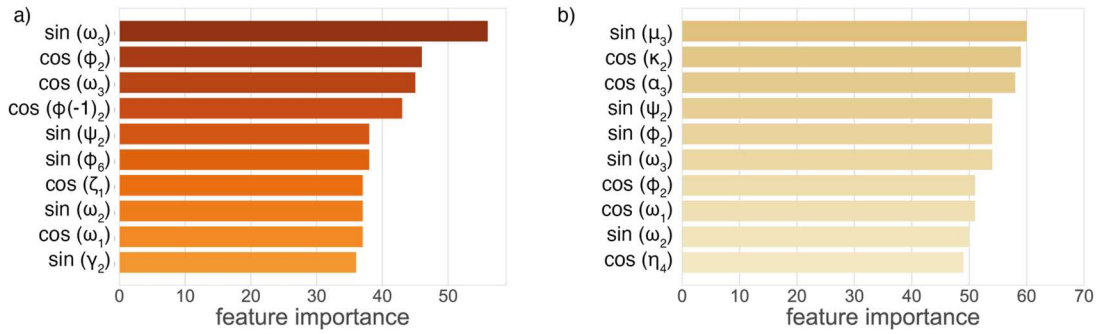

FIG. 13. Disarcosine in implicit water feature ranking plots for the two models: (a) feature importance value from LGBM method 1 and (b) feature importance value from LGBM method 2. The dihedral abbreviation is displayed in Fig. 9(a) and in Table SII.

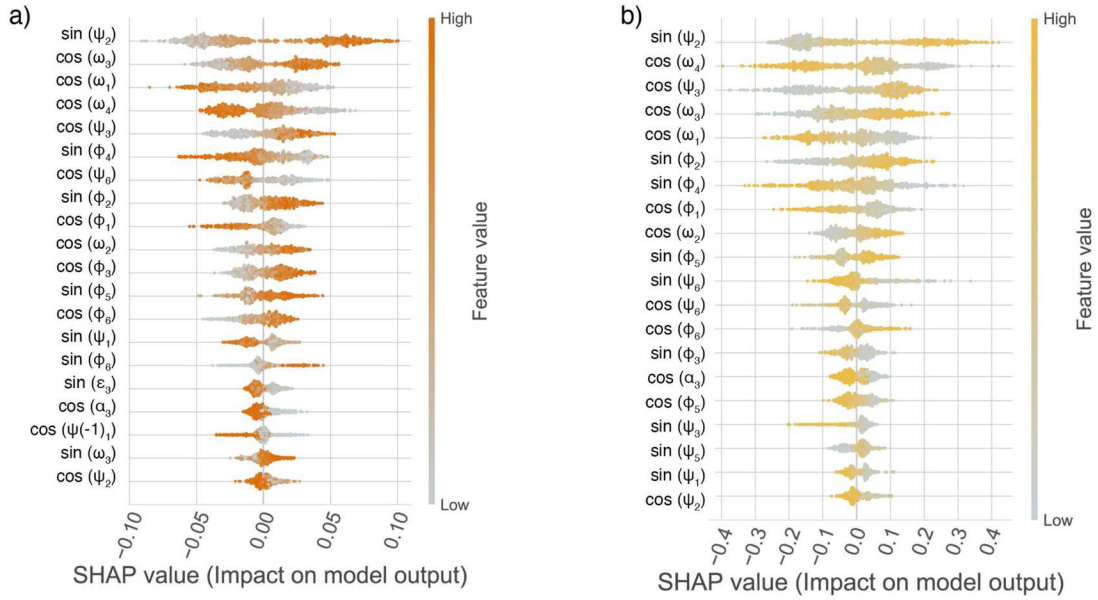

FIG. 14. Disarcosine in implicit water SHAP feature ranking plots from the LGBM model: (a) LGBM method 1 and (b) LGBM method 2. The dihedral abbreviations are displayed in Fig. 9(a) and in Table SII. The *y* axis represents the features ranked from the most important to the least.

In order to assess which feature extraction method gives rise to a more accurate description of the dynamics, we compared the impact of the N highest ranked input features of the LGBM model based on the conventional feature importance (Fig. 13) with the SHAP feature ranking (Fig. 14). Figure 15 compares the two approaches. Similar to the alanine dipeptide example, a lower RMSE is observed for the LGBM-SHAP combination compared to feature importance. This trend remains the same for both methods (method 1 and method 2).

While visualizing ten angles is challenging, the analysis in Fig. 15 and in Fig. 14 suggests that the first four features give rise to an RMSE score small enough to reduce the dimensionality further. Based on these features, we uncover a new dihedral angle, ψ_2. Interestingly, this feature does not change between the endpoint $cis\alpha_D$ and $trans\alpha_D$ states and has not been identified in earlier studies. Our approach, based on dynamic information, rather than looking at the variance at the metastable states, highlights the transient yet crucial hidden features in the transition pathway. Our approach allows for a more accurate representation of the reaction mechanism. Note that this feature might be overlooked by intuition-based or basin-based approaches.[111] In addition to ψ_2, the LGBM model identifies multiple ω dihedrals as crucial dihedrals. These features serve almost equally with ψ_2 in determining the predicted committor values. Interestingly, unlike ψ_2, these angles show variation when disarcosine transits between *cis* α_D to *trans* α_D states.

Through the coupling of LGBM and SHAP global analysis, we have identified transient features that never show variation at the metastable states and features that show variation at the end states in one framework. This robust approach can

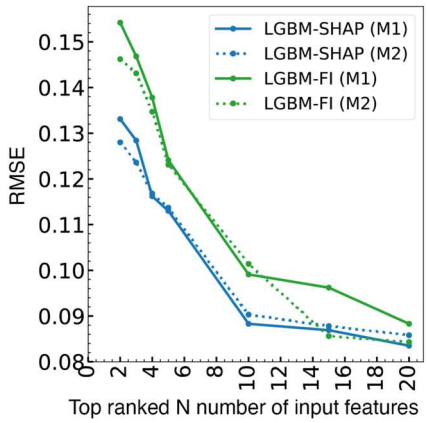

FIG. 15. Disarcosine in implicit water, *N* number of top-ranked features trained instead of the full set of features in the training dataset, and the calculated RMSE values for the LGBM model with both method 1 (M1) and method 2 (M2). The method 2 $r(q)$ values are converted to P_B values before calculating respective RMSE values. The green color is for the feature importance (FI) ranking in Fig. 13, and the blue color is for the SHAP ranking in Fig. 14. The solid lines are for method 1, and the dashed lines are for method 2.

potentially describe the features that play a key role in the transition state and identify the metastable states of the conformational transitions. To elucidate the relationships between the essential degrees of freedom, we looked at their interactions to study whether these two groups of features (ψ, ω) are correlated (Figs. S5 and S6). Interestingly, we observe a high level of correlation between the two degrees of freedom; ψ_2 interacts with ω_4, ω_3, and ω_1, likely due to its proximity or due to the geometric constraints imposed by the covalent architecture on the molecular system.

The LGBM method coupled with SHAP analysis identifies ψ_2, ω_3, and ω_1 as essential degrees of freedom, in addition to ψ_1 and ϕ_2 used to separate the *cis* and *trans* states (Table III) [Fig. 16(a)]. To visualize the committors projected onto these essential coordinates, we plot the selected angle distributions from the transition path conformations at different committor values [Fig. 16(b)]. The committor values provide a monotonic change along the reaction coordinate obtained from these angles. Based on our methodology, along the transition path from *cis* α_D to *trans* α_D, the disarcosine angles ω 1,3 rotate ~60° anticlockwise, while the angles ψ_2 change ~30° in the opposite direction.

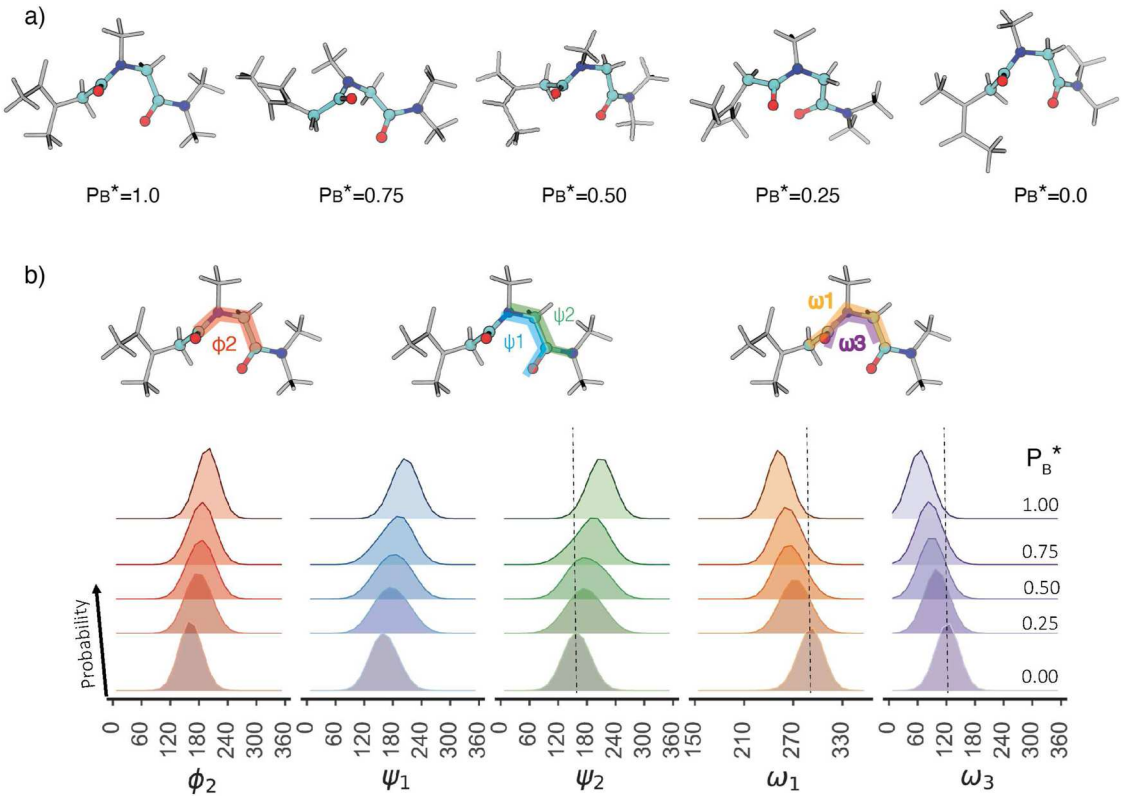

FIG. 16. Disarcosine in implicit water and top-ranked dihedrals and their change during the transition from *cis* to *trans* α_D states (as shown in the free energy plot in Fig. 9). (a) Representative starting conformations show true committor values of P_B^* of 1.0, 0.75, 0.50, 0.25, and 0.0. (b) Dihedral angle probability distribution of starting conformations at true committor values of P_B^* of 0, 0.25, 0.5, 0.75, and 1.0. The dihedral angles are highlighted above the plots with colors according to the probability distribution.

V. CONCLUSIONS AND DISCUSSIONS

We investigated the performance of various machine learning techniques to determine the essential degrees of freedom for conformational dynamics. Seventeen different ML methods were tested on two model systems. Our results suggest that decision tree approaches perform better than regression methods in our application, possibly due to their ability to capture nonlinear relationships, identify complex interactions between variables, and handle noisy data. Linear regression models assume linear, additive relationships between variables and are sensitive to outliers. LGBM, a decision tree-based method, overcomes the limitations of other gradient-boosting decision tree methods by leveraging two novel techniques: Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). Details of how these techniques improve performance can be found in Ref. 67. We collected a significant number of configurations to ensure statistical convergence and evaluate the performance of each ML method. The specific amount of data required for the task vary based on the desired level of accuracy set by the user. Our findings, as represented in Figs. 4 and 11, demonstrate that 600 configurations are adequate to achieve the necessary accuracy.

We studied the conformational transition $C7_{ax}$ to $C7_{eq}$ of the alanine dipeptide in vacuum and the transition of disarcosine from $cis\alpha_D$ to $trans\alpha_D$ in an implicit solvent. ML methods successfully identified a subset of features; however, our study showed that decision tree-based methods give rise to a better description of the dynamics with smaller prediction errors in recapitulating the committor values. Although the CREM approach shows promising results for the gas-phase alanine dipeptide dynamics, CREM did not provide an accurate description of the dynamics as the degrees of freedom increased in the case of disarcosine. Among the methods investigated, LGBM performed consistently better for different model systems and using different cost functions.

We employed feature extraction approaches to gain insights into the transition path's dynamics. In addition to the classical feature extraction functionalities of CREM and LGBM, we employed SHAP as a new tool to probe essential features and their interactions. We showed that it could provide unprecedented detail in elucidating complex molecular transitions. We also see that although CREM feature extraction and SHAP analysis provide similar features, LGBM feature extraction and SHAP analysis differ. We find that LGBM coupled with SHAP analysis provides the most robust description of the conformational dynamics of the two systems under study.

For the alanine dipeptide in vacuum, SHAP analysis for the LGBM model of both methods implies that the essential coordinate is θ_2. This feature predominantly interacts with ϕ_2, ϕ_1, and ϕ_3, which are also top features in rankings. Similarly, SHAP analysis suggests that the two dominant features (θ and ϕ) complement each other to produce a given committor value. The SHAP analysis of the CREM method chooses ϕ_3 as the top-ranked features for both mathematical representations. Similarly, the LGBM model ranks ϕ_3 and θ_2 and shows that these two variables complement each other. Our analysis further supports previous studies, suggesting the two critical degrees of freedom ϕ and θ for alanine dipeptides in vacuum.

We find that for predicting an alanine dipeptide, a linear combination of features based on Eq. (4) is enough to achieve high accuracy. However, for more complex molecules or explicit water models, neural networks may be a better option due to their versatility. By studying disarcosine in an implicit solvent, we introduce higher dimensions and complexity that allow for the critical assessment of the ML models. The LGBM model, which performs optimally with SHAP analysis, suggests ψ_2 as the most important feature. Interestingly, the change in this angle is transient, so it is impossible to be detected by end-state analysis of the metastable states. This angle is coupled with ω_4, ω_2, and ϕ_2.

In this study, we focus on vacuum and implicit solvent systems, and we examined dihedral angles as features. In a forthcoming study, we plan to investigate explicit water models with different collective variables that involve long-range contacts and solvent degrees of freedom. In our application, the committor relies on static data from geometric features. An alternative strategy, especially for the gas phase conformational transitions, is to incorporate the dynamics of atom positions into the features. This could be done using inertial likelihood maximization,[51] which we plan to explore in future studies. Peters demonstrated that binomial deconvolution, instead of the committor estimate, can significantly decrease the computational expense of committor analysis by a factor of 10 or more in maximum likelihood.[34,112] Similarly, the effectiveness of LGBM could be enhanced by training it on binary outcome data because decision trees can act as classifiers just as can be done with the maximum likelihood method. Studying complex biomolecular transitions, such as large conformational transitions occurring in enzymes or allosteric transitions, would be interesting. In addition, we plan to explore more complex machine-learning models. Our findings indicate that to represent the alanine dipeptide accurately, a linear combination of features based on Eq. (4) is sufficient. However, other models, such as neural networks, may, due to their versatility, be a better option for more complex molecules or systems involving explicit water. Specifically, models such as neural networks may provide greater accuracy, enabling the study of complex molecular transitions involving solvent degrees of freedom.

SUPPLEMENTARY MATERIAL

The supplementary material includes alanine dipeptide and sarcosine dipeptoid molecules, dihedral angle definitions with atom indices, regression model comparison tables, SHAP feature interaction plots, SHAP decision plots, learning curves, and correlation curves comparing LGBM with the MXLK-t approach.

ACKNOWLEDGMENTS

M.E.T. acknowledges support from the National Science Foundation, Grant No. CHE-1955381. S.K. and M.E.T. also acknowledge support from NYUAD REF under Grant No. RE317. S.K. and N.N. acknowledge NYUAD Faculty funding under Grant No. AED181. M.E.T. and M.T. acknowledge support from the U.S. Department of Energy, Grant No. DE-SC0020971 M0003. This research was carried out on the High-Performance Computing resources at New York University Abu Dhabi. In addition, we thank David W. H. Swenson for his assistance with the OpenPathSampling package.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Nawavi Naleem: Data curation (lead); Formal analysis (lead); Software (equal); Validation (lead); Visualization (lead); Writing – original draft (equal); Writing – review & editing (supporting). **Charlles R. A. Abreu**: Software (lead); Supervision (equal); Validation (equal); Writing – review & editing (equal). **Krzysztof Warmuz**: Data curation (supporting); Formal analysis (supporting); Investigation (supporting); Validation (supporting); Writing – review & editing (supporting). **Muchen Tong**: Data curation (equal); Investigation (equal); Validation (equal). **Serdal Kirmizialtin**: Conceptualization (equal); Funding acquisition (equal); Investigation (equal); Methodology (supporting); Project administration (equal); Resources (lead); Writing – original draft (supporting); Writing – review & editing (equal). **Mark E. Tuckerman**: Conceptualization (equal); Funding acquisition (equal); Methodology (lead); Project administration (equal); Supervision (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] W. Lechner, J. Rogal, J. Juraszek, B. Ensing, and P. G. Bolhuis, "Nonlinear reaction coordinate analysis in the reweighted path ensemble," J. Chem. Phys. **133**, 174110 (2010).

[2] S. Wu, H. Li, and A. Ma, "A rigorous method for identifying a one-dimensional reaction coordinate in complex molecules," J. Chem. Theory Comput. **18**, 2836–2844 (2022).

[3] R. T. McGibbon, B. E. Husic, and V. S. Pande, "Identification of simple reaction coordinates from complex dynamics," J. Chem. Phys. **146**, 044109 (2017).

[4] B. M. Bonk, J. W. Weis, and B. Tidor, "Machine learning identifies chemical characteristics that promote enzyme catalysis," J. Am. Chem. Soc. **141**, 4108–4118 (2019).

[5] P. Novelli, L. Bonati, M. Pontil, and M. Parrinello, "Characterizing metastable states with the help of machine learning," J. Chem. Theory Comput. **18**, 5195–5202 (2022).

[6] C. Dellago, P. G. Bolhuis, and P. L. Geissler, "Transition path sampling," in *Advances in Chemical Physics, edited by I. Prigogine and S. A. Rice* (John Wiley & Sons, Ltd., Chichester, England, UK, 2002), pp. 1–78.

[7] P. G. Bolhuis, C. Dellago, and D. Chandler, "Reaction coordinates of biomolecular isomerization," Proc. Natl. Acad. Sci. U. S. A. **97**, 5877–5882 (2000).

[8] B. Peters and B. L. Trout, "Obtaining reaction coordinates by likelihood maximization," J. Chem. Phys. **125**, 054108 (2006).

[9] J. Rogal, W. Lechner, J. Juraszek, B. Ensing, and P. G. Bolhuis, "The reweighted path ensemble," J. Chem. Phys. **133**, 174109 (2010).

[10] P. M. Piaggi, O. Valsson, and M. Parrinello, "Enhancing entropy and enthalpy fluctuations to drive crystallization in atomistic simulations," Phys. Rev. Lett. **119**, 015701 (2017).

[11] Y.-Y. Zhang, H. Niu, G. Piccini, D. Mendels, and M. Parrinello, "Improving collective variables: The case of crystallization," J. Chem. Phys. **150**, 094509 (2019).

[12] F. Giberti, M. Salvalaglio, M. Mazzotti, and M. Parrinello, "Insight into the nucleation of urea crystals from the melt," Chem. Eng. Sci. **121**, 51–59 (2015).

[13] T. S. van Erp and P. G. Bolhuis, "Elaborating transition interface sampling methods," J. Comput. Phys. **205**, 157–181 (2005).

[14] A. Ma and A. R. Dinner, "Automatic method for identifying reaction coordinates in complex systems," J. Phys. Chem. B **109**, 6769–6779 (2005).

[15] H. Jung, R. Covino, and G. Hummer, "Artificial intelligence assists discovery of reaction coordinates and mechanisms from molecular dynamics simulations," arXiv:1901.04595 (2019).

[16] H. Jung, R. Covino, A. Arjun, P. G. Bolhuis, and G. Hummer, "Autonomous artificial intelligence discovers mechanisms of molecular self-organization in virtual experiments," arXiv:2105.06673 (2021).

[17] J. Neumann and N. Schwierz, "Artificial intelligence resolves kinetic pathways of magnesium binding to RNA," J. Chem. Theory Comput. **18**, 1202–1212 (2022).

[18] J. H. Appeldorn, S. Lemcke, T. Speck, and A. Nikoubashman, "Employing artificial neural networks to identify reaction coordinates and pathways for self-assembly," J. Phys. Chem. B **126**, 5007–5016 (2022).

[19] T. Kikutsuji, Y. Mori, K.-i. Okazaki, T. Mori, K. Kim, and N. Matubayasi, "Explaining reaction coordinates of alanine dipeptide isomerization obtained from deep neural networks using Explainable Artificial Intelligence (XAI)," J. Chem. Phys. **156**, 154108 (2022).

[20] M. M. Sultan, H. K. Wayment-Steele, and V. S. Pande, "Transferable neural networks for enhanced sampling of protein dynamics," J. Chem. Theory Comput. **14**, 1887–1894 (2018).

[21] M. M. Sultan, G. Kiss, D. Shukla, and V. S. Pande, "Automatic selection of order parameters in the analysis of large scale molecular dynamics simulations," J. Chem. Theory Comput. **10**, 5217–5223 (2014).

[22] M. M. Sultan and V. S. Pande, "tICA-metadynamics: Accelerating metadynamics by using kinetically selected collective variables," J. Chem. Theory Comput. **13**, 2440–2447 (2017).

[23] B. E. Husic and F. Noé, "Deflation reveals dynamical structure in nondominant reaction coordinates," J. Chem. Phys. **151**, 054103 (2019).

[24] F. Noé, G. De Fabritiis, and C. Clementi, "Machine learning for protein folding and dynamics," Curr. Opin. Struct. Biol. **60**, 77–84 (2020).

[25] T. Karmakar, M. Invernizzi, V. Rizzi, and M. Parrinello, "Collective variables for the study of crystallisation," Mol. Phys. **119**, e1893848 (2021).

[26] H. Sidky, W. Chen, and A. L. Ferguson, "Machine learning for collective variable discovery and enhanced sampling in biomolecular simulation," Mol. Phys. **118**, e1737742 (2020).

[27] P. L. Geissler, C. Dellago, and D. Chandler, "Kinetic pathways of ion pair dissociation in water," J. Phys. Chem. B **103**, 3706–3710 (1999).

[28] R. Du, V. S. Pande, A. Y. Grosberg, T. Tanaka, and E. S. Shakhnovich, "On the transition coordinate for protein folding," J. Chem. Phys. **108**, 334–350 (1998).

[29] G. Hummer, "From transition paths to transition states and rate coefficients," J. Chem. Phys. **120**, 516–523 (2004).

[30] R. B. Best and G. Hummer, "Reaction coordinates and rates from transition paths," Proc. Natl. Acad. Sci. U. S. A. **102**, 6732–6737 (2005).

[31] S. V. Krivov and M. Karplus, "One-dimensional free-energy profiles of complex systems: Progress variables that preserve the barriers," J. Phys. Chem. B **110**, 12689–12698 (2006).

[32] P. V. Banushkina and S. V. Krivov, "Nonparametric variational optimization of reaction coordinates," J. Chem. Phys. **143**, 184108 (2015).

[33] S. V. Krivov, "Numerical construction of the p_{fold} (committor) reaction coordinate for a Markov process," J. Phys. Chem. B **115**, 11382–11388 (2011).

[34] B. Peters, "Reaction coordinates and mechanistic hypothesis tests," Annu. Rev. Phys. Chem. **67**, 669–690 (2016).

[35] B. Peters, "Common features of extraordinary rate theories," J. Phys. Chem. B **119**, 6349–6356 (2015).

[36] W. Lechner, C. Dellago, and P. G. Bolhuis, "Role of the prestructured surface cloud in crystal nucleation," Phys. Rev. Lett. **106**, 085701 (2011).

[37] M. Shah, E. E. Santiso, and B. L. Trout, "Computer simulations of homogeneous nucleation of benzene from the melt," J. Phys. Chem. B **115**, 10400–10412 (2011).

[38] G. D. Leines and J. Rogal, "Maximum likelihood analysis of reaction coordinates during solidification in Ni," J. Phys. Chem. B **122**, 10934–10942 (2018).

[39] A. Arjun and P. G. Bolhuis, "Molecular understanding of homogeneous nucleation of CO_2 hydrates using transition path sampling," J. Phys. Chem. B **125**, 338–349 (2021).

[40]G. T. Beckham, B. Peters, C. Starbuck, N. Variankaval, and B. L. Trout, "Surface-mediated nucleation in the solid-state polymorph transformation of terephthalic acid," J. Am. Chem. Soc. **129**, 4714–4723 (2007).
[41]G. T. Beckham and B. Peters, "Optimizing nucleus size metrics for liquid–solid nucleation from transition paths of near-nanosecond duration," J. Phys. Chem. Lett. **2**, 1133–1138 (2011).
[42]R. G. Mullen, J.-E. Shea, and B. Peters, "Easy transition path sampling methods: Flexible-length aimless shooting and permutation shooting," J. Chem. Theory Comput. **11**, 2421–2428 (2015).
[43]C. Leitold, C. J. Mundy, M. D. Baer, G. K. Schenter, and B. Peters, "Solvent reaction coordinate for an S_N2 reaction," J. Chem. Phys. **153**, 024103 (2020).
[44]A. Muždalo, P. Saalfrank, J. Vreede, and M. Santer, "*Cis*-to-*trans* isomerization of azobenzene derivatives studied with transition path sampling and quantum mechanical/molecular mechanical molecular dynamics," J. Chem. Theory Comput. **14**, 2042–2051 (2018).
[45]B. C. Knott, M. Haddad Momeni, M. F. Crowley, L. F. Mackenzie, A. W. Götz, M. Sandgren, S. G. Withers, J. Ståhlberg, and G. T. Beckham, "The mechanism of cellulose hydrolysis by a two-step, retaining cellobiohydrolase elucidated by structural and transition path sampling studies," J. Am. Chem. Soc. **136**, 321–329 (2014).
[46]J. S. Kretchmer and T. F. Miller, "Direct simulation of proton-coupled electron transfer across multiple regimes," J. Chem. Phys. **138**, 134109 (2013).
[47]M. N. Joswiak, M. F. Doherty, and B. Peters, "Ion dissolution mechanism and kinetics at kink sites on NaCl surfaces," Proc. Natl. Acad. Sci. U. S. A. **115**, 656–661 (2018).
[48]P. G. Bolhuis, "Transition-path sampling of β-hairpin folding," Proc. Natl. Acad. Sci. U. S. A. **100**, 12129–12134 (2003).
[49]J. Juraszek, J. Vreede, and P. G. Bolhuis, "Transition path sampling of protein conformational changes," Chem. Phys. **396**, 30–44 (2012).
[50]B. Peters, G. T. Beckham, and B. L. Trout, "Extensions to the likelihood maximization approach for finding reaction coordinates," J. Chem. Phys. **127**, 034109 (2007).
[51]B. Peters, "Inertial likelihood maximization for reaction coordinates with high transmission coefficients," Chem. Phys. Lett. **554**, 248–253 (2012).
[52]E. E. Borrero and F. A. Escobedo, "Reaction coordinates and transition pathways of rare events via forward flux sampling," J. Chem. Phys. **127**, 164101 (2007).
[53]T. Mori and S. Saito, "Dissecting the dynamics during enzyme catalysis: A case study of Pin1 peptidyl-prolyl isomerase," J. Chem. Theory Comput. **16**, 3396–3407 (2020).
[54]Y. Mori, K.-i. Okazaki, T. Mori, K. Kim, and N. Matubayasi, "Learning reaction coordinates via cross-entropy minimization: Application to alanine dipeptide," J. Chem. Phys. **153**, 054115 (2020).
[55]G. M. Torrie and J. P. Valleau, "Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling," J. Comput. Phys. **23**, 187–199 (1977).
[56]A. Laio and M. Parrinello, "Escaping free-energy minima," Proc. Natl. Acad. Sci. U. S. A. **99**, 12562–12566 (2002).
[57]L. Rosso, P. Minárý, Z. Zhu, and M. E. Tuckerman, "On the use of the adiabatic molecular dynamics technique in the calculation of free energy profiles," J. Chem. Phys. **116**, 4389–4402 (2002).
[58]L. Maragliano and E. Vanden-Eijnden, "A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations," Chem. Phys. Lett. **426**, 168–175 (2006).
[59]J. B. Abrams and M. E. Tuckerman, "Efficient and direct generation of multidimensional free energy surfaces via adiabatic dynamics without coordinate transformations," J. Phys. Chem. B **112**, 15742–15757 (2008).
[60]M. Tuckerman, B. J. Berne, and G. J. Martyna, "Reversible multiple time scale molecular dynamics," J. Chem. Phys. **97**, 1990–2001 (1992).
[61]B. M. Dickson, F. Legoll, T. Lelièvre, G. Stoltz, and P. Fleurat-Lessard, "Free energy calculations: An efficient adaptive biasing potential method," J. Phys. Chem. B **114**, 5823–5830 (2010).
[62]M. Chen, M. A. Cuendet, and M. E. Tuckerman, "Heating and flooding: A unified approach for rapid generation of free energy surfaces," J. Chem. Phys. **137**, 024102 (2012).
[63]A. Berezhkovskii and A. Szabo, "One-dimensional reaction coordinates for diffusive activated rate processes in many dimensions," J. Chem. Phys. **122**, 014503 (2005).
[64]E. Darve, D. Rodríguez-Gómez, and A. Pohorille, "Adaptive biasing force method for scalar and vector free energy calculations," J. Chem. Phys. **128**, 144120 (2008).
[65]E. Schneider, L. Dai, R. Q. Topper, C. Drechsel-Grau, and M. E. Tuckerman, "Stochastic neural network approach for learning high-dimensional free energy surfaces," Phys. Rev. Lett. **119**, 150601 (2017).
[66]C. Zhang, Y. Zhang, X. Shi, G. Almpanidis, G. Fan, and X. Shen, "On incremental learning for gradient boosting decision trees," Neural Process. Lett. **50**, 957–987 (2019).
[67]G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, edited by 0, I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (NIPS, 2017), Vol. 30.
[68]G. Csizmadia, K. Liszkai-Peres, B. Ferdinandy, Á. Miklósi, and V. Konok, "Human activity recognition of children with wearable devices using LightGBM machine learning," Sci. Rep. **12**, 5472 (2022).
[69]K. L. Goh, A. Goto, and Y. Lu, "*LGB-stack*: Stacked generalization with *LightGBM* for highly accurate predictions of polymer bandgap," ACS Omega **7**, 29787–29793 (2022).
[70]T. Shimazaki and M. Tachikawa, "Collaborative approach between explainable artificial intelligence and simplified chemical interactions to explore active ligands for cyclin-dependent kinase 2," ACS Omega **7**, 10372–10381 (2022).
[71]L. S. Shapley, "17. A value for n-person games," in *Contributions to the Theory of Games (AM-28)* (Princeton University Press, Princeton, NJ, 2016), Vol. II, pp. 307–318.
[72]S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, edited by 0, I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (NIPS, 2017), Vol. 30.
[73]K. Lindorff-Larsen, S. Piana, K. Palmo, P. Maragakis, J. L. Klepeis, R. O. Dror, and D. E. Shaw, "Improved side-chain torsion potentials for the Amber ff99SB protein force field," Proteins **78**, 1950 (2010).
[74]H. Nguyen, D. R. Roe, and C. Simmerling, "Improved generalized Born solvent model parameters for protein simulations," J. Chem. Theory Comput. **9**, 2020–2034 (2013).
[75]J.-P. Ryckaert, G. Ciccotti, and H. J. C. Berendsen, "Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of *n*-alkanes," J. Comput. Phys. **23**, 327–341 (1977).
[76]B. Leimkuhler and C. Matthews, "Efficient molecular dynamics using geodesic integration and solvent–solute splitting," Proc. R. Soc. A **472**, 20160138 (2016).
[77]C. R. A. Abreu (2023). "Unified Free Energy Dynamics with OpenMM v0.1.0, Zenodo. https://doi.org/10.5281/zenodo.8124438
[78]P. Eastman, J. Swails, J. D. Chodera, R. T. McGibbon, Y. Zhao, K. A. Beauchamp, L.-P. Wang, A. C. Simmonett, M. P. Harrigan, C. D. Stern, R. P. Wiewiora, B. R. Brooks, and V. S. Pande, "OpenMM 7: Rapid development of high performance algorithms for molecular dynamics," PLoS Comput. Biol. **13**, e1005659 (2017).
[79]D. W. H. Swenson, J.-H. Prinz, F. Noe, J. D. Chodera, and P. G. Bolhuis, "OpenPathSampling: A Python framework for path sampling simulations. 1. Basics," J. Chem. Theory Comput. **15**, 813–836 (2019).
[80]D. W. H. Swenson, J.-H. Prinz, F. Noe, J. D. Chodera, and P. G. Bolhuis, "OpenPathSampling: A Python framework for path sampling simulations. 2. Building and customizing path ensembles and sample schemes," J. Chem. Theory Comput. **15**, 837–856 (2019).
[81]P. G. Bolhuis and D. W. H. Swenson, "Transition path sampling as Markov chain Monte Carlo of trajectories: Recent algorithms, software, applications, and future outlook," Adv. Theor. Simul. **4**, 2000237 (2021).
[82]S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," Biometrika **52**, 591–611 (1965).
[83]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, "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res. **12**, 2825–2830 (2011).

[84]L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: Experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (Springer, 2013), pp. 108–122.
[85]B. Bengfort, R. Bilbro, N. Danielsen, L. Gray, K. McIntyre, P. Roman, Z. Poh *et al.* (2018). "Yellowbrick," Zenodo. https://doi.org/10.5281/zenodo.1206264
[86]S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable AI for trees," Nat. Mach. Intell. **2**, 56 (2020).
[87]S. M. Lundberg, B. Nair, M. S. Vavilala, M. Horibe, M. J. Eisses, T. Adams, D. E. Liston, D. K.-W. Low, S.-F. Newman, J. Kim, and S.-I. Lee, "Explainable machine-learning predictions for the prevention of hypoxaemia during surgery," Nat. Biomed. Eng. **2**, 749 (2018).
[88]S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems 30*, edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Curran Associates, Inc., 2017), pp. 4765–4774.
[89]Pandas Development Team (2020). "pandas-dev/pandas: Pandas," Zenodo. https://doi.org/10.5281/zenodo.7093122
[90]M. Ali (2020). "PyCaret: An open source, low-code machine learning library in Python," PyCaret, V.1.0.0.
[91]J. D. Hunter, "Matplotlib: A 2D graphics environment," Comput. Sci. Eng. **9**, 90–95 (2007).
[92]M. Waskom, "seaborn: Statistical data visualization," J. Open Source Software **6**, 3021 (2021).
[93]C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," Nature **585**, 357–362 (2020).
[94]P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental algorithms for scientific computing in Python," Nat. Methods **17**, 261–272 (2020).
[95]W. Humphrey, A. Dalke, and K. Schulten, "VMD: Visual molecular dynamics," J. Mol. Graphics **14**, 33–38 (1996).
[96]P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," Mach. Learn. **63**, 3–42 (2006).
[97]T. K. Ho, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition* (IEEE, 1995), Vol. 1, pp. 278–282.
[98]J. H. Friedman, "Greedy function approximation: A gradient boosting machine," Ann. Stat. **29**, 1189–1232 (2001).
[99]M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," J. Mach. Learn. Res. **1**, 211–244 (2001).
[100]Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers* (IEEE, 1993), Vol. 1, pp. 40–44.
[101]P.-B. Zhang and Z.-X. Yang, "A novel AdaBoost framework with robust threshold and structural optimization," IEEE Trans. Cybern. **48**, 64–76 (2018).
[102]R. Tibshirani, "Regression shrinkage and selection via the lasso," J. R. Stat. Soc. **58**, 267–288 (1996).
[103]W. Ren, E. Vanden-Eijnden, P. Maragakis, and W. E, "Transition pathways in complex systems: Application of the finite-temperature string method to the alanine dipeptide," J. Chem. Phys. **123**, 134109 (2005).
[104]J. Sun and Z. Li, "Peptoid applications in biomedicine and nanotechnology," in *Peptide Applications in Biomedicine, Biotechnology and Bioengineering* (Woodhead Publishing, Buckingham, England, UK, 2018), pp. 183–213.
[105]K. Moehle and H. J. Hofmann, "Peptides and peptoids—A quantum chemical structure comparison," Biopolymers **38**, 781 (1996).
[106]V. A. Voelz, K. A. Dill, and I. Chorny, "Peptoid conformational free energy landscapes from implicit-solvent molecular simulations in AMBER," Biopolymers **96**, 639–650 (2011).
[107]G. L. Butterfoss, P. D. Renfrew, B. Kuhlman, K. Kirshenbaum, and R. Bonneau, "A preliminary survey of the peptoid folding landscape," J. Am. Chem. Soc. **131**, 16798–16807 (2009).
[108]A. Prakash, M. D. Baer, C. J. Mundy, and J. Pfaendtner, "Peptoid backbone flexibilility dictates its interaction with water and surfaces: A molecular dynamics investigation," Biomacromolecules **19**, 1006–1015 (2018).
[109]J. L. Kessler, G. Kang, Z. Qin, H. Kang, F. G. Whitby, T. E. Cheatham, C. P. Hill, Y. Li, and S. M. Yu, "Peptoid residues make diverse, hyperstable collagen triple-helices," J. Am. Chem. Soc. **143**, 10910–10919 (2021).
[110]J. Sun and R. N. Zuckermann, "Peptoid polymers: A highly designable bioinspired material," ACS Nano **7**, 4715–4732 (2013).
[111]D. Mendels, G. Piccini, and M. Parrinello, "Collective variables from local fluctuations," J. Phys. Chem. Lett. **9**, 2776–2781 (2018).
[112]B. Peters, "Using the histogram test to quantify reaction coordinate error," J. Chem. Phys. **125**, 241101 (2006).