Published in final edited form as:
J Chem Inf Model. 2017 July 24; 57(7): 1579–1590. doi:10.1021/acs.jcim.7b00153.

Machine Learning Consensus Scoring Improves Performance Across Targets in Structure-Based Virtual Screening

Spencer S. Ericksen[†], **Haozhen Wu**[†,#], **Huikun Zhang**[‡,#], **Lauren A. Michael**[§], **Michael A. Newton**[‡,||], **F. Michael Hoffmann**[†,⊥], and **Scott A. Wildman**[*,†]

[†]Small Molecule Screening Facility, UW Carbone Cancer Center, School of Medicine and Public Health, University of Wisconsin-Madison, 1111 Highland Ave., Madison, Wisconsin 53705, United States

[⊥]McArdle Laboratory for Cancer Research, University of Wisconsin-Madison, 1111 Highland Ave., Madison, Wisconsin 53705, United States

[‡]Department of Statistics, University of Wisconsin-Madison, 1300 University Ave., Madison, Wisconsin 53706, United States

[||]Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, 1300 University Ave., Madison, Wisconsin 53706, United States

[§]Center for High Throughput Computing, Department of Computer Sciences, University of Wisconsin-Madison, 1210 W. Dayton St., Madison, Wisconsin 53706, United States

Abstract

In structure-based virtual screening, compound ranking through a consensus of scores from a variety of docking programs or scoring functions, rather than ranking by scores from a single program, provides better predictive performance and reduces target performance variability. Here we compare traditional consensus scoring methods with a novel, unsupervised gradient boosting approach. We also observed increased score variation among active ligands and developed a statistical mixture model consensus score based on combining score means and variances. To evaluate performance, we used the common performance metrics ROCAUC and EF1 on 21 benchmark targets from DUD-E. Traditional consensus methods, such as taking the mean of quantile normalized docking scores, outperformed individual docking methods and are more robust to target variation. The mixture model and gradient boosting provided further improvements over the traditional consensus methods. These methods are readily applicable to new targets in

[*]Corresponding Author: swildman@wisc.edu.

[#]**Author Contributions**
These authors contributed equally to this work.

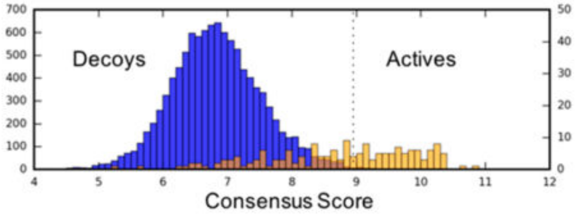
ORCID
Scott A. Wildman: 0000-0002-8598-0751

Notes
The authors declare no competing financial interest.

Supporting Information
The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.7b00153.
Mean variance consensus mixture model details, box-whisker plots comparing performance metrics of scoring methods, performance of boosting consensus score versus number of targets, standard deviation of docking scores, *p*-values comparing consensus methods to individual docking programs (PDF)

academic research and overcome the potentially poor performance of using a single docking method on a new target.

Graphical Abstract



INTRODUCTION

Protein–ligand docking is a common computational method for structure-based drug discovery, used for ligand binding pose determination, molecular design and prediction of binding affinity. Of interest, is structure-based virtual screening (VS), also referred to as *in silico* screening, or virtual high-throughput screening (vHTS), commonly done through protein–ligand docking.[1,2] A primary goal of VS is to score a large database of compounds based on the likelihood of interactions with a specific target structure. The rank order of these scored compounds is then used to identify a subset of the full database of compounds enriched for "hit" ligands that interact with the target. A sufficiently precise VS would predict hits from a library of available compounds, and experimental testing could then, with confgidence, be limited to those predicted hits, rather than "wet-lab" screening the entire library. In theory, this can save significant HTS costs as VS can routinely process millions of compounds, including yet to be synthesized virtual compounds.

In practice, however, VS approaches often do not provide sufficient enrichment to obviate the need for large-scale HTS.[2–4] Reasons for the limited efficacy of docking-based VS have been reviewed elsewhere[5–9] and commonly arise from oversimplified models of protein–ligand interactions that trade accuracy for computational speed. Docking and scoring programs intended for VS often inadequately represent ligand and target conformation space, dynamics, solvation, polarization, and other effects. On the other hand, algorithms that do consider these features lack the computational speed necessary to dock millions of compounds. Also problematic is the common use of the same scoring function for the distinct tasks of docking search, pose selection, and compound ranking, the result being that none of these tasks is performed ideally.[10]

A second major limitation of VS is the performance variability of a given docking method across targets or in new target space.[2–9] No single docking and scoring algorithm performs the best for every target. Therefore, confgidence is limited for any *a priori* selection of a docking and scoring program, especially a scoring function for accurate compound ranking, from among the large number available. A robust approach to VS includes an evaluation of algorithms for the specific target in question, though this would require a significant amount of experimental data on that specific target, in which case a docking-based VS may no longer be the most appropriate approach.

One approach that can partially compensate for these limitations of search algorithms and scoring functions is the use of data fusion methods such as consensus scoring,[11–15] which can be implemented easily given access to a methodologically diverse set of docking programs and scoring functions. This idea accepts that each scoring function or docking program may not be highly accurate for compound ranking on a specific target. Instead, it supposes that each scoring function uses different forms, terms, and parameters and that each has at least some predictive value. Thus, integrating the scores (compound rankings) from many different programs may produce a consensus score that outperforms individual programs in terms of VS enrichment while also being more robust across more targets.

A variety of consensus scoring approaches have been developed since the defining work by Charifson et al.[11] for compound scoring and soon after for pose prediction.[12] The theoretical basis for the advantage of consensus scoring strategies was elucidated by Wang et al.,[13] firmly rooted in the law of large numbers where the mean of repeated independent measures tends toward a true value. In traditional consensus scoring, values from different scoring functions were combined by either statistical methods, such as the mean, median, minimum, or maximum of ranks of scores, or by voting schemes, which classified hits based on how many of the scoring functions gave a good ranking to a particular ligand.[13] The details of each method varied in different implementations, but the performance of consensus scoring methods was often superior to individual scoring functions against a wider range of targets. In the years since consensus methods were first developed, an emphasis has been placed on pose-matching consensus.[16–20] In these approaches, ligands are docked by multiple programs and considered to be in agreement if the RMSD of ligand poses is below a particular threshold. This type of consensus can perform better than single programs, though they can still be subject to missing VS hits if individual programs dock compounds incorrectly. In contrast, Pereira et al.[21] implemented a deep-learning neural net to develop scoring models with improved VS performance on the original DUD targets. Their approach builds machine learning models based on ligand features in the binding sites and is therefore not a consensus model in the sense that it does not combine results from distinct docking/ scoring programs. Nonetheless, they achieve significant improvements in overall performance, ROCAUC = 0.48 improves to 0.74 based on DOCK6 poses and ROCAUC = 0.62 improves to 0.81 based on AutoDock Vina poses.

One drawback of many traditional consensus scoring approaches is that compounds are often docked only once using a single program, and then several scoring functions evaluate a single pose or set of poses. This can create abnormalities in some scores due to sensitivity to the exact placement of the ligand in the binding site since docking algorithms commonly use the scoring function internally for pose selection. A compound docked by FRED, for example, may not result in a favorable AutoDock score even if the pose is “correct” (within the error bars of the docking). This limitation was understood even in the early days of consensus scoring,[11–15] and some efforts to compensate by minimizing the ligand to each new scoring function were attempted. The preferred solution would be separate pose prediction with each algorithm prior to scoring in order to avoid over-reliance on a single docking engine. Though far more computationally expensive, this approach is now achievable with high-throughput computing (HTC) and cloud computing resources.

With access to campus and Open Science Grid[22] HTC resources and a variety of different academic and proprietary docking programs, we have re-examined docking-based consensus scoring methods for performance in structure-based VS. We compare individual and consensus scoring methods in terms of VS performance metrics against a set of 21 benchmark targets from DUD-E[23] using eight different docking programs. We confirm that traditional consensus methods outperform individual scoring methods and are more robust to target variation. Further, a novel mixture model consensus and a gradient boosting consensus provide additional improvements in VS performance.

Discrete mixture models are used widely in statistical applications to characterize heterogeneity of a population.[24,25] An unsupervised approach to building a consensus score is to leverage features of the multivariate distribution of multiple docking scores from each ligand/target pair. The mixture model treats this distribution as a mixture of two components, one for the actives and one for the decoys, recognizing that we do not know the active/decoy status for any arbitrary molecule. Learning is nevertheless possible via maximum likelihood and expectation-maximization.[26] A novel consensus score is constructed as the posterior probability that the ligand is active given the multiple docking scores.

Gradient boosting[27] is a common technique in machine learning and data science fields. This approach, and the related GBM, gradient boosting machines, multiple additive regression trees, and stochastic gradient boosting, involve building ensembles of decision trees, similar to the popular random forest models.[28] However, in this case, additional trees are added to the model in order to overcome errors of existing models until no further improvement is made. This approach is shown to be less sensitive to noisy input than many machine learning methods due specifically to this adaptive learning step,[29,30] where the relative contributions of individual scores are adjusted while learning the trees. The final predictive model returns a weighted average of predictions from the ensemble of trees.

In this study, mixture modeling and gradient boosting are used to develop two consensus scores from eight distinct docking scores. As these scores are based on a variety of scoring approaches, the inputs to the model are not all predictions of binding energy. The resulting consensus scores are likewise not predictions of binding energy, but rather compound ranking scores suitable to enrich a small selection with active compounds from a large library. The resultant scores provide an improvement over individual docking methods and traditional consensus scoring approaches in a VS setting.

MATERIALS AND METHODS

Target Structure Preparation

Benchmark targets with labeled compounds (actives and decoys) were obtained from the DUD-E set.[23] This set is built from experimentally verified actives and property-matched, but topologically dissimilar, decoys. A subset of DUD-E comprising 21 targets was selected to cover the major druggable target classes (GPCRs, ion channels, kinases, nuclear receptors, and proteases) shown in Table 1.

Target structure coordinates (PDB format) were obtained from the DUD-E database and processed via PDB2PQR (v2.1.0)[31] to fill missing atoms and standardize atom and residue names. The resulting PQR files were then processed using Chimera's "Dock Prep" utility[32] and saved in Mol2 format. We used this utility to choose atom coordinates based on highest occupancy where alternative locations for atoms are provided, add missing side chain atoms, protonate (at pH 7.4), and assign partial charges using AMBER FF14 SB for standard residues and Antechamber[33] (AM1-BCC) for nonstandard residues.

Compound Library Preparation

Each target's compound set was downloaded directly from DUD-E in Mol2 and SMILES formats. Compounds with undesignated stereocenters were enumerated in all possible stereochemical combinations with OpenEye OMEGA[34] to a maximum of 12 stereocenters per molecule. In later scoring, only the stereoisomer with the best score was retained for analysis and consensus methods. Table 1 therefore shows the number of actives and decoys scored by the methods herein, rather than the total number docked, as the number scored is most relevant for the analysis of performance metrics below. DUD-E Mol2 files provide AM1 partial charges for active and decoy ligands, and these were used directly unless further processing was required as noted below.

Ligand Docking

Default parameters were used for each docking and scoring program with exceptions explicitly indicated below. All docking involved static target structure representations. Run times for each program were tracked to examine relative performance and establish an appropriate number of compounds to submit for each HTC job (library chunk size). The crystal structure ligands specified by DUD-E were used to identify the binding region for each target and therefore are used to define the docking search space. No ligand binding information was used to assist compound ranking, with the exception of OpenEye HYBRID, which uses bound ligand shape and features as components of the scoring function.

AutoDock v4.2.6[35]

Target structure files were converted to PDBQT format using the prepare_receptor4.py script provided with AutoDockTools. Zn^{2+}/Ca^{2+} ions were replaced if removed during preparation, and full net charges were reassigned. Grid files were generated for every unique atom type observed within the entire ligand set. Grid dimensions were specified by default (15 Å edges with 0.375 Å lattice spacing) from the center of mass of the crystal structure ligand from DUD-E. The docking search space was defined as a box from coordinates of the crystal structure ligand with 4 Å padding in all Cartesian directions. Initial translational coordinate (tran0) was set to random, and 10 Lamarkian Genetic Algorithm dockings were performed for each compound. Compounds were converted to PDBQT format using the prepare_ligand4.py script in AutoDockTools.

Smina v1.1.2[36]

The Smina fork of AutoDock Vina[37] was used. One processing thread was specified on the Smina command line using the –cpu flag, and the box center was defined using the –autobox flag with the crystal structure ligand. The box dimensions were defined as the minimal box to enclose the crystal structure ligand coordinates, extended by 4 Å. Compounds were converted to PDBQT format using the prepare_ligand4.py script in AutoDockTools.

Dock v6.7[38]

A hydrogen-free target structure file was produced using Chimera, and the molecular surface was then generated using the DMS program in the DOCK6 suite with a probe radius of 1.4 Å. Using the molecular surface file as input, the negative binding site space was defined using the SPHGEN program, with arguments for distance, minimum radius, and maximum radius arguments set to 0.0 Å, 0.0 Å, 1.4 Å, respectively. A subset of spheres was isolated using sphere_selector program with distance cutoff of 6 Å. This sphere subset was visually inspected and edited for proper representation of the target binding site. Contact, energy, and bump grid files were generated with grid_spacing argument set to 0.3 Å, and box dimensions were set to encompass the sphere set with a 5 Å padding. Anchor and grow docking was performed with default parameters with the following exceptions: max_orientations was increased from 1000 to 5000, min_anchor_size was set to 40 to use a single anchor, pruning_clustering_cutoff was expanded from 100 to 1000, and van der Waals atom definitions were taken from vdw_AM-BER_parm99.defn file included with DOCK6 installation tree.

FRED v3.0.1 and HYBRID v3.0.1[39]

Since OEDocking algorithms operate on precomputed ligand conformers, compound libraries were converted directly from the DUD-E canonical SMILES format to stereochemically and conformationally enumerated 3D structures in a multiconformer OEBinary file format using OpenEye OMEGA with the following nondefault settings: maxconfs = 1000, flipper = true, strictstereo = false. Compounds were assigned MMFF94 charges using OpenEye molcharge. HYBRID scoring method requires bound ligand coordinates for each target for the shape and chemical feature matching aspects of its scoring function. The crystal structure ligands from DUD-E were used as the “bound_ligand” in the receptor_setup module of the OEDocking package. FRED and HYBRID were run with dock_resolution = High as the only nondefault setting.

PLANTS v1.2[40]

Protein targets and their corresponding crystal structure ligands were processed with SPORES v1.3, mode = complete, to ensure compatibility with PLANTS. The binding site space was a sphere defined by PLANTS, mode = bind, defined from the center of mass of the crystal structure ligand with 5 Å padding. PLANTS virtual screening was conducted with mode = screen. Compounds were processed using the SPORES program to ensure format compatibility with PLANTS.

rDock v2013.1[41]

Search space was defined automatically using the crystal structure ligand coordinates as a reference with the following settings: radius = 6.0 Å, small_sphere = 1.0 Å, max_cavities = 1. rDock was run with receptor_flex = 3.0 to permit some motion for target H-bond donors and acceptors. rDock developers recommend a multistaged triage protocol for VS rather than exhaustive docking of each compound, and therefore only compounds that pass a score threshold within a set number of runs move on to subsequent stages of additional runs. Since appropriate score cutoffs are highly site- and parameter-dependent, these were determined separately for each target. Cutoffs and maximum number of runs were obtained using the rbhtfinder script provided with the rDock suite based on the runtimes and triage rates from exhaustively docking a random subset of ~400 compounds. rDock developers also recommend trying both SCORE.TOTAL and SCORE.INTER for compound ranking. These scores are highly correlated, but in this work SCORE.INTER was found to perform slightly better on average and thus was used for all evaluations here.

Surflex-Dock v3.040[42]

Surflex requires a protomol to represent the negative binding site space, which was based on the crystal structure ligand, and was produced using Surflex in the “proto” mode with proto_bloat = 1.0. Protomols were visually inspected to make sure they appropriately represented the potential binding space. Docking was conducted with Surflex in “dock” mode with pgeom = on and ndock_final = 1.

Computing Resources and Job Management

Consensus docking, as a computational task, is ideally suited to the HTC resources available through the University of Wisconsin-Madison Center for High Throughput Computing (CHTC), including a campus grid of HTCondor[43] pools and the Open Science Grid (OSG).[22] HTC involves large numbers of compute nodes without the requirement for high-end, homogeneous architectures or fast node interconnections. Each compound-target docking was run as an independent process on a single core, and thus compound throughput scaled directly with the number of accessible cores. The HTCondor job management system[43] was used to run and track all docking jobs, as it easily scaled to many thousands of simultaneously running jobs, and allowed for the use of local resources and the OSG via a single, local submission computer. To run efficiently on these resources, the compound sets for each target were split into “chunks” that could run in under 2 h on a single core. We found this 2 h run time to be an efficient balance between reducing number of independent jobs and queue time with a low probability of job eviction by higher priority users. Chunks ranged from 10 to 500 compounds depending on compound throughput of each individual docking program. In this way, jobs could scavenge any open cores without high risk of job eviction when running on remote (OSG) resources.

Score Normalization

Due to the difference in raw docking score scales between different docking programs, the raw docking scores were normalized prior to use in consensus scoring schemes. For each target, the docking score distributions from each of the eight programs were transformed by quantile normalization using *R*.[44] Quantile normalization was favored over the more common *z*-score and min–max normalizations because transformed score distributions achieve a common shape, ensuring equal weights among program scores. Quantile normalization is also not sensitive to outliers (extremely bad or good docking scores), and therefore no additional treatment of outliers was performed. Missing compound scores from individual docking programs were ignored.

Traditional Consensus Scoring Methods

Using normalized scores, four common consensus scoring methods were applied, each using a different score selection method. Mean and median consensus methods set the score of each compound to the mean or median of the quantile normalized scores across the docking programs. Min and max consensus methods set the score of each compound to the minimum or maximum normalized score of the compound across the docking programs. In rare cases where less than three programs succeeded in docking or scoring a particular compound, no consensus score was computed. Note that in these rank-by-score consensus schemes, the input scores were uniformly weighted.

Mean–Variance Consensus

A set of unsupervised mixture models[24] were developed to combine the features of score mean and score variance across the eight individual docking programs, with a separate model for each target. Each target has a Gaussian distribution of mean scores and gamma distribution of score variance, taken to be independent. The compound is scored by mean–variance consensus (MVC), the posterior probability that it is active, conditional upon the mean and variance statistics, after the mixture model is estimated from the unlabeled data on the target.

The derived MVC score is calculated as

$$\mathrm{MVC}{=}(1{-}\pi_0)f_1(x,\,s)/f(x,\,s) \quad (1)$$

where π_0 is mixing proportion, $f_1(x,s)$ is the probability of the compound being active, and $f(x,s)$ is the probability density associated with score mean (x) and variance (s), conditional on the true activity of the compound–target pair. This means that MVC is the posterior probability of the compound being active in the mixture model. Free parameters are fit using expectation-maximization[26] methods on unlabeled scores from each target independently. Further details are provided in Supporting Information.

Gradient Boosting Consensus

Boosted tree models were developed using python interface of XGBoost,[45] a C++ implementation of a gradient boosting decision tree framework. Separate binary objective gbtree booster models were constructed for each target using the following specifications: η = 0.05, max_depth = 7, subsample = 0.83, colsample_bytree = 0.8, num_parallel_tree = 1, min_child_weight = 5, γ = 5, max_delta_step = 1, and scale_pos_weight = 1/(fraction of actives). The models were trained using 5-fold k-stratified cross-fold validation and optimized with respect to EF1 or ROCAUC.

For the boosting consensus scoring method, 21 individual decision tree ensemble binary classification models, one for each target, were trained by gradient boosting on labeled compound data (actives and decoys) from DUD-E using docking scores from each program as input features for each compound. Then, for each target ("on-target"), the individual docking scores were given to each of the 20 other "off-target" boosting models, and their output scores were averaged to produce the boosting consensus score (BCS). Only "off-target" models have been trained; no labeled data are required for the "on-target", and only docking scores (from the same set of programs) for compounds at the "on-target" are required as inputs to obtain a BCS value.

Evaluation of VS Performance

Several standard VS performance metrics were used in our evaluation.[46] Areas under the receiver-operator characteristic curve (ROCAUC) and precision-recall curve (PRAUC) metrics were computed using built-in functions from the scikit-learn Python module.[47] Enrichment factor at 1% was computed using the following equation:

$$\mathrm{EF1}{=}(a/n)/(A/N) \quad (2)$$

where n = number of compounds in 1% of database, a = number of actives in the top scoring 1% of database, A = number of actives in database, N = total number of compounds in database. EF1 has an upper bound based on the number of actives in the compound list for each target. This maximum EF1 is presented in Table 1 as a useful comparison for the EF1 values achieved by the VS approaches. Active and decoy distributions were generated using Python's matplotlib package.[48] Nonparametric t tests were performed using Wilcoxon sign rank (2-sided) to test for levels of significance ($p \leq 0.05$) for observed improvements when comparing compound ranking methods.

RESULTS AND DISCUSSION

The main goal of docking-based VS is to identify a subset of compounds enriched with actives from a large, chemically diverse library, based on predicted interactions with a target binding site.[1,2] The docking and scoring programs[35–42] included here are routinely used for VS, and while several additional docking packages may be available to some users, this study is focused on those most widely available to academic research groups. Our intention was to represent a single, general VS process across targets rather than to optimize performance for any single target.

A key concept in consensus approaches is the use of distinct docking and scoring tools utilizing both different algorithms for the docking pose search and different scoring function types. Table 2 summarizes the docking programs implemented in our study and their respective search and scoring strategies. Only one of these programs, HYBRID, requires and utilizes prior knowledge of the structure of a ligand bound to the target site. The tools used in this work are all commonly available, even to academic groups, and are based on both different docking techniques and scoring function types.

To evaluate consensus scoring strategies, 21 targets were selected from the DUD-E benchmark data set,[23] each accompanied by sets of active and decoy compounds (Table 1).

The 21 targets span a variety of target classes including kinases, nuclear receptors, proteases, and other enzymes and were selected to represent targets often of interest in VS efforts. We did not eliminate targets that may have been used in the development of the individual scoring functions in Table 2, as any positive effect for a single docking program is likely to have only negligible effect in the consensus methods.

To compare methods, ROCAUC and EF1 were used as performance metrics,[46,49,50] and any difference to values published in previous analyses of DUD-E targets by the same programs is likely to result from minor differences in target and ligand preparation. ROCAUC, the most widely accepted measure of VS performance, is the probability that an active compound will be scored better than a decoy, and EF1 is taken here as the enrichment of actives in the top 1% of the ranked database. Precision-recall AUC (PRAUC) was also calculated, and these values were consistent with the EF1 and ROCAUC results and are therefore not shown. Many other VS metrics have been used in other work, but those presented here are expected to provide the most direct comparison to other publications.

Individual Docking Programs

We observed that individual programs exhibit significant performance variability across targets and with respect to individual targets. The ROCAUC (Table 3) and EF1 metrics (Table 4) for each of the eight individual programs confirmed previous studies showing that no single algorithm can reliably distinguish actives from decoys for all targets.[2–9] For example, as shown in Table 4, DOCK6 was the most effective program on ADRB1 (EF1 = 25), while rDock, using a distinct algorithm, was the best for PLK1, PTN1, and FA10 (EF1 values of 10, 26, and 27, respectively). PLANTS performed best on three other targets: ACE, HIVINT, and HIVPR (EF1 values of 24, 15, and 15, respectively), Smina was the best for HDAC8 (EF1 = 32), and Surflex far outperformed the other algorithms for TRY1 (EF1 = 39). Compared to the other algorithms that do not require a bound ligand structure, FRED performed the best on average across the 21 targets, with a mean EF1 of 18 (mean ROCAUC = 0.78), and was ranked first on several targets, though it was not the best algorithm for most targets.

The ranking of individual programs by ROCAUC is different from those of EF1. The most distinct example of this is Surflex, which is the best approach not only for TRY1 but also PTN1, ADA17, and HIVPR when ranked by ROCAUC (ROCAUC values of 0.93, 0.88, 0.70, and 0.81, respectively). This difference is due to the nature of the two metrics, where ROCAUC measures performance over the full compound database, and EF1 is a more direct indication of early enrichment. Ranking programs by either metric alone is not a sufficient indication of performance,[46,49,50] and small differences in docking algorithm performance may be overcome by computational expense and ease of use. Still, these data confirm that blind selection of a single program for VS on an arbitrary target can be risky.

Tables 3 and 4 include an additional column for the “best performance.” This is a retrospective analysis of the top-performing individual program for each target, taking the maximum performance metric across each row in Tables 3 and 4. The ROCAUC and EF1 metrics for the best performance are significantly better than any single program (*p*-values $\leq 1.8 \times 10^{-04}$). The best performing values are useful for comparison purposes, but in reality are not possible to determine without prior knowledge of the active compounds for each target and docking with each of the different programs. This does not represent a new approach and is only provided here for comparison purposes.

HYBRID is the only program of the eight that explicitly utilizes prior knowledge of a ligand-target structure, which is used for a shape-matching algorithm embedded within its scoring function. The nature of the DUD-E benchmark decoys may provide some advantage for the HYBRID shape-matching due to the inclusion of only the most structurally dissimilar decoys.[23] In most cases, HYBRID provided better active versus decoy discrimination as measured by either ROCAUC or EF1 for a broad range of the targets tested. It yielded the highest EF1 for 12 of the 21 targets (DRD3, GRIA2, BRAF, CDK2, PLK1, SRC, FABP4, ESR2, GLCM, PDE5A, ADA17, and MMP13). The mean of the EF1 values for the 21 targets was 24 ± 11, significantly better than the mean EF1 of any other single program across the 21 targets (p-values $\leq 2.5 \times 10^{-03}$). Nonetheless, HYBRID did not provide the highest EF1 for every target, indicating again that it is not possible to predict *a priori* which docking algorithm will perform best for a specific target, even when a bound reference ligand is utilized. The retrospective selection of the best performance for each target indicates this point (Tables 3 and 4).

Traditional Consensus Scoring

To take advantage of the merits of all of the docking programs, a consensus of the scores may be used to predict actives rather than relying on any single algorithm. We applied four traditional consensus methods, using the minimum, maximum, median, and mean scores of either 7 or 8 docking scores for each compound (Tables 5 and 6), with p-values provided in Table 7. The traditional consensus methods largely outperformed the individual methods, which is as expected based on years of consensus scoring literature.[11–15] The best individual methods, FRED and HYBRID, had mean EF1 values of 18 and 24, respectively (Table 4), while the median, maximum, and minimum consensus methods achieved average EF1 values of 23, 18, and 14, respectively, across the 21 targets (Table 6).

The mean consensus performed the best of the traditional methods with mean ROCAUC = 0.83 and EF1 = 26. The improvement for mean consensus over HYBRID is not significant by EF1 (26 vs 24, p-value = 0.19), but is significant with respect to ROCAUC (0.83 vs 0.78, p-value = 0.019). The mean consensus performs as well as the best performance result, but is now accomplished *a priori*, without needing previous experimental results to identify the best individual program. These results based on EF1 are corroborated by ROCAUC values (Table 5).

Since HYBRID does consider a known bound ligand for scoring, we also built our consensus methods based on only seven docking programs, omitting HYBRID, in order to evaluate performance when no bound ligand structure is available, as is commonly the case. As seen in Tables 5 and 6, there is slight decrease in consensus ROCAUC and EF1 values, which can be expected since HYBRID often was the top-performing individual program. However, the consensus methods still outperform the individual programs.

Most traditional consensus methods combine a number of different docking scores by value and thereby suffer from the inclusion of poorly performing docking programs.[11–15] As these programs cannot be identified for each target *a priori*, they are included for every target. The effect of including noisy inputs, as inaccurate docking scores, reduced overall performance of the traditional consensus methods. In order to further improve our ability to distinguish actives from decoys, we implemented two additional consensus methods: a statistical mixture model of score mean and variance and a gradient boosting consensus approach. These machine-learning approaches are less subject to input noise[29,30] and as a result show more robust performance across targets.

Mean–Variance Consensus (MVC)

In our initial review of the individual docking scores, we identified the expected behavior that actives had somewhat better scores than decoys, even when this was not sufficient to produce a useful enrichment of the actives (Table 4). Importantly, we also noticed that for a given target, the active compounds had larger score variance than the decoys. Since actives and decoys are property matched in DUD-E, this discrepancy probably does not arise from the physicochemical differences, but is likely an inherent result of the docking. An active compound should have a highly negative (good) score when docked correctly, but a low negative (poor) score when docked incorrectly by a different docking program. A decoy, on the other hand, does not have the opportunity for a correctly docked good score and therefore should have predominantly poor scores. Since no docking algorithm will dock and score all actives correctly for all targets, the result is a wider range of scores for active compounds (good–poor) and a narrower range of scores (poor) for decoys.

We exploited this observation by building a statistical mixture model based on the score mean and score variance for each compound across the eight docking programs. The mixture model[24] is a common statistical technique used to identify the subpopulations within larger set. In our case, this translates to separating the active subset from the decoy subset for each target. This is done by mixing the bivariate distributions of mean and variance of the quantile normalized scores for each target. The model learns each target separately, but does not use the active or decoy labels. It determines the appropriate mixing parameters based only on the distributions of score mean and score variance using expectation-maximization. The resulting MVC score is taken as the probability of the compound being active.

The MVC achieves a mean EF1 of 26 and mean ROCAUC of 0.84 over the 21 targets (Tables 5 and 6). This easily outperforms the individual docking programs, with the exception of HYBRID, with *p*-values given in Table 7. We find that ranking by MVC gains enrichment accuracy compared to ranking by the traditional consensus methods for some protein targets. The gain is not statistically significant across targets for all of the traditional methods, and therefore more work is required to leverage the mean-variance phenomenon. We observed somewhat better enrichment of the MVC score when we eliminate all ligands for which there is at least one missing docking score (not shown). Nonetheless, MVC still performs equal to, or better than, the traditional consensus methods for 13 of 21 targets as measured by ROCAUC and 15 of 21 targets by EF1. The decrease in performance is similar to that of the traditional consensus methods when HYBRID scores are omitted. Overall, this represents similar performance to the mean consensus, with the possibility of slight improvement on some targets.

Boosting Consensus Scoring (BCS)

Of the methods considered here, a boosting consensus provides the greatest ability to distinguish between actives and decoys. The approach is based originally on random forest-style decision trees,[28] adapted into a boosting method[27] as implemented in XGBoost.[45] It has become a common approach used in many classification problems. As a machine learning approach, it does require some training data and, in the ideal docking use, would be trained on labeled compounds with activity data for the specific target.

However, given the desire to create a workflow that does not require prior activity data for the target of interest, we developed a transfer-learning approach called boosting consensus scoring (BCS). For BCS, individual tree ensemble classifiers (one for each target) were trained by gradient boosting using active and decoy compound labels from DUD-E and the docking scores from the eight programs as feature inputs. Then, for each “on-target” in the study, the BCS for each compound is determined in a leave-one-out manner, as the average score from the 20 “off-target” models. In this way, we were able to build a reliable boosting consensus model for each target without using any experimental data or compound labels from that specific target.

BCS (Tables 5 and 6) significantly outperforms both the individual programs and the traditional consensus methods, with ROCAUC = 0.85 and EF1 = 29 (*p*-values provided in Table 7). Compared to the mean consensus (ROCAUC = 0.83 and EF1 = 26), BCS performs better, with *p*-values of 2.5×10^{-03} (ROCAUC) and 3.9×10^{-03} (EF1). BCS gave the highest EF1 for 16 of 21 targets. This includes targets where several individual methods did well (ESR1 and PTN1), and also those targets that proved to be more difficult, such as DRD3 and HIVPR. Some individual programs did perform better than BCS on some specific targets (4/21 by ROCAUC, 4/21 by EF1). With respect to ROCAUC, these were FRED (PLK1 and FABP4), HYBRID (GLCM), and Surflex (PTN1), and by EF1, these were FRED (ESR2), HYBRID (PLK1, ESR2 and FA10), rDock (PLK1), and Surflex (TRY1). In these few cases, BCS still performed near the top, usually ranking second or third. This remains a more reliable performance, across a variety of targets, compared to any individual program, given that each program encountered targets for which it performed poorly (Tables 3 and 4).

In order to assess model stability, the number of “off-targets” used to build each BCS by scanning through all combinations of *n-choose-k* models from $k = 1$ to $k = 20$ (Supporting Information). Over all targets, we find $k > 5$ is often sufficient to build a reliable BCS, with only negligible improvements at $k > 15$ “off-targets”. Generating BCS from all available “off-targets” is shown to be reliable, and therefore we expect to be the common practice.

We expected that BCS would perform well when the target of interest (on-target) had a closely related target in the off-target list. Surprisingly, after grouping models by target class (according to DUD-E labels), we noted only one situation where a highly structurally related off-target model, used in isolation, performs exceptionally well for the on-target model. The

ESR1 off-target model produces ROCAUC = 0.90 and EF1 = 42 for ESR2 ligands, and the ESR2 off-target model produces ROCAUC = 0.89 and EF = 43 for ESR1 ligands. Similar occurrences were not observed for other structurally related targets: GPCR class members (ADRB1 and DRD3) or kinases (BRAF, CDK2, PLK1, SRC). Therefore, BCS performance remains high, even without the inclusion of a closely related off-target model, when the model is built using a broad set of off-targets.

Several studies present VS through docking with rescoring by MM/GBSA and mm/PBSA, which typically show improved performance compared to the individual docking methods. Virtanen and co-workers[51] studied five targets taken from the original DUD benchmark set and found average MM/GBSA EF1 = 20.1 and MM/PBSA EF1 = 20.2. These were an improvement over the initial docking, but do not meet the performance, reported here, of either MVC or BCS. A larger effort by Zhang and co-workers[52] including 38 DUD-E targets also shows MM/GBSA improvement over the initial docking methods to ROCAUC = 0.71 and EF1 = 8.99, though again not as good as the performance of MVC or BCS reported here. It is also noted that docking and MM/GBSA rescoring took on average ~6800 CPU seconds per ligand, compared to our eight docking programs which total ~1600 CPU second per ligand. While consensus docking is sufficiently expensive to warrant the use of HTC resources, it remains more accessible than MM/GB(PB)SA methods.

Variance of Scores within a Consensus

A close inspection of the scores provides additional insight into the function of the consensus methods. Compared to the individual docking methods, the score distributions of all ligands become noticeably narrower for the consensus methods. This is not the effect of normalization, but of the consensus methods themselves. An example of this effect is shown in Figure 1. For HDAC8 docked by Smina (ROCAUC = 0.86, EF1 = 32), the score distribution for decoys (blue) is shifted to the left (poorer normalized scores) compared to that of the actives (orange) in panel A. However, for the mean consensus (panel B, ROCAUC = 0.93, EF1 = 45), both distributions become narrower. This is reffected in a decrease in standard deviation of scores (σ). For all individual docking programs, the average standard deviation of scores is 2.38 for actives and 2.08 for the decoys. By applying mean consensus, these distributions are both narrowed to $\sigma = 1.74$ for actives and 1.33 for decoys. A table of changes to σ for all targets is provided in Supporting Information.

Figure 1, panels C and D show the impact on the critical top 1% of scores, where the reduction in standard deviation for the two distributions results in improved separation between the actives and decoys, now shown on the same scale. This separation is the cause of the improvement in compound ranking by consensus methods reffected in Tables 5 and 6 and is therefore somewhat larger for BCS than MVC or for mean consensus.

Timing

It is important to realize that not all of the docking and scoring programs process compounds with the same efficiency. FRED and HYBRID demonstrated highest molecule docking throughput, including the precomputation of ligand conformers by OpenEye OMEGA, and also provide the most consistent time per molecule (Table 8). Some of the eight programs have considerable variation in docking throughput, given by compute time per compound, where some molecules take much longer than the average. This situation is reffected in the standard deviation (Table 8) and often results from molecules with many rotatable bonds. Time for file transfers as well as for packing and unpacking of both programs and data was not considered, but in all cases, this was trivial compared to docking times. These calculation times reffect docking on our HTC resources using our installations of these programs, and different computing environments or changes to the search or scoring options used may greatly affect the compute time as well as docking accuracy.

Given access to our HTC resources, none of the programs or settings were too slow to implement on our benchmark set. However, higher docking throughput would free resources for other complementary VS strategies, such as ensemble docking or molecular dynamics-based postprocessing for rescoring or for pose stability assessments. Attention should be paid to both the computational costs and human effort with the increasing number of disparate docking methods in consensus approaches.

CONCLUSIONS

In traditional consensus scoring techniques, some subsets of docking programs or scoring functions can be more effective than using the broader set of available scores. Consensus performance is degraded by inputs of noisy predictors from poorly performing docking programs. Just as with individual programs, the most effective combination of docking programs cannot be determined *a priori* without sufficient training data for the target.

The machine learning consensus approaches are, however, more robust to noisy input scores, and in the methods described here, scores from all programs are included. In MVC, the poorly predicting programs become useful for generating wider variance in the scores of actives. For BCS, the boosting models learn to down-weight or selectively apply scores under specific conditions from weak predictors to compute a consensus.

The BCS outperforms all other VS methods presented here. While it does require some training data for “off targets”, it remains unsupervised for the target of interest, and additional training should not be necessary for new targets, provided the initial models were built on a sufficiently broad set of targets. An advantage of MVC is that it can be applied to a new target without use of labeled training data from any target.

It may be possible to improve overall performance by including more or different docking programs. In this study, only those commonly available to academic groups were considered, as many lack the resources to invest in a broad range of commercial packages. Even so, not every docking program was included, and integrating others should be balanced by ease of use and additional computational expense. Additional targets from DUD-E may be useful to examine the full range of molecular structures and interactions. Other sources of benchmark data may also be used, although each of these will also come with its own disadvantages, some of which may not be initially obvious.

This is the first use of a mean-variance mixture model or gradient boosting for consensus scoring in a virtual screening setting. Both of these methods outperform individual docking and scoring across a wide variety of targets. This initial study represents only a benchmark of these new approaches. The true test will come as they are applied to novel targets in the context of actual VS efforts.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

This work was funded by NIH/NCI P30 CA014520 to the UW Carbone Cancer Center, NIH/NCATS UL1TR000427 to the UW Institute for Clinical and Translational Research, NIH 1U54AI117924 to the UW Center for Predictive Computational Phenotyping, NSF awards 1321762 and 1148698 to the UW Center for High-Throughput Computing and Open Science Grid, and the UW-Madison Office of the Vice Chancellor for Research and Graduate Education with funding from the Wisconsin Alumni Research Foundation.

References

1. Lionta E, Spyrou G, Vassilatis DK, Cournia Z. Structure-Based Virtual Screening for Drug Discovery: Principles, Applications and Recent Advances. Curr Top Med Chem. 2014; 14:1923–1938. [PubMed: 25262799]
2. Cross JB, Thompson DC, Rai BK, Baber JC, Fan KY, Hu YB, Humblet C. Comparison of Several Molecular Docking Programs: Pose Prediction and Virtual Screening Accuracy. J Chem Inf Model. 2009; 49:1455–1474. [PubMed: 19476350]
3. Kitchen DB, Decornez H, Furr JR, Bajorath J. Docking and Scoring in Virtual Screening for Drug Discovery: Methods and Applications. Nat Rev Drug Discovery. 2004; 3:935–949. [PubMed: 15520816]
4. Moitessier N, Englebienne P, Lee D, Lawandi J, Corbeil CR. Towards the Development of Universal, Fast and Highly Accurate Docking/Scoring Methods: A Long Way to go. Br J Pharmacol. 2008; 153:S7–S26. [PubMed: 18037925]
5. Verdonk ML, Berdini V, Hartshorn MJ, Mooij WTM, Murray CW, Taylor RD, Watson P. Virtual Screening Using Protein–Ligand Docking: Avoiding Artificial Enrichment. J Chem Inf Comput Sci. 2004; 44:793–806. [PubMed: 15154744]
6. Cummings MD, DesJarlais RL, Gibbs AC, Mohan V, Jaeger EP. Comparison of Automated Docking Programs and Virtual Screening Tools. J Med Chem. 2005; 48:962–976. [PubMed: 15715466]
7. Warren GL, Andrews CW, Capelli AM, Clarke B, LaLonde J, Lambert MH, Lindvall M, Nevins N, Semus SF, Senger S, et al. A Critical Assessment of Docking Programs and Scoring Functions. J Med Chem. 2006; 49:5912–5931. [PubMed: 17004707]
8. Kolb P, Irwin JJ. Docking Screens: Right for the Right Reasons? Curr Top Med Chem. 2009; 9:755–770. [PubMed: 19754393]
9. Scior T, Bender A, Tresadern G, Medina-Franco JL, Martinez-Mayorga K, Langer T, Cuanalo-Contreras K, Agrafiotis DK. Recognizing Pitfalls in Virtual Screening: A Critical Review. J Chem Inf Model. 2012; 52:867–881. [PubMed: 22435959]
10. Wildman SA. Approaches to Virtual Screening and Library Design. Curr Pharm Des. 2013; 19:4787–4796. [PubMed: 23260026]
11. Charifson PS, Corkery JJ, Murcko MA, Walters WP. Consensus Scoring: A Method for Obtaining Improved Hit Rates from Docking Databases of Three-Dimensional Structures into Proteins. J Med Chem. 1999; 42:5100–5109. [PubMed: 10602695]
12. Stahl M, Rarey M. Detailed Analysis of Scoring Functions for Virtual Screening. J Med Chem. 2001; 44:1035–1042. [PubMed: 11297450]
13. Wang R, Wang S. How Does Consensus Scoring Work for Virtual Library Screening? An Idealized Computer Experiment. J Chem Inf Comput Sci. 2001; 41:1422–1426. [PubMed: 11604043]
14. Paul N, Rognan D. ConsDock: A New Program for the Consensus Analysis of Protein–Ligand Interactions. Proteins: Struct, Funct Genet. 2002; 47:521–533. [PubMed: 12001231]

15. Clark RD, Strizhev A, Leonard JM, Blake JF, Matthew JB. Consensus Scoring for Ligand/Protein Interactions. J Mol Graphics Modell. 2002; 20:281–295.

16. Feher M. Consensus Scoring for Protein–Ligand Interactions. Drug Discovery Today. 2006; 11:421–428. [PubMed: 16635804]

17. Miteva MA, Lee WH, Montes MO, Villoutreix BO. Fast Structure-Based Virtual Ligand Screening Combining FRED, DOCK, and Surflex. J Med Chem. 2005; 48:6012–6022. [PubMed: 16162004]

18. Houston DR, Walkinshaw MD. Consensus Docking: Improving the Reliability of Docking in a Virtual Screening Context. J Chem Inf Model. 2013; 53:384–390. [PubMed: 23351099]

19. Poli G, Martinelli A, Tuccinardi T. Reliability Analysis and Optimization of the Consensus Docking Approach for the Development of Virtual Screening Studies. J Enzyme Inhib Med Chem. 2016; 31:167–173.

20. Tuccinardi T, Poli G, Romboli V, Giordano A, Martinelli A. Extensive Consensus Docking Evaluation for Ligand Pose Prediction and Virtual Screening Studies. J Chem Inf Model. 2014; 54:2980–2986. [PubMed: 25211541]

21. Pereira JC, Caffarena ER, dos Santos CN. Boosting Docking-Based Virtual Screening with Deep Learning. J Chem Inf Model. 2016; 56:2495–2506. [PubMed: 28024405]

22. Pordes R, Petravick D, Kramer B, Olson D, Livny M, Roy A, Avery P, Blackburn K, Wenaus T, Wurthwein F, et al. The Open Science Grid. J Phys: Conf Ser. 2007; 78:012057.

23. Mysinger MM, Carchia M, Irwin JJ, Shoichet BK. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. J Med Chem. 2012; 55:6582–6594. [PubMed: 22716043]

24. McLachlan, G., Peel, D. Wiley Series in Probability and Statistics. John Wiley; New York, NY: 2000. Finite Mixture Models.

25. Efron, B. Institute of Mathematical Statistics Monographs. Vol. 1. Cambridge University Press; Cambridge, UK: 2012. Large-Scale Inference: Empirical Bayes Methods for Estimation, Testing, and Prediction.

26. Dempster AP, Laird NM, Rubin DB. Maximum Likelihood from Incomplete Data via the EM Algorithm. J R Stat Soc B. 1977; 39:1–38.

27. Friedman J, Hastie T, Tibshirani R. Additive Logistic Regression: A Statistical View of Boosting. Ann Stat. 2000; 28:337–407.

28. Breiman L. Random Forests. Mach Learn. 2001; 45:5–32.

29. Caruana, R., Niculescu-Mizil, A. An Empirical Comparison of Supervised Learning Algorithms. ICML –06 Proceedings of the 23rd International Conference on Machine Learning; New York: ACM; 2006. p. 161-168.

30. Hastie, T., Tibshirani, R., Friedman, J. Springer Series in Statistics. 2. Springer; New York: 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction; p. 588

31. Dolinsky TJ, Czodrowski P, Li H, Nielsen JE, Jensen JH, Klebe G, Baker NA. PDB2PQR: Expanding and Upgrading Automated Preparation of Biomolecular Structures for Molecular Simulations. Nucleic Acids Res. 2007; 35:W522–W555. [PubMed: 17488841]

32. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, Ferrin TE. UCSF Chimera – A Visualization System for Exploratory Research and Analysis. J Comput Chem. 2004; 25:1605–1612. [PubMed: 15264254]

33. Wang J, Wang W, Kollman PA, Case DA. Automatic Atom Type and Bond Type Perception in Molecular Mechanical Calculations. J Mol Graphics Modell. 2006; 25:247–260.

34. Hawkins PCD, Skillman AG, Warren GL, Ellingson BA, Stahl MT. Conformer Generation with OMEGA: Algorithm and Validation Using High Quality Structures from the Protein Databank and Cambridge Structural Database. J Chem Inf Model. 2010; 50:572–584. [PubMed: 20235588]

35. Morris GM, Huey R, Lindstrom W, Sanner MF, Belew RK, Goodsell DS, Olson AJ. AutoDock4 and AutoDockTools4: Automated Docking with Selective Receptor Flexibility. J Comput Chem. 2009; 30:2785–2791. [PubMed: 19399780]

36. Koes DR, Baumgartner MP, Camacho CJ. Lessons Learned in Empirical Scoring with Smina from the CSAR 2011 Benchmarking Exercise. J Chem Inf Model. 2013; 53:1893–1904. [PubMed: 23379370]

37. Trott O, Olson AJ. AutoDock Vina: Improving the Speed and Accuracy of Docking with a New Scoring Function, Efficient Optimization and Multithreading. J Comput Chem. 2010; 31:455–461. [PubMed: 19499576]

38. Allen WJ, Balius TE, Mukherjee S, Brozell SR, Moustakas DT, Lang PT, Case DA, Kuntz ID, Rizzo RC. DOCK 6: Impact of new Features and Current Docking Performance. J Comput Chem. 2015; 36:1132–1156. [PubMed: 25914306]

39. McGann M. FRED and HYBRID Docking Performance on Standardized Datasets. J Comput-Aided Mol Des. 2012; 26:897–906. [PubMed: 22669221]

40. Korb O, Stützle T, Exner TE. Empirical Scoring Functions for Advanced Protein-Ligand Docking with PLANTS. J Chem Inf Model. 2009; 49:84–96. [PubMed: 19125657]

41. Ruiz-Carmona S, Alvarez-Garcia D, Foloppe N, Garmendia-Doval AB, Juhos S, Schmidtke P, Barril X, Hubbard RE, Morley SD. rDock: A Fast, Versatile and Open Source Program for Docking Ligands to Proteins and Nucleic Acids. PLoS Comput Biol. 2014; 10:e1003571. [PubMed: 24722481]

42. Cleves AE, Jain AN. Knowledge-Guided Docking: Accurate Prospective Prediction of Bound Configurations of Novel Ligands Using Surflex-Dock. J Comput-Aided Mol Des. 2015; 29:485–509. [PubMed: 25940276]

43. Thain D, Tannenbaum T, Livny M. Distributed Computing in Practice: The Condor Experience. Concurr Comput. 2005; 17:323–356.

44. R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing; Vienna, Austria: 2013. http://www.R-project.org/

45. Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. 2016 arXiv:1603.02754.

46. Jain AN, Nicholls A. Recommendations for Evaluation of Computational Methods. J Comput-Aided Mol Des. 2008; 22:133–139. [PubMed: 18338228]

47. Pedregosa F, Varoquaux G, Gramforrt A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. Scikit-learn: Machine Learning in Python. J Mach Learn Res. 2011; 12:2825–2830.

48. Hunter JD. Matplotlib: A 2D Graphics Environment. Comput Sci Eng. 2007; 9:90–95.

49. Nicholls A. What do we Know and When do we Know it? J Comput-Aided Mol Des. 2008; 22:239–255. [PubMed: 18253702]

50. McGann M, Nicholls A, Enyedy I. The Statistics of Virtual Screening and Lead Optimization. J Comput-Aided Mol Des. 2015; 29:923–936. [PubMed: 26481649]

51. Virtanen SI, Niinivehmas SP, Pentikainen OT. Case-Specific Performance of MM-PBSA, MM-GBSA and SIE in Virtual Screening. J Mol Graphics Modell. 2015; 62:303–318.

52. Zhang X, Wong SE, Lightstone FC. Toward Fully Automated High Performance Computing Drug Discovery: A Massively Parallel Virtual Screening Pipeline for Docking and Molecular Mechanics/Generalized Born Surface Area Rescoring to Improve Enrichment. J Chem Inf Model. 2014; 54:324–337. [PubMed: 24358939]

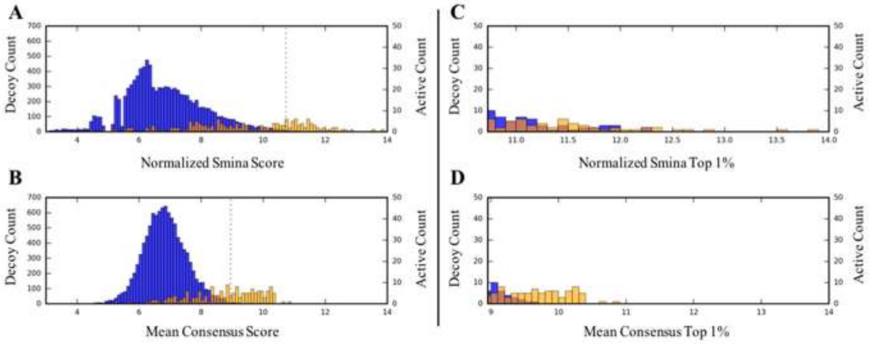


Figure 1.

Example score distributions. Full distributions of quantile normalized compound scores for HDAC8 from Smina (A) and from mean consensus (B). Distributions of decoys are shown in blue, using the left-hand scale, and actives are shown in orange, using the right-hand scale. The 99th percentile is marked by a vertical dotted line with the top scoring 1% of compounds found to the right. (C and D) The top 1% tail of the distributions, adjusted to the same scale. (Higher values are more favorable following quantile normalization.)

Table 1

Selected Targets from DUD-E and Number of Compounds Docked

				compounds scored			
target class	DUD-E	name	PDB	actives	decoys	active%	maximum EF1
GPCR	ADRB1	beta-1 adrenergic receptor	2vt4	247	15850	1.53	65
GPCR	DRD3	dopamine D3 receptor	3pbl	480	34050	1.39	72
ion channel	GRIA2	glutamate receptor ionotropic, AMPA2	3kgc	158	11845	1.32	76
kinase	BRAF	serine/threonine-protein kinase B-raf	3d4q	152	9950	1.5	67
kinase	CDK2	cyclin-dependent kinase 2	1h00	474	27850	1.67	60
kinase	PLK1	serine/threonine-protein kinase PLK1	2owb	107	6800	1.55	65
kinase	SRC	tyrosine-protein kinase SRC	3el8	524	34500	1.5	67
miscellaneous	FABP4	fatty acid binding protein adipocyte	2nnq	47	2750	1.68	60
nuclear receptor	ESR1	estrogen receptor alpha	1sj0	383	20685	1.82	55
nuclear receptor	ESR2	estrogen receptor beta	2fsz	367	20199	1.78	56
other enzymes	ACE	angiotensin-converting enzyme	3bkl	282	16900	1.64	61
other enzymes	GLCM	beta-glucocerebrosidase	2v3f	54	3800	1.4	71
other enzymes	HDAC8	histone deacetylase 8	3f07	170	10450	1.6	63
other enzymes	HIVINT	HIV type 1 integrase	3nf7	100	6650	1.48	68
other enzymes	PDE5A	phosphodiesterase 5A	1udt	398	27550	1.42	70
other enzymes	PTN1	protein-tyrosine phosphatase 1B	2azr	130	7250	1.76	57
protease	ADA17	ADAM17	2oi0	532	35900	1.46	68
protease	FA10	coagulation factor X	3kl6	537	28325	1.86	54
protease	HIVPR	HIV type 1 protease	1xl2	536	35750	1.48	68
protease	MMP13	matrix metalloproteinase 13	830c	572	37200	1.51	66
protease	TRY1	trypsin I	2ayw	449	25980	1.7	59

Table 2

Search and Scoring Strategies used by Docking Programs

docking program	search algorithm	scoring function
AutoDock v4.2	Lamarkian genetic algorithm with simulated annealing	force field
DOCK v6.7	incremental construction (anchor-and- grow)	force field
FRED v3.0.1	exhaustive rigid docking search, discretized configuration space	empirical
HYBRID v3.0.1	exhaustive rigid docking search, discretized configuration space	knowledge-based + empirical
PLANTS v1.2	ant colony optimization	empirical
rDock v2013.1	genetic algorithm, Monte Carlo, minimization	empirical
Smina 1.1.2	iterated local search global optimzer	knowledge-based
Surflex-Dock v3.040	incremental construction with matching algorithm	empirical

Table 3

ROCAUC for Individual Methods and Best Individual Performance[a]

		Individual Programs								
Target Class	**Target**	**AD4**	**DOCK6**	**FRED**	**HYBRID**	**PLANTS**	**rDock**	**Smina**	**Surflex**	**Best**
GPCR	ADRB1	0.68	0.78	0.77	0.65	0.86	0.81	0.79	0.80	0.86
GPCR	DRD3	0.69	0.59	0.79	0.81	0.69	0.66	0.68	0.71	0.81
ion channel	GRIA2	0.73	0.60	0.79	0.77	0.73	0.77	0.75	0.77	0.79
kinase	BRAF	0.73	0.60	0.75	0.69	0.54	0.79	0.86	0.71	0.86
kinase	CDK2	0.76	0.61	0.81	0.85	0.68	0.74	0.71	0.69	0.85
kinase	PLK1	0.60	0.48	0.80	0.75	0.65	0.68	0.57	0.60	0.80
kinase	SRC	0.65	0.64	0.65	0.66	0.52	0.68	0.67	0.66	0.68
miscellaneous	FABP4	0.67	0.54	0.84	0.82	0.74	0.60	0.77	0.79	0.84
receptor	ESR1	0.82	0.54	0.88	0.81	0.77	0.87	0.86	0.74	0.88
receptor	ESR2	0.77	0.48	0.89	0.89	0.69	0.80	0.79	0.68	0.89
other enzymes	ACE	0.78	0.72	0.80	0.84	0.84	0.62	0.61	0.76	0.84
other enzymes	GLCM	0.55	0.60	0.70	0.81	0.64	0.77	0.51	0.79	0.81
other enzymes	HDAC8	0.70	0.90	0.87	0.76	0.82	0.71	0.86	0.83	0.90
other enzymes	HIVINT	0.54	0.65	0.74	0.60	0.76	0.67	0.81	0.66	0.81
other enzymes	PDE5A	0.68	0.65	0.84	0.82	0.79	0.78	0.74	0.66	0.84
other enzymes	PTN1	0.66	0.76	0.76	0.78	0.72	0.76	0.66	0.88	0.88
protease	ADA17	0.51	0.40	0.59	0.69	0.58	0.58	0.54	0.70	0.70
protease	FA10	0.86	0.81	0.79	0.82	0.80	0.90	0.84	0.76	0.90
protease	HIVPR	0.63	0.66	0.74	0.78	0.79	0.64	0.74	0.81	0.81
protease	MMP13	0.67	0.60	0.77	0.87	0.71	0.67	0.67	0.76	0.87
protease	TRY1	0.79	0.82	0.80	0.83	0.81	0.74	0.75	0.93	0.93
	mean	0.69	0.64	0.78	0.78	0.72	0.73	0.72	0.75	0.84
	std. dev.	0.09	0.12	0.07	0.08	0.10	0.09	0.10	0.08	0.06

[a] Colors represent a 3-point gradient from worst (red) to best (green). “Best” indicates the best performance across docking programs.

Table 4

EF1 for Individual Methods and Best Individual Performance[a]

		Individual Programs								
Target Class	**Target**	**AD4**	**DOCK6**	**FRED**	**HYBRID**	**PLANTS**	**rDock**	**Smina**	**Surflex**	**Best**
GPCR	ADRB1	5	25	7	19	19	13	6	13	25
GPCR	DRD3	4	1	10	10	3	1	2	3	10
ion channel	GRIA2	4	13	32	47	14	4	8	13	47
kinase	BRAF	4	9	18	29	0	12	14	11	29
kinase	CDK2	15	4	18	30	4	17	10	4	30
kinase	PLK1	6	2	9	10	1	10	0	0	10
kinase	SRC	6	3	5	7	1	3	4	6	7
miscellaneous	FABP4	9	0	30	32	0	0	30	7	32
receptor	ESR1	32	8	37	36	17	29	23	20	37
receptor	ESR2	21	9	40	40	12	22	20	12	40
other enzymes	ACE	14	12	18	20	24	3	3	9	24
other enzymes	GLCM	4	12	4	35	13	17	4	31	35
other enzymes	HDAC8	0	27	31	30	15	2	32	2	32
other enzymes	HIVINT	0	11	8	10	15	7	8	5	15
other enzymes	PDE5A	7	9	20	31	15	10	12	5	31
other enzymes	PTN1	12	15	21	19	21	26	21	15	26
protease	ADA17	0	0	8	17	6	10	14	10	17
protease	FA10	26	16	17	19	12	27	18	8	27
protease	HIVPR	2	9	6	10	15	5	6	13	15
protease	MMP13	12	6	18	30	15	3	4	11	30
protease	TRY1	7	16	17	20	17	14	3	39	39
	mean	9	10	18	24	11	11	11	11	27
	std. dev.	9	7	11	11	7	9	9	9	11

[a] Colors represent a 3-point gradient from worst (red) to best (green). "Best" indicates the best performance across docking programs.

Table 5

ROCAUC for Consensus Methods and Best Individual Performance[a]

		Consensus (+HYBRID)						Consensus (−HYBRID)					
Target	**Best**	**BCS**	**MVC**	**Mean**	**Med**	**Max**	**Min**	**BCS**	**MVC**	**Mean**	**Med**	**Max**	**Min**
ADRB1	0.86	0.92	0.92	0.91	0.89	0.90	0.79	0.89	0.88	0.89	0.88	0.85	0.82
DRD3	0.81	0.81	0.75	0.79	0.78	0.74	0.73	0.80	0.74	0.76	0.75	0.73	0.71
GRIA2	0.79	0.89	0.87	0.87	0.82	0.86	0.76	0.87	0.84	0.84	0.80	0.83	0.77
BRAF	0.86	0.86	0.88	0.85	0.82	0.87	0.69	0.84	0.85	0.82	0.80	0.85	0.68
CDK2	0.85	0.91	0.87	0.85	0.81	0.89	0.77	0.83	0.79	0.79	0.77	0.80	0.75
PLK1	0.80	0.79	0.78	0.74	0.72	0.78	0.65	0.78	0.75	0.71	0.68	0.76	0.64
SRC	0.68	0.72	0.75	0.75	0.73	0.74	0.67	0.70	0.73	0.73	0.71	0.72	0.66
FABP4	0.84	0.79	0.78	0.80	0.79	0.74	0.71	0.79	0.77	0.79	0.77	0.73	0.70
ESR1	0.88	0.88	0.90	0.87	0.86	0.89	0.74	0.90	0.89	0.86	0.85	0.89	0.74
ESR2	0.89	0.91	0.89	0.85	0.82	0.89	0.68	0.89	0.87	0.82	0.79	0.88	0.67
ACE	0.84	0.85	0.83	0.83	0.83	0.81	0.78	0.82	0.80	0.81	0.81	0.77	0.77
GLCM	0.81	0.80	0.74	0.73	0.72	0.76	0.66	0.76	0.71	0.70	0.68	0.75	0.64
HDAC8	0.90	0.93	0.93	0.92	0.91	0.91	0.80	0.92	0.93	0.92	0.91	0.90	0.81
HIVINT	0.81	0.81	0.82	0.82	0.80	0.80	0.68	0.82	0.82	0.81	0.79	0.79	0.70
PDE5A	0.84	0.90	0.89	0.88	0.84	0.88	0.79	0.87	0.85	0.85	0.82	0.83	0.77
PTN1	0.88	0.85	0.85	0.81	0.81	0.86	0.65	0.83	0.84	0.80	0.78	0.86	0.65
ADA 17	0.70	0.74	0.69	0.62	0.60	0.71	0.52	0.69	0.63	0.59	0.59	0.64	0.50
FA10	0.90	0.91	0.95	0.93	0.92	0.93	0.80	0.90	0.94	0.93	0.92	0.92	0.80
HIVPR	0.81	0.88	0.85	0.84	0.82	0.83	0.74	0.85	0.82	0.82	0.81	0.79	0.73
MMP13	0.87	0.88	0.84	0.81	0.78	0.84	0.72	0.83	0.77	0.76	0.75	0.77	0.69
TRY1	0.93	0.93	0.93	0.93	0.91	0.92	0.82	0.92	0.93	0.93	0.91	0.92	0.82
mean	0.84	0.85	0.84	0.83	0.81	0.84	0.72	0.83	0.82	0.81	0.79	0.81	0.72
std. dev.	0.06	0.06	0.07	0.07	0.07	0.07	0.07	0.06	0.08	0.08	0.08	0.07	0.08

[a] Colors represent a 3-point gradient from worst (red) to best (green). “Best” indicates the best performance across docking programs.

BCS = boosting consensus score, MVC = mean-variance consensus.

Table 6

EF1 for Consensus Methods and Best Individual Performance[a]

		Consensus (+HYBRID)						Consensus (−HYBRID)					
Target	**Best**	**BCS**	**MVC**	**Mean**	**Med**	**Max**	**Min**	**BCS**	**MVC**	**Mean**	**Med**	**Max**	**Min**
ADRB1	25	31	27	28	24	21	19	22	24	25	23	19	20
DRD3	10	13	7	12	11	4	11	12	5	10	9	4	8
GRIA2	47	49	46	45	33	24	28	41	31	31	25	22	18
BRAF	29	29	23	22	15	21	6	22	20	17	11	19	4
CDK2	30	35	27	29	24	23	14	25	21	22	20	17	15
PLK1	10	8	8	5	4	7	1	4	4	3	3	5	0
SRC	7	7	9	7	6	6	5	8	9	6	6	7	4
FABP4	32	34	32	32	32	13	6	32	32	26	30	11	6
ESR1	37	38	37	34	34	32	16	37	36	32	33	31	14
ESR2	40	35	34	31	28	26	9	32	30	27	23	26	7
ACE	24	33	30	30	26	19	13	32	28	28	24	19	13
GLCM	35	37	30	30	28	19	19	22	22	22	19	15	19
HDAC8	32	45	46	46	41	20	35	32	38	38	34	16	32
HIVNT	15	19	21	17	11	13	10	15	18	16	13	12	11
PDE5A	31	32	28	26	26	19	13	24	23	23	24	14	12
PTN1	26	35	37	35	32	30	20	35	32	32	28	27	18
ADA 17	17	19	17	16	17	11	4	14	16	15	15	10	3
FA10	27	26	30	33	30	23	19	25	29	32	28	22	18
HIVPR	15	19	16	18	18	9	15	17	15	17	15	9	15
MMP13	30	34	25	26	24	20	18	23	20	22	21	16	15
TRY1	39	33	31	28	27	24	18	30	30	28	24	23	16
mean	27	29	27	26	23	18	14	24	23	22	20	16	13
std. dev.	11	11	11	11	10	8	8	10	9	9	8	7	7

[a] Colors represent a 3-point gradient from worst (red) to best (green). “Best” indicates the best performance across docking programs.

BCS = boosting consensus score, MVC = mean-variance consensus.

Table 7

p-Values Comparing Performances of Mean and Boosting Consensus to Other Methods[a]

(+HYBRID)		Mean	Median	Max	Min
	BCS	2.5E-03	3.2E-05	3.0E-03	2.5E-11
ROCAUC	**MVC**	5.7E-03	1.1E-05	6.2E-02	2.7E-10
	Mean	--	3.4E-06	3.5E-01	3.0E-12
	BCS	9.8E-04	1.8E-05	6.6E-07	1.9E-08
EF1	**MVC**	3.3E-01	1.1E-03	9.1E-06	1.6E-07
	Mean	--	7.6E-04	4.9E-05	5.1E-08
(−HYBRID)					
	BCS	3.9E-03	5.6E-05	2.6E-03	1.1E-09
ROCAUC	**MVC**	4.5E-02	8.6E-05	5.3E-02	1.8E-08
	Mean	--	3.8E-07	9.0E-01	1.2E-10
	BCS	9.4E-02	2.7E-03	6.2E-06	5.9E-06
2.7E-03	**MVC**	2.6E-01	7.3E-04	1.9E-05	3.5E-06
	Mean	--	1.9E-03	5.9E-05	2.9E-07

[a] *p*-Values computed by pairwise *t*-test. Red highlighted values indicate *p*-values that are not significant ($P > 0.05$).

Table 8

Average Timing of Docking Programs[a]

	AD4	DOCK6	FRED	HYBRID	PLANTS	rDock	Smina	Surflex
mean	435.64	719.18	15.64	9.25	43.40	49.27	250.14	78.93
std. dev.	197.05	592.90	5.68	2.94	20.48	26.73	172.79	1159.58

[a] In wall-seconds per compound per CPU-thread.