Anomaly Detection-based Recognition of Near-Native Protein Structures

Sivani Tadepalli*, Nasrin Akhter*, Daniel Barbará*, and Amarda Shehu*†‡
†Department of Computer Science, ‡Center for Advancing Human Machine Partnerships,
George Mason University, Fairfax, VA, 22030
§Corresponding Author
{stadepal, nakhter3, dbarbara, amarda}@gmu.edu

***Abstract*—The three-dimensional structures populated by a protein molecule determine to a great extent its biological activities. The rich information encoded by protein structure on protein function continues to motivate the development of computational approaches for determining functionally-relevant structures. The majority of structures generated in silico are not relevant. Discriminating relevant/native protein structures from non-native ones is an outstanding challenge in computational structural biology. Inherently, this is a recognition problem that can be addressed under the umbrella of machine learning. In this paper, based on the premise that near-native structures are effectively anomalies, we build on the concept of anomaly detection in machine learning. We propose methods that automatically select relevant subsets, as well as methods that select a single structure to offer as prediction. Evaluations are carried out on benchmark datasets and demonstrate that the proposed methods advance the state of the art. The presented results motivate further building on and adapting concepts and techniques from machine learning to improve recognition of near-native structures in protein structure prediction.**

***Keywords*-protein structure prediction, near-native decoys, anomaly detection, machine learning.**

I. Introduction

Decades of research have shown that protein molecules are inherently flexible [1]. The spatial arrangements in which atoms in a protein molecule position themselves in three dimensions, also referred to as structures, determine to a great extent the biological activities of a protein in the cell [2]. Due to the promise that tertiary structures holds to decode function, significant research in wet laboratories focuses on elucidating biologically-active/native structures.

However, advancements in protein structure determination in the wet laboratory are not keeping up with advancements in genome sequencing. While ever increasing, the number of known protein structures in the Protein Data Bank (PDB) lags the number of known protein-encoding genes by many orders of magnitude [3]. This gap motivates the development of computational methods as complementary tools to wet-laboratory techniques for protein structure determination [4].

The most challenging setting for computational approaches is template-free protein structure prediction (PSP). In PSP, no other known structure of a sufficiently-similar protein sequence is available for use as a template [5]. In the absence of a viable template, computational methods approach PSP as an optimization problem. The goal is to find tertiary structures that minimize an energy function that sums up interactions among atoms in a particular arrangement in a structure.

Currently, one of the main challenges is that energy functions are semi-empirical and give rise to an energy surface riddled with local minima. The majority of structures generated in silico are not relevant, hiding among them what is often a very small (or even negligible) subset of near-native structures. To further emphasize this point of the presence of very little signal in a sea of noise, the generated structures are referred to as decoys in PSP literature.

The body of work on computational methods for determining whether a structure is near-native or for selecting near-native structures from a set of computationally-determined ones is quite rich. In Section I-A we provide a glimpse into related work. However, while work is rapidly increasing, the the question of which decoys are near-native is an outstanding challenge in computational structural biology. This question can be inherently framed as a recognition problem and is thus prime to be approached under the umbrella of machine learning. Indeed, as we summarize in Section I-A, many machine learning methods are proposed to address this problem.

In this paper, we take a unique approach based on the premise that near-native structures are effectively anomalies in a distribution of computationally-generated structures by template-free PSP methods. Specifically, we leverage the concept of anomaly detection and propose two groups of methods, methods that automatically select subsets of (high-quality/near-native) decoys from a given set and methods automatically select one (high-quality) decoy to offer as prediction of a near-native structure. A diverse set of proteins are employed to test these methods and evaluate our premise. Decoys for them are generated via the state-of-the-art, Rosetta template-free PSP protocol. The results demonstrate that leveraging anomaly detection advances the state of the art and motivate further building on related concepts and techniques to improve the recognition of near-native protein structures.

A. Related Work

Since energy is an unreliable indicator of nativeness [6], [7], the question of which decoys are near-native is an out-

This work is supported in part by NSF FET Grant No. 1900061, NSF IIS Grant No. 1763233, and NSF DMS Grant No. 1821154.

standing challenge in computational structural biology. This is recognized by a special category in the biennial "Critical Assessment of protein Structure Prediction" (CASP) competition referred to as model assessment [4]. The term 'model' in this context refers to a tertiary structure, and 'model assessment' refers to the evaluation of a structure via a function that is a surrogate for nativeness (or functional relevance) [8].

Surrogate functions range from carefully-crafted pseudo-energy functions that combine physics-based and statistical terms [9]–[13] to functions learned from data via machine learning algorithms, such as Support Vector Machines [14], [15], Random Forest [16], Ensemble Learning [17], and even Neural Networks [18], [19]. These methods utilize features derived from statistical scoring functions [20] and/or expert-constructed structures [21].

While ML-based methods show great promise, supervised learning ones have to address several challenges, as shown in recent [22]. Computationally-generated decoys come with no labels (native or non-native); associating labels for the purpose of training relies on metrics and thresholds that introduce some arbitrariness and consequently imprecision in the process. More generally, decoy datasets are severely imbalanced, generating many decoys is a computationally-intensive process, and ML methods can additionally be prone to capturing spurious correlations when faced with many features needed to represent tertiary structures.

It is worth noting that, while 'model assessment' is often interchangeably used in literature with 'decoy selection,' the two problems are not equivalent. The latter refers to the setting, where the goal is to automatically select from a given set of tertiary structures/decoys those that are near-native (also referred to as high-quality). While model assessment may be the first step to then selecting decoys determined to be of high quality by the assessment, using model assessment is not necessary. For instance, decoy selection methods that remain popular are those based on clustering.

The success of clustering-based methods is intrinsically tied to the quality of a given decoy dataset. The majority of datasets are severely imbalanced; a small percentage of decoys are near-native. By seeking consensus, clustering-based methods cannot identify near-native decoys if they are severely under-represented. The interested reader can find several other issues that challenge clustering-based methods listed in [23], [24].

Several methods have advanced clustering-based methods by leveraging the energy surface (the structure space lifted by the additional dimension of internal energy) populated by a set of computationally-generated decoys. For instance, work in [25] utilizes components of the energy surface, known as energy basins, in lieu of clusters. After first grouping decoys into basins, the work in [25] ranks basins based on several characteristics and demonstrates that top basins contain more near-native/higher-quality decoys than top clusters [25].

B. Contributions of Proposed Methods

In this paper, we focus on the problem of decoy selection and address it in two forms, one where the goal is to select a subset of decoys, and the other where the goal is to be more precise and select a single decoy to offer as prediction of a near-native structure. Moreover, we specifically focus on the setting where the decoys are generated for a given protein molecule (from its amino-acid sequence) via template-free PSP methods. Because of the framework under which these methods operate, the issue of data imbalance, which is common in bioinformatics [26], is prominent and informs our premise in this paper. To further substantiate this point, we note that on a dataset of $54,795$ decoys that we employ in our evaluations in this paper, only 0.845% are near-native.

We posit that anomaly detection (AD) is an intuitive and natural approach in the presence of such data imbalance. AD is the task of discovering which observations are outliers (abnormal). First and most important, abnormal observations do not have to correspond to a single class. Most likely, they are captured by an unbound (yet to be known) number of classes. Supervised methods will overfit by forcing the learning of models that explain classes we already know and for which we already have training examples. AD, on the other hand, only requires that we have examples of normal instances. Second, as in most AD applications, outliers are rare and therefore a minority of observations. This kind of imbalance is difficult to address with supervised learning, since the data needs to be balanced first if one does not want the majority class to influence results negatively. Balancing leads to oversampling the minority observations which leads to overfitting, or undersampling the majority observations, which leads to loss of a great amount of data.

AD amounts to finding points that do not fit the distribution of normal data. This is accomplished by one of the following approaches. (i) Assuming a particular form of this distribution (e.g., Gaussian) and fitting the normal data to it, testing the rest of the data for fitness over the fitted distribution (e.g., [27]). (ii) Finding a way to measure the outlierness of points with respect to a baseline of normal points and ranking observations using this measure (e.g., [28]). (iii) Using a measure of outlierness and a test of fitness over an empirical distribution of normal points to determine if an observation is an outlier (e.g., [29]).

In this paper we adapt and evaluate a comprehensive and diverse list of AD techniques for decoy evaluation in PSP. We build on recent work in [25] on detection of energy basins and first identify "anomalous" basins. We show that AD allows identifying better-quality basins for decoy selection. Second, we further provide methods that directly select a decoy from the top, anomalous basins and offer as prediction. Evaluations on benchmark datasets along rigorous machine learning metrics demonstrate that the proposed methods advance the state of the art and warrant further research on adapting concepts and techniques from machine learning to improve decoy selection in template-free protein structure prediction.

II. METHODS

It may be tempting to directly apply AD techniques to a decoy dataset, effectively evaluating whether a decoy is anomalous or not. Such evaluation would constitute model

assessment, but it would require that decoys be represented via meaningful features/descriptors. While this is certainly a reasonable direction of research, our preliminary work in this direction suggests that finding informative descriptors is not trivial and relies greatly on domain-specific insight. Moreover, aiming to obtain such descriptors automatically relies on deep NNs that place demands on the size of decoy datasets and in turn place large computational demands on template-free PSP methods used to generate such datasets.

Therefore, in this paper we propose to apply AD techniques over subsets of decoys (effectively addressing decoy selection), identifying anomalous subsets and offering them as prediction. We build over recent works by our laboratory on organizing decoys into basins, which are better-quality clusters, as we summarize below. We describe novel methods that utilize basins to identify among them one or a few containing good-quality decoys. Finally, we additionally propose methods that select an individual decoy from a basin and offer it as prediction.

A. *Basins+Select*

`Basins+Select`, recently proposed and demonstrated in [25], is employed as a baseline against which we evaluate performance improvements obtained by novel, AD-based methods proposed here to select basins. Work in [25] shows that `Basins+Select` outperforms clustering-based methods that do not take into account energetics. The method relies on two concepts, a nearest-neighbor graph and energy basins, which we summarize briefly below.

The nearest-neighbor graph (nngraph) encodes the spatial proximity of decoys in the structure space probed by a template-free PSP method. The decoys constitute the vertices of the nngraph. The edges encode a local (spatial proximity) neighborhood over each decoy. The distance between two decoys is measured via Root-Mean-Squared-Deviation (RMSD), which averages the Euclidean distance among atoms in two given decoys over the atoms.

To remove differences due to rigid translation and rotation in three-dimensional Cartesian space, all decoys in a dataset are first optimally superimposed over the first decoy [30], which is chosen arbitrarily as reference. Using RMSD to compute the distance between two decoys, each vertex/decoy u is connected to other vertices v if $d(u, v) \leq \epsilon$; ϵ is a user-defined parameter. A small ϵ may result in a disconnected nngraph. This is remedied in [25] by gradually increasing the value of ϵ over a maximum number of iterations, while controlling the density of the resulting nngraph via a maximum number of nearest neighbors per vertex.

On-graph clustering algorithms leveraging the concepts of communities in social and information networks have been proposed and adapted to cluster decoys over the nngraph [31]. These algorithms result in decoy groupings that are outperformed by `Basins+Select`, which additionally takes into consideration decoy energies.

`Basins+Select` utilizes the concept of basins in the energy surface (often referred to as the energy landscape). The landscape lifts the decoy space by one additional dimension that corresponds to the internal/potential energy in a decoy. Basins "of attraction" in the landscape are local neighborhoods around local minima; the latter are referred to as focal minima.

Using the Structural Bioinformatics Library (SBL) [32], `Basins+Select` decomposes a decoy nngraph into basins. These are identified as follows. Vertices that are local (energy) minima in the nngraph are identified first. A vertex is considered a local minimum if its energy is no higher than the energies of other vertices to which it is connected via an edge. Each such identified local minimum vertex represents a basin.

Remaining vertices are assigned to basins as follows. Each vertex u is associated a negative gradient estimated by selecting the edge (u, v) that maximizes the ratio $[e(u) - e(v)]/d(u, v)$, where $e(u)$ is the energy of the decoy in vertex u. The negative gradient is followed (via the edge that maximizes the above ratio) until a local minimum is reached. Vertices that via this process reach the same local minimum are assigned to the basin associated/identified with that minimum.

Once basins are identified, they can be ranked/ordered by various characteristics based on basin size and basin energy, as described in [25]. Basin size measures the number of decoys in a basin. Basin energy can be implemented as the energy of the focal minimum in it (so, the lowest energy among the decoys in it) or the average energy over the energies of the decoys in a basin. In the interest of clarity of evaluation, in this paper we focus only on size-based ranking, ordering identified basins from largest to smallest, and selecting the top one(s) in this ranking as the decoy subset of potentially good quality. This subset is evaluated with metrics designed for the setting of decoy selection, as related later in Section II-E.

B. *Basin-based Methods for Selection of Decoy Subsets*

In this paper, we employ `Basins+Select` as a baseline method against which we evaluate three novel methods that select subsets of potentially good-quality (near-native) decoys from a decoy dataset. As described in Section I, our focus is on evaluating AD as a means of handling data imbalance in decoy datasets; i.e., a very small percentage of near-native decoys. We do so in a method to which we refer as `Basins+AD+Select`. The naming conveys the order of steps. Decoys are first grouped into basins via SBL as in [25]. The identified basins are then evaluated via various AD techniques to identify "anomalous" basins. Details of various AD techniques we employ and evaluate are related in Section II-C.

Once anomalous basins are identified, they are ranked by size, from largest to smallest. The largest l basins are offered as the decoy subset of potentially good quality. When l is specified to be larger than 1, the decoys in the l selected basins are merged into one decoy subset offered as prediction.

The proposed `Basins+AD+Select` method is additionally compared against two other novel methods proposed here, `CSSample+Basins+Select` and `CSSample+Basins+AD+Select`. Both methods evaluate

the hypothesis that additional information about the decoys, beyond their internal energies, may be leveraged to improve the quality of the selected decoy subset(s). The *CSSample* component refers to the fact that the decoy dataset is reduced prior to identification of basins. The reduction takes into account a contact-based score that evaluates the quality of a decoy in a blind setting (absence of a known native structure).

Given the amino-acid sequence alone, contacts can be predicted with various methods. We make use of RaptorX-Contact due to its good performance in CASP [33]. RaptorX-Contact weights predicted contacts by a confidence score. In this paper, we select the top 10 contacts. These are compared against contacts evaluated in a decoy; contacts are computed between CB atoms, using a distance threshold of 8Å (CA is used in glycine in lieu of a CB atom). Recall/sensitivity is used to compare contacts in a decoy to sequence-predicted contacts, using the latter as the ground truth. Sensitivity is used as a contact score for each decoy; since it ranges in $[0, 1]$, it is employed as a probability to sample decoys with high contact scores. Half of the decoys in a dataset are sampled in this manner.

Basins are then identified over the sampled/reduced dataset. In `CSSample+Basins+Select`, the largest basin is offered as prediction. In `CSSample+Basins+AD+Select`, the identified basins (over the reduced dataset) are fed to AD techniques to identify anomalous basins. The latter are ranked according to size, and the largest l are offered as prediction.

C. Anomaly Detection Techniques

We investigate a comprehensive list of 10 state-of-the-art AD techniques, applying them over detected basins. Each basin is represented with four features: size (the number of decoys in a basin), the average energy calculated over energies of decoys in a basin, Pareto Rank, and Pareto Count. These latter two features are inspired by work on multi-objective optimization and selection [25], [34], where the concept of dominance across several possibly conflicting objectives, such as basin size and energy here, is used to rank solutions (in this case, basins). Once such represented, basins are subjected to any of the following AD techniques.

In "Angle-Based Outlier Detection" (ABOD), the anomalous score of a data point is observed as the variance of its weighted cosine scores to all neighbors [35]. If a data point is an outlier, the variance of angles between the pairs of other data points is small. The main advantage of this method is independence of parameters.

In "Clustering-Based Local Outlier Factor" (CBLOF), data are classified into different clusters based on clustering [35]. The outlier score of a data point is calculated based on the size of the cluster to which a data point belongs and the distance to the nearest large cluster.

In "Feature Bagging," multiple AD methods are base detectors that are applied on different set of features on various subsamples of a dataset; the combination of the results generated is used for detecting outliers [35].

The "Histogram-Based Outlier Score (HBOS) technique is based on the assumption of independence of features and detects outliers in linear time on data size. In HBOS, for each feature (dimension), a univariate histogram is constructed with which the degree of outlierness is estimated [35].

In "k-Nearest Neighbors Detector" (kNN), the kth-nearest neighbor distance of a data point is considered its anomaly score. Two settings are considered, one in which the average of all the k-neighbors distance is the score, and another one in which the median is considered instead [35]. Different distance metrics can be used, such as cityblock, cosine, Euclidean, L1, L2, Manhattan, etc.

In "Local Outlier Factor" (LOF), local densities of particular regions are computed, and instances in low-density regions are potential anomalies [35]. LOF measures the local deviation of density of a data point in comparison to its neighbors. The LOF is computed as the ratio of average local reachability density of a data point's k-nearest neighbors and local reachability density of the data point. This LOF is the scoring criterion for outlier detection.

"Outlier detection with Minimum Covariance Determinant" (MCD) detects outliers in a Gaussian-distributed data. A minimum covariance determinant model is fit on the data. The Mahalanobis distance is then used as an estimate of the outlier degree of the data [35].

"OneClass-SVM" (OCSVM) gives useful results on high-dimensional datasets with unknown distribution with proper hyper-parameter tuning. The ν parameter is set to the proportion of outliers expected to be present in the data, and the γ parameter determines the smoothing of the contour lines [35].

Finally, the "Principal Component Analysis (PCA) for AD" technique relies on PCA to project data onto a lower-dimensional hyperplane. Data that are outliers with respect to the top few principal components (with largest eigenvalues) correspond to outliers on one or more of the original variables. Outlier scores are estimated as the sum of the projected distance of a data point on all eigenvectors [35].

D. Single-Decoy Selection Methods

Given a subset of decoys selected by a method described above, we then propose and evaluate four novel single-decoy selection methods that select a single decoy from a decoy subset. `Random-DecoySelect` is employed as a baseline, as it samples a decoy uniformly at random over a decoy subset.

`CS+E-DecoySelect` sorts decoys first by contact score (highest to lowest); decoys with the same contact score are then sorted by energy (lowest to highest). The top decoy in this ranking is offered as prediction. This ranking yields better results than an alternative one, where decoys are first ordered by energy and then by contact score (data not shown).

The third method, `AD-Random-DecoySelect`, evaluates the decoys in a subset and removes those that are deemed anomalous; in our evaluation, we focus only on 1-class SVM. Note that removing anomalous decoys is in contrast to the strategy employed in selecting a decoy subset, where anomalous basins were determined to contain potentially

good-quality decoys. Since the single-decoy selection methods operate over one or few (merged) basins, the decoys should be structurally- and energetically-similar. So, the objective changes into looking for consensus. After removing anomalous decoys, uniform random sampling is employed over the remaining ones to extract a decoy and offer it as prediction.

Alternatively, in `AD-CS+E-DecoySelect`, after removing anomalous decoys identified via 1-class SVM, the remaining decoys are ranked by their contact score first and energy second, and the highest-ranking decoy is extracted and offered as prediction. Again, this ranking yields better results than an alternative one, where decoys are first ordered by energy and then by contact score. We note that in `AD-Random-DecoySelect` and `AD-CS+E-DecoySelect` decoys are represented via three features, energy, contact score, and RMSD from the decoy with the lowest energy in the considered subset.

E. Evaluation Metrics

The methods we describe above fall in one of two categories: they select a subset of decoys or select an individual decoy from a set. In the former, such methods can be evaluated in terms of *purity*, a metric we originally introduced in [25]. In the latter, methods can be evaluated via *loss*, a classic ML metric that we adapt here.

Methods of the first category organize decoys into groups/basins. Groups can be ranked/ordered based on characteristics that can be measured over a group. For instance, one such characteristic can be size. Ordering by largest to smallest can provide groups $G_1, \ldots, G_n$, with n being the total number of identified groups. Such a method that first organizes decoys into groups and then ranks them can be used for decoy selection as follows: Provided a user-specified parameter l, the groups $G_1, \ldots, G_l$ in the ranking $G_1, \ldots, G_n$ can be selected, merged together in a set S, and the decoys in S can be offered as prediction of near-native structures.

The set S can be evaluated in terms of its *purity*; that is, how many near-native decoys are actually contained in it. On a test case, where the native structure is known, all given decoys (generated by a decoy generation algorithm) can be evaluated in terms of their dissimilarity from the native structure. We employ least Root-Mean-Squared-Deviation (lRMSD), which averages the Euclidean distance among atoms in two given structures over the atoms after removing differences due to rigid translation and rotation in 3d [30]. Provided a distance threshold dist_thresh, all decoys below the threshold are labeled as near-native; the rest are labeled as non-native. The former are positives, and the latter are negatives. So, a selected set S of decoys (consisting of decoys in groups $G_1, \ldots, G_l$) can be evaluated in terms of its purity $TP(S)/|S|$, where $TP(S)$ is the number of near-native decoys (true positives) in S. It is evident that purity is related to precision.

Methods in the second category select one decoy. We note that one can easily put together a pipeline that follows up a method from the first category with a method from the second category. For instance, after selecting first a subset S of decoys from a given dataset, uniform random sampling can be employed to select any decoy from S and offer for prediction. We propose *loss* to evaluate how good a selected decoy is. The decoy that is closest to the native structure (in terms of lRMSD) has a loss of 0. A perfect method would always find such a decoy. Let us refer to this decoy as *BestDecoy*. In the absence of such a method, any other selected decoy *SelectedDecoy* presents a loss measured as *RSMD(SelectedDecoy, NativeStructure) - RSMD(BestDecoy, NativeStructure)*.

F. Implementation Details

The Python library pyod.models is used to implement the AD methods listed above. The fraction of outliers employed in these methods is 0.05, the number of nearest neighbors of a decoy (in AD methods that rely on this parameter) varies in $[15, 40]$, and the distance function used is the Euclidean. Results reported for proposed methods relying on sampling are averages over 50 independent drawings.

III. Results

We experiment with 18 proteins of different lengths and folds. These proteins constitute a benchmark dataset often used by decoy generation algorithms [34], [36]. We used the Rosetta *template-free* (decoy generation) protocol [37] to generate around $51,000$ to $68,000$ decoys per target. For each Rosetta-generated decoy, we have its all-atom Cartesian coordinates and its all-atom internal energy (score12) measured in Rosetta Energy Units (REUs).

Table I presents all the 18 proteins arranged into 3 different categories/levels of difficulty (easy, medium, and hard). These levels have been determined using the minimum lRMSD between the generated decoys and a known native structure of the corresponding protein (obtained from the PDB). The size of the decoy ensemble $|\Omega|$ for each target is shown in Column 5. The proteins in this dataset are identified via the PDB entry id of a known native structure for them. The 4-letter PDB ids are shown in Column 2; the fifth letter identifies the chain in a multi-chain PDB entry. Column 7, which shows the percentage of near-native decoys (within dist_threshold of the known native structure), conveys the extreme imbalance of the decoy datasets; in some cases, the near-native decoys constitute less than 5% of the dataset.

A. Comparative Evaluation of Basin Purity

Figure 1 shows the purity of the decoy subset consisting of the largest basin(s) obtained by the four methods under comparison, `Basins+Select`, `Basins+AD+Select`, `CsSample+Basins+Select`, and `CsSample+Basins+AD+Select`. The top panel evaluates the largest basin, the middle panel the decoy subset that merges the largest two basins, and the bottom one the largest three basins. It is worth noting that the AD techniques that confer the best performance to the AD-employing basin selection methods vary based on the decoy datasets; consistently, however, the top-performing

TABLE I

TESTING DATASET (* DENOTES PROTEINS WITH A PREDOMINANT β FOLD AND A SHORT HELIX). THE CHAIN EXTRACTED FROM A MULTI-CHAIN PDB ENTRY (SHOWN IN COLUMN 2) TO BE USED AS THE NATIVE STRUCTURE IS SHOWN IN PARENTHESES. THE FOLD OF THE KNOWN NATIVE STRUCTURE IS SHOWN IN COLUMN 3. THE LENGTH OF THE PROTEIN SEQUENCE (#AAS) IS SHOWN IN COLUMN 4. THE SIZE OF THE ROSETTA-GENERATED DECOY DATASET IS SHOWN IN COLUMN 5. COLUMN 6 SHOWS THE MINIMUM LRMSD OVER DECOYS FROM THE KNOWN NATIVE STRUCTURE. COLUMN 7 SHOWS THE PERCENTAGE OF NEAR-NATIVE DECOYS (WITHIN DIST_THRESHOLD OF THE KNOWN NATIVE STRUCTURE).

difficulty	PDB id	fold	# aas	# decoys	min lRMSD (Å)	% near-native
Easy	1ail	α	70	58,491	0.50	6.352
	1dtd(B)	$\alpha+\beta$	61	58,745	0.51	22.827
	1wap(A)	β	68	68,000	0.60	10.192
	1tig	$\alpha+\beta$	88	60,000	0.60	15.109
	1dtj(A)	$\alpha+\beta$	74	60,500	0.68	22.435
Medium	1hz6(A)	$\alpha+\beta$	64	60,000	0.72	11.325
	1c8c(A)	β*	64	65,000	1.08	10.882
	2ci2	$\alpha+\beta$	65	60,000	1.21	22.443
	1bq9	β	53	61,000	1.30	1.565
	1hhp	β*	99	60,000	1.52	2.486
	1fwp	$\alpha+\beta$	69	51,724	1.56	5.819
	1sap	β	66	66,000	1.75	2.304
Hard	2h5n(D)	α	123	54,795	2.00	0.845
	2ezk	α	93	54,626	2.56	13.047
	1aoy	α	78	57,000	3.26	10.923
	1cc5	α	83	55,000	3.95	5.529
	1isu(A)	*coil*	62	60,000	5.53	5.304
	1aly	β	146	53,000	8.53	2.779

ones are PCA for AD, Feature Bagging, and OCSVM, followed next by CBLOF and HBOS (data not shown). The results shown in Figure 1 are those obtained with the best AD technique (in `Baseline+AD+Select` and `CsSample+Basins+AD+Select`).

Figure 1 allows making several interesting observations. First, all methods are able to extract a pure subset of decoys on the easy decoy datasets, where the percentage of near-native decoys is high. This is not surprising, as all methods build over basins, which prior work shows outperform clustering for decoy selection [25]. Two methods show larger variation in performance, `Basins+Select`, the baseline method (lower purity on 1ail) and `CsSample+Basins+AD+Select` (lower purity on 1dtd(B)). Differences in performance become more pronounced over the medium and hard datasets, where the top performing method results in higher purity on more of the datasets. Specifically, on the medium-difficulty datasets, `Basins+AD+Select` outperforms other methods on $6/7$ of the datasets on the evaluation of the largest basin, on $4/7$ of the datasets on the evaluation of the decoy subset consisting of the two largest basins, and on $3/7$ of the datasets on the evaluation of the decoy subset consisting of the three largest basins. On the harder datasets, `Basins+AD+Select` outperforms other methods on $4/6$ of the datasets on the evaluation of the largest basin, on $3/6$ of the datasets on the evaluation of the decoy subset consisting of the two largest basins, and on $4/6$ of the datasets on the evaluation of the decoy subset consisting of the three largest basins. Filtering first by predicted contact scores as in `CsSample+Basins+AD+Select` helps when the decoy subset consists of more basins; `CsSample+Basins+AD+Select` is the best-performing method on $4/7$ of the medium-difficulty decoy datasets when focusing on the three largest basins. Altogether, these results allow concluding that `Basins+AD+Select` is the better-performing method, resulting in higher-purity decoy subsets.

B. Visualization of Largest, Selected Basin

Figure 2 shows the largest basin selected by `Basins+Select` and the best method (as shown by the above evaluation), `Basins+AD+Select` on 6 representative datasets. The decoys in each dataset are plotted by their lRMSD from the native structure versus their Rosetta energy. Decoys in the largest basin obtained by `Basins+Select` are drawn in red, and those in the largest basin obtained by `Basins+AD+Select` are in yellow. Figure 2 clearly shows how `Basins+AD+Select` improves purity of the largest basins over `Basins+Select`. The largest basin obtained by `Basins+AD+Select` is more homogeneous (in terms of lRMSDs), and decoys in it have lower lRMSDs from the native structure. These characteristics together are desirable, as they may help in selecting a better decoy from the basin.

C. Comparative Evaluation of Loss

We show the loss resulting from the four methods that select an individual decoy from a decoy subset extracted from a given decoy datasets. We focus here on the subset consisting of the largest basin selected by `Basins+AD+Select`. Figure 3 shows the loss for `Random-DecoySelect`, `CS+E-DecoySelect`, `AD-RandomDecoySelect`, and `AD-CS+E-DecoySelect`. The top panel shows the loss over the entire decoy dataset (where the best/lowest-lRMSD decoy is the one over the entire dataset), whereas the bottom panel shows the loss over the basin from which these methods draw a single decoy (where the best decoy is computed over those in the basin only). We recall that the loss reported for methods that make use of uniform random sampling, shown results are averages over 50 independent drawings.

The results in Figure 3 show that, as expected, `Random-DecoySelect`, is the worst-performing method (largest loss). `CS+E-DecoySelect` and `AD-CS+E-DecoySelect` are the two top performing methods. On some of the datasets, these methods yield 0 loss with respect to loss over the top basin (with PDB ids 1dtj(A), 1fwp, 2h5n(D), and 1aly); on 2ezk, the largest basin has only 1 decoy, so all methods yield 0 loss. Moreover, `CS+E-DecoySelect` outperforms `AD-CS+E-DecoySelect` on $3/18$ of the datasets with respect to loss over the entire dataset and on $3/18$ of the datasets with respect to loss over the top basin. These two methods achieve identical loss over $14/18$ of the datasets with respect to loss over the entire dataset and loss over the top basin; this indicates that the decoy with the best contact score in the top basin is not among the anomalous ones removed on $14/18$ datasets. `AD-CS+E-DecoySelect` outperforms

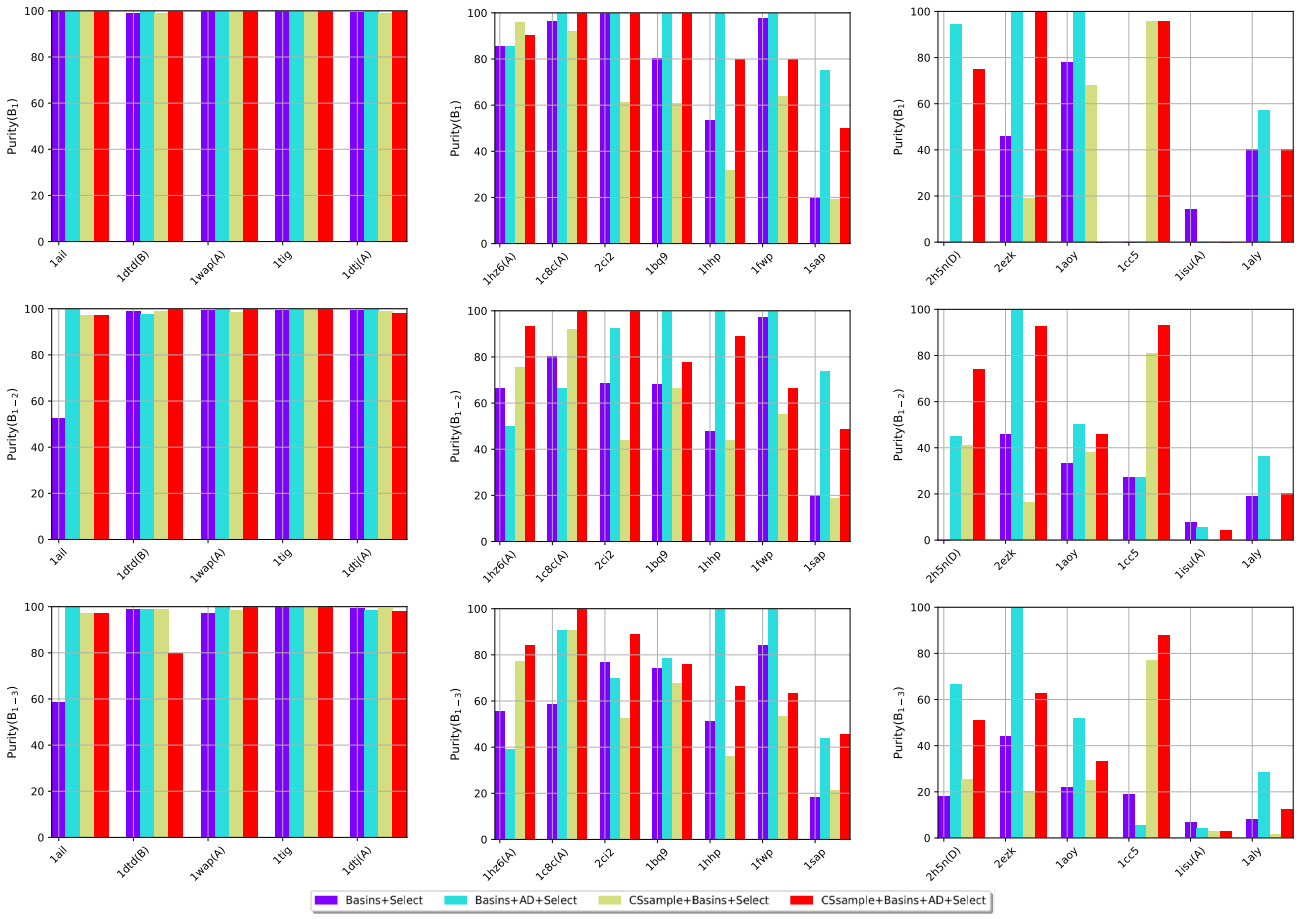

Fig. 1. Comparison of purity measured over decoys in B_1 (top row), the subset B_{1-2} (middle row), and the subset B_{1-3} (bottom row) across the four methodologies. Results obtained over the easy, medium, and hard datasets are shown separately in left, middle, and right columns, respectively, to highlight the performance comparisons depending on the dataset difficulty.

`CS+E-DecoySelect` on $1/18$ of the datasets with respect to loss over the entire dataset and loss over the top basin.

IV. Conclusion

We proposed novel methods for recognition of near-native protein structures as a decoy selection problem. We describe a two-stage methodology that first selects a subset of decoys from a given decoy dataset and then extracts a single decoy from the subset for prediction in template-free PSP. The concept of anomaly detection is employed in both stages, first to identify outlier energy basins in which decoys are grouped together and then to remove outlier decoys in order to homogenize the quality of decoys. Rigorous evaluation along metrics, such as recall and loss, shows that anomaly detection allows identifying better-quality basins and decoys.

The proposed methods show promise for decoy selection and warrant further research on anomaly-based recognition of near-native protein structures in imbalanced datasets. While our focus in this paper has been on the decoy selection problem, the proposed concepts and methods can be leveraged for model assessment. While not directly the focus of our investigation in this paper, the outlierness of a decoy can be utilized to assess its quality. Moreover, anomaly-based scores can be combined with other characteristics/features to approach model selection via supervised learning. These and other related avenues will be the focus of our future research.

Acknowledgment

Computations were run on ARGO, a research computing cluster provided by the Office of Research Computing at George Mason University.

References

[1] T. Maximova, R. Moffatt, B. Ma, R. Nussinov, and A. Shehu, "Principles and overview of sampling methods for modeling macromolecular structure and dynamics," *PLoS Comp. Biol.*, vol. 12, no. 4, p. e1004619, 2016.

[2] D. D. Boehr and P. E. Wright, "How do proteins interact?" *science*, vol. 320, no. 5882, pp. 1429–1430, 2008.

[3] H. M. Berman, K. Henrick, and H. Nakamura, "Announcing the worldwide Protein Data Bank," *Nature Structural Biology*, vol. 10, no. 12, pp. 980–980, 2003.

[4] A. Kryshtafovych, B. Monastyrskyy, K. Fidelis, T. Schwede, and A. Tramontano, "Assessment of model accuracy estimations in casp12," *Proteins: Struct, Funct, and Bioinf*, vol. 86, pp. 345–360, 2018.

[5] J. Lee, P. Freddolino, and Y. Zhang, "Ab initio protein structure prediction," in *From Protein Structure to Function with Bioinformatics*, 2nd ed., D. J. Rigden, Ed. Springer London, 2017, ch. 1, pp. 3–35.

[6] R. Das, "Four small puzzles that rosetta doesn't solve," *PLoS ONE*, vol. 6, no. 5, p. e20044, 2011.

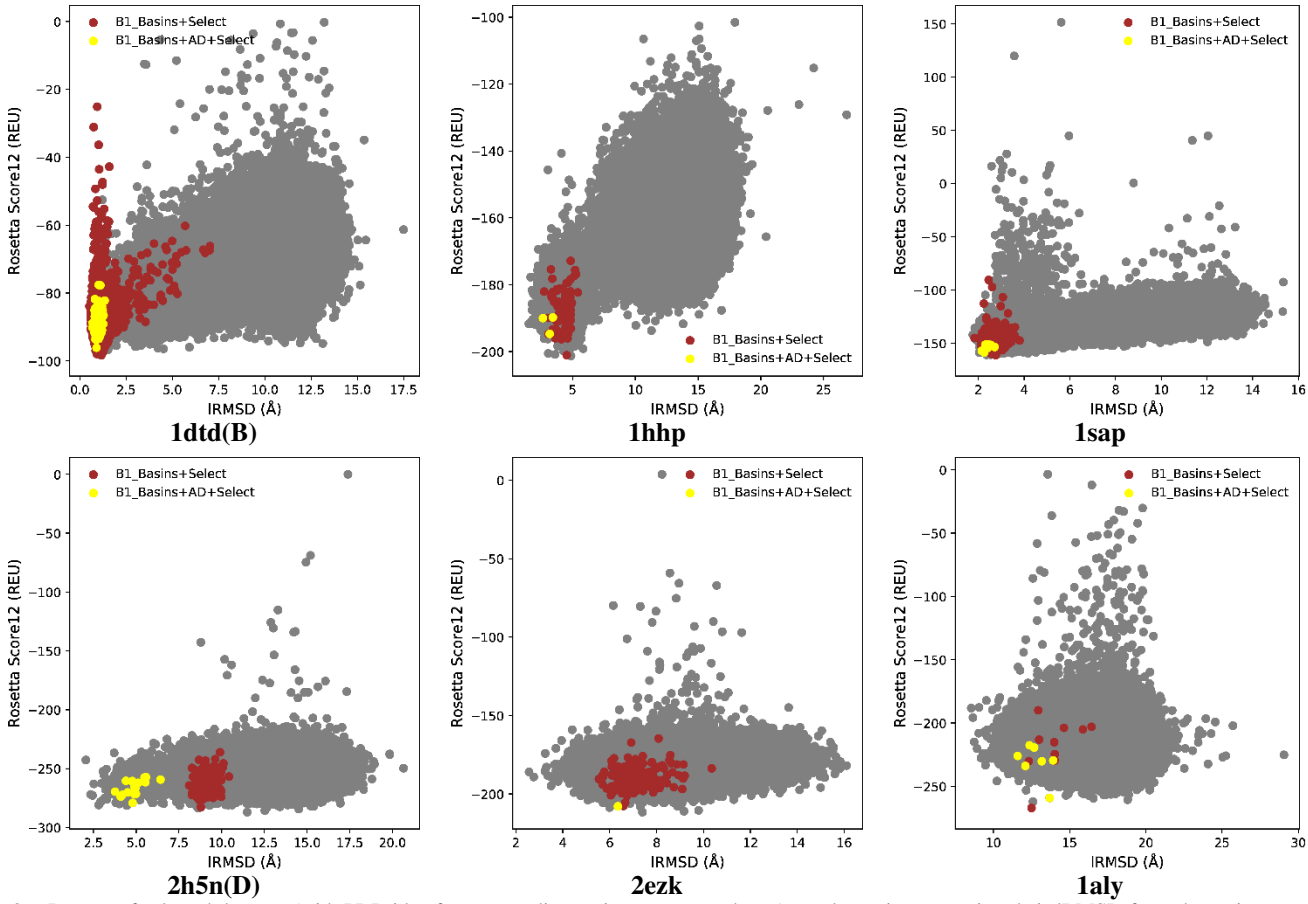

Fig. 2. Decoys of selected datasets (with PDB ids of corresponding native structures shown) are drawn in gray, using their lRMSD from the native structure and their Rosetta all-atom energy (score12, in Rosetta Energy Units – REU) as coordinates. The decoys in the largest basin obtained by `Basins+Select` are shown in red, and those in the largest basin obtained by `Basins+AD+Select` are shown in yellow.

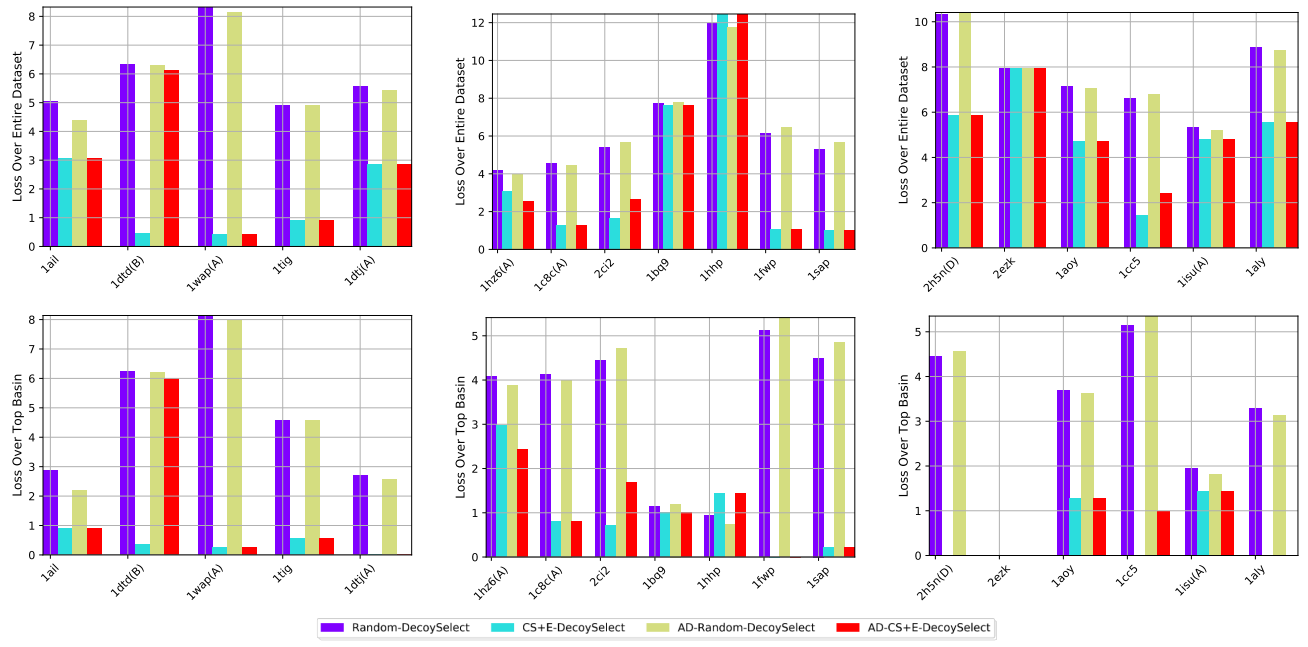

Fig. 3. The top panel evaluates various single-decoy selection methods on loss measured with respect to the best decoy over the entire decoy dataset. The bottom panel shows loss measured with respect to the best decoy in the largest basin. The left, middle, and right panels show performance over the easy, medium, and hard datasets, respectively.

[7] K. Molloy, S. Saleh, and A. Shehu, "Probabilistic search and energy guidance for biased decoy sampling in ab-initio protein structure prediction," *IEEE/ACM Trans Comput Biol and Bioinf*, vol. 10, no. 5, pp. 1162–1175, 2013.

[8] K. Uziela and B. Wallner, "Proq2: estimation of model accuracy implemented in rosetta," *Bioinformatics*, vol. 32, no. 9, pp. 1411–1413, 2016.

[9] S. Miyazawa and R. L. Jernigan, "An empirical energy potential with a reference state for protein fold and sequence recognition," *Proteins: Struct, Funct, and Bioinf*, vol. 36, no. 3, pp. 357–369, 1999.

[10] B. J. McConkey, V. Sobolev, and M. Edelman, "Discrimination of native protein structures using atom–atom contact scoring," *Proc Natl Acad Sci USA*, vol. 100, no. 6, pp. 3215–3220, 2003.

[11] K. T. Simons, I. Ruczinski, C. Kooperberg, B. A. Fox, C. Bystroff, and D. Baker, "Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins," *Proteins: Struct, Funct, and Bioinf*, vol. 34, no. 1, pp. 82–95, 1999.

[12] B. Park and M. Levitt, "Energy functions that discriminate X-ray and near-native folds from well-constructed decoys," *J Mol Biol*, vol. 258, no. 2, pp. 367–392, 1996.

[13] A. K. Felts, E. Gallicchio, A. Wallqvist, and R. M. Levy, "Distinguishing native conformations of proteins from decoys with an effective free energy estimator based on the opls all-atom force field and the surface generalized born solvent model," *Proteins: Struct, Funct, and Bioinf*, vol. 48, no. 2, pp. 404–422, 2002.

[14] S. Chatterjee, S. Ghosh, and S. Vishveshwara, "Network properties of decoys and CASP predicted models: a comparison with native protein structures," *Molecular BioSystems*, vol. 9, no. 7, pp. 1774–1788, 2013.

[15] B. Manavalan and J. Lee, "SVMQA: support–vector-machine-based protein single-model quality assessment," *Bioinformatics*, vol. 33, no. 16, pp. 2496–2503, 2017.

[16] B. Manavalan, J. Lee, and J. Lee, "Random forest-based protein model quality assessment (RFMQA) using structural features and potential energy terms," *PloS one*, vol. 9, no. 9, p. e106542, 2014.

[17] S. Mirzaei, T. Sidi, C. Keasar, and S. Crivelli, "Purely structural protein scoring functions using support vector machine and ensemble learning," *IEEE/ACM Trans Comp Biol & Bioinf*, 2016.

[18] S. P. Nguyen, Y. Shang, and D. Xu, "DL-PRO: A novel deep learning method for protein model quality assessment," in *Int Conf Neural Networks (IJCNN)*. IEEE, 2014, pp. 2071–2078.

[19] R. Cao, D. Bhattacharya, J. Hou, and J. Cheng, "DeepQA: improving the estimation of single protein model quality with deep belief networks," *BMC Bioinf*, vol. 17, no. 1, p. 495, 2016.

[20] Z. He, Y. Shang, D. Xu, Y. Xu, and J. Zhang, "Protein structural model selection based on protein-dependent scoring function," *Statistics & Interface*, vol. 5, no. 1, pp. 109–115, 2012.

[21] A. Ray, E. Lindahl, and B. Wallner, "Improved model quality assessment using proq2," *BMC Bioinf*, vol. 13, no. 1, p. 224, 2012.

[22] N. Akhter, G. Chennupati, H. Djidjev, and A. Shehu, "Decoy selection for protein structure prediction via extreme gradient boosting and ranking," *BMC Bioinformatics*, 2020, in press.

[23] S. C. Li and Y. K. Ng, "Calibur: a tool for clustering large numbers of protein decoys," *BMC Bioinf*, vol. 11, no. 1, p. 25, 2010.

[24] F. Berenger, Y. Zhou, R. Shrestha, and K. Y. Zhang, "Entropy-accelerated exact clustering of protein decoys," *Bioinformatics*, vol. 27, no. 7, pp. 939–945, 2011.

[25] N. Akhter and A. Shehu, "From extraction of local structures of protein energy landscapes to improved decoy selection in template-free protein structure prediction," *Molecules*, vol. 23, no. 1, p. 216, 2018.

[26] X. M. Zhao, X. Li, L. Chen, and K. Aihara, "Protein classification with imbalanced data," *Proteins: Struct, Funct, and Bioinf*, vol. 70, no. 4, pp. 1125–32, 2008.

[27] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, ser. Wiley Series in Probability and Statistics. John Wiley & Sons, 2003.

[28] M. M. Breunig, H. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," in *Intl Conf on Management of Data (SIGMOD)*. ACM, 2000, pp. 97–104.

[29] D. Barbará, C. Domeniconi, and J. P. Rogers, "Detecting outliers using transduction and statistical testing," in *Proc. of Intl Conf on Knowledge Discovery and Data Mining*, ser. SIGKDD. ACM, 2006, pp. 20–23.

[30] A. D. McLachlan, "A mathematical procedure for superimposing atomic coordinates of proteins," *Acta Cryst A*, vol. 26, no. 6, pp. 656–657, 1972.

[31] L. K. Kabir, L. Hassan, Z. Rajabi, and A. Shehu, "Graph-based community detection for decoy selection in template-free protein structure prediction," *Molecules*, vol. 24, no. 3, p. 741, 2019.

[32] F. Cazals and T. Dreyfus, "The structural bioinformatics library: modeling in biomolecular science and beyond," *Bioinformatics*, vol. 33, no. 7, pp. 997–1004, 2017.

[33] S. Wang, S. Sun, Z. Li, R. Zhang, and J. Xu, "Accurate de novo prediction of protein contact map by ultra-deep learning model," *PLOS Computational Biology*, vol. 13, no. 1, pp. 1–34, 01 2017.

[34] B. Olson and A. Shehu, "Multi-objective stochastic search for sampling local minima in the protein energy surface," in *ACM Conf on Bioinf and Comp Biol (BCB)*, Washington, D. C., September 2013, pp. 430–439.

[35] Y. Zhao, Z. Nasrullah, and Z. Li, "Pyod: A python toolbox for scalable outlier detection," *Journal of Machine Learning Research*, vol. 20, no. 96, pp. 1–7, 2019.

[36] G. J. Zhang, G. Zhou, X, X. F. Yu, H. Hao, and L. Yu, "Enhancing protein conformational space sampling using distance profile-guided differential evolution," *IEEE/ACM Trans Comput Biol and Bioinf*, vol. 14, no. 6, pp. 1288–1301, 2017.

[37] A. Leaver-Fay *et al.*, "ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules," *Methods Enzymol*, vol. 487, pp. 545–574, 2011.

Sivani Tadepalli is a PhD student in the Department of Computer Science at George Mason University. She received her MS in Computer Science from George Mason University in 2018. Her research is in Machine Learning and Data Mining as driven by problems in bioinformatics.

Nasrin Akhter is a PhD candidate in Computer Science at George Mason University, Fairfax, VA, USA. She obtained her MS in Computer Science and Engineering at University of Dhaka, Bangladesh. Akhter's research focuses on Machine Learning and Data mining with applications in bioinformatics and computational biology.

Daniel Barbará has taught at George Mason University since 1997. His areas of expertise are data mining and machine learning. He served as the program chair of the SIAM International Conference on Data Mining in 2003, and he has received numerous grants from the National Science Foundation, the Army, and other federal and state institutions. He obtained his Ph.D. in Computer Science from Princeton University in 1985 and MS in Computer Science from Princeton University in 1981.

Amarda Shehu is a Professor of Computer Science with affiliated appointments in the Department of Bioengineering and School of Systems Biology. She is also Co-Director of the Center for Advancing Human-Machine Partnerships (CAHMP). Shehu's research is in artificial intelligence and machine learning to bridge between computer science, engineering, and the life sciences. Shehu and her laboratory have made many contributions on elucidating the relationship between macromolecular sequence, structure, dynamics, and function.