J Mol Model (2011) 17:2133–2141
DOI 10.1007/s00894-010-0854-x

ORIGINAL PAPER

Brainstorming: weighted voting prediction of inhibitors for protein targets

Dariusz Plewczynski

Received: 10 June 2010 / Accepted: 8 September 2010 / Published online: 21 September 2010

Abstract The "Brainstorming" approach presented in this paper is a weighted voting method that can improve the quality of predictions generated by several machine learning (ML) methods. First, an ensemble of heterogeneous ML algorithms is trained on available experimental data, then all solutions are gathered and a consensus is built between them. The final prediction is performed using a voting procedure, whereby the vote of each method is weighted according to a quality coefficient calculated using multivariable linear regression (MLR). The MLR optimization procedure is very fast, therefore no additional computational cost is introduced by using this jury approach. Here, brainstorming is applied to selecting actives from large collections of compounds relating to five diverse biological targets of medicinal interest, namely HIV-reverse transcriptase, cyclooxygenase-2, dihydrofolate reductase, estrogen receptor, and thrombin. The MDL Drug Data Report (MDDR) database was used for selecting known inhibitors for these protein targets, and experimental data was then used to train a set of machine learning methods. The benchmark dataset (available at http://bio.icm.edu.pl/~darman/chemoinfo/benchmark.tar.gz) can be used for further testing of various clustering and machine learning methods when predicting the biological activity of compounds. Depending on the protein target, the overall recall value is raised by at least 20% in comparison to any single machine learning method (including ensemble methods like random forest) and unweighted simple majority voting procedures.

Keywords Ligand classification · Protein target specificity · MDL Drug Data Report · Machine-learning · Support vector machine · Random forest

Abbreviations

SVM	Support vector machine
ANN	Artificial neural nets
NB	Naïve Bayesian
TV	Trend vectors
kNN/GA	k nearest neighbors with genetic algorithm optimization
RF	Random forest
DT	Decision tree

Introduction

The number of potential drug targets is increasing, mainly through genomic initiatives [1–3], high throughput experiments [4–7] or microarray or cellular screening [8]. In the pharmaceutical industry generally, small chemical compound collections are tested for single or multiple protein targets. Typical high throughput screening (HTS) experiments allow selection of thousands of high activity leads against a given protein target by testing millions of compounds. Unfortunately, this approach can be used only with single proteins, and cannot be applied to larger number of drug targets. Using some initial information about known active molecules can reduce the

D. Plewczynski (✉)
Interdisciplinary Center for Mathematical and Computational Modelling, University of Warsaw,
Pawinskiego 5a Street,
02-106 Warsaw, Poland
e-mail: D.Plewczynski@icm.edu.pl

scope of the search for a selected protein target. The knowledge base assists HTS studies by selecting a set of compounds for further screening in virtual screening (VS). VS methods typically use 2D fingerprint measures of structural similarity depend on definition of distance in descriptor space [9–11]. Another approach, namely data fusion, combines the results of multiple similarity searches of chemical databases performed by different algorithms, or using different features [12–16]. Such models are based on frequency distributions of similarity values that are fused using integration over regions defined by the particular fusion rule. Typically, the use of binary kernel discrimination (BKD) for identifying potential active compounds in lead-discovery programs is superior to methods based on similarity searching and substructural analysis but inferior to a support vector machine [11, 13, 14, 17, 18]. New methods for ligand-based VS use data fusion and machine learning (ML) to enhance the effectiveness of identification of potential actives over typical similarity searching [19] using a single bioactive reference. This basic search protocol can be extended by the use of group fusion to combine the results of similarity searches when multiple reference structures are available. Similarity searches are typically based on the assumption that the nearest neighbors resulting from a similarity search using a single bioactive reference structure are also active [19]. Similarly, various ML techniques or more advanced ensemble methods use certain chemical descriptors in order to represent a molecule.

The most successful approaches for classification of known drugs that exploit the chemical similarity between compounds are based on ML. The goal of ML is to correctly classify the known data, and use the computed statistical model to describe the activity of a new ligand. ML methods have progressed significantly over the past few years, especially in the context of predicting the characteristics of physicochemical interactions between organic molecules and metabolic enzymes. Metabolic stability, drug metabolism and even in vivo clearance is also addressed [20]. The modeling of relationships between the chemical structure of a molecule and its metabolic effect is of great interest to the pharmaceutical industry, especially given the concurrent expansion of experimental basis. ML techniques have been used with success in massive screening, drug metabolism prediction, and in classification or quantitative prediction for large and diverse compound sets [20].

Bruce et al. [21] have carried out an assessment of different ML techniques in the context of cheminformatics, applying rigorous statistical tests and including several commonly used techniques, such as bootstrap, bagging, boosting and random forest (RF) [21]. Bruce et al. [21] used eight data sets and two different types of descriptors: 2.5D descriptors and linear fragment descriptors. The percentage of correctly classified molecules was used to validate the performance of each method on the basis of a 10-fold cross validation. Using the 2.5D descriptors, all methods correctly classify between 67% and 90% of the molecules with a large difference between individual data sets. Svetnik et al. [22] compared the performance of non-optimized standard implementations of RF, decision tree (DT) and partial least squares (PLS) for six cheminformatics data sets. In terms of prediction performance, RF ranks amongst the best algorithms. DT performs uniformly less well than RF, and PLS comes close to RF except for two datasets on which it performs less well. For some data sets the authors compared their findings with published results. For one dataset, RF was comparable to support vector machines (SVM) and artificial neural networks (ANN), where the performance of RF and SVM was similar and both outperformed ANN. In another article, Svetnik et al. [23] analyzed the performance of boosting in comparison to stochastic gradient boosting with a single DT, RF, k-nearest neighbors (kNN), PLS, naive Bayesian (NB) and SVM with both linear and radial kernels.

In our previous work, we compared results for several supervised ML algorithms, including recursive partitioning (RP), SVM, ANN, NB classification, kNN with genetic algorithm (GA)-optimized feature selection (kNN/GA), RF, DT, trend vectors (TV) and ensemble methods [24]. Here, I focus on the core question, namely how to efficiently combine these ML algorithms into a single meta-predictor (or “jury” system). In this manuscript, I present the “Brainstorming” approach, i.e., an implementation of consensus learning that combines a variety of strong and weak ML methods into a single classifier using a weighted voting procedure. This jury system approach achieves higher performance than any single method used in consensus, as confirmed by results coming from different field of bioinformatics [25–27]. Currently, the field of protein fold recognition is dominated by meta-predictors such as 3D-Jury [28, 29], Pcons [30, 31], Robetta [32–34], and many others. Multiple tests have confirmed that consensus methods are more powerful than individual prediction algorithms in terms of both sensitivity and specificity, despite the fact that some meta-predictors use as few as three methods to build a consensus model. Here, different methods are tested to estimate the quality of their performance as individual predictors, thereby avoiding some of the shortcomings of ensemble methods. The methodological details are presented in the Methods section below. The method effectively combines different powerful supervised ML methods, trained on an initial set of active compounds, into a single meta-predictor that can be used to search for unknown inhibitors.

Materials and methods

Materials

In our approach, the benchmark dataset of compounds for further training of ML algorithms was extracted from the MDL Drug Data Report (MDDR)[1]. I have selected five protein targets, namely HIV-reverse transcriptase, cyclooxygenase-2 (COX2), dihydrofolate reductase, estrogen receptor, and thrombin, from the available targets of medicinal interest. The MDDR database was used to select known inhibitors for those protein targets, then the training data was used to train a set of ML methods. Compounds annotated as "biologically tested" and ligands that have gone beyond the stage of drug discovery were excluded. Such compounds were used as positives (actives) for further training or testing. Negatives (inactives) are compounds that have not been annotated as a ligand for a given protein target, but that are considered as "launched", "Phase III", "Phase II", and "Preclinical", i.e., selective for other targets and with low cross reactivity. Such compounds are likely to be very highly active and the number of chemotypes rather restricted. Therefore, the dataset can only mimic a typical HTS situation, as a "pure" dataset is not available to the screener. However, our set, or similar subsets of the MDDR database have been used previously in various applications for method development and qualitative evaluation.

The number of negatives is much larger than the number of positives, therefore a subset of negatives with a size comparable to the size of the set of positives was randomly selected. Then all datasets (positives and negatives) were divided into training and test sets by randomly selecting two-thirds of actives and a similar number of negatives for training. The rest, i.e., one-third of the available compounds, were used for testing. Debate surrounding the influence of this procedure of selection of negatives for training of ML methods is still on-going, and some remarks regarding the five protein targets chosen here can be found in my previous works [24, 35].

To represent ligands, DRAGON chemoinformatics software was used[2]. The core idea of the brainstorming approach is not only to probe differences between types of ML algorithms, but also to simultaneously describe input data by different chemical descriptors. Both approaches, namely probing different features, or different classification methods, are valuable. Our rationale is that combining multiple representations with multiple ML methods can give additional variability to construct better meta-predictor or jury systems. Each chemical descriptor attempts to describe a ligand using a different approach, or set of features. The DRAGON software covers 1,630 descriptors of various types, including 0D (constitutional descriptors), 1D (charge descriptors and various molecular properties), 2D (walk and path counts, information indices, edge adjacency, topological charge indices, topological descriptors, connectivity indices, 2D autocorrelations, Burden eigenvalues, eigenvalue-based indices), and 3D (Randic molecular profiles, RDF and WHIM descriptors, geometrical, 3D-MoRSE, GATEWAY descriptors), as well as others, such as functional group counts, and atom-centered fragments. In the first step of this work, the whole set of descriptors was calculated for actives for all analyzed protein targets together (i.e., not separately for each protein target). Then, a principal component analysis (implemented in DRAGON software) was undertaken in order to remove dependent descriptors. This procedure divides the large set of descriptors into clusters that contain only those that are statistically dependent. In order to avoid the prohibitive high-dimensionality of chemical descriptor space, we selected only a subset of such clusters to be used for training of ML algorithms. Seven clusters were used, where chemical descriptors are correlated with: (1) regular atom pair (AP) descriptors, (2) SQ types, (3) TT (regular topological torsion), (4) DP (pairs using SQ types), (5) DT (torsions using SQ types), (6) DRUGBITS (substructures), and (7) a ROF6 set of descriptors. SQ types typically consider only non-hydrogens atoms [36]. For example, each atom has a composite "SQ type" that includes information about atomic number, hybridization, and physiochemical types (1=cations, 2=anions, 3=neutral 9H-bond donors, 4=neutral H-bond acceptors, 5=polar, unspecified H-bonding group, 6=hydrophobic, 7=other). These physiochemical types are meant to represent ionization states at physiological pH. The results of training ML methods using this diverse set of seven descriptors were previously analyzed for the five protein targets used here in order to remove those descriptors that do not provide any significant advantage in terms of recall/precision values over the simplest atom pair (AP) descriptors [24, 35]. Finally, after carefully checking the quality of the trained models, only regular AP descriptors were selected as the final representation of ligand chemical space [37, 38]. These seem to provide overall results similar to more advanced APs, and are also easy to use, and highly

[1] MDDR Drug Data Report, version 2006, MDL, 2006.Coverage: 1988–present; updated monthly. Focus: Drugs launched or under development, as referenced in the patent literature, conference proceedings, and other sources; descriptions of therapeutic action and biological activity; tracking of compounds through development phases. Size: 132,726 molecules, 129,459 models. Updates add approximately 10,000 new compounds per year.

[2] DRAGON, version 5.0, TALETE srl, Italy, 2006. An application for the calculation of molecular descriptors originally developed by the Milano Chemometrics and QSAR Research Group. These descriptors can be used to evaluate molecular structure-activity or structure-property relationships, as well as for similarity analysis and HTP screening of molecule databases. Version 5.0 was used in the present study.

interpretable. Our analysis is confirmed by other works in which AP descriptors were used with success in classifying compounds for different ML methods [24, 34, 39]. These descriptors encode molecule structures by counting, for each AP, the number of bonds that join them. Therefore, a compound is represented as a binary vector with 1 for all types of AP present, and 0 for those that are absent in the molecule.

The benchmark dataset (available at http://bio.icm.edu.pl/~darman/chemoinfo/benchmark.tar.gz) can be used for further testing of various clustering and ML methods. Previously, we compared the accuracy of compound classification by several single ML methods (including SVM, RF, ANN, k-NN with GA-optimized feature selection, TV, NB classification, DT, and others), and significant differences in the performance of these methods were observed [24].

As mentioned in the previous section, authors evaluate different ML methods in terms of the performance for a given classifier. Typically, the classification error (E), precision (P) and recall (R) values are reported, sometimes followed by receiver operating characteristic (ROC) curves analysis. E, P and R are given by following equations:

$$\begin{aligned} E &= 100\% \tfrac{FP+FN}{TP+FP+TN+FN}, \\ R &= 100\% \tfrac{TP}{TP+FN}, \\ P &= 100\% \tfrac{TP}{TP+FP} \end{aligned} \tag{1}$$

where TP is the number of true positives, FP the number of false positives, TN the number of true negatives and FN the number of false negatives. The classification error, E, provides an overall error measure, whereas recall, R, measures the percentage of correct predictions (the probability of correct prediction), and precision, P, gives the percentage of observed positives that are correctly predicted (the measure of the reliability of positive instances prediction).

Methods

The major goal of this study was to determine how much one could expect from the weighted voting technique in the context of virtual HTS. Above, I presented the results of various supervised ML approaches that are capable of learning and predicting the target-specific inhibition likelihood for different chemical compounds. The typical procedure begins initially with some experimental knowledge on inhibitors for a selected protein target; in addition, data from the MDDR, propetiary compounds collections, or commercial libraries can be used here. Now, I would like to compare those single ML algorithms with our brainstorming approach, i.e., to compare its performance within the same or similar computational setup. Therefore I decided to use a previously studied benchmark dataset [24, 40, 41]. I previously collected results for classification of different supervised ML methods for this standardized benchmark and compared their performance with other studies. In addition, I provided the overall value for classification error and precision/recall values for those ML algorithms [24]. Here, I show how far one can go with boosting the accuracy of multiple ML classification models by building a consensus between them. The meta-learning procedure is repeated separately for each protein target, therefore weights are not universal.

Let us assume that all methods have equal recall and precision values, i.e., all methods have identical quality. If the number of methods predicting a given input as a member of the positive class is equal to the number of methods predicting it as negative example, then the actual probability of success will be zero. If the negative-predicting methods have weaker quality than the actual prediction that would be given by stronger ML algorithms, the item will be classified as active one. Even if we have only a single high precision learning algorithm, it will still force the classification, as all other methods are much weaker in terms of their precision and recall values.

The model of meta-learning is based on several assumptions, as detailed in the following sections.

Binary logic

I assume the binary logic of individual predictors, i.e., we are dealing with N different ML algorithms. For the single prediction, each algorithm gives one of two opposite decisions (“YES” or “NO”), described here by the variable $\sigma_j = \pm 1$. Typically, based on trained models, ML algorithms such as SVM, DT, TV, ANN, and RF predict two classes for incoming data. Therefore, the prediction of an ML algorithm addresses a single question: is a query ligand active (“YES”) or nonactive (“NO”) for a selected protein target.

Strength parameters

Each ML algorithm is characterized typically by two parameters: $p_j = f(precision, j)$ and $s_j = f(recall, j)$ that describe the quality of predictions for the individual algorithm (described by the j index). This depends of course on the training dataset used, the values of which will be different for each protein target. Therefore, those values should be averaged over different protein targets in order to make them data-independent. The quality of the brainstorming approach depends on mean values $p = \sum \frac{p_j}{N}$ and $s = \sum \frac{s_j}{N}$ calculated over the learning algorithms used.

Probability of success

The weighted majority–minority balance in the system is given by the equation:

$$m = \frac{\sum_j \frac{(s_j+p_j)\sigma_j}{N(s+p)} + 1}{2}. \tag{2}$$

The normalized and non-negative value of m describes the probability of correct prediction, i.e., we assume here the modified or weighted vote rule. Each learner votes for the final prediction outcome, all votes are gathered, and the relative probability of correct answer is calculated, as given by the set of individual learners.

Brainstorming: the procedure of consensus learning

The global preference toward each selected solution in the brainstorming method is described as the global order parameter that is calculated using all ML algorithms used. Each algorithm (so called *learner*, or *intelligent agent*) performs its own and independent training on the available input data (both the training and testing datasets are identical for all learners). In the prediction step, a query test inhibitor is analyzed independently by each agent, which predicts the query ligand classification (active or nonactive). Then, all predictions performed by a set of learners are gathered and integrated into a single prediction via majority rule. This view of a consensus between various ML algorithms is especially useful for artificial intelligence, or robotic applications, where adaptive behavior is given by the integration of results from a set of ML methods. The consensus building between various ML algorithms, or, in other words, various predictions outcomes, is similar to the weakly coupled statistical systems known from physics. Phase transitions can be observed in the system, the global new phase emerging when the system reaches a critical point in terms of its order parameter. Changes between phases of the system are induced by certain external factors that can be modeled as a bias added to the local fields.

The binary classification, i.e., brainstorming outcome r of a prediction, is given by the sign of weighted majority–minority difference for the whole system of individual learning algorithms:

$$r = sign\left(\sum_j \frac{(s_j+p_j)\sigma_j}{N(s+p)}\right), \tag{3}$$

with the probability of success given by the parameter:

$$\langle m \rangle = \frac{\sum_j \frac{(s_j+p_j)\sigma_j}{N(s+p)} + 1}{2} \tag{4}$$

Let us assume that all methods have equal recall and precision values, i.e., all methods have identical quality. If the number of methods predicting a given input as a member of the positive class is equal to the number of methods predicting it as a negative example, then the actual probability of success will be 0.5. If the negative-predicting methods have weaker quality than the actual prediction given by stronger ML algorithms, the item will be classified as active. Even a single, high precision, learning algorithm, can force the classification, if all the other methods are much weaker in terms of their precision and recall values.

The Brainstorming implementation of the consensus learning protocol is presented in Fig. 1. The first step is focused on supervised ML training. An input set of inhibitors is first analyzed by several methods in order to represent them efficiently. The resulting numerical representations for the training data are then decomposed into their most important features using clustering algorithms and principal component analysis, and selecting the subset of representations that are not statistically dependent from each cluster. Training data prepared in this way is then used to train several different machine learning methods (SVM, ANN, RF, DT and others). The second step is the actual prediction protocol. Here, the heterogeneous predictors classify the training data differently; therefore, a consensus is needed to fuse their results. The consensus meta-learner (jury system) prepared in the classification phase can further predict the activity of a novel compound using its chemical descriptors representation.

Results

The results of this approach in the context of activity prediction for small chemical molecules are presented in Table 1.

First, I have prepared two training datasets for each protein target: positives and negatives. Then, I have represented the training examples using a predefined set of chemical descriptors (see Materials and methods for details). For each type of descriptor, a different ML algorithm was selected as the best performing. In most cases, SVM and RF were among the best. The recall and precision values for these best performing ML algorithms were then compared to a simple voting procedure. The "voting 1" method predicts a ligand as positive if at least one ML algorithm predicts it as being active. In the case of "voting **n**", at least **n** ML algorithms must be in agreement in order to predict a given ligand as active. Finally, the results were compared to the brainstorming approach, when each method votes for the final decision, with weight given by the method's recall and precision values, as computed for the training dataset. The results for different voting schemes are adopted from our previous studies [24]. Here, I compare those findings with the results of the novel

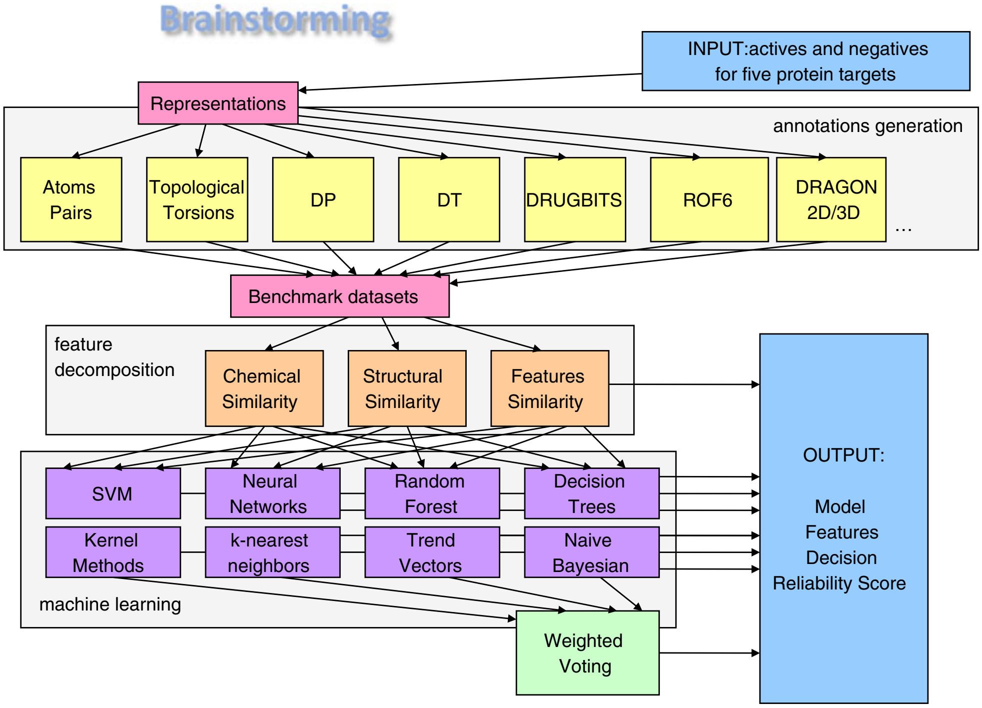

Fig. 1 Input ligands for each protein target are characterized by a set of chemical descriptors. Thus, each ligand is represented as a vector of real or binary numbers in a high dimensional abstract space of features. All training inhibitors, their features, and some additional information are then processed by feature decomposition module in order to evaluate the statistical significance of each chemical descriptor or representation, and to find some similarities between features, or annotations. In this way, the algorithms prepare a set of benchmark datasets with which to probe different representations of training data. Such preprocessed datasets are then used for training seven different machine learning (ML) methods [support vector machines (SVM), random forest (RF), artificial neural networks (ANN), k-nearest-neighbor (kNN) classification with genetic-algorithm (GA)-optimized feature selection, trend vectors (TV), naïve Bayesian (NB) classification, and decision trees (DT)]. Each ML is used independently to predict class membership for a query object. The results of such ensemble prediction are then fused into the single consensus prediction by a simple weighted voting procedure. The final output includes predicted class membership, a statistical model with performances of each learning module, trained consensus and reliability scores for prediction (calculated as described in Methods)

brainstorming procedure (summarized in last row of Table 1). Almost all the compounds were retrieved by at least one of the seven ML methods for each target; almost all were found to be actives. However, because the recall values obtained with the best performing ML method alone are already close to or above 90%, any improvement compared to this approach is small. In addition, a significant reduction in precision compared to the best

Table 1 The recall (R) and precision (P) values for the brainstorming method, as compared with the simple vote procedure or single machine learning (ML) prediction

Prediction method	Cyclooxygenase-2		Dihydrofolate reductase		Thrombin		HIV reverse transcriptase		Estrogen receptor	
	R	P	R	P	R	P	R	P	R	P
Best performing single ML method	92	71	96	79	96	82	83	55	92	61
Voting 1	93	28	100	19	99	35	94	20	93	21
Voting 2	85	57	97	54	95	67	87	47	94	53
Voting 3	87	79	96	76	98	79	84	74	86	73
Voting 4	82	90	98	86	91	82	79	88	83	78
Voting 5	75	93	91	92	89	89	64	90	76	82
Voting 6	72	98	89	97	83	92	49	96	33	86
Voting 7	64	100	75	99	52	94	24	100	23	100
Brainstorming	87	92	100	92	95	89	84	89	89	82

ML method is observed. On the other hand, if the “voting 7” procedure is applied, recall values drop to around 20% for HIV reverse transcriptase and estrogen receptor. Yet, even with this method, the number of false positives is very small and the precision is over 95% for all targets. With intermediate simple voting methods, an improvement in precision is observed when going from compounds found at least 1, 2, 3 and so on to those found 4, 5 or 7 times. At the same time, recall does not decrease significantly. In the case of brainstorming, one is able to retrieve active compounds with very high efficiency, exceeding both simple voting procedures and the best performing single ML method. Thus, whereas the consensus learning approach offer a significant advantage in terms of recall compared to single ML algorithms, at the same time precision improves significantly for the consensus results compared to any ML method applied alone. The brainstorming approach is able to boost the overall recall value by at least 20–50% with only a very small drop in precision (~8–18%).

Here, the brainstorming algorithm was compared with seven simple voting procedures. All differ in the number of methods assumed to be in agreement when activity prediction is performed. The most conservative method identifies a query compound as active if all ML methods agree with their predictions. The most liberal predict activity if at least one ML method predicts it as being active. As reported in our previous studies [24], the most liberal voting scheme achieves a very high improvement in recall value (~25%) as compared to any single ML method (even the best performing methods, i.e., SVM and RF); however, precision is lowered significantly (~30%). This method is quite useful in cases in which only a small number of actives is known—selecting all actives identified by at least one method helps to improve recall. On the contrary, when all methods are thought to agree with their predictions, the actual value of precision for that method reaches a maximal value (~95–100%), but recall is significantly lower (~20%). At intermediate numbers of agreed voting methods, an improvement in precision is observed when going from compounds found at least once to those found twice and three times. At the same time, recall decreases significantly. This consensus approach is particularly successful in identifying false positives and thus improving precision. In addition, I reported that, whereas the consensus approach does not offer any significant advantage in terms of recall compared to SVM, precision improves significantly for the consensus results compared to SVM or RF applied alone. In the case of the combination of two methods, the results somehow fall in the middle between the two extremes [24]. In order to avoid false negatives, the results for any pair of methods are combined in such a way that compounds identified by any two methods as active are considered to be active. Similarly, only compounds predicted to be inactive by both methods were counted as nonactives. Most combinations did not show any significant improvement in recall compared to results obtained from SVM or RF alone. Some combinations led to improved recall, yet for others no improvement in recall was observed, while precision was lowered significantly.

When one method is significantly better in predicting activity, then combining it with a set of much weaker methods does not improve quality unless we do not adjust its relative weight. For example, in the case where SVM alone yields excellent recall values for all protein targets, precision is reduced for each simple combination with respect to SVM alone [24]. The improvement in recall is explained by the fact that more than one ML method is used to select active compounds. Precision is retained as compared to the best performing method, because weaker ones are weighted less. If, within the whole set of ML algorithms, there are more better performing algorithms, the effective combinations of several best performing methods demonstrate a more substantial improvement of recall. However, if simple voting only is performed, the lowering of the number of false negatives will always reduce precision with respect to the best performing SVM alone. On the contrary, if votes of methods are weighted by their quality, then the overall gain in recall is accompanied by reasonable stability of precision. A combination of methods seems to be particularly useful for datasets poor in active compounds, where high values of recall are somehow crucial for further virtual screening procedures.

Conclusions

Ensemble methods that use several different ML algorithms together with several types of chemical descriptors to properly classify groups of compounds were proven here to be the more successful in terms of recall and precision in comparison to single ML methods and vote counting. Consensus learning appears to enrich datasets more than any single scoring function. Multiple scoring functions are similar to repeated samplings (the mean is closer to the true value than any single value). Considering all compounds retrieved by at least one of the seven methods for each target, almost all actives were found in the dataset of all active compounds. The number of false negatives for each target is smaller than 10% (recall>90%). However, the sample size is large since, for each target, more than 60% of the compounds predicted to be active by at least one method are false positives. If only compounds retrieved by all methods are considered, recall drops but is still substantial. At intermediate consensus numbers, a large improvement in precision is observed when going from compounds found at least once to those found twice and three times. At the same time, recall does not decrease significantly. Thus, for compounds retrieved by at least three methods, the average recall is 90%, whereas precision is 75% on average. Considering higher consensus numbers, recall values are lower than for SVM or RF applied alone, but a significant improvement in precision is observed, suggesting that a consensus approach is particularly effective at reducing the number of false positives. Two possible applications of the consensus approach are proposed. In cases in which only a small number of actives is known, selecting all actives identified by at least one method helps to reduce the number of false negatives. If the objective is to reduce the number of compounds to be tested, considering compounds that were found by several methods helps to reduce the number of false positives. Summarizing, we observed significant differences in the performance of the methods used; however, the consensus learning that integrates these classification schemes is able to boost precision and recall values independently of the protein target, or compound class. I conclude that brainstorming is a more efficient way to predict a compound's biological activity that any single ML algorithm.

Some shortcomings of this method should be stressed. First, it requires extensive calculation of weights for used ML methods, which is very elaborate to perform. Second, results depend on the diversity of learning algorithms. Third, when the number of known actives is too low, it is impossible to train a set of ML algorithms. In such practical situations, I would suggest performing more elaborate QSAR studies and detailed analysis of the protein target active site, or that docking software should be used to prepare an initial dataset of possible positives for further application of ML algorithms as described in some of our previous publications [35, 40, 41].

Acknowledgments This work was supported by the Polish Ministry of Education and Science (N301 159735 and others), PEPLASER (HEALTH-2007-223243) and OxyGreen (KBBE-2007-212281) 6FP projects. The author would like to thank Uwe Koch and Stephane Spieser for cooperation in preparing benchmark datasets, and extensive training of the various machine learning methods. The results of their work were used here for training and testing the brainstorming approach. The author would like to thank an anonymous reviewer for fruitful comments and suggestions that strongly enhanced the results of this work.

References

1. Pang YP (2007) Clin Pharmacol Ther 81:30–34
2. Wagner BK, Haggarty SJ, Clemons PA (2004) Am J Pharmacogenomics 4:313–320
3. Pillutla RC, Fisher PB, Blume AJ, Goldstein NI (2002) Expert Opin Ther Targets 6:517–531
4. Li Z, Wang RS, Zhang XS, Chen L (2009) IET Syst Biol 3:523–533
5. Monfregola L, Vitale RM, Amodeo P, De Luca S (2009) Bioorg Med Chem 17:7015–7020
6. den Besten M, Thomas AJ, Schroeder R (2009) J Biomed Discov Collab 4:5–20
7. Williams C, Schreyer SK (2009) Comb Chem High Throughput Screen 12:424–439
8. Peifer M, Weiss J, Sos ML, Koker M, Heynck S, Netzer C, Fischer S, Rode H, Rauh D, Rahnenfuhrer J, Thomas RK PLoS ONE 5:e8919
9. Ewing T, Baber JC, Feher M (2006) J Chem Inf Model 46:2423–2431
10. Hert J, Willett P, Wilton DJ, Acklin P, Azzaoui K, Jacoby E, Schuffenhauer A (2004) Org Biomol Chem 2:3256–3266
11. Willett P (2006) Drug Discov Today 11:1046–1053
12. Whittle M, Gillet VJ, Willett P, Alex A, Loesel J (2004) J Chem Inf Comput Sci 44:1840–1848
13. Whittle M, Gillet VJ, Willett P, Loesel J (2006) J Chem Inf Model 46:2206–2219
14. Whittle M, Gillet VJ, Willett P, Loesel J (2006) J Chem Inf Model 46:2193–2205
15. Parikh D, Polikar R (2007) IEEE Trans Syst Man Cybern B Cybern 37:437–450
16. Salim N, Holliday J, Willett P (2003) J Chem Inf Comput Sci 43:435–442
17. Wilton DJ, Harrison RF, Willett P, Delaney J, Lawson K, Mullier G (2006) J Chem Inf Model 46:471–477
18. Willett P, Wilton D, Hartzoulakis B, Tang R, Ford J, Madge D (2007) J Chem Inf Model 47:1961–1966
19. Hert J, Willett P, Wilton DJ, Acklin P, Azzaoui K, Jacoby E, Schuffenhauer A (2006) J Chem Inf Model 46:462–470
20. Fox T, Kriegl JM (2006) Curr Top Med Chem 6:1579–1591
21. Bruce CL, Melville JL, Pickett SD, Hirst JD (2007) J Chem Inf Model 47:219–227
22. Svetnik V, Liaw A, Tong C, Culberson JC, Sheridan RP, Feuston BP (2003) J Chem Inf Comput Sci 43:1947–1958
23. Svetnik V, Wang T, Tong C, Liaw A, Sheridan RP, Song Q (2005) J Chem Inf Model 45:786–799
24. Plewczynski D, Spieser SA, Koch U (2006) J Chem Inf Model 46:1098–1106
25. Sen TZ, Cheng H, Kloczkowski A, Jernigan RL (2006) Protein Sci 15:2499–2506
26. Sen TZ, Kloczkowski A, Jernigan RL, Yan C, Honavar V, Ho KM, Wang CZ, Ihm Y, Cao H, Gu X, Dobbs D (2004) BMC Bioinform 5:205–216
27. Basu S, Plewczynski D (2010) BMC Bioinform 11:210–225
28. Ginalski K, Elofsson A, Fischer D, Rychlewski L (2003) Bioinformatics 19:1015–1018
29. von Grotthuss M, Pas J, Wyrwicz L, Ginalski K, Rychlewski L (2003) Proteins 53(Suppl 6):418–423
30. Bujnicki JM, Elofsson A, Fischer D, Rychlewski L (2001) Bioinformatics 17:750–751
31. Lundstrom J, Rychlewski L, Bujnicki J, Elofsson A (2001) Protein Sci 10:2354–2362
32. Chivian D, Kim DE, Malmstrom L, Bradley P, Robertson T, Murphy P, Strauss CE, Bonneau R, Rohl CA, Baker D (2003) Proteins 53(Suppl 6):524–533
33. Chivian D, Kim DE, Malmstrom L, Schonbrun J, Rohl CA, Baker D (2005) Proteins 61(Suppl 7):157–166
34. Kim DE, Chivian D, Baker D (2004) Nucleic Acids Res 32(Web Server issue):W526–W531
35. Plewczynski D, von Grotthuss M, Rychlewski L, Ginalski K (2009) Comb Chem High Throughput Screen 12:484–489

36. Miller MD, Sheridan RP, Kearsley SK (1999) J Med Chem 42:1505–1514
37. Carhart RE, Smith DH (1985) J Chem Inf Comput Sci 25:64–73
38. Sheridan RP, Nachbar RB, Bush BL (1994) J Comput Aided Mol Des 8:323–340
39. Sheridan RP (2000) J Chem Inf Comput Sci 40:1456–1469
40. Plewczynski D, Spieser SAH, Koch U (2009) Comb Chem High Throughput Screen 12:358–368
41. Plewczynski D, von Grotthuss M, Spieser SA, Rychewski L, Wyrwicz LS, Ginalski K, Koch U (2007) Comb Chem High Throughput Screen 10:189–196