ChemFlow_py: A Flexible Toolkit for Docking and Rescoring

Luca Monari
Institut de Chimie de Strasbourg, UMR7177, CNRS, Université de Strasbourg
Katia Galentino
Institut de Chimie de Strasbourg, UMR7177, CNRS, Université de Strasbourg
Marco Cecchini (mcecchini@unistra.fr)
Institut de Chimie de Strasbourg, UMR7177, CNRS, Université de Strasbourg

Research Article

Keywords:

Posted Date: June 13th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3035134/v1

Version of Record: A version of this preprint was published at Journal of Computer-Aided Molecular Design on August 24th, 2023. See the published version at https://doi.org/10.1007/s10822-023-00527-z.

Abstract

The design of accurate virtual screening tools is an open challenge in drug discovery. Several structure-based methods have been developed at different levels of approximation. Among them, molecular docking is an established technique with high efficiency, but typically low accuracy. Moreover, docking performances are known to be target-dependent, which makes the choice of docking program and corresponding scoring function critical when approaching a new protein target. To compare the performances of different docking protocols, we developed ChemFlow_py, an automated tool to perform docking and rescoring. Using four protein systems extracted from DUD-E with 100 known active compounds and 3000 decoys per target, we compared the performances of several rescoring strategies including consensus scoring. We found that the average docking results can be improved by consensus ranking, which emphasizes the relevance of consensus scoring when little or no chemical information is available for a given target. ChemFlow_py is a free toolkit to optimize the performances of virtual high-throughput screening. The software is publicly available at https://github.com/IFMlab/ChemFlow_py.

Introduction

The development of a new drug is a complex and time-consuming process with costs exceeding billions of dollars [1]. To improve the efficiency of drug discovery, many computational tools have been developed to help identify new drug candidates [2]. Among them, virtual screening based on high-throughput docking has become popular to search for hit compounds [3]. Although approximated, this technique is advantageous over more rigorous approaches due to its efficiency, which allows for covering a large portion of the chemical space in a short amount of time [4].

Starting with one or more high-resolution structures of a pharmacological target (e.g., a protein molecule or a nucleic acid) in complex with a known modulator, molecular docking aims at creating virtual protein-ligand complexes, starting from the 3D structure of the target and of a set of ligands under investigation [3]. Molecular docking represents the initial step of a vHTS campaign, especially in structure based drug design (SBDD) [2], [5]. Using a search algorithm, many conformations of the ligand are generated automatically in the binding site of the protein and their fitness is evaluated by means of a simplified energy function or score. The conformational search is usually carried out by a stochastic algorithm, while pose ranking relies on physics-based, knowledge-based, or empirical scoring [3]. Recently, machine learning methods have been introduced to improve the docking scores, e.g., to account for the flexibility of the ligand and the protein [6].

To preserve computational efficiency, docking methods introduce approximations. In many cases, for instance, solvation effects and/or the entropy loss on binding are neglected [4]. These contributions, however, are critical for a rigorous evaluation of the ligand-binding affinity, making docking predictions inaccurate. To improve the final ranking of the compounds, the rescoring of a limited number of docking poses is often envisaged [7]–[10].

This involves: i. re-ranking of the docking poses by more accurate scoring functions that account for solvation-free energy contributions [11] and/or entropy corrections [12]; or ii. the combination of docking results from different programs for ranking by consensus scoring [9]. The latter was shown to increase the success rate of virtual screening at a relatively low additional cost, e.g., improving enrichment factors and/or the area under the Receiver Operating Characteristic (ROC) curve [13]–[15], and has become quite popular. However, the origin of the benefit(s) remains to be understood and what docking programs or scoring functions should be used for the consensus is not yet known. In addition, a wide range of docking programs and rescoring methodologies exist [9], [16], each of them requiring knowledge and expertise, which makes consensus docking not straightforward for non-expert users.

To ease the implementation of consensus scoring for virtual screening, we developed Chemflow_py, a Python interface for docking and rescoring. Chemflow_py provides straightforward access to five popular docking programs, eight scoring functions, seven consensus methods, and free energy rescoring powered by Amber or Gromacs [17]. The code is based on customizable Python modules, so that other docking programs can be easily integrated. In the following, we present a benchmark of Chemflow_py to analyze the performance of docking with consensus ranking on four different protein targets extracted from Directory of Useful Decoys Enhanced (DUD-E) [18]. Chemflow_py is a development of ChemFlow, our 'in house' software, which was designed to bridge the gap between 2D chemical libraries and protein-ligand binding free energies calculations [19]. In this study, the focus is on docking and no free-energy rescoring was considered.

Method:

Datasets

Four biological targets and corresponding ligands were retrieved from the Directory of Useful Decoys Enhanced [18]. We chose proteins belonging to different protein classes: the β1 adrenergic receptor (ADRB1, PDB: 2vt4), the cyclin-dependent kinase 2 (CDK2, PDB: 1h00), the human immunodeficiency virus type 1 protease (HIVPR, PDB: 1xl2), and HMG-CoA reductase (HMDH, PDB: 3ccw). DUD-E provides the PDB structure of the receptor along with MOL2 structures for experimentally active compounds, decoys and the crystal structure of (at least) one ligand co-crystallized with the protein [18]. For each receptor, we randomly selected 100 known actives and 3000 decoys.

Docking Programs

Five non-commercial docking programs are currently supported in ChemFlow_py: Autodock4 [20], Autodock Vina [21], PLANTS [22], Smina [23] and QVina2.1 [24]. In this study, we analyzed the performances of **Autodock4**, one of the first open-source docking programs that uses a Lamarckian Genetic Algorithm and an empirical free-energy scoring function as defaults; **Autodock Vina** that relies on an Iterated Local Search global optimizer and uses a simpler scoring function for a more efficient conformational search [21]; **PLANTS**, the first docking program based on the ant colony optimization algorithm that supports three empirical scoring functions, i.e. chemplp, plp and plp95 [25]; and **Smina**, a

fork of Autodock Vina designed to support custom scoring functions and improved energy and flexibility features [23].

Rescoring

For rescoring, only the best-fitting docking pose per ligand was considered and its fitness re-evaluated using a different scoring function. For this purpose, eight scoring functions were considered: autodock [26], chemplp [25], dkoes_fast, dkoes_scoring, plp [25], plp95, vina [21], and vinardo [27]. Essentially, PLANTS was used for rescoring with chemplp, plp, plp95, Smina for rescoring with vina, vinardo, dkoes_fast and dkoes_scoring, and Autodock for rescoring with its native scoring function.

Consensus

In consensus docking, a new score is assigned to each ligand by combining scores or rankings from different docking programs. Many consensus protocols have been developed to improve docking performances. In this work, we used seven consensus ranking methods as described in Ref. [9]. These methods can be classified depending on whether the consensus is based on the score or the rank.

A score-based method relies on the combination of scores obtained by docking. Those implemented in ChemFlow_py are Rank by Number (RbN), Auto-Scaled Score (ASS), Auto-Normalized Score (ANS) and Z-Score. RbN consists in averaging the score by each docking program per ligand. Since the scores of different docking programs may have significantly different magnitudes, new methods like ANS and ASS, which work with normalized docking scores, were introduced. The former divides the docking scores by the maximum value, the latter normalizes them between 0 and 1. A more sophisticated scaling is performed with Z-score: the docking results are normalized to variance units and centered according to the mean value [28].

Consensus protocols based on ranking do not rely on the docking score but only on the position of the ligand in the final ranking. The rank-based methods implemented in ChemFlow_py are Rank by Rank (RbR), Rank by Vote (RbV) and Exponential Consensus Ranking (ECR). The first method computes an average of the ranking positions per ligand. It is equivalent to Rank by Number with scores substituted by the rank order. Rank by Vote requires fixing a threshold in the ranking (e.g., top 5% of the dataset). If the ligand is positioned above this value, it gets a vote (+ 1), and the votes are summed up for all rankings. This approach elicits ligands with high ranks in all docking programs. ERC assigns to each ligand an exponential score based on the ligand ranking and a normalizing parameter sigma (σ). As a rule of thumb, sigma is the number of molecules to analyze. Finally, the exponential scores are summed up and normalized over sigma. In our work, we set both the RBV threshold and sigma value to 5% of the dataset (155 molecules).

In all consensus rankings, only the best-fitting docking pose per ligand was used according to the merging and shrinking procedure described in Ref. [29].

Metrics

Area Under the Curve (AUC): The Receiver Operating Characteristic (ROC) curve is used to evaluate the performance of a classification method. In drug discovery, ROC curves are used to compare the performances of different models and evaluate how good they are in discriminating between true positives (active compounds) and false positives (decoys). The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various fractions of the dataset. The area under the curve (AUC) provides a measure of the classification performance: an AUC close to 1 indicates a perfect classifier, and an AUC close to 0.5 indicates a random classifier [30].

Enrichment Factor (EF)

the enrichment factor provides another measure of the classification performance. The EF at a given fraction of the dataset, i.e., at 1%, indicates if the fraction of active compounds in the top 1% of the ranked dataset is higher or lower than the fraction of hits over the full dataset. It is calculated as

$$EFx\% = \frac{Hits\left(x\%\right)}{N\left(x\%\right)} \cdot \frac{N\left(tot\right)}{Hits\left(tot\right)}$$

where $N\left(tot\right)$ and $N\left(x\%\right)$ indicate the number of active compounds in the dataset and its x% top-ranked fraction. An EF of 10 indicates that the probability of finding hits in the top-ranked x% fraction is ten-fold increased. Viceversa, an EF of 1 or 0 indicates that the classification method does not improve the hit rate or that no active compound could be found in the top-ranked x%.

Method Score (MS)

The Method Score (MS) is a metric to compare the performances of rescoring or consensus scoring methods with the average docking performance. For a given protein target, if the EF1% of a given rescoring method is higher than the average EF1% from docking, its MS value is increased by one (+ 1). Vice versa, if the consensus EF1% is lower, the MS is decreased by one (-1). The MS remains unchanged if the two EF1% are the same. Since the docking performances are typically target-dependent, the MS value indicates whether a given rescoring strategy outperforms or underperforms the average docking. Considering our benchmark made of four targets, an MS of + 4 indicates better performance for each dataset, 0 implies no average improvement, and − 4 indicates that the docking average systematically leads to better results.

Implementation

ChemFlow_py is a software conceived to improve the outcome of vHTS workflow by running and postprocessing docking simulations with simple commands. It contains default settings for each docking program, which are ready for use by non-experts, but it also allows customizations for advanced users. The implementation of different docking programs often requires going through time-consuming ligand preparations and/or post processing. ChemFlow_py efficiently automatizes these steps, so that the user can focus on the workflow rather than its implementation.

Chemflow_py is an evolution of ChemFlow [19]. ChemFlow_py maintains the original features of ChemFlow, i.e., support for Vina, Smina, Plants and Qvina. In addition, it supports Autodock and Gromacs MD simulations. While ChemFlow is written in bash, ChemFlow_py was developed in Python. ChemFlow_py is based on Python modules, which allows one to integrate new docking and/or rescoring programs following a module template. Furthermore, it can be used from the command line or imported into a Python script, to build a completely customizable workflow. Another relevant feature in ChemFlow_py is multiprocessing on multiple compute nodes, e.g., when running on a cluster. Last, ChemFlow_py supports seven consensus ranking methods (see Methods). The average execution time per protocol is shown in Table 1, using a CPU Intel(R) Core(TM) i7-8700K (3.70GH) on a ligands dataset with an average molecular weight of 375.58 g/mol. Docking is the step which requires more execution time, while all the consensus methodologies are extremely fast. Rescoring requires an intermediate amount of time due to a short optimization of the structure according to the new scoring function. Even though performing consensus or rescoring is fast, docking must be run first, which remains the bottleneck of the pipeline.

Table 1
Average execution time per ligand of different methods.

VS step	Program/method	Time (sec)
Autodok	Autodock	51.2
	Plants	6.79
	Smina	7.47
	Vina	12.9
Scoring function rescoring	Autodock	3.46
	Smina-vinardo	0.0230
	Plants-plp	0.554
Consensus ranking	ANS	$1.33\ 10^{-6}$
	ASS	$4.98\ 10^{-6}$
	ECR	$1.72\ 10^{-6}$
	RBN	$1.72\ 10^{-6}$
	RBR	$1.67\ 10^{-6}$
	RBV	$6.69\ 10^{-6}$
	Z-score	$2.69\ 10^{-6}$

Results

We performed docking, rescoring and consensus ranking at four protein targets (see Methods). The docking results are given in Table 2. For each docking program, two metrics were calculated: the Area Under the ROC Curve (AUC) and the Enrichment factor at 1% of the dataset (EF1%). The former quantifies the ability to prioritize active compounds over decoys, the latter is related to the probability of finding active compounds in a fraction of the dataset. Because experimental testing is expensive and time consuming, EF is generally more meaningful than AUC for drug discovery, as it is more directly related to the probability of finding hits in the fraction of compounds that can be tested experimentally. In this work, we analyzed classification performances based on EF1%, i.e. the larger the EF1%, the better the classifier. For comparison, AUC scores are given in SI.

The docking results in Table 2 do not follow a clear trend. Autodock, for instance, is best for CDK2 but it is the worst for ADRB1 and HIVPR. Smina is best-performing for ADRB1 and HMDH, but is poor for HIVPR. Although limited to four proteins, this benchmark shows that the docking performances are highly system-dependent and that there is no docking method that is better than the others.

Since large and diverse datasets of known active compounds for a protein target are rarely available before virtual screening, the performances of different rescoring or consensus scoring strategies (see *Methods*) were evaluated on the quest for more robust classification methods. To this aim, we took a statistical approach and evaluated the rescoring/consensus performances by comparing the EF1% per protein target with the average EF1% over the four docking methods. The results are shown in Table 3. By highlighting in red and blue EF1% results above and below the docking average, respectively, the results show that no method nor combination of methods is absolutely best. However, an analysis of the method score (MS), which allows to compare rescoring strategies with the docking average, provides clearer indications. In fact, albeit soring-function rescoring provides no global improvement over the average docking performances, i.e. the EF1% is higher for HIVPR and HMDH but lower for ADRB1 and CDK2 for a MS of zero, two consensus methods outperform average docking with MS of + 2 for Z-score and + 1 for RbV. Interestingly, the latter is based on ranking (it relies only on the order of the ligands classification), while Z-score relies on the docking score per ligand. The other consensus scoring strategies have MS equal zero or below. In the limit of the dataset explored, these results indicate that docking can be improved by consensus ranking. Although consensus comes at no cost per se (Table 1), being able to evaluate consensus requires collecting results from multiple docking programs, which introduces an additional cost that is approximately linear with the number of docking programs in use. In this respect, docking methods based on machine learning, such as DeepDocking [31], which decrease the computational time quite extensively, make consensus ranking more appealing.

Table 2: Docking performances evaluated by EF 1%. The results are highlighted in blue when below the docking average, and highlighted in red when above it.

EF 1%	Autodock	Vina	Smina	PLANTS	Mean EF 1%
ADRB1	2	6	14	10	8
CDK2	14	9	13	5	10.25
HIVPR	0	4	3	6	3.25
HMDH	2	3	11	0	4

Table 3: For each protein target, we calculate the average enrichment factor at 1% for docking and rescoring. Below, we added the EF 1% value obtained by each consensus method. Consensus results are highlighted in blue when below the average of docking, highlighted in red when above the average. In the end we evaluated a score for each method, assigning +1 if the EF 1% is above the average, -1 if under the average and 0 if matching it.

EF 1%	ADRB1	CDK2	HIVPR	HMDH	Method score
Docking mean	8	10.25	3.25	4	
Rescoring mean	4.04	6.25	5.32	4.57	0
ASS	6	10	5	5	0
ANS	6	10	5	4	-1
RbN	9	8	8	0	0
Z-Score	8	15	5	4	2
ECR	6	10	4	3	-2
RbR	7	9	5	3	-2
RbV	6	11	7	4	1

Discussion

The use of consensus scoring to improve the hit rate of a dataset is not a novel strategy. By performing an idealized computer experiment under certain conditions, i.e. assuming perfect conformational search and independent scoring functions, Wang et al. demonstrated that consensus ranking enhances the hit

rate when more than three scoring functions are used [32]. Subsequent work by Ericksen *et al* corroborated this conclusion showing that by combining commercial and open-source software consensus ranking outperformed any docking program [6]. Most recently, using an artificial intelligence-driven platform (Deep Docking) with five docking programs to screen for noncovalent inhibitors of SARS-CoV-2, it was shown that consensus scoring increased the EF1% five fold relative to docking [33]. However, other studies in the literature appear to reach different conclusions. For instance, using 20 protein targets from DUD-E, Masters et al found that consensus scoring based on open-source docking programs like Vina, Smina, and Idock was systematically worse than Smina [34]. Therefore, the question of whether consensus ranking should be systematically applied or not in virtual screening remains open. Moreover, different ways of doing consensus scoring exist [9] and it is not yet established which one is best for filtering drug candidates for a given target.

In this work, using only non-commercial software and a statistical approach, we found that consensus ranking outperforms docking for the classification of active compounds. The results collected on four protein targets from DUD-E confirm that docking performances are highly system dependent, such that it is impossible to know a priori which docking program and/or scoring function are best for the protein target of interest. By introducing a metric, termed Method Score (MS), that allows comparing different consensus scoring methods with the average docking performances, we found that consensus ranking based on Z-scores or votes (ranking-by-vote) outperforms any other method yielding higher enrichment factors at 1%. These results emphasize the relevance of consensus ranking when little or no chemical information is available for a given target, which is often the case in virtual screening. On the other hand, if chemical information is available, the identification of the optimal docking program and scoring strategy is absolutely critical for the success of a screening campaign. In this case, tools and workflows that allow benchmarking multiple docking programs and rescoring strategies methods in a standardized and automated manner offer a competitive advantage.

ChemFlow_py was developed to make the implementation of docking and rescoring consensus ranking straightforward. The code is written in Python3 and can be used both as a standalone or imported as a Python module. The current version of ChemFlow_py supports five open-source popular docking programs, eight scoring functions, and seven consensus scoring methods. In addition, the standardization of protocols provides access to comparing performances of different docking programs even to non-expert users. Since it is impossible to know a priori which docking method would be best for ranking compounds for the target of interest, toolkits such as ChemFlow_py are expected to increase the success rate of virtual screening.

Declarations

Software availability

ChemFlow_py is a freely available software package that can be downloaded from the following link: https://github.com/IFMlab/ChemFlow_py. This software can be used either from the command line (on

Linux or MacOS) or as a library that can be imported into a Python script. A tutorial on how to install and use ChemFlow_py is available at https://github.com/IFMlab/ChemFlow_py/blob/main/tutorial.md. The tutorial covers different aspects from docking to rescoring using the protein CDK2 from the DUD-E database.

ChemFlow_py interfaces with several programs whose availability is outlined below. Conda is an open-source package manager that is available free of charge for not-for-profit institutions. The docking software Autodock4, Autodock Vina, QVINA, and SMINA are all open-source and available free of charge. The docking program PLANTS is available to academic not-for-profit users under a specific license agreement. The Open Babel tool is open-source and available free of charge.

Acknowledgements

This work was funded by the French National Research Agency (ANR) through the Programme d'Investissement d'Avenir under contract 17-EURE- 0016 and received financial support from the Fondation pour la Recherche Medicale (Grant DBI20141231319). Computational resources and support at the high-performance computing center (Mesocentre) of the University of Strasbourg are gratefully acknowledged.

References

1. Hughes J, Rees S, Kalindjian S, Philpott eK (2011) «Principles of early drug discovery: Principles of early drug discovery», *Br. J. Pharmacol.*, vol. 162, fasc. 6, pp. 1239–1249, mar. doi: 10.1111/j.1476-5381.2010.01127.x
2. Sliwoski G, Kothiwale S, Meiler J, Lowe eEW (2014) «Computational Methods in Drug Discovery», *Pharmacol. Rev.*, vol. 66, fasc. 1, pp. 334–395, gen. doi: 10.1124/pr.112.007336
3. Stanzione F, Giangreco I, Cole eJC (2021) «Use of molecular docking computational tools in drug discovery». Progress in Medicinal Chemistry. Elsevier, pp 273–343. doi: 10.1016/bs.pmch.2021.01.004.
4. Montalvo-Acosta JJ, Cecchini eM (2016) «Computational Approaches to the Chemical Equilibrium Constant in Protein-ligand Binding», p. 13,
5. Lionta E, Spyrou G, Vassilatis D, Cournia eZ (2014) «Structure-Based Virtual Screening for Drug Discovery: Principles, Applications and Recent Advances», *Curr. Top. Med. Chem.*, vol. 14, fasc. 16, pp. 1923–1938, ott. doi: 10.2174/1568026614666140929124445
6. Crampon K, Giorkallos A, Deldossi M, Baud S, Steffenel eLA (2022) «Machine-learning methods for ligand–protein molecular docking», *Drug Discov. Today*, vol. 27, fasc. 1, pp. 151–164, gen. doi: 10.1016/j.drudis.2021.09.007
7. Majeux N, Scarsi M, Apostolakis J, Ehrhardt C, Caflisch eA (1999) «Exhaustive docking of molecular fragments with electrostatic solvation», *Proteins Struct. Funct. Genet.*, vol. 37, fasc. 1, pp. 88–105,

ott. doi: 10.1002/(SICI)1097-0134(19991001)37:1<88::AID-PROT9>3.0.CO;2-O

8. McNutt AT et al (2021) «GNINA 1.0: molecular docking with deep learning», *J. Cheminformatics*, vol. 13, fasc. 1, p. 43, dic. doi: 10.1186/s13321-021-00522-2
9. Palacio-Rodríguez K, Lans I, Cavasotto CN, Cossio eP (2019) «Exponential consensus ranking improves the outcome in docking and receptor ensemble docking». Sci Rep 9:5142. fasc. 110.1038/s41598-019-41594-3
10. Kurkinen ST, Lätti S, Pentikäinen OT, Postila ePA (2019) «Getting Docking into Shape Using Negative Image-Based Rescoring», *J. Chem. Inf. Model.*, vol. 59, fasc. 8, pp. 3584–3599, ago. doi: 10.1021/acs.jcim.9b00383
11. Launay G et al (2020) «Evaluation of CONSRANK-Like Scoring Functions for Rescoring Ensembles of Protein–Protein Docking Poses», *Front. Mol. Biosci.*, vol. 7, p. 559005, ott. doi: 10.3389/fmolb.2020.559005
12. Pereira GP, Cecchini eM (2021) «Multibasin Quasi-Harmonic Approach for the Calculation of the Configurational Entropy of Small Molecules in Solution», *J. Chem. Theory Comput.*, vol. 17, fasc. 2, pp. 1133–1142, feb. doi: 10.1021/acs.jctc.0c00978
13. Charifson PS, Corkery JJ, Murcko MA, Walters eWP (1999) «Consensus Scoring: A Method for Obtaining Improved Hit Rates from Docking Databases of Three-Dimensional Structures into Proteins», *J. Med. Chem.*, vol. 42, fasc. 25, pp. 5100–5109, dic. doi: 10.1021/jm990352k
14. Oda A, Tsuchida K, Takakura T, Yamaotsu N, Hirono eS (2006) «Comparison of Consensus Scoring Strategies for Evaluating Computational Models of Protein – Ligand Complexes», *J. Chem. Inf. Model.*, vol. 46, fasc. 1, pp. 380–391, gen. doi: 10.1021/ci050283k
15. Kukol A (2011) «Consensus virtual screening approaches to predict protein ligands», *Eur. J. Med. Chem.*, vol. 46, fasc. 9, pp. 4661–4664, set. doi: 10.1016/j.ejmech.2011.05.026
16. Pinzi e L, Rastelli G (2019) «Molecular Docking: Shifting Paradigms in Drug Discovery», *Int. J. Mol. Sci.*, vol. 20, fasc. 18, p. 4331, set. doi: 10.3390/ijms20184331
17. Abraham MJ et al (2015) «GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers», *SoftwareX*, vol. 1–2, pp. 19–25, set. doi: 10.1016/j.softx.2015.06.001
18. Mysinger MM, Carchia M, Irwin JJ (2012) e B. K. Shoichet, «Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking», *J. Med. Chem.*, vol. 55, fasc. 14, pp. 6582–6594, lug. doi: 10.1021/jm300687e
19. Barreto Gomes DE, Galentino K, Sisquellas M, Monari L, Bouysset C, Cecchini eM (2023) «ChemFlowFrom 2D Chemical Libraries to Protein–Ligand Binding Free Energies», *J. Chem. Inf. Model.*, vol. 63, fasc. 2, pp. 407–411, gen. doi: 10.1021/acs.jcim.2c00919
20. Morris GM et al (2009) «AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility», *J. Comput. Chem.*, vol. 30, fasc. 16, pp. 2785–2791, dic. doi: 10.1002/jcc.21256
21. Trott e O, Olson AJ (2009) «AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading», *J. Comput. Chem.*, p. NA-NA, doi:

10.1002/jcc.21334

22. Korb O, Stützle T, Exner eTE (2006) «PLANTS: Application of Ant Colony Optimization to Structure-Based Drug Design», in *Ant Colony Optimization and Swarm Intelligence*, M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, e T. Stützle, A c. di, in Lecture Notes in Computer Science, vol. 4150. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 247–258. doi: 10.1007/11839088_22
23. Koes DR, Baumgartner MP, Camacho eCJ (2013) «Lessons Learned in Empirical Scoring with smina from the CSAR 2011 Benchmarking Exercise», *J. Chem. Inf. Model.*, vol. 53, fasc. 8, pp. 1893–1904, ago. doi: 10.1021/ci300604z
24. Alhossary A, Handoko SD, Mu Y, Kwoh eC-K (2015) «Fast, accurate, and reliable molecular docking with QuickVina 2», *Bioinformatics*, vol. 31, fasc. 13, pp. 2214–2216, lug. doi: 10.1093/bioinformatics/btv082
25. Korb O, Stützle T, Exner eTE (2009) «Empirical Scoring Functions for Advanced Protein – Ligand Docking with PLANTS», *J. Chem. Inf. Model.*, vol. 49, fasc. 1, pp. 84–96, gen. doi: 10.1021/ci800298z
26. Guedes IA, Pereira FSS, Dardenne eLE, «Empirical Scoring Functions for Structure-Based Virtual Screening: Applications, Critical Aspects, and, Challenges» (2018) *Front. Pharmacol.*, vol. 9, p. 1089, set. doi: 10.3389/fphar.2018.01089
27. Quiroga e R, Villarreal MA, «Vinardo: A Scoring Function Based on Autodock Vina Improves Scoring, Docking, and, Screening» V (2016) *PLOS ONE*, vol. 11, fasc. 5, p. e0155183, mag. doi: 10.1371/journal.pone.0155183
28. Liu S, Fu R, Zhou L-H, Chen eS-P (2012) «Application of Consensus Scoring and Principal Component Analysis for Virtual Screening against β-Secretase (BACE-1)», *PLoS ONE*, vol. 7, fasc. 6, p. e38086, giu. doi: 10.1371/journal.pone.0038086
29. Cozzini P et al (2008) «Target Flexibility: An Emerging Consideration in Drug Discovery and Design», *J. Med. Chem.*, vol. 51, fasc. 20, pp. 6237–6255, ott. doi: 10.1021/jm800562d
30. Mandrekar JN (2010) «Receiver Operating Characteristic Curve in Diagnostic Test Assessment», *J. Thorac. Oncol.*, vol. 5, fasc. 9, pp. 1315–1316, set. doi: 10.1097/JTO.0b013e3181ec173d
31. Gentile F et al (2020) «Deep Docking: A Deep Learning Platform for Augmentation of Structure Based Drug Discovery», *ACS Cent. Sci.*, vol. 6, fasc. 6, pp. 939–949, giu. doi: 10.1021/acscentsci.0c00229
32. Wang R, Wang eS (2001) «How Does Consensus Scoring Work for Virtual Library Screening? An Idealized Computer Experiment», *J. Chem. Inf. Comput. Sci.*, vol. 41, fasc. 5, pp. 1422–1426, set. doi: 10.1021/ci010025x
33. Gentile F et al (2021) «Automated discovery of noncovalent inhibitors of SARS-CoV-2 main protease by consensus Deep Docking of 40 billion small molecules», *Chem. Sci.*, vol. 12, fasc. 48, pp. 15960–15974, doi: 10.1039/D1SC05579H
34. Masters L, Eagon S, Heying eM (2020) «Evaluation of consensus scoring methods for AutoDock Vina, smina and idock». J Mol Graph Model 96:107532. 10.1016/j.jmgm.2020.107532

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- GA.png
- SIChemFlowpy.pdf