Development of Machine-Learning Models to Explore the Scoring Function Space with SAnDReS 2.0

Walter F. de Azevedo, Jr.[1], Marcos A. Villarreal[2], Rodrigo Quiroga[2], Nelson José Freitas da Silveira[3], Ihosvany Camps[1,4], Gabriela Bitencourt-Ferreira[5], Amauri Duarte da Silva[5,6], Martina Veit-Costa[7], Patricia R. Oliveira[8], Kathia M. Honorio[8,9], Marcelo S. Lauretto[8], Marco Tutone[10], Nadezhda Biziukova[11], Vladimir Poroikov[11], Olga Tarasova[11], Stéphaine Baud[12].

[1]Department of Physics. Institute of Exact Sciences. Federal University of Alfenas. Av. Jovino Fernandes de Sales 2600, Bairro Santa Clara. Alfenas. MG. Brazil. 37133-840.

[2]Instituto de Investigaciones en Fisicoquímica de Córdoba (INFIQC), CONICET-Departamento de Matemática y Física, Facultad de Ciencias Químicas, Universidad Nacional de Córdoba, Ciudad Universitaria, Córdoba, Argentina.

[3]Laboratory of Molecular Modeling and Computer Simulation - MolMod-CS, Institute of Chemistry, Federal University of Alfenas - UNIFAL-MG, Alfenas, Brazil.

[4]High Performance & Quantum Computing Labs, Waterloo, Canada.

[5]Laboratory of Computational Systems Biology, Pontifical Catholic University of Rio Grande do Sul (PUCRS), Ipiranga Avenue, 6681 Partenon, 90619-900, Porto Alegre/RS, Brazil.

[6]Specialization Program in Bioinformatics, Pontifical Catholic University of Rio Grande do Sul (PUCRS), Ipiranga Avenue, 6681 Partenon, 90619-900, Porto Alegre/RS, Brazil.

[7]Western Michigan University, Kalamazoo MI 49008-5200, USA.

[8]School of Arts, Sciences and Humanities, University of São Paulo, São Paulo 03828-000, SP, Brazil.

[9]Federal University of ABC, Santo Andre, Sao Paulo, Brazil.

[10]Dipartimento di Scienze e Tecnologie Biologiche Chimiche e Farmaceutiche (STEBICEF), Università di Palermo, Via Archirafi 32, 90123 Palermo, Italy.

[11]Institute of Biomedical Chemistry, Pogodinskaya Str., 10/8, Moscow 119121, Russia.

[12]Université de Reims Champagne Ardenne. Laboratoire SiRMa, UMR CNRS/URCA 7369. UFR Sciences Exactes et Naturelles. Moulin de la Housse. 51687 Reims Cedex 2. France.

Abstract

Classical scoring functions from docking programs exhibit low accuracy in determining protein-ligand binding affinity. The availability of protein structures with affinity data makes it possible to create machine-learning models focused on specific protein systems with superior predictive performance. Here, we report a new methodology that merges the AutoDock Vina 1.2 with 54 regression methods available in Scikit-Learn to calculate the binding affinity based on protein-ligand structures. This approach explores the concept of scoring function space. SAnDReS allows the development of machine-learning models based on crystal, docked, and AlphaFold-generated structures. As a proof of concept, we examine the performance of SAnDReS-generated models in six case studies. In the cases studied here, our models outperformed classical scoring functions. Also, SAnDReS-generated models showed predictive performance close to or better than other machine-learning models such as Taba, K_{DEEP}, CSM-lig, and $\Delta_{Vina}RF_{20}$. SAnDReS 2.0 is available to download at https://github.com/azevedolab/sandres.

Introduction

Protein-ligand docking simulations rely on two computational methods: search algorithm and scoring function [1]. Search algorithms try to position a ligand into the binding pocket of a protein target. Scoring functions evaluate the binding of ligands into proteins [2]. This combination of algorithms allows the scanning of large small-molecule libraries, which makes docking the first choice to try to find a hit to bind to a protein target. The application of docking simulations to large datasets of potential binders is a fundamental step in the early stages of drug discovery [3, 4].

Analysis of several protein-ligand docking programs revealed that they most likely work to fit ligands into a binding pocket of a protein target [5, 6]. On the other hand, the prediction of binding affinity based on these atomic coordinates has low accuracy compared with experimental data. This lack of accuracy in estimating binding affinity strongly indicates the need for new computational approaches to address this problem. Most of the scoring functions employed in docking programs (e.g., AutoDock4 [7], Molegro Virtual Docker [8], and AutoDock Vina [9, 10]) rely on training a polynomial equation against a fixed training set. We named these scoring functions classical scoring functions [11] or universal scoring functions [12, 13]. These polynomial equations may have energy terms for physical intermolecular interactions [7, 8] or terms derived from a machine learning perspective [9, 10].

Research in the field showed that the targeted scoring functions [12] outperform classical scoring functions available in most protein-ligand docking programs [11, 12]. Also, machine-learning models to predict binding affinity showed promising results (e.g., models based on the random forest method) [11]. In this scenario, new theoretical studies provided some framework to address this problem. The most notable being the scoring function space (SFS) concept [13]. The SFS is a mathematical set composed of infinite scoring functions. Each scoring function can predict the binding affinity based on the atomic coordinates of a protein-ligand complex. This complex could be an experimental structure (e.g., crystal structure) or a protein-pose complex generated through docking simulations [14-16].

The SFS concept adopts a systems-level approach to address protein-ligand interactions. Research focused on biological systems considers the connections of abstract mathematical sets in contrast to the reductionist interpretation applied in biochemistry [17]. Computational simulations to find novel ligands for a particular protein target benefit from the SFS concept [18-25]. Its usage enables it to address protein-ligand binding affinity as a system involving protein structures and binding-affinity data. Evaluation of the associations of the chemical [26] and protein [27, 28] spaces can contribute to the study of protein-ligand interactions. This view of machine-learning models to calculate binding affinity relies intensely on the concept of SFS.

We envisage the SFS as a set of mathematical models to address the relationship of the protein and chemical spaces. To understand this concept, we may take one element of the protein space - an enzyme target. Then, we select a region of the chemical space composed of inhibitors of this protein target. Elements of the SFS are hypothesis functions ($h_i(\vec{\mu}_i, \vec{f}_i)$) (or simply scoring functions) that take the model's parameter vectors ($\vec{\mu}_i$) and instance's feature vectors ($\vec{f}_i$) and estimate the binding affinity. We have a set indicated as *SFS* composed of infinite hypothesis functions ($h_i(\vec{\mu}_i, \vec{f}_i)$) as follows.

$$SFS = \{h_1(\vec{\mu}_1, \vec{f}_1), h_2(\vec{\mu}_2, \vec{f}_2), h_3(\vec{\mu}_3, \vec{f}_3), \cdots, h_i(\vec{\mu}_i, \vec{f}_i), \dots\}$$

Each hypothesis function ($h_i(\vec{\mu}_i, \vec{f}_i)$) is a scoring function to estimate the binding affinity for this targeted protein. Figure 01 illustrates the SFS concept with a few proteins and their respective hypothesis functions ($h_i(\vec{\mu}_i, \vec{f}_i)$). Each dot (•) represented in the SFS is a hypothesis function developed for the protein target. The closer the dot is to the protein target, the better the predictive performance of this model.

In Figure 01, we represent classical scoring functions (a.k.a. universal scoring function) as *x*, *+*, and Δ. Since these functions did not take any specific target to generate their hypothesis functions ($h_{\times}(\vec{\mu}_{\times}, \vec{f}_{\times})$, $h_{+}(\vec{\mu}_{+}, \vec{f}_{+})$, and $h_{\Delta}(\vec{\mu}_{\Delta}, \vec{f}_{\Delta})$), they are not close to any protein target. We may think of these classical scoring functions as protein-averaged models with an average performance for most of the proteins used in the training process. Therefore, their predictive performance is poor compared to targeted-scoring functions. For instance, let us consider we seek to develop a scoring function targeted to cyclin-dependent kinase 2 (CDK2) with K_i data (our protein system). Our approach generated a hypothesis function indicated as $h_{CDK2}(\vec{\mu}_{CDK2}, \vec{f}_{CDK2})$. In the SFS, $h_{CDK2}(\vec{\mu}_{CDK2}, \vec{f}_{CDK2})$ is closer to CDK2 than any other hypothesis function, including those available in docking programs ($h_{\times}(\vec{\mu}_{\times}, \vec{f}_{\times})$, $h_{+}(\vec{\mu}_{+}, \vec{f}_{+})$, and $h_{\Delta}(\vec{\mu}_{\Delta}, \vec{f}_{\Delta})$). We do not expect classical scoring functions to outperform the CDK2-targeted scoring function ($h_{CDK2}(\vec{\mu}_{CDK2}, \vec{f}_{CDK2})$).

When building $h_{CDK2}(\vec{\mu}_{CDK2}, \vec{f}_{CDK2})$, we took ligands from the chemical space. Specifically, a sub-space of the chemical space made of inhibitors of CDK2 for which K_i is available. The challenge is to find an adequate hypothesis function (scoring function) in the SFS. This equation models a relationship between one element of the protein space (CDK2) and a limited region of the chemical space (CDK2 inhibitors with Ki data). Since we have infinite hypothesis functions in the SFS, it would be necessary to have an infinite computational time to find an ideal $h_{CDK2}(\vec{\mu}_{CDK2}, \vec{f}_{CDK2})$. On the other hand, we could devise strategies to investigate finite parts of the SFS using intelligent approaches as in machine-learning methods. Using supervised machine learning techniques, we can explore this SFS to build a computational model targeted to a specific protein system.

We have the principles we defined above implemented in the program SAnDReS 2.0, which is an acronym for Statistical Analysis of Docking Results and Scoring Function. SAnDReS 2.0 can explore the SFS concept through the integration of recent progress in the fields of machine learning and protein-ligand docking simulations. This new version brings together AutoDock Vina (version 1.2) [10] and regression methods available in the Scikit-Learn (version 1.3) [29]. We have a collection of 54 regression methods available in SAnDReS 2.0 to generate new scoring functions. Also, we use statistical metrics designed to evaluate biological systems described in the DOME (Data, Optimization, Model, and Evaluation in Machine Learning) study [30]. We named these metrics the DOME strategy.

This freedom to play with the features (energy terms, descriptors, and additional parameters) and regression methods makes it possible to explore a wider region of the SFS [13-15], increasing the chances of finding an adequate model for a protein system. Although on computational modeling, some argue that the data is more relevant than the machine learning algorithms for complex problems [31]. Several studies showed that variations of the machine learning algorithms may generate scoring functions with superior predictive power compared with classical scoring functions [32-37]. These results highlight the importance of trying different computational methodologies when studying complex systems. In SAnDReS, we adopt this idea

of freedom to search unexplored regions of the SFS. We seek to find a machine-learning model just right for our protein system.

One key aspect of SAnDReS is the freedom of choice. You can test several machine-learning models (hypothesis functions) and choose the model you find is adequate for the protein you are studying. With this view, we combine the holistic approach of systems biology with machine-learning methods to contribute to the early stages of drug discovery projects [38]. As far as we know, SAnDReS 2.0 is the first computational tool to integrate the most recent versions of AutoDock Vina (version 1.2) and Scikit-Learn (version 1.3) in one package. Also, SAnDReS 2.0 is the first computational tool to include the DOME strategy to evaluate all machine-learning models developed to calculate protein-ligand binding affinity. In the following sections, we describe the main new aspects of SAnDReS 2.0 and analyze six case studies exploring different aspects of SAnDReS 2.0. We compare SAnDReS-generated models against similar computational tools.

Methods

Overview

The central idea behind SAnDReS is the SFS concept [14]. SAnDReS navigates the SFS and builds a machine-learning learning model ($h(\vec{\mu}, \vec{f})$) targeted to one protein system. However, the flexibility of SAnDReS also allows the development of universal scoring functions to predict pK_i, pK_d, and pIC_{50} (see case study 06). When targeting one protein system with SAnDReS, we gave up a one-size-fits-all approach [13] taken by the classical scoring functions employed in protein-ligand docking programs (e.g., AutoDock Vina). Another concept to understand how SAnDReS works is the idea of a protein system.

SAnDReS considers a protein system composed of *M* structures of a specific protein for which experimental affinity data is available. We rely on experimental binding affinity data from the following databases: BindingDB [39] and PDBbind [40]. For SAnDReS, a protein system has two types of data: experimental binding affinity and structures of protein-ligand complexes. These *M* protein-ligand complexes are structures for which we have affinity data. They could be crystal structures (e.g., case studies 01 and 06) or docked poses (e.g., case studies 03 and 05). We could also take computational models (e.g., models developed using AlphaFold [41]) of the protein targets to create a targeted scoring function (see case study 05). Taking this view, we focus on creating an adequate model from the SFS for one protein target. With this perception, we can apply the targeted scoring function (with *N* features) to choose protein-ligand complexes in virtual screening projects.

SAnDReS 2.0 incorporates two main programs: SAnDReS tools and MLRegMPy (Machine-Learning Regression Methods in Python). The first handles downloading of structures, file format conversion (PDB->PDBQT and mol2->PDBQT), protein-ligand docking simulations with AutoDock Vina 1.2 [10], statistical analysis (using SciPy and Scikit-Learn [29]), and plot generation (using Matplotlib). MLRegMPy carries out machine-learning modeling using Scikit-Learn [29]. Figure 02 shows a flowchart with all tasks available in SAnDReS 2.0.

In a typical project developed with SAnDReS, we aim to create a machine-learning model ($h(\vec{\mu}, \vec{f})$) using protein-ligand complexes and binding affinity data. In the first step, we set up a project directory. We expect to have one project directory for each protein system. Then, we add

the PDB access codes used in the project. After, we prepare the ligand data and download the structures (PDB files). In the following, we generate PDBQT files for docking simulations and calculate energy terms from the AutoDock Vina 1.2 force field (VinaFF) [10]. Then, we can carry out protein-ligand docking simulations. The next step calculates energy terms (VinaFF), descriptors, and additional parameters (described later in the text) for docked and crystallographic structures. This part provides the potential features ($\vec{f}$) we will employ to generate our hypothesis functions ($h(\vec{\mu}, \vec{f})$).

In the following, we use the virtual screening option. In the sequence, we have the machine-learning box encapsulating all regression methods. Then, we may apply the machine-learning models to docking results and virtual screenings. Finally, we can perform statistical analysis of machine-learning models and evaluate intermolecular contacts for protein-ligand structures in the dataset.

We do not need to use all tasks shown in Figure 02 for a project using SAnDReS. The flexibility of SAnDReS allows the users to choose the set adequate to their project. To illustrate this flexibility, we describe here six case studies highlighting which tasks are necessary for each type of project. Also, to facilitate its use, SAnDReS 2.0 has five out of six case studies integrated as tutorials to its code. These tutorials allow the users to have a self-learning platform to study how to apply SAnDReS to a wide range of drug discovery projects.

SAnDReS 2.0 has more regression methods (54 techniques) than the previous version (9). Also, it incorporated novel developments in the statistical analysis of the machine-learning models. Recently, a consortium of machine learning researchers (ELIXIR Machine Learning Focus Group) recommended a set of metrics to assess machine learning models applied to biological systems [30]. For regression models developed for biological systems, they indicated the application of the following metrics: coefficient of determination (R^2), root mean squared error (RMSE), and mean absolute error (MAE). SAnDReS adds the DOME strategy to the set metrics commonly used to evaluate the performance of machine-learning models. With this set of metrics, we aim to establish standards to validate machine-learning models focused on protein systems. Also, SAnDReS 2.0 relies on the most recent version of AutoDock Vina (version 1.2). This new version has a tool to explore the SFS and create a hypothesis function (model) for a specific protein system (our main goal) or a universal scoring function.

Exploring the SFS

The main novelty of SAnDReS 2.0 is a tool to explore the SFS (explore-sfs tool). Figure 03 illustrates a flowchart of the explore-sfs tool. Initially, we define a set of features based on VinaFF energy terms, ligand descriptors, and additional parameters (e.g., B-factors) (supplementary material 01). Taking this set of input features, we generate a combination of possible models to estimate the binding affinity. SAnDReS determines a regression model (for each combination) that takes input features, as shown in the following equation,

$$PBA = h(\mu_0, \mu_1, \mu_2, .., \mu_n, f_0, f_1, f_2, \dots, f_n)$$

where *PBA* is the predicted binding affinity value, *n* is the number of features, f_i is the i-th feature value, μ_i is the i-th model parameter (including the bias term μ_0 and the feature weights μ_1, μ_2, …, μ_n), and *h* is a hypothesis function involving μ_i and f_i.

In a vectorized form, we can express the above equation as follows,

$$PBA = h(\vec{\mu}, \vec{f})$$

where $h(\vec{\mu}, \vec{f})$ (hypothesis function) is the SAnDReS-generated model using $\vec{\mu}$ and $\vec{f}$, $\vec{\mu}$ is the model's parameter vector (μ_0, μ_1, μ_2, …, μ_0), $\vec{f}$ is the instance feature vector (f_0, f_1, f_2, …, f_n where $f_0 = 1$).

With 14 variables, we could evaluate all hypothesis functions with eight input features ($\vec{f}$): $C_{14,8}$ combinations of $\vec{f}$. Then, SAnDReS trains each combination of $\vec{f}$ using 54 regression methods available in the Scikit-Learn library (supplementary material 02). For each machine-learning model, SAnDReS apply the DOME strategy [30]. For the example of 14 features taken eight at a time without repetition, we have 54x3003 = 162,162 machine-learning models ($h(\vec{\mu}, \vec{f})$). Since the regression methods are fast in Scikit-Learn, we can generate a statistically relevant number of models with modest computational resources. For instance, using an Intel Core i5-10300H processor, it took 11 hours and 33 minutes to create 162,162 regression models ($h(\vec{\mu}, \vec{f})$) to predict pIC_{50} for CDK2 (EC 2.7.11.22) (Case Study 01).

SAnDReS reads the file *ml.in* to define the parameters to explore the SFS. We describe the main command lines (input file ml.in) in Table 01. SAnDReS 2.0 aims to develop machine-learning models to predict binding affinity. This version of SAnDReS focuses on six possible target functions to represent binding affinity. Table 02 shows all target functions available in SAnDReS. Further details are available in a User Guide provided along with the code: https://github.com/azevedolab/sandres.

Dataset

Here, we prepare all protein-ligand structures for docking, energy terms evaluation, and virtual screening. This part involves downloading from the protein data bank (PDB) and converting it to PDBQT format. SAnDReS has predefined ligand data with experimental binding affinities (K_i, K_d, and IC_{50}) and generated PDBQT files for ligand structures. We previously created ligand structures using ADFRsuite [42]. The users can add any ligand structures not present in the current dataset. We employed the affinity data in the PDBbind version 2020 [40] to generate files with ligand information (*bind_IC50.csv*, *bind_Kd.csv*, and *bind_Ki.csv*). SAnDReS adds hydrogen atoms to protein coordinates using the program reduce [43] and automatically converts PDB protein structures to PDBQT using ADFRsuite [42].

Docking Hub

SAnDReS can carry out redocking for all crystallographic structures in a dataset. The goal is to validate a docking protocol using AutoDock Vina 1.2 [10]. Then, we can apply the best docking protocol to virtual screening (VS). Also, we may employ a SAnDReS-generated scoring function to sort docking results and predict binding for poses generated during the docking simulations.

SAnDReS creates biological units of the protein structure because we may have a protein for which the active site lays between monomers (e.g., human purine nucleoside phosphorylase). The asymmetric unit of purine nucleoside phosphorylase is a monomer, and the biological assembly is a trimer. Also, the binding pocket of this enzyme sits between the monomers (see structure 1V2H) [44].

At the end of docking simulations, SAnDReS takes all results and determines the root-mean-squared deviations (RMSD) and docking accuracies *DA1(a, b)* and *DA2(a, b, c)* defined as follows.

$$DA1(a,b) = f_a + 0.5(f_a - f_b)$$

$$DA2(a,b,c) = DA1(a,b) + 0.25(f_c - f_b)$$

where f_a is the fraction poses for which the docking RMSD is less than *a* and f_b is the fraction poses for which the docking RMSD is less than *b*, where *a<b* and f_c is the fraction poses for which the docking RMSD is less than *c*, where *a<b<c* [45]. In the current version of SAnDReS, the values for *a*, *b*, and *c* are 2.0, 3.0, and 4.0 Å, respectively. We evaluate the docking RMSD by the following equation.

$$RMSD = \sqrt{\frac{\sum_{i=1}^{N}[(x_e - x_t)^2 + (y_e - y_t)^2 + (z_e - z_t)^2]}{N_a}}$$

where x_e, y_e, and z_e are the experimental coordinates (e.g., crystallographic structure) for the ligand, and x_t, y_t, and z_t are the atomic coordinates for the position generated by the docking simulation (pose). When we calculate the summation, we consider the N_a nonhydrogen atoms in the ligand structure.

Scoring Function

Now, we calculate the energy terms, descriptors, and additional parameters for the crystallographic positions and poses generated during the docking simulations. SAnDReS employs energy terms available on the VinaFF [10]. We use these terms as features of our machine-learning models ($h(\vec{\mu}, \vec{f})$). Also, SAnDReS calculates a few descriptors, such as the number of atoms found in the structure of the ligands (e.g., C, N, O). SAnDReS calculates additional parameters based on crystallographic information (e.g., Ligand B-factor). It is possible to generate hybrid scoring functions involving energy terms (VinaFF), descriptors, and additional parameters (see case study 01).

Virtual Screening

Here, we use AutoDock Vina 1.2 [10] integrated into SAnDReS 2.0 to perform docking screens for a set of potential ligands or known ligands against our protein target. We may carry out docking simulations for known inhibitors to use protein-pose complexes to build novel scoring functions. We start this part with a selection of a mol2 file. This file has all small molecules intended for screening. SAnDReS has a built-in tool to split this mol2 file and generate PDBQT files for individual molecules. In the following, SAnDReS checks the previous redocking results for all structures in the dataset and selects the *config.txt* file for the best result. We consider the best result the one with the lowest RMSD. SAnDReS uses the coordinates of the receptor for which we have the lowest RMSD to run the virtual screening. The user may change it, but the default strategy is to take the structure with the lowest RMSD.

Machine Learning Box (For Modeling)

Here, we generate a machine-learning model for a targeted protein and save it for further use. SAnDReS 2.0 relies on Scikit-Learn [29] to create multiple machine-learning models to predict binding affinity. We may employ crystallographic structures or docked poses (receptor-pose complexes) to develop our scoring functions. Also, we have two sources of experimental data: the crystallographic structures and the affinity data (e.g., pIC_{50}). We can have pIC_{50} (or pK_i or pK_d) as our target functions. SAnDReS employs the structures to calculate the VinaFF energy terms (Gauss 1, Gauss 2, Repulsion, Hydrophobic, Hydrogen, Torsional), descriptors, and additional parameters. These are the features used in machine learning modeling.

We have 54 regression methods (Supplementary Material 02) available in SAnDReS to generate new scoring functions. This freedom to play with the features and regression methods makes it possible to explore a wider region of the SFS [38], increasing the chances of finding an adequate model for our protein system.

Half of the regression methods available in SAnDReS use cross-validation (CV). We implemented the Kfold class from Scikit-Learn to perform cross-validation. The Kfold class builds an n-fold cross-validation loop and tests the generalization ability of regression. Cross-validation better estimates of how well we could generalize to predict unseen data. Scikit-Learn [29] provides some regression classes with built-in cross-validation implementation, e.g., ElasticNetCV. However, this inclusion of built-in CV is not available for all regression methods (e.g., AdaBoostRegressor). Therefore, we adopted the same CV approach [46] for the regression methods in SAnDReS 2.0. The MLRegMPy package has a class (ValidationLoop) that carries out cross-validation for all CV methods.

For all machine-learning models generated in this part, we have an evaluation of the predictive performance using the DOME strategy. Once we select an adequate machine-learning model, SAnDReS saves it as a joblib file. Then, it is possible to apply it to other structures of the same protein system. With SAnDReS, we may employ a previously generated machine-learning model in our protein system. It is possible to share these models and make them available on GitHub.

Machine Learning Box (For Docking Results)

Now, we use the docking results and apply our machine-learning models against them. We may calculate binding affinity using the machine-learning models generated in the previous section. It is also possible to download a previously built machine-learning model and use it to predict binding affinity for docked poses. Here, SAnDReS does not perform machine learning regression. It only applies the created models (joblib files) to the docking results. In this part, SAnDReS evaluates the predictive performance of the machine-learning models used for docked poses. It also verifies the performance of the SAnDReS-generated scoring functions to rank poses and determine docking RMSD and docking accuracy (*DA1* and *DA2*).

Machine Learning Box (For Virtual Screening Results)

Here, we apply a SAnDReS-generated machine-learning model to the results of our VS simulation. We employ one regression model only. The goal is to use a machine-learning model to sort VS poses, selecting the most promising ones (lowest score).

Statistical Analysis

DOME strategy calculates an Euclidean distance (called L2-Norm) using the metrics specified by Walsh et al. 2021 for regression models of biological systems [30]. Our goal is to evaluate the Euclidean distance of a machine-learning model from an ideal model with the following coordinates: Root-mean squared error (RMSE) = 0.0, mean absolute error (MAE) = 0.0, and coefficient of determination (R^2) = 1.0. We also have derived metrics merging the DOME strategy with the evaluation of Spearman's (ρ) and Pearson's (r) correlations named EDOME (Extended DOME).

We define the expressions for ρ, r, RMSE, MAE, R^2, DOME, EDOMEr^2, EDOMEρ, and EDOME in Table 03. The metrics EDOMEr^2 and EDOMEρ add r^2 (squared Pearson correlation) and ρ (Spearman correlation) to the Euclidean distance equation (DOME), respectively. The ideal model has $r^2 = 1.0$ and $\rho = 1.0$. The last metric is EDOME. We have a space composed of r^2, ρ, RMSE, MAE, and R^2. Our goal is to evaluate the distance of a machine-learning model from an ideal model with the following coordinates: $r^2 = 1.0$, $\rho = 1.0$, RMSE = 0.0, MAE = 0.0, and $R^2 = 1.0$. An ideal model has DOME = EDOMEr^2 = EDOMEρ = EDOME = 0.0.

For every model generated to explore the SFS, SAnDReS determines the following nine metrics: RMSE, MAE, R^2, DOME, r^2, ρ, EDOMEr^2, EDOMEρ, and EDOME. We will apply this statistical analysis for all SAnDReS-generated models described in the following case studies. For benchmark purposes, we will determine the predictive performance for models to estimate pK_i using alternative approaches. We focused on the CASF-2016 test set [47] to evaluate the predictive performance of SAnDReS-generated models against external scoring functions (case study 06). Amongst these scoring functions, we have classical scoring functions and machine-learning models (K_{DEEP} [48], CSM-lig [49], and $\Delta_{Vina}RF_{20}$ [50]).

Case Studies

Here, we find five case studies focused on a protein system and one study to build a universal scoring function (case study 06). A protein system has one target and binding affinity data. Taking this definition, CDK2 with IC_{50} (case study 01) is a protein system different from CDK2 with K_i data (case study 03). In case study 06, we do not focus on one protein system. We train our scoring functions using a data set of high-resolution crystallographic structures. Our goal in case study 06 is to check the predictive performance of a universal scoring function generated with SAnDReS and other functions against the CASF-2016 [47] benchmark with a test with K_i data.

Case Study 01: CDK2 with IC_{50} Data

Our goal in this case study is to build a machine-learning model based on the crystallographic positions of the ligands. We did not use docked poses to generate our machine-learning model. We also run re-docking simulations of the structures in this dataset to evaluate the predictive performance of the SAnDReS-generated scoring functions to rank poses. We determined docking accuracy and RMSD to reranked poses using our machine-learning models. Figure 04 Indicates all steps of this case study. We set up a dataset of CDK2 structures complexed with different inhibitors. We eliminated repeated ligands from the dataset. Then, we used SAnDReS to generate PDBQT files for all structures in this dataset and carried out docking using the Docking Hub of SAnDReS. Later, we calculated the affinity using the Scoring Function interface of SAnDReS. We

intend to compare the ranking of poses using a SAnDReS machine-learning model with the Affinity function of AutoDock Vina 1.2. In the following, we generated machine-learning models based on the energy terms calculated using VinaFF and additional descriptors. The best machine-learning model was applied to evaluate docking results. We determined the predictive performance of machine-learning models for all case studies using previously described metrics.

Case Study 02: Application of a Machine Learning Model to CDK2 Docked Structures with IC_{50}

This case study shows the application of a machine-learning model built in case study 01 to docked structures. We saved the previously generated regression model as CDK2_IC50_ExtraTreeRegressor. This model is to predict binding affinity (pIC_{50}) for CDK2 structures based on the atomic coordinates of protein-ligand complexes. Figure 05 illustrates the steps we followed for this case study. Here, we employed affinity data available in BindingDB [39] to test the predictive power of our machine-learning model. We prepared a mol2 file with a toy dataset from the BindingDB for which IC_{50} data is available but no crystal structures of the protein-ligand complexes. We randomly selected 50 ligands from the BindingDB with pIC_{50} ranging from 4.25 to 8.7. We generated the complexes using docked structures of the ligands against the structure 2DS1 [51]. We employed AutoDock Vina 1.2 to perform a short docking screen of known inhibitors against our protein target, the CDK2. We restricted our docking simulations to a cube (16 Å x 16 Å x 16 Å) centered at the ATP-binding pocket with the following coordinates: x = -9.060 Å, y = 10.360 Å, and z = 13.454 Å. SAnDReS generated a file with virtual screening results (*virtual_screening.csv*) for these complex structures of CDK2 inhibitors. This file has descriptors, additional parameters, and VinaFF energy terms for each pose calculated for all ligands used in the dataset, which allow us to apply the CDK2_IC50_ExtraTreeRegressor model to predict pIC_{50} for these structures. We applied the DOME strategy to evaluate all models.

Case Study 03: CDK2 Docked Structures with K_i Data

Here, we studied docked structures of CDK2 complexed with known inhibitors for which K_i data is available. We built a machine learning model to predict pK_i using docked structures. Figure 06 highlights all the steps followed for this case study. We used binding affinity data and ligand structures from the BindingDB (search performed on November 20, 2023). We prepared a mol2 file with molecules available in the BindingDB. We docked all inhibitors against the structure of CDK2 (PDB access code: 2DS1) as previously described in case studies 01 and 02. The resulting docked structures and binding affinity data were the source to train our models to predict pK_i. We selected the best model based on the performance metrics for the test set.

Case Study 04: Application of a Machine Learning Model to CDK2 Structures with K_i

Here, we applied the machine-learning model generated using docked structures (case study 03) to another dataset of docked structures of the same protein target. Figure 05 shows all the steps of this case study. We run our docking simulations using the ligands extracted from the crystallographic structures of the Taba test set [35]. We employed the same docking protocol described in case studies 02 and 03. We used the coordinates of 2DS1 [51]. In previous case studies, we compared the predictive performance of SAnDReS-generated models against classical scoring functions. Now, we applied the machine-learning model generated in case study

03 to predict binding affinity for structures in a test set used in Taba development. Taba uses regression methods to create a machine-learning model based on a spring-mass system to assess intermolecular interactions [35]. We applied the CDK2_Ki_DecisionTreeRegressorCV model to the ligands used as a test set in the development of Taba.

Case Study 05: AlphaFold Model of CDK19 with IC_{50} Data

This case study focuses on developing a machine-learning model to predict inhibition (pIC_{50}) of cyclin-dependent kinase 19 (CDK19). Figure 07 illustrates all the steps used in this case study. The experimental structure for this enzyme is not available (PDB search performed on November 20, 2023). Therefore, we employed a model (PDB access code: (AF_AFQ9BWU1F1) generated using AlphaFold [41]. We superposed the structure of CDK19 (AF_AFQ9BWU1F1) onto the crystal structure 2DS1 [51]. Then, we transferred the inhibitor of 2DS1 (ligand CD1) to the superposed CDK19. Finally, we carried out model optimization of the CDK19-1CD structure using the minimization of sidechain positions. We employed Molegro Virtual Docker [8] to optimize the CDK19-CD1 complex. We validated the docking (re-docking) using the protocol described in Cases 02-04 for structure 2DS1 using AutoDock Vina 1.2. The RMSD (docking) between the docked and the model of the CDK19-CD1 complex is 1.133 Å. We employed ligand structures (mol2 format) and IC_{50} data in the BindingDB. We identified 127 unique molecules for which IC_{50} data for CDK19 is available in the BindingDB (search performed on November 20, 2023). We followed the same procedures described in Case Study 03. We run all docking simulations using AutoDock Vina 1.2 integrated into SAnDReS. We created our machine-learning models employing these docked structures and set up 12 features, taking eight at a time without repetition. We also used the DOME strategy to choose our best machine-learning model.

Case Study 06: CASF-2016 with Ki Data

Here, we develop a universal scoring function to predict inhibition (pK_i). Figure 08 shows all the steps used in this development. We selected crystallographic structures for which K_i data is available. We filtered our training set by choosing the structures with a resolution of 1.8 Å or better. Also, we eliminated structures for which active ligands (small-molecule inhibitors) presented an occupation factor below 1.0. In doing so, we focused on ligand structures with one position for the active ligands. These structures comprised our training set. Our goal in the case study is to evaluate the predictive performance of SAnDReS-generated scoring functions against 36 external scoring functions, including three obtained using machine learning approaches ((K_{DEEP} [48], CSM-lig [49], and $\Delta_{Vina}RF_{20}$ [50])). Supplementary material 03 has all scoring functions for the CASF-2016 test set with K_i data. The original CASF-2016 test set comprises crystallographic structures with K_i and K_d data. We focus our study on PDBs with K_i information.

Results and Discussion

We applied SAnDReS 2.0 to six case studies to highlight its flexibility and ability to automatize docking simulations and machine-learning modeling. In the first five case studies, we focused on CDKs due to structural and binding data availability for these proteins. Another reason to focus on CDK2 and CDK19 is their importance as protein targets for drug development. CDK2 and CDK19 are targets of anticancer drugs [52-55]. Many CDKs had their structures determined due

to their role in cell cycle progression. For instance, CDK2 inhibition causes the blockage of cell cycle progression. This halt of cell cycle progression may lead to apoptosis of DNA-damaged cells [52].

So far, we have over four hundred CDK2 structures available in the PDB (search performed on November 20, 2023). This richness of structural information has paired with binding affinity data. On the other hand, CDK19 has no crystallographic structure. Nevertheless, we found an AlphaFold model for this protein (case study 05). In case study 06, we have a test of the SAnDReS scoring function against CASF-2016 test set structures with K_i data.

We organized the following sections focusing on different protein systems (crystallographic structures plus binding data) with one exception: case study 06. This last case study dealt with a benchmark based on the CASF-2016 test set. The first five case studies reported scoring functions developed for specific protein systems: CDKs. They presented situations faced in the early stages of drug discovery projects. They involved machine learning modeling using crystal structures with IC_{50} (case study 01). We also developed models based on docked poses (case studies 03 and 05). We described the application of previously generated machine learning models in case studies 02 and 04.

Case Study 01: CDK2 with IC_{50} Data

Our approach to chemical inhibition of kinase activity considers that ATP-competitive inhibitors of CDK2 reduce activity by physically blocking the ATP-binding pocket. For activation, CDK2 needs binding to a partner protein named cyclin and phosphorylation of residue Thr160 by CDK7 [52]. Also, it is necessary to dephosphorylation of residue Tyr15 by the phosphatase CDC25A [56, 57]. A previous computational study [58] focused on Xenopus eggs indicated that activation cdc2 behaves as an oscillatory system. We may take the cdc2-cyclin complex in Xenopus eggs as a prototype to evaluate the activation of cell-cycle human CDK2. In the Xenopus system, cyclin level varies during cell cycle progression, and the peak of kinase activity occurs after cdc2-cyclin complex formation in the late phase of the modeled cycle [58].

We propose that the inhibitors bind to cdc2 (or CDK2) early in the cell cycle, which blocks the ATP-binding pocket. Inhibitors prevent enzyme activation even with the binding of the cyclin partner. In this case study, we used the crystallographic structure of CDK2 not complexed with cyclin (monomeric CDK2 structure). We also filtered our dataset to have unique CDK2-inhibitors. After data filtering, we ended up with a dataset composed of 104 crystallographic structures of CDK2 complexed with inhibitors for which binding data is available. We split this dataset into training (74 structures) and test (30 structures) sets. SAnDReS downloaded the PDB structures listed in the supplementary material 04.

We used eight independent variables out of 14 features to generate our machine-learning models. SAnDReS built 162,162 scoring functions (54 x $C_{14,8}$ = 54 x 3003 = 162,162 machine-learning models). Taking the lowest EDOME among machine-learning models, we selected the ExtraTreeRegressor model with the following features: Torsions, B-factor ratio (Ligand/Receptor), Q, Gauss 1, Ligand Occupation Factor, Gauss 2, Ligand B-factor(A2), Receptor B-factor(A2). This machine learning model (r^2 = 0.297416, ρ = 0.547917, RMSE = 1.23548, and EDOME = 1.99604) shows superior predictive performance compared with AutoDock Vina scoring function (r^2 = 0.0711395, ρ =-0.341382, RMSE = 14.1781, and EDOME = 158.617) for test set structures. Figures 09A and 09B show the predictive performance using previously defined metrics (r^2, ρ, and

EDOME) and the scattering plot (Predicted pIC_{50} vs. Experimental pIC_{50}) for all structures in the test set.

We applied the ExtraTreeRegressor model to evaluate its performance in predicting affinity and ranking poses. Figures 10A and 10B show the predictive performance (r^2, ρ, and EDOME) and the scattering plot (Predicted pIC_{50} vs. Experimental pIC_{50}) for test set structures. Our machine learning model (ExtraTreeRegressor) shows the best predictive performance for docked poses for prediction of pIC_{50} (r^2 =0.200877, ρ =0.414735, RMSE = 1.34298, and EDOME = 2.27428) against Affinity (AutoDock Vina scoring function) (r^2 = 0.0280475, ρ =-0.291977, RMSE = 15.4249, and EDOME =187.519). Supplementary material 05 brings the predictive performance for all terms and machine learning models determined for the selected set of features in the scoring function.

We have the analysis of docking accuracy (DA) [34] in supplementary material 06. The ExtraTreeRegressor model performs better than the Affinity function (see docking accuracy) [34]. The performance is also better for r^2, ρ, RMSE, and EDOME for the predicted affinity. Amongst all these metrics, we may say that the most demanding for a machine learning model trained against binding affinity data and crystal structures is the docking accuracy (DA), and our model has an improvement from 21.6667 to 26.6667 % for *DA1* compared with the Affinity scoring function of AutoDock Vina. Additionally, we may take a machine-learning model to predict binding affinity and a different model to sort docking poses. In summary, the overall performance of the ExtraTreeRegressor model is better for sorting poses and predicting the binding affinity (pIC_{50}).

We used all structures in this dataset to evaluate intermolecular interactions involving all ligands and their respective protein coordinates. SAnDReS took all structures and determined all contacts, splitting the plots into the main chain, side chain, and all atoms (supplementary materials 07, 08, and 09, respectively). Figure 11 shows the intermolecular contacts for main-chain atoms of CDK2. We find a concentration of peaks in the region close to residues Glu81, Phe82, and Leu83. The main-chain atoms of these residues comprise the molecular fork of CDK2 responsible for most of the contacts observed in CDK2-inhibitors complexes [59, 60].

Case Study 02: Application of a Machine Learning Model to CDK2 Docked Structures with IC_{50}

In this case study, we focused on the previously generated machine-learning model (CDK2_IC50_ExtraTreeRegressor) to estimate binding affinity based on docked structures. As a proof-of-concept project, we randomly selected 50 CDK2 inhibitors for which experimental IC_{50} data is available at the BindingDB. There are no crystal structures of CDK2 in the complexes with these selected ligands. This docking protocol generated an RMSD of 0.234 Å for the ligand 1CD. Then, we docked these 50 ligands against the structure of CDK2 (PDB access code: 2DS1) using the docking protocol chosen in case study 01. Figure 12 shows the docked results for all inhibitors against the ATP-binding pocket of CDK2. We see all docked structures inside a sphere (radius of 12 Å) centered at the ATP-binding pocket (including the molecular fork). These poses exhibit the same pattern of intermolecular interactions identified in crystal structures of CDK-inhibitor complexes (case study 01).

SAnDReS calculated the same group of descriptors, additional parameters, and VinaFF energy terms for each pose obtained for all inhibitors docked to CDK2 structure, which let us apply the CDK2_IC50_ExtraTreeRegressor model to predict pIC_{50} for these complexes. Some may argue

that the set of features employed in the CDK2_IC50_ExtraTreeRegressor model is inadequate to predict pIC_{50} based on docked structures since it relies on crystallographic-derived parameters such as B-factor ratio (Ligand/Receptor), Ligand B-factor(A^2), and Receptor B-factor(A^2). For the protein coordinates, we took the crystallographic-related parameters from the crystal structure (PDB: 2DS1). For the docked ligands, we may determine B-factors using molecular dynamics [61-63]. To avoid running molecular dynamics simulations for 50 complexes, we may set all ligand B-factors to 20.0 $Å^2$ and ligand occupation factors to 1.0. Alternatively, we may eliminate the contribution of these features and set ligand B-factors to 0.0 $Å^2$ and ligand occupation factors to 0.0. We adopted the last one for the ligands. To avoid any problems with crystallographic-related parameters, we may omit them from the machine learning modeling (see case studies 03 and 05).

Figures 13A and 13B present the predictive performance (r^2, ρ, and EDOME) and the scattering plot (Predicted pIC_{50} vs. Experimental pIC_{50}) for all structures. Our ExtraTreeRegressor model exhibits a superior predictive performance to predict pIC_{50} (r^2 = 0.108508, ρ = 0.316343, RMSE = 1.38903, and EDOME = 2.61879) compared with the Affinity function of AutoDock Vina (r^2 = 0.0636156, ρ =-0.151142, RMSE = 6.89501, and EDOME = 45.8168) and all other energy terms and descriptors analyzed for this dataset (supplementary material 10). Although the machine-learning model shows superior predictive performance compared with the Affinity function, the precited values concentrated around two values (4.8 and 7.3) with poor predictive performance. In case study 03, we will build a machine-learning model based on docking poses.

In summary, we highlighted the application of a machine learning model developed using crystallographic structures to predict the binding affinity (pIC_{50}) of docked ligands against the same protein system. Furthermore, we showed the potential of integration of SAnDReS into ongoing docking projects. Once one research group develops a machine-learning model for a protein system, another researcher can integrate it into a docking screen project. To predict pIC_{50} of CDK2, we made available this machine learning model in the GitHub (https://github.com/azevedolab/sandres/blob/master/CDK2_IC50_ExtraTreeRegressor.zip). We suggest that all potential new users of SAnDReS take this strategy and make available their models to researchers working on the same protein systems.

Case Study 03: CDK2 Docked Structures with K_i Data

Now, we have changed the focus to scoring functions based on docked poses against CDK2. In case study 01, we generated a machine learning model to predict pIC_{50} and applied it to predict the affinity of docked poses (case study 02). We based our machine learning modeling on two sources of experimental data: crystallographic structures of protein-inhibitor complexes and binding affinity data. It is worth noting that crystallographic coordinates for protein-ligand complexes may insert the uncertainties inherent to this source of experimental data: the most notable is the crystal packing effects on ligand position [64]. Energy minimization of protein-ligand structures may reduce these crystal-packing effects. The same holds for molecular dynamics simulations [63] before machine learning modeling.

To partially avoid these crystal packing effects on ligand coordinates, we designed SAnDReS to accommodate machine learning modeling using docking results. In doing so, we provide the flexibility to choose the most adequate approach to the protein system under study. Also, we overcome the limitation of experimental data for crystallographic structures of protein-ligand

complexes with binding affinity data. We have $2.8.10^6$ compounds with binding data available in the BindingDB [39] against $2.2.10^5$ structures determined in the PDB [65] (search performed on November 20, 2023). The difference is even higher since only a fraction of the data deposited in the PDB has ligands bound to their structures [1].

We prepared a mol2 file with molecules for which K_i data is available in the BindingDB. In the data preparation, we eliminated ligands for which binding affinity data showed undefined values for the affinity (e.g., >1000 or < 3.5). Our final dataset set has 97 unique inhibitor molecules (supplementary material 11). SAnDReS ran all docking simulations using AutoDock Vina 1.2 integrated into it. We used these docked structures to generate a set of machine-learning models. In the explore-sfs option of SAnDReS, we set up 12 features taken eight at a time without repetition. We built a total of 495x54 (26,730) regression models. SAnDReS selected these 12 features from a pool of 14 potential features (Torsions, Q, Average Q, C, N, O, S, Affinity(kcal/mol), Gauss 1, Gauss 2, Repulsion, Hydrophobic, Hydrogen, Torsional). Among these 14 features, SAnDReS chooses the top 12 with higher correlation (r^2) with experimental affinity (e.g., pK_i). We employed these 12 features to generate the combinations ($C_{12,8}$) during the explore-sfs phase.

Figures 14A and 14B present the map of metrics (r^2, ρ, and EDOME) and the scattering plot (Predicted vs. Experimental values) for the test set. Taking the EDOME as a selection criterion, the best model is the DecisionTreeRegressorCV with the following features: Gauss 2, C, Average Q, Gauss 1, Q, Torsional, Torsions, and Repulsion. This machine-learning model (r^2 = 0.627354, ρ = 0.598595, RMSE = 0.698289, and EDOME = 1.1024) exhibits superior metrics compared with the Vina Affinity function (r^2 = 0.179336, ρ = 0.351028, RMSE = 15.775, and EDOME = 278.755) for the test set. This machine-learning model is available on GitHub (https://github.com/azevedolab/sandres/blob/master/CDK2_Ki_DecisionTreeRegressorCV.zip).

Also, comparing the predictive performance of the models generated in case studies 01 and 03, we see a substantial improvement when using docked structures (this case study). We suggest the following possible causes for this improvement in the predictive performance. Firstly, the use of docked structures seems to pay off. The computationally adjusted positions of the ligands in the ATP-binding pocket of CDK2 capture a more realistic view of these protein-ligand interactions. Another possibility is the target function employed in the machine learning modeling. IC_{50} is notoriously noisier than K_i [66]. We should expect a more reliable model using pK_i as a target function.

Case Study 04: Application of a Machine Learning Model to CDK2 Structures with K_i

In this study, we applied the machine learning model generated in case study 03 (CDK2_Ki_DecisionTreeRegressorCV model) to predict binding affinity for all structures in a test set used in the development of Taba [35]. Figures 15A and 15B present a map of selected metrics (r^2, ρ, and EDOME) and the scattering plot (Predicted pK_i vs. Experimental pK_i) for all structures in the Taba test set [35], respectively. Figure 15A shows the predictive performance for all regression methods used to generate models that predict pK_i for CDK2 with SAnDReS and Taba scoring function (supplementary material 12).

The CDK2_Ki_DecisionTreeRegressorCV model has predictive performance to estimate pK_i (r^2 = 0.157647, ρ = 0.655468, RMSE = 1.5449, and EDOME = 2.28673) inferior to the Taba scoring function (r^2 = 0.657051, ρ = 0.766667, RMSE = 1.52644, and EDOME = 2.3505) for almost all

metrics. The Taba scoring function shows the best metrics for r^2 and ρ. They have almost the same RMSE. On the other hand, the CDK2_Ki_DecisionTreeRegressorCV model has the best performance considering EDOME (2.28673 against 2.3505 for Taba) and MAE metrics (1.13555 against 1.33171 for Taba). Since the EDOME metric has more dimensions (RMSE, MAE, R^2, r^2, and ρ), we may say that the CDK2_Ki_DecisionTreeRegressorCV model has a slightly superior performance in predicting pK_i for CDK2. In case study 06, we carried out a complete benchmark study comparing the predictive performance of a SAnDReS-generated model against 36 external scoring functions using the CASF-2016 test set for pK_i.

Case Study 05: AlphaFold Model of CDK19 with IC_{50} Data

Here, we focus on the AlphaFold model of CDK19 complexed with inhibitors obtained through docking simulation. We generated the complex structures using AutoDock Vina integrated into SAnDReS. We employed these docked structures to create our machine-learning models with SAnDReS. Figures 16A and 16B show a map of selected metrics (r^2, ρ, and EDOME) and the scattering plot (Predicted vs. Experimental values) for all protein-ligand complexes in the test set (supplementary material 13). We also took the EDOME as a selection criterion, the best model is ExtraTreeRegressor with the following features: C, O, Hydrophobic, Gauss 2, Hydrogen, Torsional, Gauss 1, S. This machine learning model (r^2 = 0.378834, ρ = 0.601846, RMSE = 0.878856, and EDOME = 1.48754) shows superior metrics compared with the Vina Affinity function (r^2 = 0.00113771, ρ = -0.170391, RMSE = 15.5999, and EDOME = 290.232) for the test set. This machine-learning model is available on GitHub (https://github.com/azevedolab/sandres/blob/master/CDK19_IC50_ExtraTreeRegressor.zip). As observed for case study 03, there is a substantial improvement in using docked structures to generate machine learning models.

Case Study 06: CASF-2016 with K_i Data

SAnDReS is an open-access computational tool to generate machine-learning models for targeted protein systems. Nevertheless, due to the easiness of use and flexibility of SAnDReS, we proposed an ultimate test: a universal scoring function to predict K_i. Our goal with this last case study is to develop a scoring function without targeting any protein system. We employed a diverse dataset with 991 crystallographic structures with the filtering defined in the methods section (supplementary material 14). To test the predictive performance of our machine learning models generated using SAnDReS, we predicted pK_i of the CASF-2016 test set [47].

The original CASF-2016 dataset has structures with both K_d and K_i. One way to represent the relationship between K_i and K_d is using the Cheng-Prusoff equation [67], where we may determine K_i from K_d and IC_{50}. But they are not the same (for instance, the inhibition of CDK2 by Roscovitine) [60] indicates different values for K_i and K_d (K_i = 250 nM and K_d in the range of 2900 nM to 3400 nM). In summary, the distinction between them is that K_d is a more general term. K_i also represents a dissociation constant, but more narrowly for the binding of an inhibitor to an enzyme. That is a small-molecule inhibitor whose binding decreases the catalytic activity of a target enzyme. The value of K_i depends on the specific kinetic mechanism of the enzyme inhibition (e.g., competitive and uncompetitive inhibitors) [68]. To develop a more realistic machine learning model, we focused on K_i. We suggest that any scoring function should follow this path: not using mixed datasets with K_i and K_d.

We used structures of the CASF-2016 test set with K_i data. The CASF-2016 has 285 structures combining both K_i and K_d data. Specifically for Ki, we have 175 structures. We further filtered the CASF-2016 test set to eliminate structures with ligands showing occupancy factors below 1.0 and those with weak electron density for part of the ligand structures. The final test set has 155 structures (supplementary material 15). We name this dataset as the CASF-2016 K_i test set.

SAnDReS employed high-resolution crystallographic structures as the training set. We created our universal scoring function with 14 independent variables out of 16 features. SAnDReS built 6,480 scoring functions ($54 \times C_{16,14} = 54 \times 120 = 6{,}480$). Figures 17A and 17B present the predictive performance using defined metrics (r^2, ρ, and EDOME) and the scattering plot (Predicted pK_i vs. Experimental pK_i) for all structures in the test set (CASF-2016 K_i test set). Taking the lowest EDOME among SAnDReS models, we selected the ExtraTreesRegressorCV model (supplementary material 15). We named this model KiETR_F14 (a model to calculate K_i using Extra Trees Regressor with 14 variables). KiETR_F14 has the following features: Gauss 2, C, Gauss 1, Hydrophobic, N, Torsional, B-factor ratio (Ligand/Receptor), Torsions, S, Receptor B-factor(A2), Q, Average Q, Hydrogen, O. This machine learning model has following metrics: $r^2 = 0.737876$, $\rho = 0.854758$, RMSE = 1.23658, and EDOME = 1.65883.

In Figure 17A, we show the performance of 107 scoring functions and features (supplementary material 15). Among the scoring functions, we have 54 regression models generated with SAnDReS, 34 scoring functions available for the CASF-2016 Ki test set, 16 features, and three additional external machine learning models (K_{DEEP} [48], CSM-lig [49], and $\Delta_{Vina}RF_{20}$ [50]). All metrics calculated here used the same 155 structures of the CASF-2016 K_i test set. KiETR_F14 model shows superior predictive performance compared with classical scoring functions for test set structures. Taking EDOME as a criterion, the top-ranked models are K_{DEEP}, $\Delta_{Vina}RF_{20}$, and KiETR_F14). On the other hand, using r^2 to evaluate all models, the KiETR_F14 outperforms all other models. Strictly comparing them, K_{DEEP} and KiETR_F14 have similar performance, followed by $\Delta_{Vina}RF_{20}$.

Future Studies

We intend to keep incorporating the new developments of machine learning regression methods. These new developments of SAnDReS will focus mainly on ensemble methods [29, 69, 70]. Some of these ensemble methods are already among the 54 techniques available in SAnDReS (e.g., Voting Regressor). These novel amendments will combine machine learning regressors and return the average predicted binding affinities. Also, we expect a rise in the number of SAnDReS users. The increasing number of validated machine-learning models will provide the raw data to create a database of scoring functions. We are developing a scoring function database to have a set of ready-to-use models for a wide range of protein targets.

Conclusion

SAnDReS is an open-source computational to explore the SFS concept based on advanced machine-learning methods. This new version is a different computational tool that allows us to computationally navigate through the SFS, finding an adequate machine-learning model for a protein system of interest. We explained SAnDReS functioning and described six new case studies highlighting the different functionalities. Four case studies focused on CDK2, using

crystallographic and pose positions to train our models. At least for CDK2, the machine learning models based on docked structures showed superior predictive performance, most likely due to the use of energy-minimized positions of ligands resulting from docking simulations. Also, we developed a model using the atomic coordinates of CDK2 generated using AlphaFold. This study of a SAnDReS-generated model focused on an AlphaFold structure paves the way to expand the application of targeted scoring functions to deep-learning models of protein structures. Finally, we highlight the adequacy of SAnDReS to generate a universal scoring function with predictive performance superior to classical scoring functions and at least the same performance as other machine-learning scoring functions (K_{DEEP}, CSM-lig, and $\Delta_{Vina}RF_{20}$).

Acknowledgments

WFA is a researcher for CNPq (Brazil) (Process Number: 309029/2018-0). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) – Finance Code 001. OT, NB and VP thank Russian Foundation for Basic Research grant No. 20-04-60285 for the support.

Keywords: scoring function space; machine learning; protein-ligand interactions; binding affinity; drug discovery.

References

1. Veit-Acosta M, de Azevedo Junior WF (2021) The Impact of Crystallographic Data for the Development of Machine Learning Models to Predict Protein-Ligand Binding Affinity. Curr Med Chem 28(34):7006–7022.
2. Yang C, Chen EA, Zhang Y (2022) Protein-Ligand Docking in the Machine-Learning Era. Molecules 27(14):4568.
3. Takada Y, Fujita M, Takada YK (2023) Virtual Screening of Protein Data Bank via Docking Simulation Identified the Role of Integrins in Growth Factor Signaling, the Allosteric Activation of Integrins, and P-Selectin as a New Integrin Ligand. Cells 2023 12(18):2265.
4. Potlitz F, Link A, Schulig L (2023) Advances in the discovery of new chemotypes through ultra-large library docking. Expert Opin Drug Discov 18(3):303–313.
5. Sulimov VB, Kutov DC, Sulimov AV (2019) Advances in Docking. Curr Med Chem. 26(42):7555–7580.
6. Jiang H, Wang J, Cong W, Huang Y, Ramezani M, Sarma A, Dokholyan NV, Mahdavi M, Kandemir MT (2022) Predicting Protein-Ligand Docking Structure with Graph Neural Network. J Chem Inf Model 62(12):2923–2932.
7. Morris GM, Huey R, Lindstrom W, Sanner MF, Belew RK, Goodsell DS, Olson AJ (2009) AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. J Comput Chem 30(16):2785–2791.
8. Thomsen R, Christensen MH (2006) MolDock: a new technique for high-accuracy molecular docking. J Med Chem 49(11):3315–3321.
9. Trott O, Olson AJ (2010) AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. J Comput Chem 31(2):455–461.
10. Eberhardt J, Santos-Martins D, Tillack AF, Forli S (2021) AutoDock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings. J Chem Inf Model 61(8):3891–3898.
11. Wójcikowski M, Siedlecki P, Ballester PJ (2019) Building Machine-Learning Scoring Functions for Structure-Based Prediction of Intermolecular Binding Affinity. Methods Mol Biol 2053:1–12.
12. Seifert MH (2009) Targeted scoring functions for virtual screening. Drug Discov Today 14(11-12):562–569.
13. Ross GA, Morris GM, Biggin PC (2013) One Size Does Not Fit All: The Limits of Structure-Based Models in Drug Discovery. J Chem Theory Comput 9(9):4266–4274.
14. Heck GS, Pintro VO, Pereira RR, de Ávila MB, Levin NMB, de Azevedo WF (2017) Supervised Machine Learning Methods Applied to Predict Ligand- Binding Affinity. Curr Med Chem 24(23):2459–2470.
15. Bitencourt-Ferreira G, de Azevedo WF Jr. (2019) Exploring the Scoring Function Space. Methods Mol Biol 2053:275–281.
16. Veríssimo GC, Serafim MSM, Kronenberger T, Ferreira RS, Honorio KM, Maltarollo VG (2022) Designing drugs when there is low data availability: one-shot learning and other approaches to face the issues of a long-term concern. Expert Opin Drug Discov 17(9):929–947.
17. Westerhoff HV, Palsson BO (2004) The evolution of molecular biology into systems biology. Nat Biotechnol 22(10):1249–1252.
18. Limbu S, Dakshanamurthy S (2022) A New Hybrid Neural Network Deep Learning Method for Protein-Ligand Binding Affinity Prediction and De Novo Drug Design. Int J Mol Sci 23(22):13912.
19. Hahn DF, Bayly CI, Macdonald HEB, Chodera JD, Mey ASJS, Mobley DL, Benito LP, Schindler CEM, Tresadern G, Warren GL (2022) Best practices for constructing,

preparing, and evaluating protein-ligand binding affinity benchmarks. Living J Comput Mol Sci 4(1):1497.
20. Scott OB, Gu J, Chan AWE (2022) Classification of Protein-Binding Sites Using a Spherical Convolutional Neural Network. J Chem Inf Model 2022 62(22):5383–5396.
21. Sauer S, Matter H, Hessler G, Grebner C (2022) Optimizing interactions to protein binding sites by integrating docking-scoring strategies into generative AI methods. Front Chem 10:1012507.
22. Bieniek MK, Cree B, Pirie R, Horton JT, Tatum NJ, Cole DJ (2022) An open-source molecular builder and free energy preparation workflow. Commun Chem 5(1):136.
23. Mudedla SK, Braka A, Wu S (2022) Quantum-based machine learning and AI models to generate force field parameters for drug-like small molecules. Front Mol Biosci 9:1002535.
24. Murugan NA, Muvva C, Jeyarajpandian C, Jeyakanthan J, Subramanian V (2020) Performance of Force-Field- and Machine Learning-Based Scoring Functions in Ranking MAO-B Protein-Inhibitor Complexes in Relevance to Developing Parkinson's Therapeutics. Int J Mol Sci 21(20):7648.
25. Ballester PJ, Schreyer A, Blundell TL (2014) Does a more precise chemical description of protein-ligand complexes lead to more accurate prediction of binding affinity? J Chem Inf Model 54(3):944–955.
26. Bohacek RS, McMartin C, Guida WC (1996) The art and practice of structure-based drug design: a molecular modeling perspective. Med Res Rev 16(1):3–50.
27. Smith JM (1970) Natural selection and the concept of a protein space. Nature 225(5232):563–564.
28. Hou J, Jun SR, Zhang C, Kim SH (2005) Global mapping of the protein structure space and application in structure-based inference of protein function. Proc Natl Acad Sci U S A 102(10):3651-3656.
29. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Verplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: Machine Learning in Python. J Mach Learn Res 12:2825–2830.
30. Walsh I, Fishman D, Garcia-Gasulla D, Titma T, Pollastri G; ELIXIR Machine Learning Focus Group; Harrow J, Psomopoulos FE, Tosatto SCE (2021) DOME: recommendations for supervised machine learning validation in biology. Nat Methods 18(10):1122–1127.
31. Halevy A, Norvig P, Pereira F (2009) The Unreasonable Effectiveness of Data. IEEE Intelligent Systems 24(2):8–12.
32. Ballester PJ, Mitchell JB (2010) A machine learning approach to predicting protein-ligand binding affinity with applications to molecular docking. Bioinformatics 26(9):1169–1175.
33. Ain QU, Aleksandrova A, Roessler FD, Ballester PJ (2015) Machine-learning scoring functions to improve structure-based binding affinity prediction and virtual screening. Wiley Interdiscip Rev Comput Mol Sci 5(6):405–424.
34. Xavier MM, Heck GS, Avila MB, Levin NMB, Pintro VO, Carvalho NL, Azevedo WF (2016) SAnDReS a Computational Tool for Statistical Analysis of Docking Results and Development of Scoring Functions. Comb Chem High Throughput Screen 19(10):801–812.
35. Da Silva AD, Bitencourt-Ferreira G, de Azevedo WF Jr (2020) Taba: A Tool to Analyze the Binding Affinity. J Comput Chem. 2020 41(1):69–73.
36. Veit-Acosta M, de Azevedo Junior WF (2022) Computational Prediction of Binding Affinity for CDK2-ligand Complexes. A Protein Target for Cancer Drug Discovery. Curr Med Chem 29(14):2438–2455.
37. Evans GB, Schramm VL, Tyler PC (2015) The Immucillins: Design, Synthesis, and Application of Transition- State Analogues. Curr Med Chem 22(34):3897–3909.

38. Bitencourt-Ferreira G, Villarreal MA, Quiroga R, Biziukova N, Poroikov V, Tarasova O, de Azevedo Junior WF (2023) Exploring Scoring Function Space: Developing Computational Models for Drug Discovery. Curr Med Chem doi: 10.2174/0929867330666230321103731.
39. Gilson, M.K.; Liu, T.; Baitaluk, M.; Nicola, G.; Hwang, L.; Chong, J. BindingDB in 2015: A public database for medicinal chemistry, computational chemistry and systems pharmacology. Nucleic Acids Res., 2016, 44(D1), D1045-D1053.
40. R. Wang, X. Fang, Y. Lu, S. Wang, J. Med. Chem. 2004, 47, 2977.
41. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, Tunyasuvunakool K, Bates R, Žídek A, Potapenko A, Bridgland A, Meyer C, Kohl SAA, Ballard AJ, Cowie A, Romera-Paredes B, Nikolov S, Jain R, Adler J, Back T, Petersen S, Reiman D, Clancy E, Zielinski M, Steinegger M, Pacholska M, Berghammer T, Bodenstein S, Silver D, Vinyals O, Senior AW, Kavukcuoglu K, Kohli P, Hassabis D. Highly accurate protein structure prediction with AlphaFold. Nature. 2021 Aug;596(7873):583-589.
42. Ravindranath PA, Forli S, Goodsell DS, Olson AJ, Sanner MF. AutoDockFR: Advances in Protein-Ligand Docking with Explicitly Specified Binding Site Flexibility. PLoS Comput Biol., 2015, 11(12), e1004586.
43. Word JM, Lovell SC, Richardson JS, Richardson DC. Asparagine and glutamine: using hydrogen atom contacts in the choice of side-chain amide orientation. J Mol Biol. 1999; 285(4):1735-47.
44. de Azevedo WF Jr, Canduri F, dos Santos DM, Pereira JH, Bertacine Dias MV, Silva RG, Mendes MA, Basso LA, Palma MS, Santos DS. Crystal structure of human PNP complexed with guanine. Biochem Biophys Res Commun. 2003; 312(3):767-72.
45. Ballante F, Marshall GR. An Automated Strategy for Binding-Pose Selection and Docking Assessment in Structure-Based Drug Design. J Chem Inf Model. 2016; 56(1):54-72.
46. Coelho LP, Richert W. (2015) Building Machine Learning Systems with Python. 2nd ed. Packt Publishing Ltd. Birmingham UK, 301 pp.
47. Su M, Yang Q, Du Y, Feng G, Liu Z, Li Y, Wang R. Comparative Assessment of Scoring Functions: The CASF-2016 Update. J Chem Inf Model. 2019 Feb 25;59(2):895-913.
48. Jiménez J, Škalič M, Martínez-Rosell G, De Fabritiis G. K_{DEEP}: Protein-Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks. J Chem Inf Model. 2018 Feb 26;58(2):287-296.
49. Pires DE, Ascher DB. CSM-lig: a web server for assessing and comparing protein-small molecule affinities. Nucleic Acids Res. 2016 Jul 8;44(W1):W557-61.
50. Wang C, Zhang Y. Improving scoring-docking-screening powers of protein-ligand scoring functions using random forest. J Comput Chem. 2017 Jan 30;38(3):169-177.
51. Kawanishi N, Sugimoto T, Shibata J, Nakamura K, Masutani K, Ikuta M, Hirai H. Structure-based drug design of a highly potent CDK1,2,4,6 inhibitor with novel macrocyclic quinoxalin-2-one structure. Bioorg Med Chem Lett. 2006 Oct 1;16(19):5122-6.
52. Volkart PA, Bitencourt-Ferreira G, Souto AA, de Azevedo WF. Cyclin-Dependent Kinase 2 in Cellular Senescence and Cancer. A Structural and Functional Review. Curr Drug Targets. 2019;20(7):716-726.
53. De SK. Pyrazolo[4,3-H] quinazolines as Cyclin-dependent Kinase Inhibitors for Treating Cancer. Curr Med Chem. 2023 May 25. doi: 10.2174/0929867330666230525160458.
54. Dai D, Yu J, Huang T, Li Y, Wang Z, Yang S, Li S, Li Y, Gou W, Li D, Hou W, Fan S, Li Y, Zhao Y. PET imaging of new target CDK19 in prostate cancer. Eur J Nucl Med Mol Imaging. 2023 Sep;50(11):3452-3464.
55. Mao M, Zheng X, Sheng Y, Chai J, Ding H. Evodiamine inhibits malignant progression of ovarian cancer cells by regulating lncRNA-NEAT1/miR-152-3p/CDK19 axis. Chem Biol Drug Des. 2023 Jul;102(1):101-114.
56. Ray D, Kiyokawa H. CDC25A phosphatase: a rate-limiting oncogene that determines genomic stability. Cancer Res. 2008 Mar 1;68(5):1251-3.

57. Chytil A, Waltner-Law M, West R, Friedman D, Aakre M, Barker D, Law B. Construction of a cyclin D1-Cdk2 fusion protein to model the biological functions of cyclin D1-Cdk2 complexes. J Biol Chem. 2004 Nov 12;279(46):47688-98.
58. Goldbeter A. Oscillatory enzyme reactions and Michaelis-Menten kinetics. FEBS Lett. 2013 Sep 2;587(17):2778-84.
59. De Azevedo WF Jr, Mueller-Dieckmann HJ, Schulze-Gahmen U, Worland PJ, Sausville E, Kim SH. Structural basis for specificity and potency of a flavonoid inhibitor of human CDK2, a cell cycle kinase. Proc Natl Acad Sci U S A. 1996 Apr 2;93(7):2735-40.
60. De Azevedo WF, Leclerc S, Meijer L, Havlicek L, Strnad M, Kim SH. Inhibition of cyclin-dependent kinases by purine analogues: crystal structure of human cdk2 complexed with roscovitine. Eur J Biochem. 1997 Jan 15;243(1-2):518-26.
61. Fuchs JE, Waldner BJ, Huber RG, von Grafenstein S, Kramer C, Liedl KR. Independent Metrics for Protein Backbone and Side-Chain Flexibility: Time Scales and Effects of Ligand Binding. J Chem Theory Comput. 2015 Mar 10;11(3):851-60.
62. Bitencourt-Ferreira G, de Azevedo WF Jr. Molecular Dynamics Simulations with NAMD2. Methods Mol Biol. 2019; 2053:109-124.
63. de Azevedo WF Jr. Molecular dynamics simulations of protein targets identified in Mycobacterium tuberculosis. Curr Med Chem. 2011;18(9):1353-66.
64. Zuo K, Capelli R, Rossetti G, Nechushtai R, Carloni P. Predictions of the Poses and Affinity of a Ligand over the Entire Surface of a NEET Protein: The Case of Human MitoNEET. J Chem Inf Model. 2023 Jan 23;63(2):643-654.
65. Berman HM, Vallat B, Lawson CL. The data universe of structural biology. IUCrJ. 2020 May 28;7(Pt 4):630-638.
66. de Ávila MB, Xavier MM, Pintro VO, de Azevedo WF Jr. Supervised machine learning techniques to predict binding affinity. A study for cyclin-dependent kinase 2. Biochem Biophys Res Commun. 2017 Dec 9; 494(1-2):305-310.
67. Cheng Y, Prusoff WH. Relationship between the inhibition constant (K1) and the concentration of inhibitor which causes 50 per cent inhibition (I50) of an enzymatic reaction. Biochem Pharmacol. 1973 Dec 1;22(23):3099-108.
68. Stoddart LA, White CW, Nguyen K, Hill SJ, Pfleger KD. Fluorescence- and bioluminescence-based approaches to study GPCR ligand binding. Br J Pharmacol. 2016 Oct;173(20):3028-37.
69. Sinha K, Ghosh J, Sil PC. Machine Learning in Drug Metabolism Study. Curr Drug Metab. 2022 Dec 27. doi: 10.2174/1389200224666221227094144.
70. Jung Y, Geng C, Bonvin AMJJ, Xue LC, Honavar VG. MetaScore: A Novel Machine-Learning-Based Approach to Improve Traditional Scoring Functions for Scoring Protein-Protein Docking Conformations. Biomolecules. 2023 Jan 6;13(1):121.

Comments:

Dear Walter,

First of all, thank you for involving me in this project.

I read the draft of the manuscript several times. The computational work behind this draft is very hard and challenging. The idea is original, and the final goal of the work is clear, but in my humble opinion the draft is not too easy to read and follow the workflow. I think that it needs several corrections before submitting it.

Following some suggestions, I hope can be useful for the better comprehension of the text:

1) Introduction section, it would be useful to add one figure and some equations equations to explain better the concept of scoring functions. This could make the text easier to read for non-expert researchers in the field or people approaching docking and scoring functions.
2) It would be useful to explain the difference between knowledge-based, force field-based, and empirical scoring functions. Related to this one, when you write of universal or classical scoring functions, do you refer to empirical scoring function? Don't you?
3) Hi (ui, fi) are defined in a not so clear way. The ui parameters and fi instance's vectors should be better defined.
4) Sometimes the text is redundant in the introduction, material and methods, and results and discussion. This makes reading the draft not so smooth
5) Figure 1 should be improved to allow a better ocomprehension of SFS.
6) The limit of this kind of approach is the knowledge of experimental data. This should be underlined.
7) The statistical metrics used in DOME should be reported not only cited to help the comprehension of the workflow.
8) Figure 2: captions do not explain the workflow
9) The validation of the docking protocol: why did you use the nucleoside phosphorylase?
10) The definition of the indipendent varaiables, the descriptors used are not clear in each case study. For instance, C14,8, What the meaning of
11) What kind of descriptors have been used? The number? Are they constitutional 1D descriptors? In any case, could these kinds of descriptors be too simple to describe the ligands.
12) Page 8, VS paragraph. I am a bit confused when you use the term Virtual screening. Usually, I intend the screening of unknown molecules on a computational model (Pharmacophore model, docking model obtained by the validation of the scoring functions) I am sorry, but it is difficult to understand for me as it is written at this moment
13) Machine learning for modeling, please improve it
14) I suggest a more ordered definition of the case study
15) Case 02: why did you conserve the previously generated model? What the meaning of VS in this workflow
16) Why the use of 2DS1?
17) Why in Case 02 is there no model generation and in Case 03 yes?
18) Case 01: the predictive performance is quite low, could it be an issue?
19) Figures of the workflow are not so clear and not well-explained in the captions.

20) I suggest adding a Table to compare the predictive performance of all the case studies, and a cumulative figure of EDome plots.
21) As I have understood, the use of Ki gives more predictive performance than IC50, and crystal structure are more reliable of docking poses. It should be stressed.
22) What is TABA? A citation is not enoughin my humble opinion.

I understand the hard work you do to put your idea of SFS in practice creating SanDres, and how difficult it is to explain what is behind a computational approach.

I hope these suggestions could be useful to improve the manuscript.

I have just the regret that I have not the time to test Sandres with a dataset of mine. Maybe I would have understood better all the workflow. I hope to make this in the next future and involve you in a new project.