The Journal of Chemical Physics

RESEARCH ARTICLE | APRIL 13 2023

KIF—Key Interactions Finder: A program to identify the key molecular interactions that regulate protein conformational changes

Special Collection: New Views of Allostery

Rory M. Crean ; Joanna S. G. Slusky ; Peter M. Kasson ; Shina Caroline Lynn Kamerlin

J. Chem. Phys. 158, 144114 (2023)
https://doi.org/10.1063/5.0140882

The Journal of Chemical Physics ARTICLE scitation.org/journal/jcp

KIF—Key Interactions Finder: A program to identify the key molecular interactions that regulate protein conformational changes

Cite as: J. Chem. Phys. 158, 144114 (2023); doi: 10.1063/5.0140882
Submitted: 31 December 2022 • Accepted: 28 March 2023 •
Published Online: 13 April 2023

Rory M. Crean,[1] Joanna S. G. Slusky,[2,3] Peter M. Kasson,[4,5] and Shina Caroline Lynn Kamerlin[1,6,a)]

AFFILIATIONS

[1] Department of Chemistry – BMC, Uppsala University, BMC Box 576, S-751 23 Uppsala, Sweden
[2] Center for Computational Biology, University of Kansas, Lawrence, Kansas 66047, USA
[3] Department of Molecular Biosciences, University of Kansas, Lawrence, Kansas 66045, USA
[4] Departments of Molecular Physiology and Biomedical Engineering, University of Virginia, Charlottesville, Virginia 22908, USA
[5] Department of Cell and Molecular Biology, Uppsala University, BMC Box 596, Uppsala 751 24, Sweden
[6] School of Chemistry and Biochemistry, Georgia Institute of Technology, 901 Atlantic Drive NW, Atlanta, Georgia 30332-0400, USA

Note: This paper is part of the JCP Special Topic on New Views of Allostery.
[a)] **Author to whom correspondence should be addressed:** skamerlin3@gatech.edu

ABSTRACT

Simulation datasets of proteins (e.g., those generated by molecular dynamics simulations) are filled with information about how a non-covalent interaction network within a protein regulates the conformation and, thus, function of the said protein. Most proteins contain thousands of non-covalent interactions, with most of these being largely irrelevant to any single conformational change. The ability to automatically process any protein simulation dataset to identify non-covalent interactions that are strongly associated with a single, defined conformational change would be a highly valuable tool for the community. Furthermore, the insights generated from this tool could be applied to basic research, in order to improve understanding of a mechanism of action, or for protein engineering, to identify candidate mutations to improve/alter the functionality of any given protein. The open-source Python package Key Interactions Finder (KIF) enables users to identify those non-covalent interactions that are strongly associated with any conformational change of interest for any protein simulated. KIF gives the user full control to define the conformational change of interest as either a continuous variable or categorical variable, and methods from statistics or machine learning can be applied to identify and rank the interactions and residues distributed throughout the protein, which are relevant to the conformational change. Finally, KIF has been applied to three diverse model systems (protein tyrosine phosphatase 1B, the PDZ3 domain, and the KE07 series of Kemp eliminases) in order to illustrate its power to identify key features that regulate functionally important conformational dynamics.

INTRODUCTION

Non-covalent interactions modulate the structure, dynamics, and ultimately function of biomolecules. Molecular dynamics (MD) simulations of biomolecules contain an abundance of information about these interactions, including how specific interactions or interaction networks regulate the structure and, therefore, the function of a given biomolecule. A deeper insight into specific interactions and interaction networks responsible for any given conformational change would be highly valuable to the scientific community. This is true from both a basic research perspective to better understand a biological mechanism of action and in the context of rational or semi-rational structure-based design where specific interactions could be targeted to drive a desired conformational change.

Several tools such as PyContact,[1] RIP-MD,[2] and gRINN[3] can be readily applied to MD simulations of biomolecules to identify and score the strength of non-covalent interactions between residues on

a per frame basis. In cases where a specific residue or a small subset of residues are known to be of interest, directly studying those residues interactions is likely to be sufficient. However, in the case of identifying unknown structurally important residues/interactions, including those that are distal/allosteric, and/or ranking their relative importance/contribution, some form of computational processing is likely required in order to sift through the potentially thousands of non-covalent interactions within any given protein.

Methods using both machine learning (ML) and statistics have been successfully applied to large datasets obtained from MD simulations of biomolecular systems,[4–21] with these approaches typically involving the generation of large feature sets through determining all the C_α–C_α (or C_β–C_β) distances or backbone dihedrals on a per frame basis. One such application of these feature sets is to use them to describe slow timescale processes occurring in biological systems, enabling the development of collective variables (i.e., reaction coordinates) that can describe this process in a coarse-grained manner. With these collective variables, complex processes can be well described with one or two dimensions and/or applied to methods such as metadynamics in order to enhance the sampling of molecular dynamics simulations.[19,22–31] Furthermore, prior work has demonstrated how these large datasets can be exploited to identify (conformationally) important residues or groups of residues distributed throughout an entire biomolecular system.[17,18,32–37]

Here, we describe a Python package called "KIF" (short for Key Interactions Finder), which we have developed to provide a largely automated workflow that allows researchers to identify key non-covalent interactions and residues that describe a change in any user defined conformational state. As we will demonstrate in this article, these changes can be anything from relatively large-scale loop conformational changes all the way through to much more subtle changes in a single side chain conformation. To do this, we built upon the popular MD simulation analysis tool PyContact,[1] which is an MD simulation engine-agnostic method to identify and score all non-covalent interactions present in MD simulation. By combining the per frame interaction data with a user defined target variable, one can apply KIF to identify the key non-covalent interactions distributed throughout the protein that are the most sensitive to the target variable's value. Furthermore, KIF allows for the conformational change of interest to be described as a classification problem (i.e., state A vs state B) or as a regression problem (i.e., a continuous target variable that describes the conformational change), and the end user has complete flexibility in terms of deciding how to generate this target variable to describe their conformational change of interest. The software outputs a relative score for each interaction based on its relationship to the target variable, and alongside this, per residue scores can also be determined (by summing and normalizing the per interactions scores obtained for each residue). The per feature and per residue scores generated by our program can be outputted in the tabular form or as PyMOL[38] compatible scripts in order to visualize the results on a 3D structure.

In addition, our program also enables researchers to easily use the non-covalent interaction data as inputs to many available graph theory-based approaches to describe protein interaction networks.[39,40] As some examples, graph theory methods, such as Weighted Implementation of Suboptimal Paths (WISPs),[39] community network analysis (CNA),[40] shortest path map (SPM),[41] and Ohm,[42] have been extensively used to study protein interaction networks and allostery, among other approaches.[41,43–53] These methods work by describing proteins/biomolecular systems as a network, with residues being the nodes in the network and the connections between the nodes being generated by physical descriptors, with the descriptor most commonly used being the correlation between C_α or C_β distances of each residue. Instead, we use the non-covalent interaction data to generate a per residue linear correlation matrix, which is linked directly to linear correlations between interacting residue pairs. This enables us to easily identify the specific non-covalent interactions that are likely responsible for allosteric signaling within a given protein network.

KIF is a computationally efficient approach, which can be executed on a standard laptop, with execution times for standard post-processing of MD simulation datasets typically requiring only several minutes or less. Here, we will describe the computational framework underlying KIF and show its successful application to three model systems. The chosen model systems include a large, allosterically regulated, conformational change in a protein loop; a study on the impact of peptide binding on a neighboring protein domain; and characterizing how mutations alter side chain conformations inside an enzyme active site for two engineered enzymes. Based on these illustrative systems, it can be seen that our package KIF provides an efficient and largely automated approach to identify the key interactions and residues that modulate both subtle and large-scale conformational changes distributed throughout the protein and even rank their relative contributions. Furthermore, our approach provides users with the flexibility to describe any conformational change; however, they see fit. As such, KIF provides a widely applicable and freely available tool for the community for the analysis of conformational switching using biomolecular simulations.

METHODOLOGY

Preparation and simulation of the model systems

The setup of all the model systems used in this paper is described below with further details provided in the supplementary material. Unless otherwise stated, the following applies to all systems simulated: The Amber ff14SB force-field[54] and TIP3P[55] water model were used to describe protein atoms and water molecules, respectively, and relevant starting structures were obtained from the Protein Data Bank.[56] A time step of 2 fs was used, and production MD simulations were performed in an octahedral water box under constant temperature and pressure (298 K and 1 atm).

MD simulations of the PDZ3 domain with the CRIPT peptide bound were performed using PDB ID: 1BE9,[57] with simulations of the unbound form performed by removing the CRIPT peptide from the structure. For peptide bound simulations, the N- and C-termini of the peptide side chain were capped with an acetyl group and a N-methyl amide group, respectively, and the partially missing side chains for PDZ3 residue F301 and peptide residue K5 (using the numbering from PDB ID: 1BE9[57]) were completed using PyMOL.[38] The protonation states of all residues were kept at their standard states, and histidine tautomerization states (Table S1) and any necessary Asn or Gln "flips" were determined using MolProbity.[58] MD simulations were performed using AMBER18[59] and run for 200 ns

with the last 100 ns of each simulation used in PyContact calculations (to enable unliganded simulations to first relax after removing the peptide). PyContact was performed on 20 000 equally spaced frames obtained from both bound and unbound MD simulations (0.1 ns time step), with all PDZ3 domain residues used in the calculation (i.e., the peptide was not included in the bound simulation PyContact calculation).

We have previously[60–62] performed molecular dynamics (MD) simulations of protein tyrosine phosphatase (PTP1B) in the phospho-enzyme intermediate state, which is the reactant state for the second, rate limiting cleavage step of the catalytic mechanism (Fig. S1). In this study, we used our previously prepared structures (based on PDB ID: 6B90[63]) to simulate both closed and open conformations of the WPD-loop with the AMBER18[59] simulation package. We generated 10 × 100 ns long MD simulation replicas of PTP1B starting from both closed and open WPD-loop conformational states (2 μs in total). From these simulations, a total of 10 000 equally spaced frames (0.2 ns time step) were extracted from simulations and used in PyContact[1] calculations (to generate the features required for analysis with our program KIF).

Hamiltonian replica exchange MD (HREX-MD) simulations[64] on two designed Kemp eliminase variants from the KE07 series[65–67] (R1 and R4) were performed using GROMACS 2018.4[68] patched with PLUMED v2.5.[69] We used the same protocol as described in the prior work,[70] with simulations performed using six replicas with λ values scaled between 1.0 and 0.667 were used for each system (giving an effective temperature range of 300–450 K). Both KE07 variants were simulated for 200 ns each using their available crystal structures, as summarized in Table S1. PyContact calculations were performed on 10 000 equally spaced frames (0.02 ns time step) obtained from the neutral replica of HREX-MD simulations.

Preparation of the features and target variables

PyContact[1] analysis was performed on all residues present in each system in order to generate the features (i.e., the non-covalent interactions) needed for our approach. PyContact[1] analyzes a given trajectory file to identify and calculate the strength of all non-covalent interactions present in the trajectory on a per frame basis. Here, we used the default cutoffs and settings provided by PyContact[1] to label each interaction as one of the following: a hydrogen bond, a salt bridge, a hydrophobic interaction, and a van der Waals interaction.

Depending on the analysis performed, not all features were retained for analysis; for instance, features were discarded if they had a low occupancy and/or low average interaction strength (Tables S2–S4 provides complete details on all the filtering performed for each system). Furthermore, in the case of PTP1B, we tested removing all features that included a residue on the WPD-loop (as opposed to including all residues); see Table S2.

C_α-atom root mean square deviations (RMSDs) and the first two side chain dihedral angles of W50 (χ_1 and χ_2) that were required to determine the per frame target variable value for simulations of PTP1B and the two KE07 variants were determined using CPPTRAJ.[71] A shell script was written to determine the per frame conformational state, given the RMSD value (see the supplementary material). In the case of simulations of the PDZ3 domain, the frames were classified simply by if they were from peptide bound or unliganded simulations.

Per feature and per residue scoring

For the construction of the classification and regression machine learning (ML) models described in this paper, three different ensemble methods were used: (1) Random Forest (RF),[72] (2) categorical boosting (CatBoost),[73] and (3) eXtreme Gradient Boosting (XGBoost).[74] These methods were chosen because of their insensitivity toward feature sets with high levels of multi-collinearity (see the supplementary material). In all cases, a random selection of 85% of each dataset was used for training and testing using repeated stratified k-fold cross validation (three repeats and five folds/splits) as implemented in scikit-learn. The remaining 15% of each dataset was used to evaluate the quality of each model (the holdout/validation set) with either the F_1-score (if classification) or the mean absolute error and root mean squared error (if regression). The performance metrics obtained during training (from the cross validation runs) were compared to the results obtained by the validation dataset to validate that overfitting was not occurring. All features were normalized with scikit-learn's[75] min–max scaler. Feature importance was taken directly from the models generated.

For binary classification models made with our statistical analysis module, the two metrics used to score each feature were the mutual information[76,77] (MI) and the Jensen–Shannon distance (JS-distance).[78] The MI was calculated using the function available from scikit-learn.[75] In preparation for JS-Distance calculations, features (already separated into their classes) were described using a Gaussian kernel density estimation (KDE) using a bandwidth of 0.02. The implementation of the JS-distance calculation available from SciPy[79] was then used. For the case of a continuous target variable (i.e., regression), the two metrics used were the MI or linear correlation (LC) using the implementations available from scikit-learn.[75]

Per residue scores were obtained by summing together all the per feature scores in which a given residue was found and then scaling the results so that the top residue had a score of 1. If the per feature/interaction scores were non-linearly scaled (such as those obtained from the MI calculations), they were first rescaled to be linearly scaled before calculating the per residue scores.

Correlation/network analysis

The linear correlation between each feature to every other feature was first determined in order to generate a correlation matrix. This correlation matrix was then converted into a per residue matrix (i.e., the dimensions changed from a size of the number of features squared to the number of residues squared) by identifying the largest correlation (in absolute values) between a pair of residues found in any interaction. The correlation of a residue to itself (i.e., the values along the diagonal of the matrix) was set to 1.

To generate contact maps/matrices, MDTraj[80] was used to calculate all the heavy atom distances between each pair of residues for a given crystal structure. The minimum heavy atom distance between each residue pair was determined from this, and a cutoff of 6 Å was used to define if the two residues were in contact. A binary matrix of size the number of residues squared was then constructed

with "1" used to signify if the residue pair was within the cutoff and "0" if not in contact. In the case of PTP1B, both open and closed state conformations (provided in PDB ID: 6B90[63]) were used to determine if a residue was in contact by taking the smallest heavy atom distance found between a residue pair in either structure. The correlation and contact matrices were used as inputs to Bio3D[81] in order to perform Weighted Implementation of Suboptimal Path (WISP)[39] analysis. WISP is a technique that enables one to describe how distal regions of a protein allosterically communicate with one another by describing a protein as a graph/network. In WISP, one defines a "source" residue and a "sink" residue, which are distal from one another, and then determines the shortest pathways of communication between these two residues. This allows for the identification of the key residues and residue–residue interactions between the two selected residues that enable allosteric communication between them. A correlation cutoff of |0.25| was used to determine the 500 shortest paths between each source and sink. From these 500 paths, the node and edge degeneracies were determined alongside the path lengths.

Bootstrapping and random subsampling

In order to estimate the reproducibility of the calculations we performed using our package (KIF) on our simulations of PTP1B and the PDZ3 domain, we turned to bootstrapping with replacement alongside random subsampling. Bootstrapping with replacement is a resampling technique that involves generating many resamples of a dataset through randomly selecting combinations of observations (with the possibility to duplicate individual observations). With bootstrapping, summary statistics can be calculated, and in our case, we used bootstrapping to generate probability densities describing the convergence/reproducibility of our calculations. Random subsampling is a variant of bootstrapping in which fewer observations than were used are combined for each bootstrap calculation. For example, if there were 20 replicas performed in total, bootstrapping would mean each resample would be of size 20 (replicas), while random subsampling would be of size N, where N can be any integer value between 1 and 19. Performing these methods allowed us to investigate how much sampling would be needed for reproducible results. These calculations on PTP1B and the PDZ3 domain were performed in the same manner by using the per residue scores obtained from the Jensen–Shannon distance metric. First, the complete datasets were separated into their individual 20 replicas, with PTP1B having ten replicas from both the "closed" and "open" WPD-loop conformations and the PDZ3 domain having ten replicas from both the "bound" and "unliganded" states. For the different scenarios to bootstrap or sub-sample from, a random selection of X replicas (where X was 1, 2, 3, 5, 8, or 10) from both states were extracted (giving 2, 4, 6, 10, 16, or 20 replicas in total, respectively) and the per residue scores were determined for each sample.

For our second set of bootstrapping/sub-sampling calculations on the PTP1B and PDZ3 systems, we focused on the scenario of only using ten randomly selected replicas (five replicas from each conformation/state simulated) and, instead, using a different spacing between the frames used. For the PDZ3 domain, we evaluated nine different time frame spacing scenarios between 0.1 and 10 ns, while for PTP1B, we evaluated eight different time frame spacing scenarios between 0.2 and 10 ns (frames for PTP1B were initially collected with a time spacing of 0.2 ns).

Bootstrapping with replacement and random subsampling were performed 5000 times for every scenario described above, and the per residue scores obtained from each sample were compared to the top 100 residue scores obtained using all 20 unique replicas with the two correlation metrics: the Spearman's rank and the Pearson correlation coefficient. The 5000 correlation scores obtained for each scenario were used to construct Gaussian kernel density estimates using a bandwidth of 0.02.

Software and data availability

The software developed for this work is freely available for download from GitHub at the following link: https://github.com/kamerlinlab/KIF, and is provided under the GNU General Public License v2.0. Included in the package are several Jupyter Notebook tutorials,[82] which show how to use the software toward various research questions. The parameter files, MD simulation input files, input datasets, and additional analysis scripts are available from Zenodo at the following link: http://DOI:10.5281/zenodo.7104965.

RESULTS AND DISCUSSION

Package overview

The package was written in Python and utilizes several well-known packages for data science, such as pandas,[83,84] NumPy,[85] scikit-learn,[75] and SciPy.[79] The MDTraj[80] Python library (for the analysis of MD simulations) was used to provide some additional functionality where necessary. The general workflow for using this package is depicted in Fig. 1 and can be described as follows. (1) Run MD simulations (or any other conformational sampling approach) on your system(s) of interest. (2) Post-process the resulting trajectories with PyContact[1] (which is MD simulation engine agnostic). While processing the trajectories with PyContact,[1] one can optionally identify a target variable to label each simulation frame for the available supervised methods. Among others, example target variables could be a reacting atom distance for an enzyme or a set of RMSD cutoffs to define a protein's conformational state. (3) Load the features calculated by PyContact[1] (all the non-covalent interactions) into the program and perform some basic pre-processing (standardize formatting, merge multiple files, etc.). (4) Prepare the dataset for further analysis by filtering which features to include and optionally merging a target label if one was generated during step 2.

Following from this, a user can choose between performing one or more of the following analysis techniques on the system of interest: machine learning, statistical modeling, or network analysis (step 5, pathway 1). For machine learning and statistical modeling pathways, one can apply KIF to identify the key non-covalent interactions distributed throughout the protein that are the most sensitive to the target variable's value (step 6, pathway 1). Furthermore, the per feature (i.e., per interaction) scores generated can be readily outputted and/or further post-processed to generate per residues scores by summing and normalizing the per feature scores that each residue is present in (step 7, pathway 1). Alongside outputting the per feature

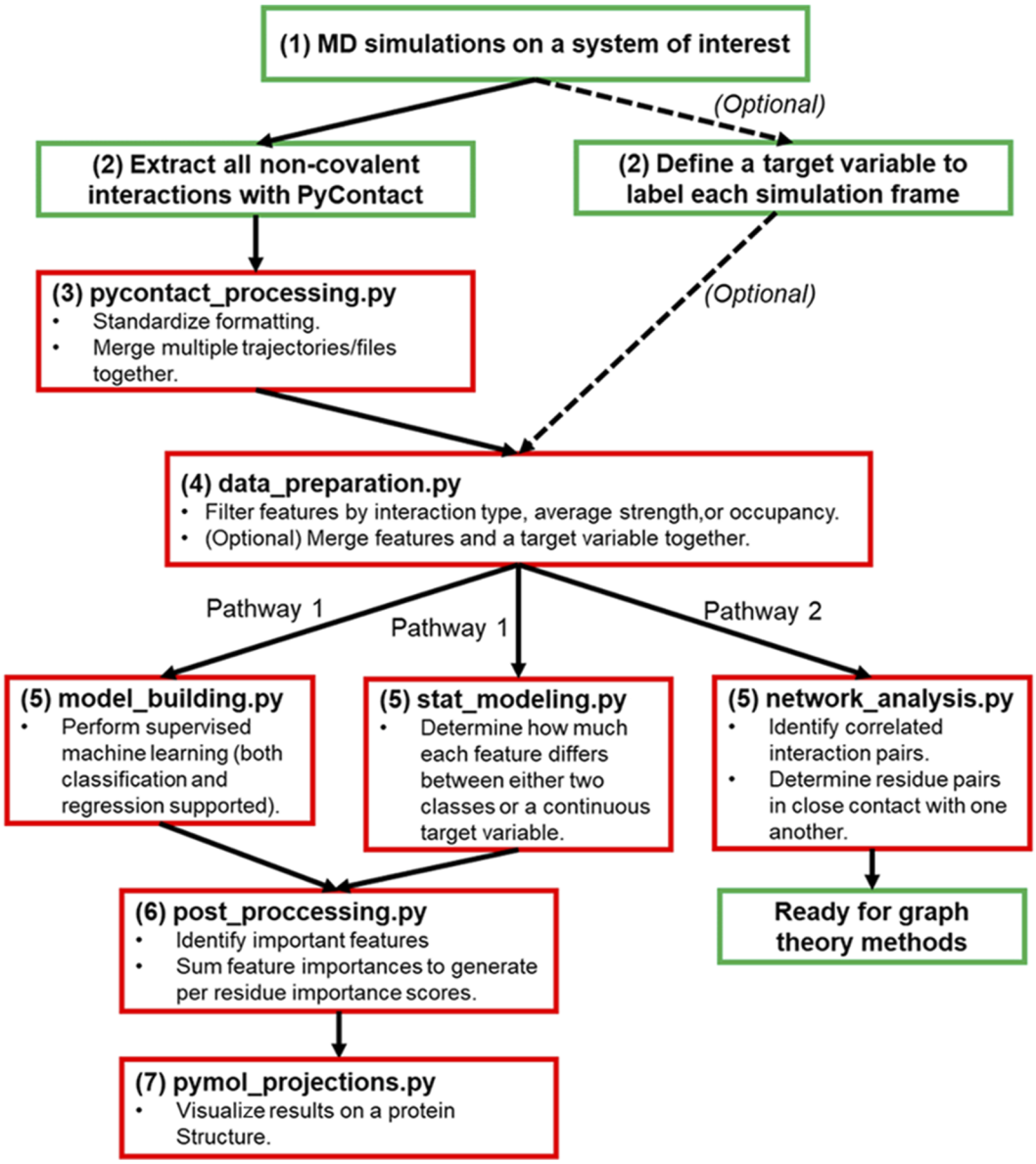

FIG. 1. Generic workflows available for applying the Key Interactions Finder (KIF) package on a given biomolecular system of interest. Each step is labeled to match with the description provided in the text. The boxes in green represent steps that are performed outside of the program (i.e., using other tools). The red boxes represent the modules in the program and the main responsibilities of each module. In practice, a user would create a main.py file or a Jupyter notebook[82] and import each module required for their workflow, making use of the available classes and functions provided by each module. Several tutorials that provide example workflows and possible ways to analyze the data generated are included in KIF's GitHub repository at: https://github.com/kamerlinlab/KIF.

and per residue scores in a tabular format, these results can be visualized on a protein structure by creating PyMOL[38] compatible scripts (step 5, pathway 2). For the network analysis pathway, a per residue correlation matrix and per residue distance matrix are generated, which can be readily applied to the many different graph theory-based methods available in many pre-existing programs (such as WISP,[39] BIO3D,[81] and NetworkX[86]).

In addition to the main parts of the workflow described above, several additional useful tools exist. These include being able to calculate and plot each residue's score against its distance from a user defined site of interest or in the case of binary classification, whether each interaction appears to strengthen or weaken a given conformational state. Here, we will demonstrate the application of this package to several well-characterized biomolecular systems in order

to evaluate the performance of the approaches available within KIF with respect to available experimental data and demonstrate some of the insights that are possible to obtain from this software package.

Selection of model systems

We applied our package to three model systems (Fig. 2) to show and test the capabilities of our package toward several different objectives (assessing the impact of ligand binding, describing protein conformational changes, characterizing allosteric signaling pathways, and understanding the origin of enhanced catalytic rates). The model systems were also chosen, in part, due to the available experimental data with which we could compare our results to.

The chosen model systems will be described in more detail in the section titled "Results and Discussion," but Fig. 2 demonstrates that we have included both subtle and large-scale conformational changes. For example, Fig. 2(b) shows a relatively large (~10 Å) conformational change between two protein loop conformations, in which (due to the size of the change) novel interactions would be expected to be found in one conformation and not the other. On the other hand, Fig. 2(c) shows a model system where the difference is subtle changes in side chain conformations of an active site, meaning that the differences between states are likely to be substantially more subtle and potentially harder to capture. Furthermore, all three systems selected have some degree of experimentally characterized allostery that we can compare our results to. This will enable us to evaluate our package's ability to find both local and distal interactions that are associated with a conformational change of interest.

Finally, in order to validate that the methods contained within KIF were correctly implemented and work as expected, we generated a synthetic dataset whereby the important interactions/features were known *a priori*. This was used to test the ability of all the methods within KIF to identify these important interactions/features. In the interest of space, the generation of this synthetic dataset and the subsequent validation of KIF are detailed in the supplementary material, Sec. S3.

Impact of peptide binding on the PDZ3 domain

Our first model system focuses on the protein-peptide binding interaction between the PSD95-PDZ3 domain with the CRIPT peptide [Fig. 2(a)]. The PSD95-PDZ3 domain (from now on referred to as just the PDZ3 domain) is an ~100 residue long protein that contains a peptide binding site for the five residue CRIPT peptide, with the peptide binding known to allosterically modulate the PDZ3 domain.[87–89] One such study on the impact of peptide binding used exhaustive single and double point mutant screening to calculate the average mutational sensitivity of each PDZ3 residue to ΔG_{bind} as a way to probe what residues throughout the PDZ3 domain are perturbed upon peptide binding [Fig. 3(a)].[87] This study enabled the authors to classify every PDZ3 domain residue as "orthosteric," "allosteric," or "remainder" with respect to peptide binding based on each residue's mutational sensitivity and the proximity to the peptide.[87]

We applied KIF to see if we could identify those residues on the PDZ3 domain that are sensitive to peptide binding by comparing the protein interaction networks generated from MD simulations of the PDZ3 domain with (1) the CRIPT peptide bound and (2) without the CRIPT peptide bound. To this end, we performed 10 × 100 ns long MD simulations of the PDZ3 domain both in the unliganded state and with the peptide bound in order to generate 20 000 frames for feature generation (the non-covalent interactions) with PyContact[1] (see the section titled "Methodology"). We then classified each frame as either "peptide bound" or "unliganded," simply according to whether the given frame was from peptide bound or unliganded MD simulations. We applied both our ML and statistical modeling modules to evaluate the differences between the two classes (binary classification).

We generated three classification models using three different ensemble ML algorithms (Random Forest,[72] CatBoost,[73] and XGBoost[74]), with ensemble methods chosen over others due to their speed and ability to deal with multicollinearity (see the supplementary material). We validated our models on a hold-out/validation dataset representing 15% of the total data and used

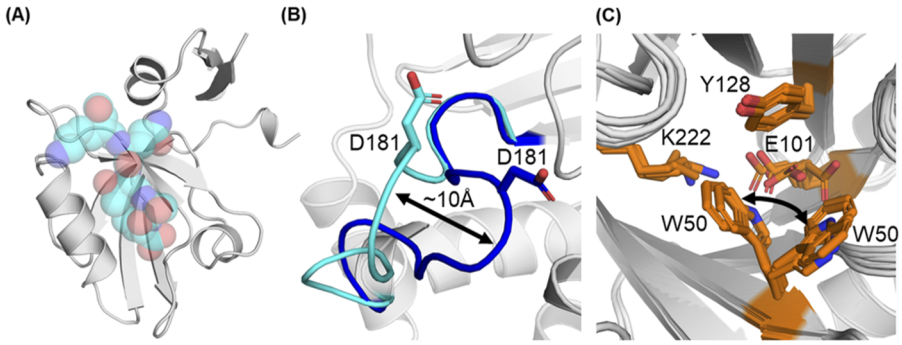

FIG. 2. An overview of the model systems selected for this work with a description of the nature of the conformational change(s) that will be studied for each model system. (a) Structure of the PDZ3 domain with the CRIPT peptide bound (shown as semi-transparent spheres). We will study the impact peptide binding has on the PDZ3 domain by simulating the peptide bound and unliganded states of the PDZ3 domain (binary classification problem). (b) Crystal structures of the WPD-loop open and closed states of the enzyme PTP1B, with the "closed" (and catalytically active) WPD-loop conformation colored dark blue and the "open" conformation colored light blue. We will use binary classification to identify how the protein interaction network differs in the "open" and "closed" conformational states of the WPD-loop. (c) The active site of several Kemp eliminase variants from the KE07 series[65–67] over the course of directed evolution (DE). Mutations both local and distal across this series of variants altered the active site side chain conformations of several residues, which lead to enhanced catalytic activity.[70] We will study how the protein interaction network was altered as a result of these mutations, which led to a population shift and ultimately enhanced activity, using regression analysis.

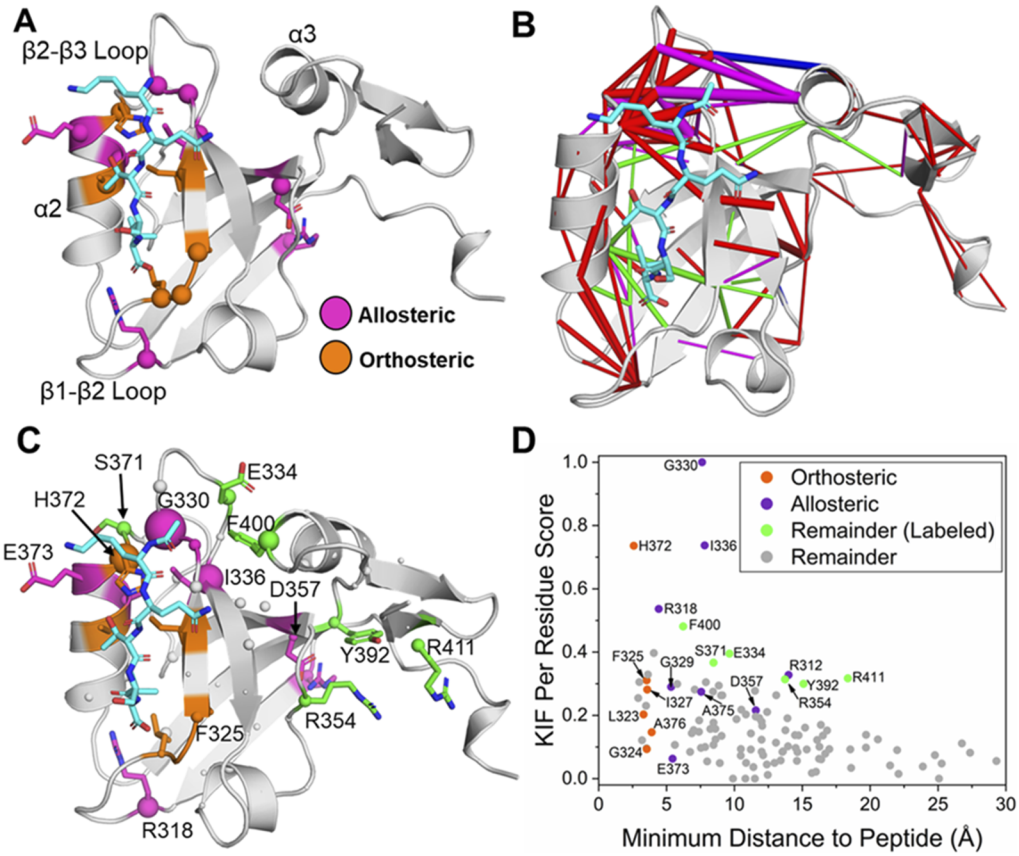

FIG. 3. (a) Structure of the PDZ3 domain with the CRIPT peptide bound (cyan sticks). Residues identified by Faure *et al.*[87] as either allosteric or orthosteric are colored accordingly, with their C_α atoms shown as a spheres and their side chains shown as sticks. Calculated per feature (interaction) (b) and per residue (c) scores for distinguishing between the peptide bound and unliganded states of the PDZ3 domain, obtained from our statistical analysis module using the JS-distance metric.[87] For (b), the thickness of the cylinder indicates the relative score of the feature/interaction. Cylinders are colored according to their interaction type, with red indicating a hydrogen bond, blue indicating a salt bridge, green indicating a hydrophobic interaction, and pink indicating a van der Waals interaction. For (c), the size of the sphere represents the per residue score, and each residue is colored using the same scheme as in (a), with the exception that some "remainder" residues' side chains are shown and colored green if their per residue score is large, considering their distance from the peptide. (d) Minimum heavy atom distance of each residue to any peptide residue against the per residue scores, as in (c).

the F_1 score (a metric that considers both the precision and recall of a classifier) to measure each model's quality. The F_1 scores on the validation dataset for the three ML models generated were all very high, with values between 0.98 and 0.99 (see Table S5).

We compared both the rank ordering and linear correlation of the five models we generated for the PDZ3 domain (three from our ML module and two from our statistical analysis module) by constructing a scatter matrix of the per residue scores (Fig. S2). While the scatter matrix demonstrated all methods to be positively correlated with one another (Fig. S2), there were clear differences in the results obtained between the methods, with the CatBoost[73] ML algorithm seeming to give rise to the most different per residue scores between the methods tested (at least according to both metrics used; see Fig. S2). In terms of the top ranking residues, all five approaches consistently identified the residues G330, I336, R318 (classified as allosteric by Faure *et al.*[87]), and H372 (classified as orthosteric by Faure *et al.*[87]) as highly important (Fig. S3). Furthermore, by comparing the closest heavy atom distance of each residue to any peptide residue, we observed that all five approaches were able to identify residues that were distal to the peptide binding site, but considered important (Fig. S3).

While the three ML models generated were highly accurate, their per feature scores (i.e., their feature importance) are limited by the fact that only some of the features that are notably associated with the target variable are required to accurately predict the target variables value. While this would be perfectly fine (if not overkill) if one were applying these ML models toward prediction, this is undesirable if the goal is to identify many important features that describe the conformational differences. This limitation for the ML module is shown in Sec. S2 of the supplementary material for our analyses on both the PDZ3 domain and PTP1B model systems.

For the purposes of further analysis, we will now focus on the results generated through our statistical analysis package (instead of the ML package for the aforementioned reasons) using the JS-distance metric,[78] as the results obtained were highly similar to those obtained from the MI calculations (Fig. S2, the other method available from our statistical analysis package), which would be expected, given their similarity.

The top four residues obtained with the JS-distance metric[78] are identified as either allosteric or orthosteric [Fig. 3(a)], and the two most distal allosteric residues (R312 and D357) are both among the top ranked residues when compared to other residues with similar minimum heavy atom distances to the peptide (Fig. S4). The residue E373 is the lowest ranked residue when compared to other residues with a similar minimum distance (Fig. S4) and was consistently identified as unimportant by all five approaches used (Fig. S3). This residue's experimentally observed mutational sensitivity toward peptide binding is perhaps unsurprising, given the location of the N-terminal lysine on the peptide [Fig. 3(b)]. It may therefore be more accurate to characterize this residue as "orthosteric" as opposed to "allosteric" (the original authors characterized this residue as allosteric due to its distance from the peptide[87]). Alongside E373, several orthosteric residues (such as L323, G324, and A376) are identified to have a relatively low per residue score, given their close proximity to the binding site [Fig. 3(a)]. It is important to consider that the features/interactions we used to build our models contain no protein-peptide interactions. Our results would therefore suggest that while these residues, of course, play a key role in regulating the binding of the peptide (through direct interactions with the peptide), they have a limited role in any subsequent allosteric signaling to the remainder of the PDZ3 domain. In contrast, H372 is an orthosteric residue that is consistently identified as one of the most important residues present in the PDZ3 domain across all five models (Fig. S3). H372 is identified as such due to its interactions with five residues on the neighboring β2-β3 loop (see Table S6), suggesting a more substantial role in allosteric signaling for this residue compared to the other binding site residues, as discussed in detail in, e.g., Ref. 89.

Beyond those residues labeled as important from the work of Faure *et al.*,[87] our approach also identified several other residues with potentially interesting roles in allosteric communication (Fig. 3). For example, F400 sits on the α3-helix and is identified as the fifth most important residue due to its differing van der Waals interactions with several residues on the β2-β3 loop, including G330 and G329 (Table S6), both of which were determined as allosteric by Faure *et al.*[87] Interestingly, the α3-helix (which F400 is located on) has previously been implicated in having a role in allosteric signaling upon peptide binding,[90,91] supporting our observations herein.

In Fig. 3, we were able to validate our model's ability to identify allosterically sensitive residues on the PDZ3 domain. We will now focus on the interactions between the β2-β3 loop and the α3-helix to demonstrate how our approach can be used to gain deeper insights at a more local level. By calculating the difference in the C_α-atom root mean squared fluctuation (ΔRMSF) across the entire PDZ3 domain for unliganded and peptide bound simulations (Fig. S5), we were able to identify that the N-terminal portion of the α2-helix, the β2-β3 loop, and the α3-helix were all destabilized in the absence of the peptide [Fig. 4(a)], although the destabilization in the helices is comparatively minor.

By analyzing our per feature (per interaction) scores, we were then able to identify several interactions, which likely resulted in this destabilization, which are depicted in Fig. 4(a). This interaction network structurally begins from the α2-helix and passes through the β2-β3 loop to end at the α3-helix [Fig. 4(a)]. By measuring the correlation between interacting pairs in these regions [Fig. 4(b)], we are also able to confirm a high degree of correlation between these regions. This was especially true in the case of interacting pairs, such as H372, G329, G330, and F400 [Fig. 4(b)]. Beyond this, we can refer back to the PyContact[1] determined interaction strength for any given interaction and identify how this interaction differed for unliganded and peptide bound simulations over time. In this case, we focused on the salt bridge between E334 on the β2-β3 loop and R399 on the α3-helix [Fig. 4(c)]. By extracting the kernel density estimations generated for each feature in unliganded and peptide bound states (a part of the process to calculate the JS-distance for each feature; see the section titled "Methodology"), we identified this salt bridge to be clearly weakened in the unliganded state [Fig. 4(d)]. This observation also helps to rationalize the observed destabilization of this region in the absence of the peptide [Fig. 4(a)].

Differences in the interaction networks for the closed and open states of PTP1B

Our second model system is the enzyme Protein Tyrosine Phosphatase 1B (PTP1B), which contains a catalytic loop that undergoes a large scale (~10 Å) conformational change in order to regulate its catalytic activity [Fig. 5(a)].[60] The WPD-loop can adopt two major conformations: "closed" (catalytically active) and "open" (catalytically inactive) conformations [Fig. 5(a)]. How the conformational state of the WPD-loop is allosterically regulated has been the focus of many prior studies, including those using NMR,[92–94] x-ray crystallography, mutagenesis, and molecular simulations (see, e.g., Refs. 60–63 and 92–108, among many others). For PTP1B, we applied KIF to predict the key residues and interactions that are associated with the WPD-loops' conformational state and compared our predictions to the aforementioned available experimental data. Furthermore, we will utilize our package to characterize the allosteric signaling pathways from PTP1B's two known allosteric drug binding sites[94,98] to the active site [Fig. 5(a)]. In order to do this, we performed a total of 20 × 100 ns MD simulations of PTP1B, starting ten of these from the closed and ten from open WPD-loop conformations of this enzyme. We then extracted 10 000 equally spaced frames from these simulations and used PyContact[1] to identify all the non-covalent interactions present within PTP1B for subsequent analysis with our program. To go alongside the interactions (i.e., the features) generated from PyContact[1] calculations, we classified the conformation of the WPD-loop in each frame in order to generate the target variable (for classification). For this, we used a simple rule-based scheme to define the conformation of the WPD-loop for each frame: "closed" (if the WPD-loop RMSD to the closed loop crystal structure was ≤1.5 Å) or "open" (if the WPD-loop RMSD to the open loop crystal structure was ≤1.5 Å) and "neither" (if the WPD-loop RMSD to both the closed and open crystal structure conformations was ≤1.5 Å or if the RMSD was >1.5 Å to both crystal structures). This is an approximate classification as the WPD-loop conforma-

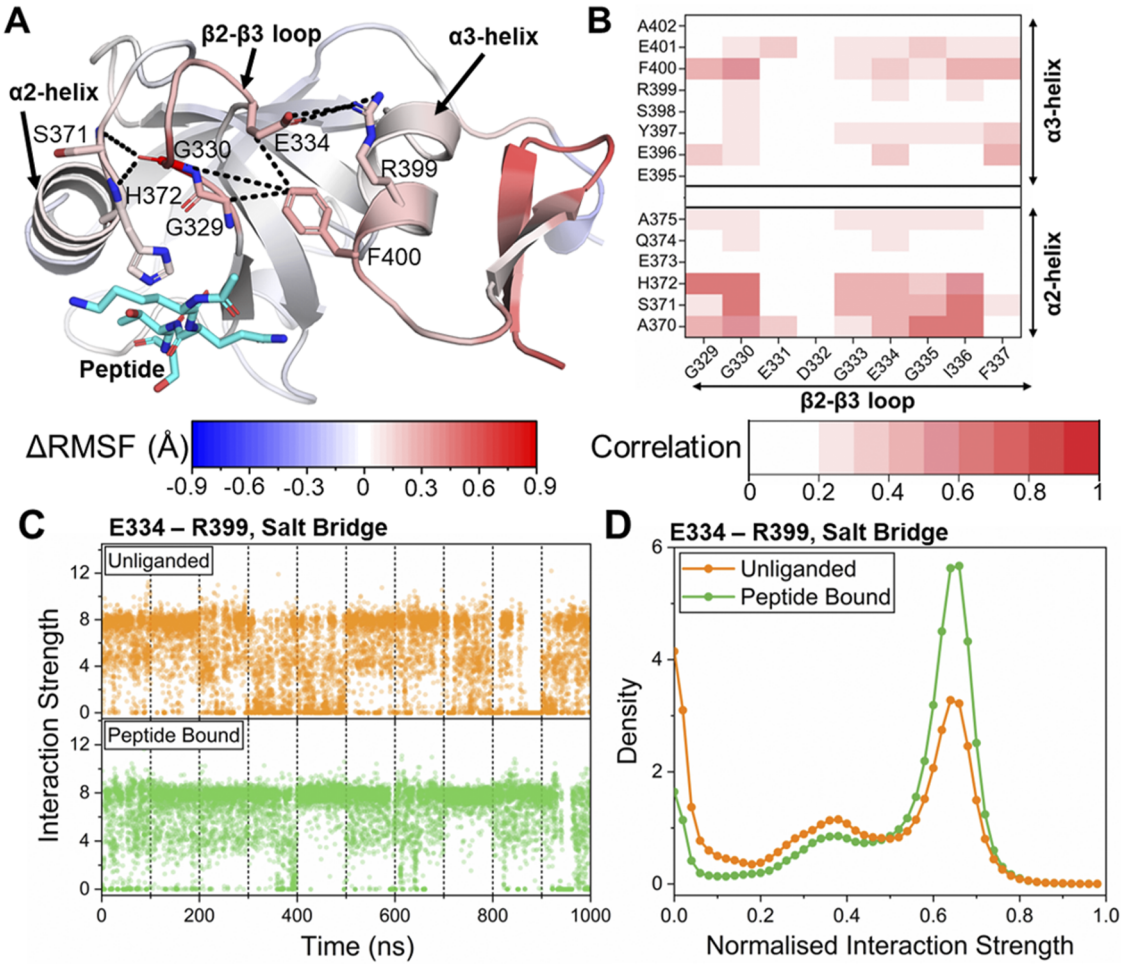

FIG. 4. A focus on allosteric signaling from the α2-helix to the α3-helix via the β2-β3 loop. (a) Projection of the difference in calculated C_α-atom root mean squared flexibility (ΔRMSF, Å) for each PDZ3 domain residue. Residues are color mapped from red (more flexible in unliganded simulations) through white (equally flexible in both simulations) to blue (more flexible in peptide bound simulations). Key non-covalent interactions identified by our package as important (see Table S6) between the α2-helix, β2-β3 loop, and α3-helix are indicated with black dashed lines. (b) The linear correlation of the β2-β3 loop residues to the α2- and α3-helices of the PDZ3 domain determined from our MD simulations with the peptide bound (equivalent results for the unliganded simulations are presented in Fig. S6). (c) PyContact[1] calculated interaction strength against simulation time for unliganded and peptide bound simulations for the salt bridge interaction between E334 and R399. The dotted lines drawn every 100 ns represent the start of a new replica. (d) Kernel density estimations of the normalized interaction strength of the E334–R399 salt bridge for both unliganded and peptide bound MD simulations.

tion can fall on a spectrum between "open" and "closed states," occupying multiple stable conformations along that spectrum.[62] However, such a simplified definition enabled us to perform binary classification using both the ML and statistical analysis modules by first discarding any frames that were classified as "neither" (348 of 10 000 frames) according to the aforementioned protocol. The results from the ML module reflected the same issue as was described for the PDZ3 domain (only describing some and not all of the key interactions/residues; see Sec. S2 of the supplementary material), so herein, we focus on our results generated only from the statistical analysis module.

We built two models describing the differences in the interactions between closed and open WPD-loop conformations using two different approaches, the Jensen–Shannon distance (JS-distance)[78] and the mutual information (MI);[76,77] see the section titled "Methodology." As would be expected, given that the two metrics are highly related, MI and JS-distance calculations gave highly similar results, with R^2 of 0.82 when comparing their per residue scores (Fig. S7). Furthermore, of the top 50 residues identified using both methods, 45 of these residues are shared among both lists. For this reason, we focused the remainder of our analysis on only one of these approaches, the JS-distance,[78] which is summarized in Fig. 5.

Our approach using the JS-distance[78] was able to capture both local and several distal interactions, which are associated with the WPD-loop's conformational state [Figs. 5(b) and 5(c)]. Analysis of

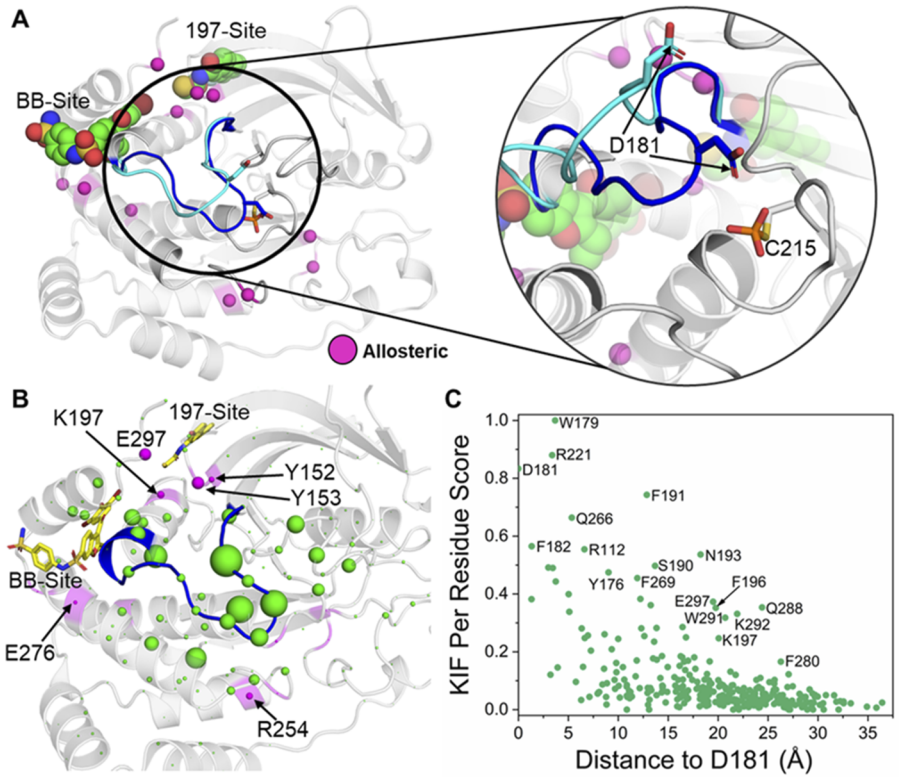

FIG. 5. Results obtained from our statistical modeling module for describing the differences in the open and closed WPD-loop conformational states of the enzyme PTP1B. (a) Crystal structures of the WPD-loop open and closed states of the enzyme PTP1B, with the closed (and catalytically active) WPD-loop conformation colored dark blue and the open conformation colored light blue. The two known allosteric drug binding sites are indicated with an exemplar drug molecule bound to each, and residues with mutations known to impact k_{cat} or K_m by ≥ 50% are indicated with pink spheres.[60,63,98] Also shown is a close-up view of the PTP1B active site, demonstrating the conformational change that occurs, with the catalytic residue D181 (on the WPD-loop) and phosphorylated C215 indicated. (b) Projection of the calculated per residue scores onto the structure of PTP1B, with a larger sphere size indicating a greater score and, therefore, greater association with the target variable. As in (a), the two known allosteric drug binding sites are depicted and the non-WPD-loop mutations that alter k_{cat} and/or K_M by ≥ 50% have their cartoon shown in magenta, with other residues colored in green. (c) Plot of the minimum heavy atom distance to the catalytic residue D181 on the WPD-loop against the calculated per residue scores described in (b), with a selection of high scoring residues at different distances from D181 labeled.

Fig. 5(b) shows several residues identified as important for regulating the WPD-loop conformation located around the two known allosteric drug binding sites, alongside important residues found on or around residues with mutations that are known to alter k_{cat} or K_M of PTP1B by ≥ 50%. We used a helper function available within our package to calculate each residue's minimum heavy atom distance from the catalytic WPD-loop residue D181 and compared this to the obtained per residue scores [Fig. 5(c)]. While unsurprisingly some of the residues closest to D181 have the largest score, our analysis highlights several residues that have both a relatively high score and a relatively large distance to D181, suggesting that our method can capture both local and allosteric interactions, which regulate the WPD-loop. This is further backed up by the weak linear correlation between the per residue scores and the distance to D181 (R^2 = 0.24). Those residues that are relatively distal from D181 but are found by our model to have a significant impact on the WPD-loop are labeled in Fig. 5(c) and include residues on the α4, α6, and α7 helices (residues 189–201, 263–282, and 285–298, respectively) and loop 11 (residues 149–154), which have all previously been identified as part of PTP1B's allosteric network.[63,93]

Identifying allosteric communication pathways in PTP1B

Using the same dataset as described above for PTP1B, we used our program to generate per residue correlation and contact matrices as inputs for a Weighted Implementation of Suboptimal Path (WISP)[39] calculation, as implemented in the Bio3D[81] package. The per residue correlation matrix is generated by identifying the largest linear correlation (in absolute values) between a pair of residues found in any interaction (if one exists), and this is used to create and weight the edges (connections) between the nodes (the residues). The contact matrix is a binary and symmetrical (along the diagonal) matrix that defines if a given residue pair is in contact or not, with residues that have a minimum heavy atom distance ≤6 Å in their crystal structures, considered in contact. The contact matrix

is used to filter the edges so that only those residues considered in contact are included; see the section titled "Methodology" for further details.

WISP can be used to identify how distal regions of a protein allosterically communicate with one another. In this method, the protein is described as a graph or network, in which residues are treated as nodes (points on the graph/network) and residue–residue correlations are the edges (connections between the nodes). By defining a "source" and a "sink" (a start residue and an end residue, respectively), the shortest pathways of communication are determined between these two residues to identify the key residues and key residue–residue interactions that are responsible for signal propagation. The pre-processing we described above gave rise to a graph/network where the nodes are residues; the edges are the residue pairs in contact with one another; and the edge weights are the correlations determined between each residue pair, enabling us to perform WISP calculations on our PTP1B dataset. As PTP1B has two known allosteric drug binding sites named the BB-site and K197-site [see Fig. 5(a)], we focused our WISP analysis on these two sites, picking a residue at each site to be used as the "source" for the WISP analysis, with D181 defined as the "sink."

Figure 6 represents the results obtained from the WISP analysis performed using residue G283 as the "source" to represent the BB-site [see Fig. 5(a)] and D181 to represent the "sink," with equivalent results describing the second allosteric binding site [K197-site; see Fig. 5(a)] provided in Fig. S8. Analysis of the network close to the source shows a primary pathway of communication from G283 through M282, F280, and Q288 to K197 (all residues on the α4, α6, and α7-helices of PTP1B) before the network begins to branch out. Key residues further along the network include those on the α4-helix (N193), which shows a strong coupling to Y152 on loop 11, which is, in turn, coupled to the N-terminus of the WPD-loop through T177. Another branch point from K197 heads to E297 and S295 on the α7-helix and, in turn, to the α4-helix (A188 and F191) and then finally to the central portion of the WPD-loop (W179) and the P-loop catalytic residue R221.

The observations made above are consistent with prior experimental studies on the allosteric regulation of PTP1B's WPD-loop. For example, our observation that the allosteric communication from the α7-helix is communicated through loop 11 and then to the N-terminus of the WPD-loop is consistent with prior NMR studies.[63,93] Furthermore, NMR experiments analyzing the differences in the closed and open states identified a "hydrophobic core network" in PTP1B, which showed altered chemical shifts depending on the WPD-loop state; this network included residues F280, Y152, F191, Y152, and W179, all of which were found on our pathway.[93]

Our WISP analysis also shows that many of the edges (interactions) in the network are between side chains, meaning that they could be modified by natural or artificial evolution, making them potential targets for protein engineering. These results are interesting to consider, given that the standard way to perform WISP is to generate a dynamic cross correlation matrix (DCCM), using the C_α–C_α distances between residues, which may not be able to capture such relationships. Finally, we note that we have so far used a more qualitative approach to label the key residues and interactions along the allosteric network [i.e., by examining Figs. 6(a) and 6(c)]; however, the node degeneracy plots shown in Fig. 6(b) can be used instead if a more quantitative approach is desired.

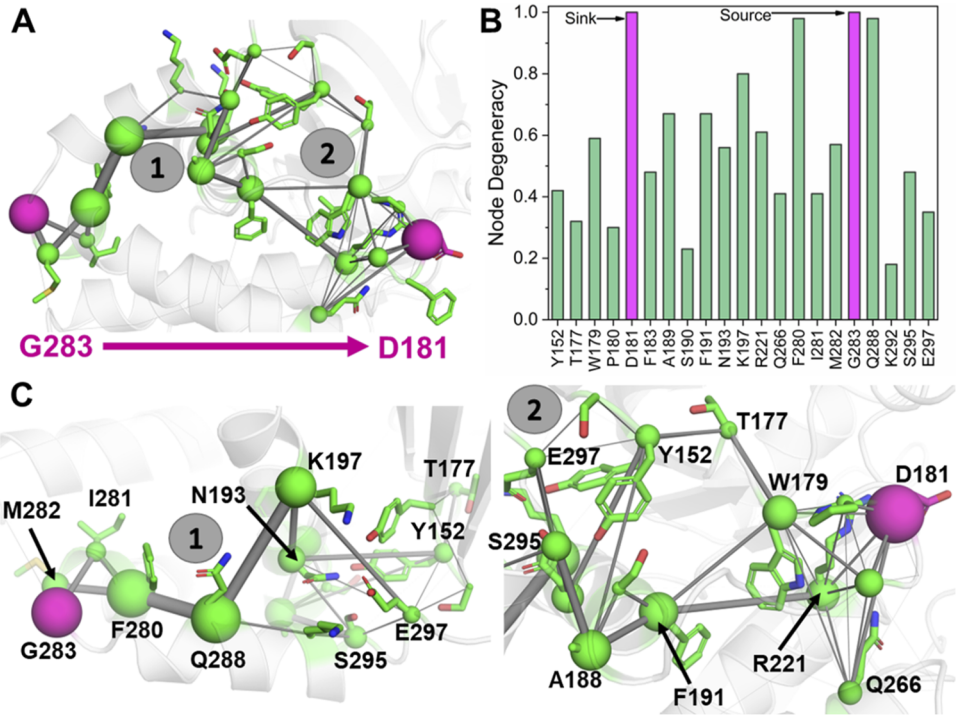

FIG. 6. Overview of WISP[39] calculations performed on PTP1B using G283 as the source and D181 as the sink. (a) Visualization of key residues and interactions along the 500 pathways determined, with the source and sink residues colored pink. How frequently a node in the network (i.e., a residue) is present in the 500 paths generated is represented by spheres, and likewise, how frequently an edge in the network (i.e., a residue–residue interaction) is present in the 500 paths generated is represented by the cylinders. In both cases, a larger size means an increased frequency and, therefore, increased importance in allosteric communication. (b) Node degeneracies (the fraction of pathways in which a given residue is present) for the 500 paths generated from G283 to D181. Only residues with node degeneracies ≥0.1 are shown for clarity. (c) A close-up view of two sections of the path depicted in (a), with the major residues and interactions present along the path shown. Equivalent results for the 500 pathways generated between E200 to D181 are shown in Fig. S8.

Conformational selection in the directed evolution of a Kemp eliminase

Our final model system focuses on two engineered Kemp eliminase variants (Fig. 7) from the KE07 series, with k_{cat} and k_{cat}/K_m differences of about 35-fold between the two variants studied here [Fig. 7(b)] and ~100-fold between the initial design and the final optimized variant from the series.[65–67] A prior study that combined x-ray crystallography, MD simulations, and fluorescence spectroscopy to study these variants identified conformational selection as playing a key role in enhancing enzymatic activity,[70] with the emergence of multiple active site conformations along the directed evolution trajectory, during which the lowest energy active site conformation changed from the "A" to "C" conformation as the enzyme activity increased [Fig. 7(c)]. Four key residues make up part of the enzyme active site [Fig. 7(c)], with W50 used to bind the substrate and stabilize the transition state; E101 used to act as base (for C–H bond cleavage); Y128 used to bind the substrate; and K222 used as a hydrogen-bond donor to stabilize the phenoxide intermediate.[65] The transition from the "A" to "C" conformational state relates to a flip of the W50 ring to hydrogen bond with E101 and a breaking of the E101-K222 hydrogen bond [Fig. 7(c)].

To sample active site conformations of the two KE07 variants described above, we performed eight Hamiltonian replica exchange MD (HREX-MD)[64] simulations of 200 ns length each using the same protocol as described in our prior work.[70] HREX-MD is a form of enhanced sampling simulation, and with these simulations, we were able to generate free energy landscapes representing the relative stability of the two major active site conformations observed in crystal structures of these variants using the χ_1 and χ_2 angles of W50 [Fig. 8(a)].[70] We have selected the R4 variant, in particular, as this is the first variant in which a sizable population of the catalytically important C conformation is observed.[70] Note that for conciseness, we do not discuss here the conformational flip of the E101 side chain shown in Fig. 7(c); for further discussion of this shift, see Ref. 70.

Analysis of Fig. 8(a) shows a population shift toward the "C" state and away from the "A" state from R1 to R4, in line with prior work.[70] Note that, as pointed out in prior work, the "B" state

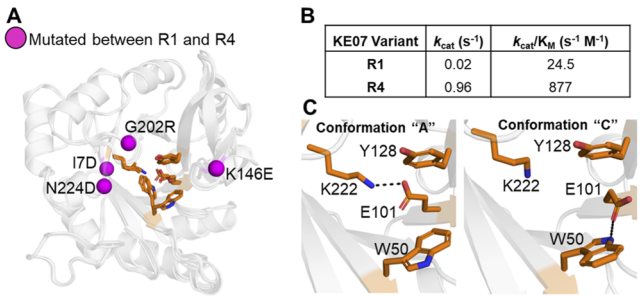

FIG. 7. Designed Kemp eliminase variants from the KE07 series[65–67] over the course of directed evolution (DE). (a) Corresponding crystal structures of the selected KE07 variants, with mutation sites indicated by spheres and labeled. (b) k_{cat} and k_{cat}/K_M of the two KE07 variants that will be evaluated in this study.[65–67] (c) The two major active site conformations of the selected KE07 variants,[70] with the key catalytic residues that remain unchanged throughout the DE process labeled.

appears to be a transient conformation that appears in the transition from the "A" and "C" states.[70] Confident that our simulations could reproduce the previously observed population shift, we applied our package to help understand how this population shift had occurred through the four mutations that separate R1 and R4 (I7D, K146E, G202R, and N224D). For both systems, we performed regression analysis using the W50 χ_2 angle as the target variable [which clearly separates the "A" and "C" states; see Fig. 8(a)]. To help focus the regression analysis on the population shift of interest (away from the "A" state and toward the "C" state), we filtered frames to only include those with W50 χ_1 angles $\geq 160^\circ$ and $\leq 240^\circ$, effectively removing some "outlier" frames or those in the "B" conformational state, as described in the section titled "Methodology." This left us with 9281 and 9714 frames (from the initial 10 000 each) for R1 and R4, respectively, with which to build our models; see Table S7 for the relative populations of each state.

As with our prior model systems, we utilized both the ML and statistical analysis modules available in our package to generate five models for both the R1 and R4 variants (three models with the ML module and two with the statistical analysis module). We quantified the error in our regression ML models by calculating the mean absolute error (MAE) and root mean squared error (RMSE) on a validation set containing 15% of the data (Table S9). For all six ML models generated (three for each of the R1 and R4 variants), MAEs and RMSEs were substantially smaller than the differences in W50 χ_2 dihedral angle values separating the "A" and "C" conformational states (Table S9), suggesting that the models produced for both KE07 variants can easily discriminate between the two states. As our goal was to compare the differences in per feature and per residue scores obtained for two different enzyme variants (i.e., the R1 vs R4 variants), highlighting only some of the important features/residues instead of all (as we observed occur for our ML results in both PTP1B and PDZ3) would give rise to misleading comparisons. We therefore focused our analysis on the results obtained from the statistical analysis package, in particular those generated using the mutual information (MI) metric[76,77] (as both metrics gave very similar per residue scores, with R^2 values of ~0.8; see Fig. S9).

We extracted the per residue scores for the R1 and R4 variants generated by our models (Fig. S10) and used this to determine the difference (Δ) in per residue scores for these variants [Figs. 8(b) and S11]. We note that another approach could be to calculate the "mutual information gain" by identifying the largest differences in the MI values for each feature/contact between the two enzymes. That said, the approach taken still allows for the easy identification of those residues that have notably changed. By analyzing these differences, we identified four residues that became (relatively) "more important" in R4 (K222, I7D, H201, and G202R) alongside four residues that were (relatively) "more important" in R1 (W50, T104, N224D, and H228). Furthermore, while the catalytic residues K222 and W50 show a notable change in score, the corresponding values for E101 and Y128 are close to 0 [Fig. 8(b)], suggesting that the role these residues play in regulating the W50 conformational state was not notably impacted by these mutations. Finally, by comparing the top per feature scores obtained for the R1 and R4 variants, we observed that the top features in R4 are substantially more correlated with the target variable than those in R1 and that this is true for both metrics used (Fig. S12). This would

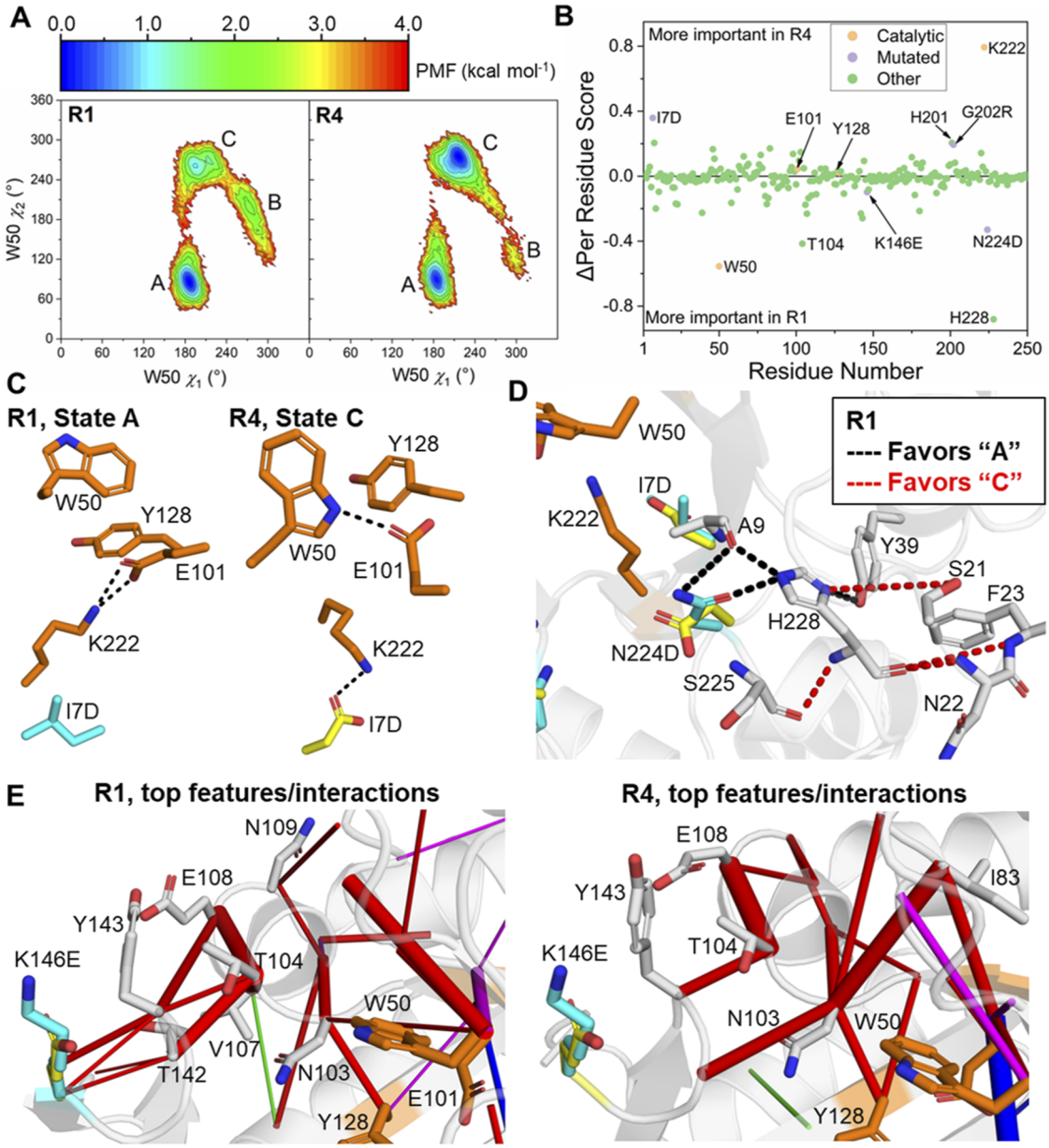

FIG. 8. (a) Free energy landscapes obtained from HREX-MD[64] simulations of the R1 and R4 KE07 variants,[65–67] as described by the first two side chain dihedral angles of W50, with the three conformational states described in Fig. 7(c) and Ref. 70 indicated. (b) Difference (Δ) in per residue scores for the R1 and R4 KE07 variants, obtained by performing regression analysis on the W50 χ_2 angle and using the mutual information (MI) as the metric.[76,77] Positive scores depict an increased (relative) importance in the R4 variant, while negative scores depict an increased (relative) importance in the R1 variant. (c) Illustration of how the I7D mutation enhances the favorability of the "C" conformational state by enabling E101 to engage W50. (d) The interaction network surrounding N224D in R1. Interactions notably coupled to W50's conformation (see Table S8) are indicated with black dashed lines, indicating interactions which when strengthened favor the "A" conformational state, with red dashed lines favoring the "C" conformational state. (e) Projection of the per feature scores determined for the regression analysis (on the W50 χ_2 angle using the MI as the metric) onto the structures of R1 and R4, focusing on features/interactions near the K146E mutation site. The thickness of the cylinder indicates the relative score of the feature/interaction. Cylinders are colored according to their interaction type, with red indicating a hydrogen bond, blue indicating a salt bridge, green indicating a hydrophobic interaction, and purple indicating a van der Waals interaction.

suggest that the overall enzyme scaffold is playing a more active role in regulating the conformational state of W50 in R4 than in the R1 variant.

Following from this, we shift our focus to characterizing the role each mutation has had toward altering the conformational sampling of the W50 side chain, starting with the active site mutation I7D. By analyzing the strength of several interactions against the distributions of the W50 χ_2 dihedral angle in simulations of R1 and R4 variants (Fig. S13), we can identify how the I7D mutation favors the "C" conformational state [as summarized in Fig. 8(c)]. Specifically, in the R4 variant, the I7D and K222 side chains form a salt bridge, which in turn disrupts K222's salt bridge with E101, which, in addition to elevating pK_a of E101, also enables the W50 side chain to form a hydrogen bond with E101 [Fig. 8(c)], stabilizing the new conformation. The I7D-K222 interaction was ~6 times stronger in the "C" state and K222-E101 was ~12 times weaker in the "C" state, with both interactions having a mutual information score of ~0.5 and a linear correlation of ~|0.7| (Fig. S13). In contrast, the R1 K222-E101 salt bridge is substantially less correlated with the conformation of W50, with a mutual information score of just 0.09, and with a linear correlation of −0.31 (Fig. S13).

From our calculations, the G202R mutation does not appear to directly impact the conformational sampling of the W50 side chain, as no notable non-covalent interactions were identified that differ between the "A" and "C" conformational states for either the R1 or R4 variants (Fig. S14). Instead, this mutation has been shown by EVB calculations to provide transition state (TS) stabilization,[70] which would not be detected by our approach (given that these are classical MD simulations and that the conformation of W50 is the target variable).

Similarly to the G202R mutation, the N224D mutation is likely to have an electrostatic impact on TS stabilization due to its location near the active site and the associated change in charge, as indicated also by calculations of Arrhenius parameters for the R1 and R4 variants.[70] That said, our calculations show that the mutation also impacts the conformational sampling of the active site. By analyzing the per feature scores obtained for the region around the N224D mutation site, we identified a network of hydrogen bonds present in the R1 variant that are correlated with the sampling of the W50 χ_2 dihedral angle [Fig. 8(d) and Table S10]. Four of these interactions favor the "A" conformation and are all native interactions (i.e., they are present in the R1 x-ray structure, PDB ID: 5D2W[70]), while 2/4 of the interactions that favor the "C" conformation are non-native interactions and require the H228–N224D interaction to break in order for them to be able to form [Fig. 8(d) and Table S10]. By comparing the per feature scores for these interactions in the R1 and R4 variants, we identified that the N224D mutation effectively abolishes this allosteric network, as the value of many of these interactions is close to 0 in the R4 variant (Table S10). Given that native interactions in the R1 variant would appear to favor the "A" conformational state, the removal of this allosteric network would seem likely to help shift the population of W50 toward the "C" state.

The K146E mutation lies on the tip of the loop outside of the active site, and the absence of clear electrostatic or structural impact on the scaffold has made it challenging to pin-point a reason for enhanced activity as a result of this mutation.[65,70] By projecting the per feature scores onto the structures of the R1 and R4 variants, we were able to identify a network of hydrogen bonds with a strong relationship to the W50 χ_2 dihedral angle, as shown in Fig. 8(e). Figure 8(e) depicts how the loop K146E is located on interacts with a neighboring helix (residues 103–109) through several hydrogen bonds. By analyzing the linear correlation of these interactions to the W50 χ_2 dihedral angle (i.e., the target variable), we observed that the interactions between the loop and the helix and those within the helix that are shown in Fig. 8(e) are all inversely correlated with the χ_2 dihedral angle in both the R1 and R4 variants (Table S10). This means that the strengthening of these interactions would be expected to drive the conformation of W50 toward the "A" conformational state. We calculated the root mean squared fluctuations (RMSFs) of both the R1 and R4 variants to determine the difference in RMSF as a result of the four mutations (ΔRMSF, Fig. S15). Our results show that not only is the R4 variant generally more flexible, but that both the loop on which K146E is located and the neighboring helix are destabilized in this variant (Fig. S15). Taken together, our results suggest that the K146E mutation destabilizes its local environment, reducing the strength of these interactions and ultimately their ability to drive W50 toward the "A" conformational state.

Evaluating the reproducibility of our simulations

While our simulations on KE07s were enhanced through Hamiltonian replica exchange simulations,[64] our simulations of PTP1B and the PDZ3 domain were performed using conventional MD and were multi-replica, with ten independent replicas performed from each state ("closed" and "open" for PTP1B and "bound" and "unliganded" for the PDZ3 domain). This enabled us to evaluate how reproducible using differing numbers of replicas would have been for our analyses of PTP1B and the PDZ3 domain and, therefore, the minimum number of replicas needed to obtain reproducible results. This is important to consider as it has been previously demonstrated that false positive (or false negative) conclusions can be drawn from analyzing a single or only a few replica(s) of conventional MD simulations.[109,110] To do this, we turned to statistical techniques: bootstrapping with replacement and random subsampling. These techniques are highly similar and involve generating many resamples of a dataset through randomly selecting combinations of observations (with it also being possible to duplicate individual observations).

We applied this method to our KIF calculations on PTP1B and the PDZ3 domain to evaluate how many MD simulation replicas would have been needed to obtain reproducible results. For several different numbers of replicas, we generated 5000 resamples and determined the per residue scores for each resample. We then calculated the correlation between the per residue scores obtained for all 20 unique replicas against each sample's per residue scores, and we used this to generate KDEs of the distribution of correlation scores possible for a given number of replicas. Results for this analysis using the Spearman's rank correlation coefficient (which measures the ability to rank order the per residue scores correctly) are represented in Fig. 9(a), while those using the Pearson correlation coefficient (which measures the linear correlation) are represented in Fig. S16(a). We note that an even number of simulations of each state were used for all bootstrap resamples. For example, in the scenario for PTP1B with ten replicas, five replicas each would

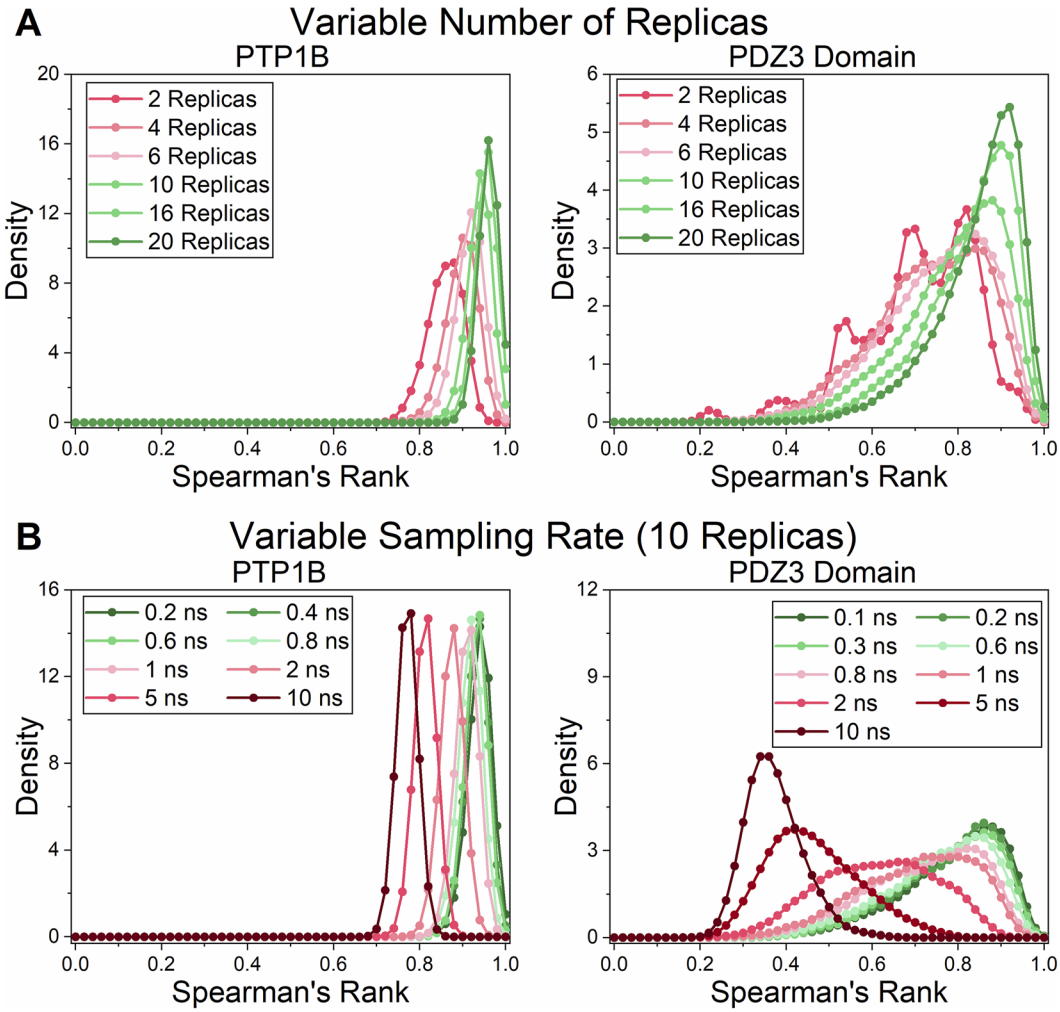

FIG. 9. Gaussian kernel density estimates (KDEs) obtained from performing bootstrapping with replacement and random subsampling (5000 samples were generated for each scenario) for the per residue scores calculated for PTP1B and the PDZ3 domain. (a) Bootstrapping with replacement and random subsampling performed using a differing number of simulation replicas. The Spearman's rank correlation coefficient between each of the bootstrap samples per residue scores against those obtained from the 20 unique replicas was used to build each KDE shown. (b) Bootstrapping with replacement and random subsampling using a random selection of ten replicas for each bootstrap sample with the time step between each frame used in the analysis altered (meaning that those with larger time steps also have correspondingly less frames available). For both PTP1B and the PDZ3 domain, an equal number of simulations from both conformational states ("closed" and "open" WPD-loop or "peptide bound" and "unliganded," respectively) were used in bootstrapping/subsampling calculations; see the section titled "Methodology" for further details. Equivalent results using the Pearson correlation coefficient are provided in Fig. S16.

be obtained from both the "closed" and "open" WPD-loop conformation replicas; see the section titled "Methodology" for further details.

In the case of PTP1B, even a single pair of replicas (one from the "closed" conformation and one from the "open" conformation) shows quite reproducible results according to both correlation metrics used [Figs. 9(a) and S16(a)]. This contrasts strongly with the PDZ3 domain whereby highly misleading results could be obtained from just a single pair of replicas [Figs. 9(a) and S16(a)]. This contrast is likely a result of two factors, the first being that magnitude of the difference in the conformational states of PTP1B is larger than that of the PDZ3 domain. That is, while in PTP1B, the largest differences will be from interactions present in one state and not present at all in the other, for the PDZ3 domain, interactions will generally be present in both states but stronger in one state, therefore requiring more sampling to reliably differentiate. The second is the available crystal structure of PTP1B in both the closed and open states, enabling us to confidently assign and simulate both states. In contrast, simulations of the PDZ3 domain's unliganded state were performed using the peptide bound crystal structure (after a 100 ns equilibration simulation; see the section titled "Methodology"). Another consideration is that the sampling

of each state was obtained using different simulations, for example, PDZ3 domain peptide bound simulations were used to obtain frames for the peptide bound state and vice versa for the unliganded state. For simulations in which both (or more) states are obtained from the same set simulations (i.e., sampling back and forth between states in each replica), the sampling required would also likely be affected by decorrelation times associated with transitioning from one state to another, that is, the time taken for the structure to fully adapt to the new conformation/configuration after undergoing a conformational change away from another conformation.

Based on the results represented in Figs. 9(a) and S16(a), we would argue that ten replicas (five pairs of replicas from each state) to be enough to obtain quite reliable and reproducible results for both systems and decided to use this number of replicas for an additional bootstrapping/subsampling investigation. In this investigation, we tested altering the sampling rates for KIF calculations. For example, in the case of PTP1B, frames were originally analyzed with PyContact[1] and then KIF using a time spacing of 0.2 ns. Bootstrapping/subsampling was used to investigate if using a larger frame spacing could still lead to reproducible results for these systems. This would also practically mean a reduced number of frames to analyze (e.g., a 0.4 ns spacing bootstrap calculation has half the number of frames as a 0.2 ns calculation). The results obtained for the bootstrapping with replacement and random subsampling calculations for both PTP1B and the PDZ3 domain are represented in Figs. 9(b) and S16(b). Our results show that a reduced rate of sampling could indeed be used as the per residue scores would still be reproducible. Naturally, there are limits to how large the sampling rate can be, which is most clearly demonstrated for the PDZ3 domain [Figs. 9(b) and S16(b)]. That said, it would appear reasonable that for both test systems, a reduced sampling rate of 0.6 ns would be enough to obtain reliable and reproducible results (when performed alongside using five pairs of replicas).

CONCLUSIONS

With often thousands of non-covalent interactions present in standard MD simulation, identifying the key interactions responsible for regulating a given process of interest can be challenging affair, especially in the case of interactions between residues distant from the site of interest. By building on and interfacing with well-established tools within the biomolecular simulation community,[1,38,81] our package provides a largely automated approach for a user to identify and rank these key interactions for further investigation. The novelty of this is threefold. (1) Exploiting non-covalent interactions allows KIF to directly evaluate the features that regulate protein conformational changes (the non-covalent interaction network). (2) KIF provides a new tool for the community that makes this analysis much more straightforward and streamlined. (3) We demonstrate the effectiveness of this approach on a range of model systems. Specifically, we have applied our package to three different exemplary systems to demonstrate potential use cases and evaluate how the methods within perform relative to one another. We note that in all three cases, KIF was applied to "unbiased" simulation datasets; this includes the KE07 analysis, in which the neutral replica from HREX-MD simulations was used for analysis with KIF; see the section titled "Methodology." If one wishes to analyze biased simulation datasets with KIF, this can still be done; however, one has to do so with the caveat that the relative weight of each frame/observation is no longer identical. It is therefore recommended to use KIF with unbiased simulation data if possible. It should be noted that we were able to obtain our results on all three model systems used using ≤ 2 μs of simulations per system, which is routinely accessible with modern computing resources. Overall, and in all three cases, we were able to gain detailed insights into each conformational change studied. This includes both large scale conformational changes (PTP1B's WPD-loop conformations) and much more subtle conformational changes (differences in active site side change conformations for the KE07 variants). Furthermore, we were able to apply our package to study not only conformational changes but also the impact of the removal of a peptide on its protein binding partner (the PDZ3 domain).

Any user of KIF needs to consider how to describe their target variable and if their simulations adequately sample along this target variable. In all the examples shown here, we were able to use between 10 and 20 000 frames obtained from between 1 and 2 μs of MD simulation sampling, which is easily routinely available especially with the GPU code available from MD simulation programs, such as AMBER[111,112] and GROMACS.[113] In our opinion, binary classification is likely to be a good approach to describe the target variable because it allows one to clearly define the problem. In contrast, multi-class classification could be liable to identifying features important for distinguishing between two classes, but have no relevance between, for example, other two classes. Likewise, in the case of regression, the results obtained can be sensitive to outliers/extremes in the target values depending on the distribution of the target values. We note that for the analysis of PTP1B, we used cutoffs to convert the continuous values obtained from the RMSD and dihedral calculations into a binary classification problem.

The classification ML models generated for both the PDZ3 domain and PTP1B were highly accurate but demonstrated a clear limitation with ML to generate per feature scores (directly from the model's per feature importance). That is, the ML models did not need all the "important" features in order to accurately distinguish between the classes. On the other hand, our statistical analysis module was able to identify both local and distal features that compared well with the experiment, and the metrics used in this module to calculate per feature scores do not suffer from this issue. While the ML methods can provide one with the confidence that the dataset contains the information needed to accurately distinguish between the different classes/values of the target variable, we would recommend turning to the statistical analysis package for both classification and regression. One possible application of the ML package in KIF would be to identify a minimal set of interactions needed to describe a conformational change of interest. The interactions identified from this approach could then be described by something quick and easy to measure, such as a set of distances, and used as collective variable(s) for enhanced sampling simulations (such as metadynamics simulations) to rapidly simulate the conformational change of interest. This approach could also benefit from approaches such as RAVE (Reweighted autoencoded variational Bayes for enhanced sampling),[13] which would enable further reduction of the number of collective variables. This idea would require further study beyond the scope of this paper, however.

KIF performed well at identifying both local and distal interactions/residues associated with each conformational change studied. By tuning the question being asked (i.e., what is the target variable) and the feature/interaction selection (whether to include/exclude certain residues), a user can to some extent control whether the output of KIF will focus on more local interactions or more distal interactions. The per interaction and per residue scores that can be calculated with KIF enable analysis at different levels of detail. As we showed with the PDZ3 domain, we believe that the per residue scores are best suited to providing an overview of the system and for making comparisons with experimental data (if available). On the other hand, the per interactions scores can be used to focus on the understanding a region of interest in greater detail.

By generating correlation and distance matrices from PyContact[1] datasets, we were able to generate the necessary inputs for WISP[39] calculations on PTP1B's two known allosteric sites. This allowed us to identify the interaction networks responsible for allosteric communication between each of the allosteric sites to the active site. Furthermore, with the aforementioned matrices generated, numerous other graph theory approaches, such as community network analysis[40] and shortest path map,[41] could be readily applied depending on the specific goals of the project.

We utilized bootstrapping and random subsampling to evaluate the reproducibility of our calculations for PTP1B and the PDZ3 domain. The different sensitivities to the number of replicas for both model systems demonstrate the reality that reproducibility is system dependent and also dependent on the question(s) being asked (i.e., what the target variable is). That said, we showed one such way to evaluate if a set of simulations are reproducible, and furthermore, it is likely reasonable that for a set of highly similar systems (such as a wild-type enzyme and a series of variants to study), the procedure required would be similar. This would mean, in practice, that bootstrapping and random subsampling would only need to be performed on one system to figure out an appropriate protocol, before being applied to the other systems.

Beyond assisting basic research efforts to increase the understanding of specific protein interaction networks, our package has clear potential uses in the field of protein engineering. Firstly, being able to identify the key interactions that regulate a conformational state can be used to assist (semi-)rational design efforts toward enhanced activity by locating specific interactions/residues to target/alter. Furthermore, our statistical analysis module can be used to determine if (1) the given feature's interaction strength is positively or negatively correlated with the target value (for regression) or (2) if the feature stabilizes or destabilizes a given conformational state (in the case of classification). For the analysis of both PTP1B and the PDZ3 domain, we were able to identify allosteric residues/sites by plotting the per residue score against the distance from the site of interest. This approach could therefore be used to identify novel allosteric sites or better characterize existing allosteric sites in drug discovery efforts.

SUPPLEMENTARY MATERIAL

See the supplementary material for (1) additional information about the methodology used in this work; (2) supplementary results about the machine learning module; (3) validation of the implementation of KIF using a synthetic dataset; (4) protonation and histidine tautomerization states used in MD simulations; (5) descriptions of the feature filtering performed prior to KIF calculations on all systems; (6) measures of the performance of all ML models built; (7) selected per feature scores to support arguments made in the main text regarding the PDZ3 domain and KE07s; (8) a schematic of the chemical mechanism of PTP1B catalysis; (9) a scatter matrix comparing all the calculation methods used to calculate per residue scores for the PDZ3 domain; (10) visualization of the per residue scores for the PDZ3 domain; (11) RMSFs and correlation matrices for the PDZ3 domain; (12) comparisons of per residue scores obtained for various model systems; (13) WISP calculation results on PTP1B with E200 as the source and D181 as the sink; (14) free energy landscapes obtained for all KE07 variants from our HREX-MD simulations; (15) projection of per residue and delta per residue scores obtained for the KE07 variants; (16) top per feature scores for the regression analysis on the R1 and R4 KE07 variants; (17) scatter plots describing the role of the I7D mutation in the R1 and R4 KE07 variants; (18) projection of the per feature scores and RMSFs for the R1 and R4 KE07 variants; and (19) bootstrapping and random subsampling results obtained for PTP1B and the PDZ3 domain obtained using the Pearson correlation coefficient.

ACKNOWLEDGMENTS

This work was supported by the Carl Tryggers Foundation for Scientific Research (postdoctoral fellowship to R.M.C., Grant No. CTS 19:172), the Knut and Alice Wallenberg Foundation (Wallenberg Academy Fellowship and Wallenberg Scholar Grants to S.C.L.K., Grant Nos. 2018.0140 and 2019.0431), the Swedish Research Council (Grant No. 2019-03499), and an American–Scandinavian Foundation Fellowship to J.S.G.S. Computational resources were provided by the Swedish National Infrastructure for Computing (Grant Nos. 2019/2-1, 2019/3-258, and 2020/5-250), and simulations were performed on the BerzeLiUs and Tetralith clusters at NSC Linköping.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Rory M. Crean: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (equal); Validation (equal); Visualization (equal); Writing – original draft (lead); Writing – review & editing (equal). **Joanna S.G. Slusky**: Conceptualization (equal); Formal analysis (equal); Methodology (equal); Writing – review & editing (equal). **Peter M. Kasson**: Formal analysis (equal); Methodology (equal); Writing – review & editing (equal). **Shina Caroline Lynn Kamerlin**: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Funding acquisition (equal); Methodology (equal); Project

administration (equal); Resources (equal); Supervision (equal); Writing – original draft (supporting); Writing – review & editing (equal).

DATA AVAILABILITY

The data (PyContact and KIF generated results) that support the findings of this study are openly available in Zenodo at http://doi.org/10.5281/zenodo.7104965. The raw MD simulation data were generated at the BerzeLiUs and Tetralith clusters at NSC Linköping. These data are available from the corresponding author upon reasonable request. The code for KIF is freely available for download from GitHub at the following link: https://github.com/kamerlinlab/KIF, under a GNU General Public License v2.0.

REFERENCES

[1] M. Scheurer *et al.*, Biophys. J. **114**, 577 (2018).
[2] S. Contreras-Riquelme *et al.*, PeerJ **6**, e5998 (2018).
[3] O. Serçinoğlu and P. Ozbek, Nucleic Acids Res. **46**, W554 (2018).
[4] L. Molgedey and H. G. Schuster, Phys. Rev. Lett. **72**, 3634 (1994).
[5] J. D. Chodera, N. Singhal, V. S. Pande, K. A. Dill, and W. C. Swope, J. Chem. Phys. **126**, 155101 (2007).
[6] Y. Naritomi and S. Fuchigami, J. Chem. Phys. **139**, 215102 (2013).
[7] J. D. Chodera and F. Noé, Curr. Opin. Struct. Biol. **25**, 135 (2014).
[8] J. Preto and C. Clementi, Phys. Chem. Chem. Phys. **16**, 19181 (2014).
[9] M. M. Sultan *et al.*, J. Chem. Theory Comput. **10**, 5217 (2014).
[10] C. Wehmeyer and F. Nóe, J. Chem. Phys. **148**, 241703 (2017).
[11] B. E. Husic and V. S. Pande, J. Am. Chem. Soc. **140**, 2386 (2018).
[12] A. Rodriguez *et al.*, J. Chem. Theory Comput. **14**, 1206 (2018).
[13] J. M. L. Ribeiro *et al.*, J. Chem. Phys. **149**, 072301 (2018).
[14] Y. Wang, J. M. L. Riberio, and P. Tiwari, Nat. Commun. **10**, 3573 (2019).
[15] W. Chen, H. Sidky, and A. L. Ferguson, J. Chem. Phys. **150**, 214114 (2019).
[16] Y. Wang, J. M. L. Ribeiro, and P. Tiwary, Curr. Opin. Struct. Biol. **61**, 139 (2020).
[17] O. Fleetwood *et al.*, Biophys. J. **118**, 765 (2020).
[18] P. Ravindra, Z. Smith, and P. Tiwary, Mol. Syst. Des. Eng. **5**, 339 (2020).
[19] L. Bonati, V. Rizzi, and M. Parrinello, J. Phys. Chem. Lett. **11**, 2998 (2020).
[20] A. Mardt and F. Noé, J. Chem. Phys. **155**, 214106 (2021).
[21] G. Diez, D. Nagel, and G. Stock, J. Chem. Theory Comput. **18**, 5079 (2022).
[22] G. Bussi *et al.*, J. Am. Chem. Soc. **128**, 13435 (2006).
[23] A. Barducci, G. Bussi, and M. Parrinello, Phys. Rev. Lett. **100**, 020603 (2008).
[24] A. Barducci, M. Bonomi, and M. Parrinello, Wiley Interdiscip. Rev.: Comput. Mol. Sci. **1**, 826 (2011).
[25] C. Abrams and G. Bussi, Entropy **16**, 163 (2014).
[26] P. Tiwary and M. Parrinello, J. Phys. Chem. B **119**, 736 (2015).
[27] A. Pérez de Alba Ortíz *et al.*, J. Chem. Phys. **149**, 072320 (2018).
[28] M. M. Sultan and V. S. Pande, J. Chem. Phys. **149**, 094106 (2018).
[29] H. Sidky, W. Chen, and A. L. Ferguson, Mol. Phys. **118**, e1737742 (2020).
[30] M. Chen, Eur. Phys. J. B **94**, 211 (2021).
[31] F. Hooft, A. Pérez de Alba Ortíz, and B. Ensing, J. Chem. Theory Comput. **17**, 2294 (2021).
[32] T. Lenaerts *et al.*, BMC Struct. Biol. **8**, 43 (2008).
[33] C. L. McClendon *et al.*, J. Chem. Theory Comput. **5**, 2486 (2009).
[34] U. Doshi *et al.*, Proc. Natl. Acad. Sci. U. S. A. **113**, 4735 (2016).
[35] G. A. Cortina and P. M. Kasson, Bioinformatics **32**, 3420 (2016).
[36] S. Singh and G. R. Bowman, J. Chem. Theory Comput. **13**, 1509 (2017).
[37] X.-Q. Yao, M. Momin, and D. Hamelberg, J. Chem. Inf. Model. **58**, 1325 (2018).
[38] E. Lilkova *et al.*, The PyMOL Molecular Graphics System, Version 2.0, Schrödinger, LLC, 2015.
[39] A. T. van Wart *et al.*, J. Chem. Theory Comput. **10**, 511 (2014).
[40] I. Rivalta and V. S. Batista, Methods Mol. Biol. **2253**, 137 (2021).
[41] A. Romero-Rivera, M. Garcia-Borràs, and S. Osuna, ACS Catal. **7**, 8524 (2017).
[42] J. Wang *et al.*, Nat. Commun. **11**, 3862 (2020).
[43] K. V. Brinda and S. Vishveshwara, Biophys. J. **89**, 4159 (2005).
[44] C. Chennubhotla and I. Bahar, Mol. Syst. Biol. **2**, 36 (2006).
[45] C. Chennubhotla and I. Bahar, PLoS Comput. Biol. **3**, e172 (2007).
[46] A. R. Atilgan, D. Turgut, and C. Atilgan, Biophys. J. **92**, 3052 (2007).
[47] C. Atilgan and A. R. Atilgan, PLoS Comput. Biol. **5**, e1000544 (2009).
[48] M. S. Vijayabaskar and S. Vishveshwara, Biophys. J. **99**, 3704 (2010).
[49] M. De Ruvo *et al.*, Biophys. Chem. **165–166**, 21 (2012).
[50] I. J. General *et al.*, PLoS Comput. Biol. **10**, e1003624 (2014).
[51] V. A. Feher *et al.*, Curr. Opin. Struct. Biol. **25**, 98 (2014).
[52] B. R. C. Amor *et al.*, Nat. Commun. **7**, 12477 (2016).
[53] S. Bowerman and J. Wereszczynski, Methods Enzymol. **578**, 429 (2016).
[54] J. A. Maier *et al.*, J. Chem. Theory Comput. **11**, 3696 (2015).
[55] W. L. Jorgensen *et al.*, J. Chem. Phys. **79**, 926 (1983).
[56] H. M. Berman *et al.*, Nucleic Acids Res. **28**, 235 (2000).
[57] D. A. Doyle *et al.*, Cell **85**, 1067 (1996).
[58] V. B. Chen *et al.*, Acta Crystallogr., Sect. D: Biol. Crystallogr. **66**, 12 (2010).
[59] D. A. Case *et al.*, AMBER 18, University of California, San Francisco, 2018.
[60] R. M. Crean *et al.*, J. Am. Chem. Soc. **143**, 3830 (2021).
[61] R. Shen *et al.*, JACS Au **1**, 646 (2021).
[62] R. Shen *et al.*, Chem. Sci. **13**, 13524 (2022).
[63] D. A. Keedy *et al.*, eLife **7**, e36307 (2018).
[64] G. Bussi, Mol. Phys. **112**, 379 (2014).
[65] D. Röthlisberger *et al.*, Nature **453**, 190 (2008).
[66] O. Khersonsky *et al.*, J. Mol. Biol. **396**, 1025 (2010).
[67] O. Khersonsky *et al.*, J. Mol. Biol. **407**, 391 (2011).
[68] D. van der Spoel *et al.*, J. Comput. Chem. **26**, 1701 (2005).
[69] G. A. Tribello *et al.*, Comput. Phys. Commun. **185**, 604 (2014).
[70] N.-S. Hong *et al.*, Nat. Commun. **9**, 3900 (2018).
[71] D. R. Roe and T. E. Cheatham, J. Chem. Theory Comput. **9**, 3084 (2013).
[72] L. Breiman, Mach. Learn. **45**, 5 (2001).
[73] A. V. Dorogush, V. Ershov, and A. Gulin, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'16)* (ACM, 2016), pp. 785–794.
[74] T. Chen, and C. Guestrin, arXiv:1603.02754 (2016).
[75] F. Pedregosa *et al.*, J. Mach. Learn. Res. **12**, 2825 (2011), http://jmlr.org/papers/v12/pedregosa11a.html.
[76] C. E. Shannon, Bell Syst. Tech. J. **27**, 379 (1948).
[77] J. Kreer, IRE Trans. Inf. Theory **3**, 208 (1957).
[78] J. Lin, IEEE Trans. Inf. Theory **37**, 145 (1991).
[79] P. Virtanen *et al.*, Nat. Methods **17**, 261 (2020).
[80] R. T. McGibbon *et al.*, Biophys. J. **109**, 1528 (2015).
[81] B. J. Grant *et al.*, Bioinformatics **22**, 2695 (2006).
[82] T. Kluyver *et al.*, in *20th International Conference on Electronic Publishing*, edited by F. Loizides and B. Schmidt (IOS Press, 2016).
[83] The Pandas Development Team(2020), Zenodo. https://doi.org/10.5281/zenodo.7741580
[84] W. McKinney, in *Proceedings of the Ninth Python in Science Conference* (ScyPy, 2010), p. 56.
[85] C. R. Harris *et al.*, Nature **585**, 357 (2020).
[86] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics and function using NetworkX," in *Proceedings of the seventh Python in Science Conference* (SciPy, Pasadena, CA, 2008), pp. 11–15.
[87] A. J. Faure *et al.*, Nature **604**, 175 (2022).
[88] R. N. McLaughlin, Jr. *et al.*, Nature **491**, 138 (2012).
[89] C. Gautier *et al.*, Protein Eng., Des. Sel. **31**, 367 (2019).
[90] C. M. Petit *et al.*, Proc. Natl. Acad. Sci. U. S. A. **106**, 18249 (2009).
[91] O. Bozovic, B. Jankovic, and P. Hamm, Nat. Commun. **11**, 5841 (2020).

[92]S. K. Whittier, A. C. Hengge, and J. P. Loria, Science **341**, 899 (2013).
[93]D. S. Cui *et al.*, J. Mol. Biol. **429**, 2360 (2017).
[94]M. S. Choy *et al.*, Mol. Cell **65**, 644 (2017).
[95]Z. Y. Zhang and J. E. Dixon, Adv. Enzymol. Relat. Areas Mol. Biol. **68**, 1 (1994).
[96]D. A. Erlanson *et al.*, J Am. Chem. Soc. **125**, 5602 (2003) .
[97]N. K. Tonks, FEBS Lett. **546**, 140 (2003).
[98]C. Wiesmann *et al.*, Nat. Struct. Mol. Biol. **11**, 730 (2004).
[99]J.-P. Sun *et al.*, J. Biol. Chem. **278**, 12406 (2003).
[100]S. K. Hansen *et al.*, Biochemistry **44**, 7704 (2005).
[101]T. A. S. Brandão, A. C. Hengge, and S. J. Johnson, J. Biol. Chem. **285**, 15874 (2010).
[102]L. Ren *et al.*, Biochemistry **50**, 2339 (2011).
[103]T. A. S. Brandão, S. J. Johnson, and A. C. Hengge, Arch. Biochem. Biophys. **525**, 53 (2012).
[104]M. Feldhammer *et al.*, Crit. Rev. Biochem. Mol. Biol. **48**, 430 (2013).
[105]P. Xiao *et al.*, Int. J. Biochem. Cell Biol. **57**, 84 (2014).
[106]M. V. V. V. Sekhar Reddy *et al.*, Protein Pept. Lett. **21**, 90 (2014).
[107]G. Moise *et al.*, Biochemistry **57**, 5315 (2018).
[108]K. R. Torgeson *et al.*, Sci. Adv. **8**, eabo5546 (2022).
[109]B. Knapp, L. Ospina, and C. M. Deane, J. Chem. Theory Comput. **14**, 6127 (2018).
[110]V. Gapsys and B. L. de Groot, eLife **8**, e44718 (2019).
[111]A. W. Götz *et al.*, J. Chem. Theory Comput. **8**, 1542 (2012).
[112]R. Salomon-Ferrer *et al.*, J. Chem. Theory Comput. **9**, 3878 (2013).
[113]M. J. Abraham *et al.*, SoftwareX **1**, 19 (2015).