

Mixed learning algorithms and features ensemble in hepatotoxicity prediction

Chin Yee Liew · Yen Ching Lim · Chun Wei Yap

Received: 7 February 2011 / Accepted: 23 August 2011 / Published online: 6 September 2011
© Springer Science+Business Media B.V. 2011

Abstract Drug-induced liver injury, although infrequent, is an important safety concern that can lead to fatality in patients and failure in drug developments. In this study, we have used an ensemble of mixed learning algorithms and mixed features for the development of a model to predict hepatic effects. This robust method is based on the premise that no single learning algorithm is optimum for all modelling problems. An ensemble model of 617 base classifiers was built from a diverse set of 1,087 compounds. The ensemble model was validated internally with five-fold cross-validation and 25 rounds of y-randomization. In the external validation of 120 compounds, the ensemble model had achieved an accuracy of 75.0%, sensitivity of 81.9% and specificity of 64.6%. The model was also able to identify 22 of 23 withdrawn drugs or drugs with black box warning against hepatotoxicity. Dronedarone which is associated with severe liver injuries, announced in a recent FDA drug safety communication, was predicted as hepatotoxic by the ensemble model. It was found that the ensemble model was capable of classifying positive compounds (with hepatic effects) well, but less so on negatives compounds when they were structurally similar. The ensemble model built in this study is made available for public use.

Keywords Ensemble · Consensus · Meta-learner · Mixed variables · Mixed algorithm · Prediction · Drug-induced liver injuries · Hepatotoxicity · Drug discovery · Support vector machine · k-Nearest neighbor · Naive Bayes · QSTR

Introduction

The liver is highly susceptible to the insults of drugs and chemicals as it has an important role in metabolizing xenobiotics. It was estimated that around 5–10% of adverse drug reactions resulted in liver injuries [1]. The degree of drug-induced liver injuries (DILI) can vary from damage that is mild, such as transient elevation of liver enzymes, to severe injuries such as liver cirrhosis and fulminant hepatic failure. Approximately 50% of fulminant hepatic failure was caused by adverse reaction of ingested medicaments. The rate of mortality or liver transplantation for these patients was estimated at 9.2%. Considering the morbid consequences of DILI, it is unsurprising that liver injury is one of the drug safety aspects that can prevent the registration of drugs or results in the withdrawal of marketed drugs such as Troglitazone, Bromfenac and Ticrynafen.

The occurrence of hepatotoxicity is a result of multiple factors. The drug may be inherently hepatotoxic or its metabolite is reactive causing undesirable consequences in the human body [2]. Moreover the level of exposure, environmental factors, and genetic factors could play a role in hepatotoxicity [3]. The multitude of factors may confound human judgment and require expert interpretation in hepatotoxicity prediction. Consequently, the prediction of hepatotoxicity in the preclinical stages is often difficult [4]. Although automated prediction tools are very much needed in drug development, the accuracy of many currently

Electronic supplementary material The online version of this article (doi:10.1007/s10822-011-9468-3) contains supplementary material, which is available to authorized users.

C. Y. Liew · Y. C. Lim · C. W. Yap (✉)
Department of Pharmacy, Pharmaceutical Data Exploration
Laboratory, National University of Singapore, Singapore,
Singapore
e-mail: phayapc@nus.edu.sg

available *in silico* methods. For example, global models for prediction of diverse compounds are relatively poor [5, 6]. This could be caused by the lack of toxicity data and the difficulty in building a predictive model for an effect which has many underlying mechanisms and factors [7, 8].

Preclinical tools such as, DEREK, METEOR [9], and MetabolExpert [10], which can predict metabolism or reactive metabolite formation, can be used to sieve out potential toxicant early. However, they may sometimes give high false positive or false negative rates in their predictions [11]. Efforts to improve the prediction performance have been attempted. Currently, hepatotoxicity can be predicted by a variety of cell-based (in vitro) systems [12], biochemical pathway kinetics [13], or through the use of *in silico* models of in vitro measurements such as gene profiling [14, 15] and metabonomics [16]. These methods were made possible by the many causative aspects in hepatotoxicity, such as the molecular structure, genetics, metabonomics, and environmental factors which may be explored for their predictive value.

There are a variety of cell-based (in vitro) methods that evaluate a mixture of endpoints which can lead to DILI, for example, cell necrosis, metabolite-mediated toxicity, inhibition of biliary transporters, etc. [12, 17–21]. These methods can achieve a sensitivity of 1–70% [12, 17, 20, 21]. Therefore, cell-based methods with acceptable predictivity are useful replacement or complementary to in vivo methods because they are cheaper and easier to handle. However, they are neither cheaper nor more efficient alternatives compared to pure *in silico* methods because of the in vitro component. Biological or chemical experimentations are still required to generate the data needed as inputs into these prediction systems. Although in vitro methods are established techniques that complement or substitute the use of animal testing, these methods are not truly identical to in vivo systems. There may be species specific toxicity, e.g., toxicity in rats which may not occur in humans, or differences in drugs concentration required to elicit a toxic response between in vitro and in vivo. In other cases, absence of organ-specific heterotypic cell–cell interactions, deterioration of key metabolism genes expression, or inadequate supply of human tissues may restrict the use of in vitro methods [19, 20]. Therefore, a number of pure *in silico* hepatotoxicity prediction methods had been reported. These predictive models were generated from a variety of data sets of different endpoints related to hepatotoxicity and the models were made of different algorithms and methodologies [4, 22–27]. Two of these hepatotoxicity studies reported the use of consensus of support vector machine (SVM) or k-nearest neighbor (kNN) models trained from mixed training sets and optimized (mixed) features [26–29]. Here, we report an alternative consensus method which

involves the ensemble of models of mixed features and mixed learning algorithms.

Ensemble method, or consensus modelling, is a technique introduced to modelling studies to improve the accuracies of individual classifiers. These individual (constituent) classifiers, which were referred to as *base classifiers* or *base models* in the manuscript, form the “bottom layer” where the ensemble method is applied on. The ensemble method has shown better performances of varying degree in quantitative structure–activity relationship (QSAR) studies [30–34] and quantitative structure–toxicity relationship (QSTR) studies [35–38] compared to predictions from a single classifier. There is an assortment of approaches to generate multiple models that form the base classifiers of an ensemble model [39]. Other than using various learning algorithms (Al_{var}), one may generate multiple models by varying the training set (T_{var}) through sampling methods like bagging and boosting. One may also generate many models from the same training set but using different subset of features (F_{var}). Most of these approaches have been explored, but frequently used were ensemble of fixed learning algorithm with varied features and (or) varied training set ($T_{fix}Al_{fix}F_{var}$ or $T_{var}Al_{fix}F_{var}$) [35, 36, 40–48]. For example, a technique that uses mixed training set and features ($T_{var}Al_{fix}F_{var}$) is the Random Forest [49] technique which is an ensemble of many decision trees built from a variation of sample and features.

In this study, we have used an ensemble method ($T_{fix}Al_{var}F_{var}$) that combines different models not only generated by a variety of feature subsets, but also a few learning algorithms on the basis of no sole learning algorithm can best model a variety of problems [29]. The method uses a fixed amount of training data on the basis that a model should learn from as many sample as possible to exploit all available information. To the best of our knowledge, there were eight other QSAR studies on ensemble of mixed features and mixed algorithms, applied on training set size of 48–816 compounds [33, 50–56]. The application of ensemble method had improved the final performances in a majority of these studies when compared with the best performing individual model [52, 55]. Nevertheless, the ensemble of a few base classifiers was preferred over individual model as ensemble could be more robust. The single classifier may have been selected by chance and it may represent a smaller solution space. To the best of our knowledge, this study is the first hepatotoxicological study that applied the proposed ensemble method, known as $T_{fix}Al_{var}F_{var}$ ensemble from hereafter, to a medium-large data size (1,087 training compounds) validated with at least 120 compounds. We had used a range-based applicability domain on the ensemble method in this study. The model built from diverse compounds was

validated through internal validation, y-randomization, and a few external validation sets.

Methods

Modelling set

The U.S. FDA Orange Book [57] was used to obtain a list of available drugs in the market. These drugs were checked for adverse hepatic effects using the Micromedex® Healthcare Series [58] which has reports on adverse reactions in each drug's monograph. In this study, adverse hepatic effects were grouped into different levels according to the severity: (level 1) transient and asymptomatic liver function abnormalities (level 2) liver function abnormalities, hyperbilirubinaemia (level 3) hepatitis, jaundice, cholestasis (level 4) fulminant hepatitis, liver failure, and (level 5) fatality. When any of these effects was associated with a drug, even with one case report of transient liver function abnormalities, the drug was labeled as “positive”, that is, with adverse hepatic effects in our data set. We had taken an extremely reserved approach in the labeling so that any drug with the potential to cause any adverse liver effects was flagged as “positive”. If a drug was not associated with any adverse hepatic effects, it was labeled as “negative”, that is (level 0) without hepatic effects. Besides the list of drugs from the FDA Orange book, other pharmaceutical and non-pharmaceutical compounds were added into the data set by searches using keywords like hepatic effect, hepatitis, and jaundice in Micromedex. The Merck Index [59] and the book, Drug-Induced Liver Disease [60], were used as sources for more compounds.

In total, 1,685 compounds (not processed) were collected. Compounds with unclear hepatic effects reports, duplicates, combination products, inorganic compounds, and compounds with molecular weight of greater than 5,000 were removed as molecular descriptor calculation does not handle them well. A total of 1,274 descriptor-calculable compounds were available for the subsequent analysis and modelling processes (Online Resource 1). Three independent external validation sets, with a total of 187 compounds, were drawn out from the 1,274 collected compounds. The remaining 1,087 compounds (654 positives and 433 negatives) were used for model building. The 2D structures of all collected compounds were downloaded from PubChem [61] or drawn using ChemDraw [62]. Subsequently, Pipeline Pilot Student Edition [63] was used to standardize the structures by adding hydrogens and removing salts, while the 3D coordinates were generated by using Corina [64].

Validation sets

The first validation set, valBLACK, contained 47 compounds. The positive compounds consisted of 23 drugs withdrawn from the market or those with black box warning for hepatotoxicity [65]. This is to validate the model's ability to predict “severely” toxic compounds. A comparable number of negative compounds were added to this data set to enable the calculation of precision for the positive (toxic) class, that is, the correctness of classifications predicted as positives. These 24 nontoxic compounds were obtained through the process as shown in Fig. 1. Greene et al. have reported 152 compounds that have no evidence for hepatotoxicity in human and animal in their validation set. This list was further reduced by checking for compounds that were duplicated in our collected data which were also not associated with hepatotoxicity. From this refined set, Kennard-Stone sampling was applied to select training compounds that gave the balance of 24 nontoxic compounds in valBLACK.

In a recent FDA drug safety communication [66], the heart medication dronedarone was associated with rare cases of severe liver injuries including two cases of acute liver failure. Dronedarone was approved in July 2009 in the United States by the FDA. The announcement came at the end of the experiments; hence, this compound was not present in our training set and was tested by the ensemble model.

The second validation set, valPAIR, consisted of 20 compounds from 10 pairs of structurally similar compounds but of opposing toxicity status. For example, doxorubicin and epirubicin which are hepatotoxic and not hepatotoxic respectively. The 20 compounds in Fig. 2 are the top ten most similar compound pairs measured by 3-nearest neighbor in Manhattan distance.

The third validation set, valRANDOM, consisted of 120 compounds obtained through stratified sampling of the data set. Stratified sampling was used to keep the original ratio of positive to negative compounds in the training set, thus, the resultant valRANDOM has 48 negative compounds and 72 positive compounds.

Molecular descriptors

Molecular descriptors are quantitative representations of structural and physicochemical features of molecules. The program, PaDEL-Descriptor version 2.0, was used in the calculation of molecular descriptors and Klekota-Roth substructures in this study. The list of molecular descriptors is available in the PaDEL-Descriptor website [67].

PaDEL-Descriptor [68] is an open source Java-based software developed using the Chemistry Development Kit for the calculation of molecular descriptors and fingerprints.

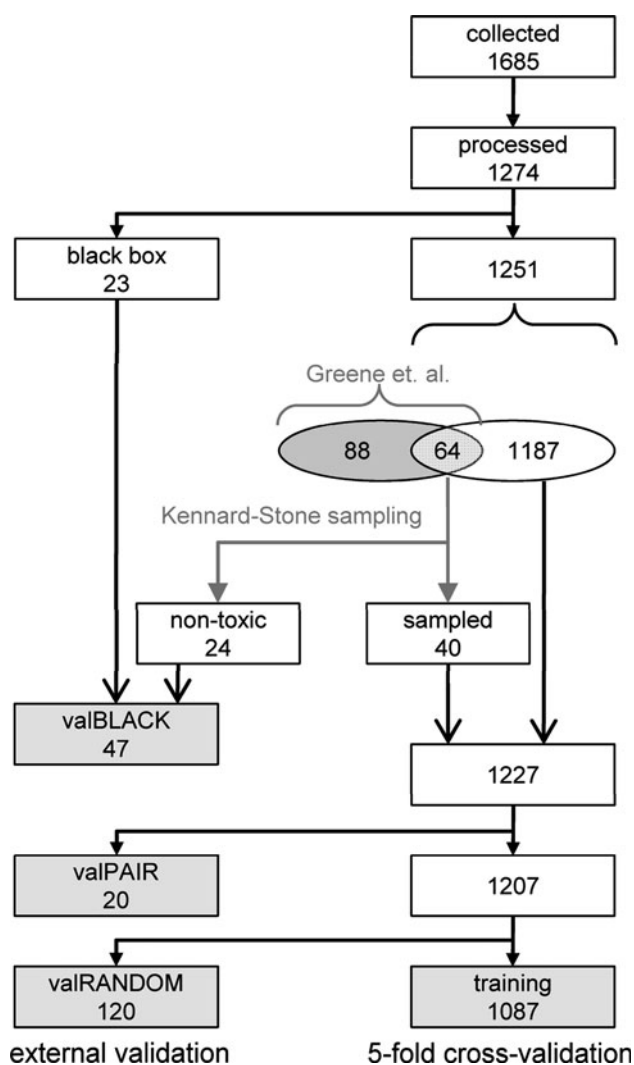


Fig. 1 The number of compounds in each data set. The compounds set aside for external validation were never used during the modelling process

It has a graphical user interface, a command line interface and can be used as an extension to RapidMiner. Currently, it can calculate 797 descriptors and 10 types of fingerprints which includes 1D, 2D and 3D descriptors, e.g., atom type electrotopological state descriptors, McGowan volume, molecular linear free energy relation descriptors, ring counts, WHIM, Petitjean shape index, count of chemical substructures identified by Lagner, and binary fingerprints/count of chemical substructures identified by Klekota and Roth.

k-Nearest neighbor

k-Nearest Neighbor (kNN) is a type of lazy learner whereby it delays the learning of the training data until it is needed to classify a test sample [39]. kNN does not produce a model and it classifies a test compound by searching

for the training compounds that are similar in characteristics to the test compounds. The class of the test compound will be determined by the majority class of its neighbors. For this work, the best kNN model was obtained by optimizing simultaneously: the data normalization method, the number of nearest neighbor, k , and the distance measures, for example cosine similarity, Euclidean, or Manhattan distance.

Support vector machine

Support vector machine (SVM) is a machine learning method based on statistical learning theory [69]. It is a classifier that is less affected by duplicated data and has lower risk of model overfitting [39]. In linearly separable data, SVM tries to build a maximal margin hyperplane to separate positive compounds from negative compounds.

Nonlinear SVM is useful for classifying compounds of diverse structures which are usually not linearly separable. SVM maps the input vectors into a higher dimensional feature space by using a kernel function. The Gaussian radial basis function kernel which has been widely used and had consistently shown better performance [70, 71] were used in this study,

$$K(x_i, x_j) = e^{-\|x_i - x_j\|^2 / 2\sigma^2} \quad (1)$$

For the SVM model in this study, a margin of $C = 100,000$ was used and Brent's minimization algorithm [72] was used to find the optimal gamma in RapidMiner.

Naive Bayes

Naive Bayes (NB) is a simple classifier derived from the well known Bayes' theorem. It assumes independence among the molecular descriptors. In training, the classifier tries to learn the relationship between the class label and the molecular descriptors probabilistically, after which the class of an unknown compound is found by maximizing its conditional probability [39].

Performance measures

The classification performance of machine learning methods can be assessed by the quantity of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) [73]. The prediction accuracy for positive compounds (with adverse hepatic effects) and negative compounds (without adverse hepatic effects) are sensitivity:

$$SEN = \frac{TP}{(TP + FN)} \quad (2)$$

and specificity respectively:





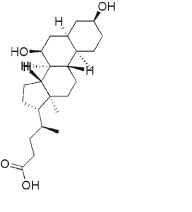
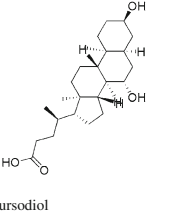
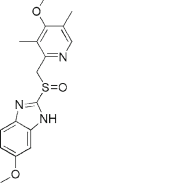
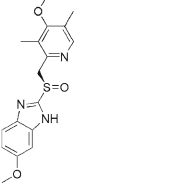
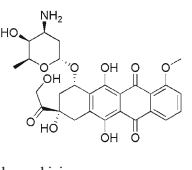
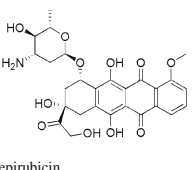






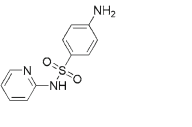
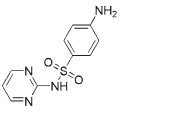
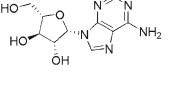
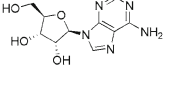
pair	hepatotoxic compounds (positive compounds)	prediction	non-hepatotoxic compounds (negative compounds)	prediction
1	 niacin	positive (out of AD)	 benzoic Acid	positive
2	 ethylenediamine	positive	 ethanolamine	positive
3	 chenodiol	negative	 ursodiol	negative
4	 omeprazole	positive	 esomeprazole	positive
5	 doxorubicin	positive	 epirubicin	positive
6	 tranlycypromine	positive	 amphetamine	negative
7	 fluoroacetic acid	positive (out of AD)	 acetohydroxamic	positive (out of AD)
8	 methylphenidate	negative	 dexmethylphenidate	negative
9	 sulfapyridine	positive	 sulfadiazine	positive
10	 vidarabine	positive	 adenosine	positive

Fig. 2 Prediction results of structurally similar pairs but of opposing hepatic effect potential

$$SPE = \frac{TN}{(TN + FP)} \quad (3)$$

The overall prediction performance can be calculated by the overall prediction accuracy (ACC):

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

Matthew's correlation coefficient [74] (MCC):

$$MCC = \frac{TP \times TN - FN \times FP}{\sqrt{(TP + FN)(TP + FP)(TN + FN)(TN + FP)}} \quad (5)$$

and the geometric mean (GMEAN):

$$GMEAN = \sqrt{SEN \times SPE} \quad (6)$$

The precision for positive prediction is:

$$PRE = \frac{TP}{(TP + FP)} \quad (7)$$

which is the ratio of actual hepatotoxic compounds to all compounds predicted as toxic. The area under a ROC curve (AUC), which has been widely used as a general measure of model predictiveness in many fields [75, 76] was reported. Due to its calculation algorithm, there are three types of ROC curves that may be reported: optimistic, expected and pessimistic ROC curves [75]. As the names suggest, the performance in terms of optimistic ROC will appear better than the pessimistic performance for the same prediction exercise. For this study, the pessimistic AUC (AUC(pes)) were reported and used during the model optimization process. The AUC value falls between 0 and 1, of which useful classifiers should not have an AUC of less than 0.5.

Modelling

All models were built and optimized using RapidMiner [77]. The model building process is illustrated in Fig. 3. The gist of the process is to generate many base classifiers to form an ensemble model when they satisfy a cutoff criterion. The full data set of 1,087 compounds was used for every step and the main steps are:

1. Generate different training data sets which had different subsets of molecular descriptors, i.e. vary(MDes) as shown in Fig. 3.
2. Produce different kNN models for each of the training data sets generated in step 1 with different combination of *k*, distance measures and normalization method.

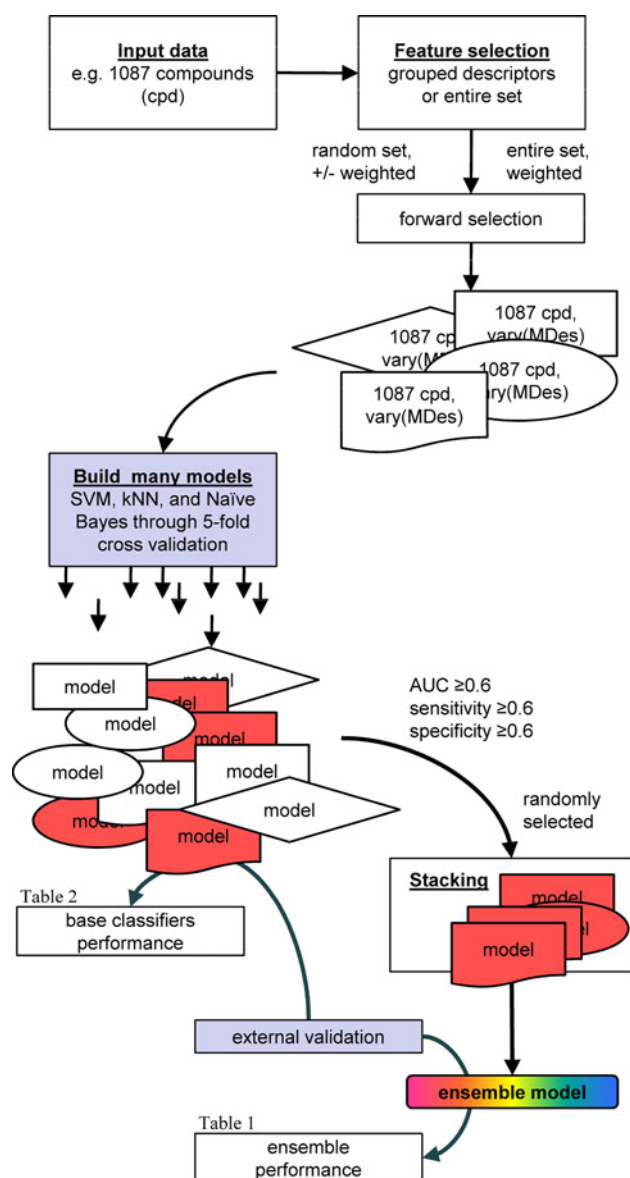


Fig. 3 General flow of the modelling process. Many SVM, kNN or naive Bayes models were generated from the same number of compounds but with differing molecular descriptor set referred to as *vary(MDes)* in the figure

- Repeat step 2 with SVM models of different gamma optimized by the Brent's minimization algorithm.
- Repeat step 2 with naive Bayes models.
- Select models produced from step 2, 3, and 4 which fulfill the criteria of $AUC \geq 0.6$, $SEN \geq 0.6$ and $SPE \geq 0.6$.
- From the pool of models produced from step 5, eliminate models with duplicated molecular descriptors set or those with only one molecular descriptor. Subsequently, apply the ensemble method, stacking with naive Bayes, on the selected base classifiers to give an ensemble model.

To obtain a large number of training sets with *vary(MDes)* for step 1, two methods were used in this study. The first method to generate *vary(MDes)* was to take the full set of molecular descriptors and weigh each molecular descriptor with respect to the class label. Molecular descriptors that have no influence on the class label will receive a weight of 0, while the most influential descriptor will receive a weight of 1. The remaining descriptors will receive weights between 0 and 1 depending on their influence on the class label. In this study, the symmetrical uncertainty method [78] was used to weigh the descriptors. Subsequently, six training sets with *vary(MDes)* were obtained by varying the cutoff weights from ≥ 0 to ≥ 0.5 at an increment of 0.1. Each of these six sets of 1,087 compounds with *vary(MDes)* underwent the modelling process in step 2, 3, and 4, where SVM, kNN and NB models were built.

The benefits of an ensemble method can be potentiated if the base classifiers are diverse [30, 79]. To make each training set of *vary(MDes)* more distinct, the second method was to categorize the full set of molecular descriptors into 13 groups according to their descriptor types. Example groups include electrotopological state descriptors, charged partial surface area descriptors, autocorrelation descriptors, WHIM descriptors and 2D miscellaneous descriptors; the complete list of grouping with their descriptors is available in Table 2 of Online Resource 2. For each of the groups, a random number of descriptors will be selected. Subsequently, the descriptors were passed on to the weighting procedure (as in the first method) before further refinement through forward selection with NB. Therefore, the number of training sets with *vary(MDes)* is the product of the number of descriptor groups used, the number of times of random sampling of descriptors, and the number of cutoff in the weighting procedure. Although many combinations are possible, we have restricted the combination to 8 descriptor groups, 10 rounds of random sampling and 6 cutoff weights for this study due to the limitation in computation resources. Each of these 480 sets of 1,087 compounds with *vary(MDes)* underwent step 2, 3, and 4 where a pool of models were built.

From the pool of individual models selected at step 5, the number of these models was further reduced at step 6 to retain unique models. These shortlisted models were sampled at an increasing number to compile the constituent models, also known as the base classifiers set (nBase), for the ensemble method. For example, starting with random sample of 5 base models, an ensemble was built. Next, 9 models were randomly sampled and a new ensemble was built. The ensemble size was increased by four until all base classifiers (largest odd number) were included into an ensemble. Random sampling of models was used because

of its efficiency and ease of use. In addition, the random method had shown to be effective in methods such as Random Forest and Random Decision Trees [50, 80]. This process was repeated for 50 times, that is, there were 50 ensemble models built from each combination of 5 base classifiers, 9 base classifiers, etc. Subsequently, the averages of the 50 training set performance values, for each base classifier combination ($n_{\text{Base}} = 5, 9, \dots, 793$) were obtained. The number of base classifiers where the average AUC(pes) starts to plateau (i.e., no increase in average AUC(pes) for five consecutive combinations) was taken as the minimum number and only one ensemble model (highest AUC(pes)) among the 50 replicates was selected as the final model.

Ensemble method

Stacking or stacked generalization [81] with NB was used as the ensemble method. Briefly, in stacking, the NB learns from the predictions of the selected base classifiers to produce the final ensemble model. That is, in this study, modelling occurred at two levels: first, when the base classifiers (N_{base} individual models) were generated; second, when another learner or rule was used to build a model from the (N_{base}) predictions of the base classifiers combined with the attributes in the training set as features. For example, if the training set has 100 molecular descriptors and there were 50 base classifiers selected, the learner at the second level will have at least 150 features for its modelling process. Any type of learners or rules may be applied at the second level, for example, applying a majority voting method or linear regression. For this study, the naive Bayes algorithm had been used because it is fast and minimal optimization is required.

Y-randomization

Y-randomization was carried out to establish the statistical significance and robustness of the ensemble model [82]. The performance of the y-scrambled models should be significantly lower than the models generated from unaltered data, therefore, it was expected that the number of base classifiers fulfilling the cutoff criteria to form an ensemble model will be significantly lesser. In this study, we have adopted the procedure where the y (with or without adverse hepatic effects) of the data is randomly permuted while the molecular descriptors were kept unaltered. The y-scrambled data set underwent the same model building process, that is, generation of a pool of models and selection of models with $\text{AUC} \geq 0.6$, $\text{SEN} \geq 0.6$ and $\text{SPE} \geq 0.6$ for the final ensemble model to be validated with the 120 compounds in external validation. The y-randomization was repeated for 25 times as per

recommended by Rücker et al. in a y-randomization study [82].

Applicability domain

The applicability domain (AD) of the ensemble model was calculated based on the range [83] of the individual features. That is, the minimum and maximum values of each molecular descriptor in consideration of all the compounds in the training set were used. For this work, the AD is defined by a hyper-rectangle and compounds that violated one or more of the molecular descriptor ranges were highlighted in the prediction process.

Results

Hepatic effects prediction

From the 1,087 training set, 17,012 models (14580 kNN, 1946 SVM and 486 NB) were generated and examined. Only 794 unique models achieved the cutoff of $\text{AUC} \geq 0.6$, $\text{SEN} \geq 0.6$ and $\text{SPE} \geq 0.6$ in five-fold cross-validation, and thus were included in the pool of base classifiers for the building of ensemble models. Starting from randomly selected 5 base classifiers, the number of base classifiers increased at a step of 4 (e.g. 5, 9, ..., 793 base models) until an ensemble made of 793 base classifiers was generated. A total of 198 ensemble models were produced and their AUC(pes), MCC, and GMEAN, calculated using the training set, were determined. This process was repeated for 50 times, hence, the average performance for each combination of base classifiers (n_{Base}) was obtained and shown in Fig. 4. The minimum number of base classifiers needed before the average AUC(pes) starts to plateau was 617. Among the ensemble with 617 base classifiers (n_{Base}), the ensemble in replicate 28 had achieved the best AUC(pes) value, and its performance is shown in Table 1.

The prediction performance of the ensemble model ($n_{\text{Base}} = 617$) is shown in Table 1. The model has achieved an AUC(pes) of 0.822, ACC of 87.6%, SEN of 91.9%, SPE of 81.1%, MCC of 0.739, and GMEAN of 86.3% on the training set. When the ensemble model was tested on the external validation set (valRANDOM), it has achieved an AUC(pes) of 0.595 and GMEAN of 72.7%. In set valBLACK, the ensemble model achieved an AUC(pes) of 0.924, GMEAN of 79.9% and precision for positive classification of 73.3%. Dronedarone was predicted as hepatotoxic. For valPAIR, the ensemble model achieved an AUC(pes) of 0.450 and GMEAN of 49%. The detailed performance of the ensemble model on valPAIR is shown in Fig. 2.

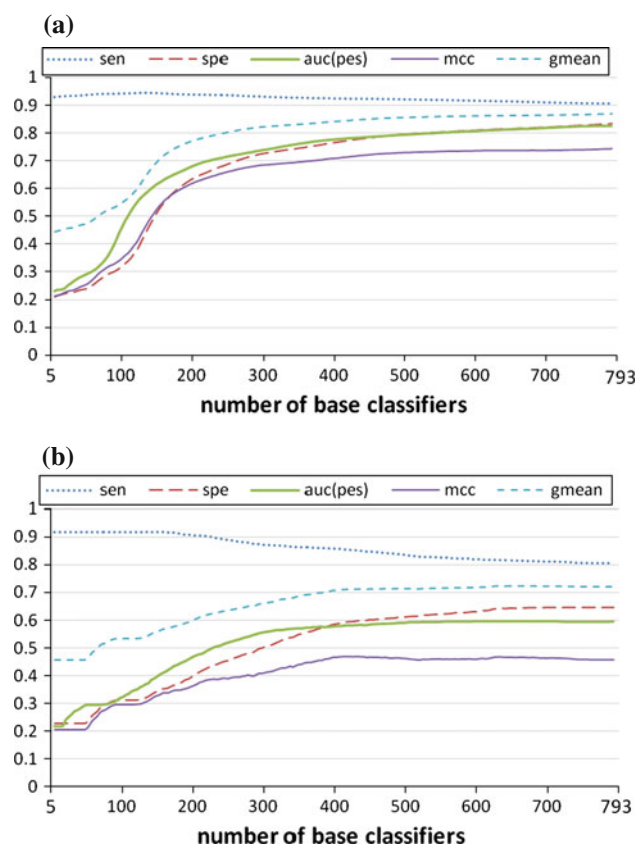


Fig. 4 Graph of performances against the number of base classifiers (nBase) in ensemble models. **a** Training set performances, **b** valRANDOM performances. SEN, SPE or GMEAN of 0.8 is equivalent to 80%

The average performance of the 617 base classifiers shortlisted for the ensemble model were examined and reported in Table 2. The 617 models were made out of 408 kNN (2.8% of 14,580 models), 195 SVM (10% of 1,946 models), and 14 NB (2.9% of 486 models) base classifiers. The average five-fold cross-validation results for these 617 base classifiers are: 0.676 ± 0.001 for AUC(pes), $63.8 \pm 0.1\%$ for accuracy, $64.1 \pm 0.1\%$ for sensitivity, $63.3 \pm 0.1\%$ for specificity, 0.269 ± 0.001 for MCC, and $63.7 \pm 0.1\%$ for GMEAN. The detailed average performance for the three validation sets are shown in Table 2.

Figure 4 consists of (a) training set and (b) valRANDOM performances against the number of base classifiers (nBase). The graphs consisted of 198 data points, starting from the performance of the ensemble built from 5 base classifiers and ending with the ensemble built with 793 base classifiers. The y-axis is applicable for SEN, SPE and GMEAN after conversion of percentages to its corresponding ratios, for example, 68.8% is equivalent to 0.688 in the y-axis.

The best performing model among the 617 base classifiers in the ensemble was a 9-NN model in replicate

number 28. It had achieved an AUC(pes) of 0.743, accuracy of 68.1%, sensitivity of 66.8%, specificity of 70.0%, MCC of 0.361 and GMEAN of 68.4% in five-fold cross-validation. The detailed performance for the three validation sets, valRANDOM, valBLACK, and valPAIR, are included in Table 3. From the 617 models, we have examined the top 10 models based on their AUC(pes) in five-fold cross-validation and valRANDOM to check if the top scorers in training also scores well in valRANDOM. It was found that only three of the top ten models in five-fold cross-validation appeared in the top 10 of valRANDOM performance. The best valRANDOM performance was not achieved by any of the top 10 scorers in cross-validation. Furthermore, the best model, 9-NN in Table 3, produced an AUC(pes) value at rank 8 of valRANDOM performances.

Applicability domain

The number of compounds that exceed the range of one or more descriptors in the external validation sets were 19 (14 positives and 5 negatives), 3 (3 positives), and 3 (2 positives and 1 negative), in valRANDOM, valBLACK and valPAIR respectively. The application of AD on the validation sets did not change the overall prediction significantly. Small improvement on the sensitivity or specificity value was observed for valRANDOM in Table 1. The prediction accuracy for the compounds that fall outside of the domain were ACC = 68.4% (SEN = 71.4%, SPE = 60%) for valRANDOM, ACC = 100% (SEN = 100%, SPE = NA) for valBLACK, and ACC = 66.7% (SEN = 100%, SPE = 0%) for valPAIR.

Y-randomization

Twenty-five rounds of y-randomization were conducted on the training set with 1,087 compounds. On average, approximately 16,650 base classifiers were built for each round of y-randomization. The mean \pm standard deviation of the average AUC from five-fold cross-validation of the 25 rounds of y-randomization was 0.374 ± 0.009 , $51.7 \pm 0.8\%$ for sensitivity and $48.9 \pm 0.8\%$ for specificity. None of the base models generated in the 25 rounds of y-randomization satisfy the cutoff criteria of $AUC \geq 0.6$, $SEN \geq 0.6$ and $SPE \geq 0.6$ to form an ensemble model. Hence, there were no prediction results for all external validation sets.

Substructures with hepatic effects potential

The Klekota-Roth substructures were calculated using the PaDEL-Descriptor program for the 1,274 compounds in this study. Substructures that were unique to the positive compounds and have occurred in more than 5 compounds

Table 1 Performance of the selected ensemble model (made of 617 base classifiers) in training and various external validation sets

Validation	No. of compounds	AUC (pes) ^a	ACC (%)	SEN (%)	SPE (%)	MCC	GMEAN (%)
Without applicability domain							
Training	1,087	0.822	87.6	91.9	81.1	0.739	86.3
valRANDOM	120	0.595	75.0	81.9	64.6	0.473	72.7
valBLACK	47	0.924	80.9	95.7	66.7	0.648	79.9
valPAIR	20	0.450	55.0	80.0	30.0	0.115	49.0
Within applicability domain							
Training	1,087	0.822	87.6	91.9	81.1	0.739	86.3
valRANDOM	101	0.610	76.2	84.5	65.1	0.509	74.2
valBLACK	44	0.913	79.5	95.0	66.7	0.631	79.6
valPAIR	17	0.417	52.9	75.0	33.3	0.091	50.0

^a AUC pessimistic**Table 2** Five-fold cross-validation and external validation performance (average \pm standard deviation) of all 617 base classifiers used in the final ensemble model

Validation	No. of compounds	AUC (pes)	ACC (%)	SEN (%)	SPE (%)	MCC	GMEAN (%)
Five-fold cross-validation	1,087	0.676 \pm 0.001	63.8 \pm 0.1	64.1 \pm 0.1	63.3 \pm 0.1	0.269 \pm 0.001	63.7 \pm 0.1
valRANDOM	120	0.668 \pm 0.002	62.2 \pm 0.2	62.4 \pm 0.3	61.8 \pm 0.3	0.240 \pm 0.004	61.8 \pm 0.2
valBLACK	47	0.757 \pm 0.003	69.6 \pm 0.3	67.9 \pm 0.3	71.1 \pm 0.5	0.396 \pm 0.006	68.9 \pm 0.3
valPAIR	20	0.489 \pm 0.003	50.8 \pm 0.2	64.5 \pm 0.6	37.2 \pm 0.5	0.021 \pm 0.004 ^a	47.2 \pm 0.3

^a The average MCC was calculated from 615 base classifiers; 2 cases where TN + FN = 0 were excluded**Table 3** Five-fold cross-validation and external validation results of the top base classifier (kNN, $k = 9$) among the 617 models selected for the ensemble

Validation	No. of compounds	AUC (pes)	ACC (%)	SEN (%)	SPE (%)	MCC	GMEAN (%)
Five-fold cross-validation	1,087	0.743	68.1	66.8	70.0	0.361	68.4
valRANDOM	120	0.762	70.8	68.1	75.0	0.422	71.4
valBLACK	47	0.842	83.0	82.6	83.3	0.659	83.0
valPAIR	20	0.490	50.0	70.0	30.0	0	45.8

are reported in Fig. 5. A few of the substructures in Fig. 5 coincide with the drug design guideline on structural alerts for bioactivation which may lead to toxicity, that is, halogenated aromatics and arylacetic fragments [84]. Note that the presence of one or more of these substructures may predispose a compound to cause hepatotoxicity, but it does not imply that the compound will definitely have hepatic effects as multiple factors are involved in a toxic event.

Hepatotoxicity prediction program

A program that uses the ensemble model (nBase = 617) trained from 1,087 compounds for prediction of hepatotoxicity is available for download at <http://padel.nus.edu.sg/software/padelddpredictor>. The total set of compounds was not used for training as testing sets were needed to validate the best performing ensemble model; hence,

ensuring its usability. Compounds in the form of molecule structural files e.g. MDL SDF, MOL or SMILES format, can be used as inputs into the program. The molecular descriptors will be calculated by the program which then makes a prediction of the hepatotoxicity potential.

Discussion

Level 1 compounds

Compounds that cause transient and asymptomatic liver function abnormalities, labeled level 1, were included into the training set as toxic compounds (positive class). These 56 compounds were left in the training set to minimize the risk of false negatives. Consequently, producing a “pessimistic” model which learned that level 1 compounds are

KR21	KR662	KR1124	KR1165
KR1575	KR3084	KR3540	KR4003
KR4018	KR4192	KR4232	KR4491
KR4556		KR4689	

Fig. 5 SMARTS substructures (captioned with PaDEL-Descriptor identification) absent in negative set but present in more than 5 instances of positive compounds. *[#1] is any atom not with atomic number of 1

toxic. This is so that an unknown similar compound will have a higher chance of being predicted as positive rather than negative. These predictions will then alert the user of the toxic potential, as it is more detrimental to overlook a potential toxic compound and permitted to be further developed into medicament.

Nevertheless, we have applied the same modelling processes onto a training set without these 56 compounds (results not shown) to check the effects of their removal. Only 48 base classifiers fulfilled the cutoff criteria of $AUC \geq 0.6$, $SEN \geq 0.6$ and $SPE \geq 0.6$. Hence, an ensemble (named minus-1) was built and applied on the validation sets. From the results of the base classifiers and three validation set, the removal of these level 1 compounds was detrimental. First, the number of individual models satisfying the cutoff of $AUC \geq 0.6$, $SEN \geq 0.6$ and $SPE \geq 0.6$ had dropped greatly from 794 to 48. This shows that the overall quality of all individual models had decreased. Furthermore, when the new ensemble, minus-1, was applied on the three validation sets, the results showed very poor predictions for non-toxic compounds. For example, valRANDOM and valBLACK gave SEN of $\sim 94\%$ and SPE of $\sim 20\%$, while none of the non-toxic compounds in valPAIR were predicted as negative. The results suggest that the 56 compounds were critical in defining the boundary between positive and negative compounds. Figure 6 illustrates the proposed importance of level 1 compounds in a model's decision boundary. When level 1 compounds were included, it was probable

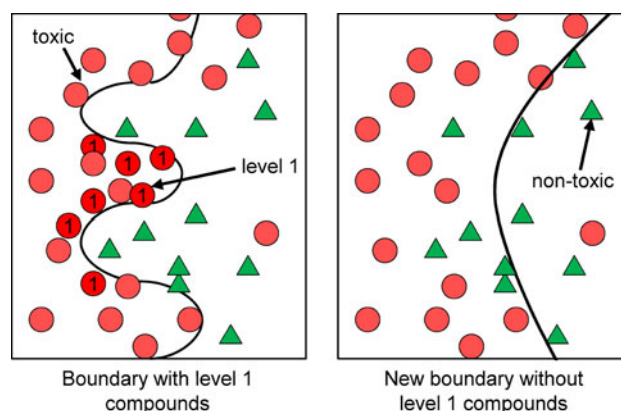


Fig. 6 An illustration of the hypothetical importance of level 1 compounds in defining the boundary between toxic and non-toxic compounds. When the boundary changed, less non-toxic compounds were predicted correctly

that it gave a more refined boundary. After their removal, it may have changed to a more generalized boundary that had predicted some non-toxic compounds as positives. Hence, this suggests that it is important to include level 1 compounds as toxic compounds in the training set.

Applicability domain

The applicability domain (AD), or some may prefer to use the term optimum prediction space, is used to assess the reliability of a prediction from a QSAR model [85]. The AD can be defined from the input perspective through the training data set with range-based, distance-based, geometrical-based, and density-based methods [83]. It can also be defined from the output perspective through the use of a meta-model which takes into consideration the prediction outcome of a model [86, 88]. For this study, where a potentially large amount of base classifiers were used in the final ensemble model, it is not trivial to defined the AD from the training set of each base classifiers that contain a different set of descriptors. Therefore, we have adopted to calculate the ranges from all available descriptors, prior to feature selection, to define the AD in this study.

A model learns from existing training examples. Therefore, it is commonly acknowledged that predictions of dissimilar compounds are less reliable as the model might not do well on unfamiliar descriptor values. Therefore, the prediction for an unknown compound that falls outside of the domain is deemed unreliable, but it does not mean that it is wrong. One would normally expect the prediction of compounds to improve with the application of AD. However, from Table 1, the overall accuracy and sensitivity did not change significantly after the application of AD. In some cases, the performance decreased although a small improvement can be seen in valRANDOM for the compounds within AD. Moreover, the accuracies of the

prediction of compounds out of AD were 68.4, 100, and 66.7% for valRANDOM, valBLACK, and valPAIR respectively. This shows that the prediction performance was still good even for compounds outside of AD and therefore suggests that the ensemble model in this study is robust. Hence, for compounds that fall outside of AD, their predictions should not be discarded entirely. But it is prudent to keep in mind that not all predictions for compounds within the domain are 100% reliable; it is very difficult to separate highly similar compounds although they have differing activities as encountered in valPAIR of this study and the study by Rodgers et al. [27].

The model

The $T_{fix}Al_{var}F_{var}$ ensemble method improves the outcome of prediction compared to the prediction from base classifiers. The performance of the ensemble model ($nBase = 617$) in this study was unlikely to have occurred by chance as 25 rounds of y-randomization did not manage to produce any ensemble model. In external validation of 120 compounds, valRANDOM, the $T_{fix}Al_{var}F_{var}$ ensemble had improved the average of the 617 base classifiers with accuracy of 75.0% from 62.2%, geometric-mean of 72.7% from 61.8%, MCC of 0.473 from 0.240, sensitivity of 81.9% from 62.4% and specificity of 64.6% from 61.8%, although the AUC(pes) appeared to have dropped from 0.668 to 0.595 (Tables 1, 2). It was observed that the $T_{fix}Al_{var}F_{var}$ method has a bias for positive predictions because the sensitivity was greatly improved (from 62.4 to 81.9%). When the external validation performance of the best (one) base classifier was examined, the $T_{fix}Al_{var}F_{var}$ ensemble's preference for positive prediction was more obvious. The best base classifier has achieved a sensitivity of 68.1% and specificity of 75.0%. Comparing the three validation results in Table 1 with 3, the ensemble method has improved the sensitivity greatly by 10–14%, while the specificity decreased by 0–17%. However, there were no significant changes for the other performance measures where AUC(pes) (decreased), accuracy (increased), MCC (increased), and geometric-mean (increased). In spite of the mixed results, the ensemble model is expected to be more robust than using one single classifier as the “best” single model may have been chosen by chance. Furthermore, the best single model did not achieve the best valRANDOM performance (among 617 base classifiers) although it was the top performer from cross-validation results. This observation is unsurprising as a study has shown that training results may not correlate well with actual model performance [88]. Therefore, this provides the motivation to apply the ensemble method which can minimize the risk of selecting a sub-optimal single model.

The significant improvement in sensitivity may have occurred at the ensemble level due to the class imbalance in the training set, i.e., 654 positive and 433 negative compounds. The NB can be susceptible to class imbalance. This class imbalance had probably introduced prediction bias for the majority (positive) class in stacking as the meta-model (NB) was given predictions of base classifiers as well as the training set for learning. Therefore, the sensitivity increased greatly but the specificity decreased. However, at the cutoff of $AUC \geq 0.6$, $SEN \geq 0.6$ and $SPE \geq 0.6$, a larger number of base models have seemed to mitigate the effect of class imbalance. Additionally, the mitigating effect was also seen by increasing the ensemble size in Fig. 4. Consequently, acceptable performance was still achieved by the final ensemble model. Nevertheless, future studies should look into adjusting the class imbalance as it may improve the NB in stacking, i.e., smaller ensemble size and improved sensitivity values without affecting the specificity significantly.

In summary, although the specificity had deteriorated with the introduction of ensemble method, the overall value in 2 out of 3 validations are still above 50% (64.6 and 66.7%), which is better than random guesses. The improvement of one indicator (sensitivity) which causes the deterioration of another indicator (specificity) is not uncommon as it is adjustable by the parameters of a model, depending on the intended use of the model. Furthermore in this study, the preference for positive prediction is a desirable effect as it is more detrimental to overlook a toxic compound which can cause harm when ingested and also failure of drug development. More importantly, although $T_{fix}Al_{var}F_{var}$ ensemble has a bias for positive prediction and small improvements compared to the “best” single model, it still managed to improve all indicators except AUC(pes) and specificity of the averages of the 617 base classifiers. This outcome agrees with the aim of this study to explore ensemble method to produce a more robust solution for hepatotoxicity prediction compared to a single model which may not cover a large enough solution space.

Model validation

There were three validation sets prepared for this study. The first validation set, valRANDOM, was randomly selected from the training set by keeping the ratio of positive compounds to negative compounds constant. This set was probably the most reliable validation set. It was expected to be the most representative of the training set as uniform random sampling was applied. For valBLACK, this validation task was expected to be easier than valRANDOM because the compounds were probably well separated by the nature that one class consisted of withdrawn drugs, while the other was non-toxic. For valPAIR,

the validation task was expected to be much tougher than valRANDOM because they were made of highly similar pairs. Moreover, the majority of the base classifiers in the ensemble were made of kNN models which are dependent on similarity of compounds in their predictions. Therefore, the performance of the ensemble was expected to be less than that of valRANDOM and valBLACK. In summary, the performance of valRANDOM should be the most reflective of the ability of the ensemble model, whereas valBLACK and valPAIR are discussed below.

For the external validation valBLACK on withdrawn drugs or those with black box warning, the $T_{\text{fix}}A_{\text{var}}F_{\text{var}}$ ensemble successfully identified 22 out of the 23 toxic drugs (95.7%) and 16 out of 24 nontoxic drugs (66.7%), with a precision for positive predictions of 73.3%. This shows that approximately three-quarters of positive predictions made by the ensemble model were truly toxic compounds. It is desirable to have a model with good precision in predicting toxic compounds so that compounds with toxic potential can be identified without having too many false alarms (false positives). This will minimize the cost and time needed for further toxicity confirmation of these false positives using animals or in vitro testing. In valBLACK, the only toxic compound that was not identified is naltrexone. It is interesting to note that the black box warning on naltrexone was recommended for removal. This is because the benefit of the drug outweighs the risk in the treatment of opiate dependence and alcoholism. Moreover, hepatotoxicity caused by this drug at the clinical dose was low [89, 90]. Nevertheless, we would again stress that it is important not to overlook toxic compound. On the other hand, it is not ideal to have high false positive rate, e.g., in valPAIR, of which potentially useful compounds might be excluded from further development.

In the external validation valPAIR, of similar pairs but of opposing activity, 80% of the toxic compounds and 30% of the nontoxic compounds were identified correctly. For the 8 pairs that were within AD, only 1 pair was separated correctly. The inability of the model to separate the nontoxic compounds was probably due to the similarity of the actual negative compounds to positive training compounds, and the inherent difficulty to separate highly similar compounds. It was found that 6 of the negatives in the validation set were most similar to positive training compounds and all 6 of them were predicted as positives; 4 were most similar to negative training compounds and 3 were predicted as negatives. The outcome was expected because the dogma of QSAR expects structurally similar compounds to have similar activities. In addition, very similar compounds like stereoisomers might overshadow each other and introduce noise into the training data. If molecular descriptors that can distinguish these compounds were lacking, it would be more difficult for the model to

separate them. In addition, forceful separation of these compounds may produce an overfitted model because of high misclassification penalty to develop the model. Therefore, a model can identify negatives better only when there are large enough negative samples to learn from. As the number of negative compounds in the training set increases, the chance of most similar training compound which is a negative also increases. Hence, it is important to update the training set whenever new compounds become available, so that more information is provided for better class distinction. Nevertheless, the challenge of distinguishing structurally similar compounds was also encountered in the study by Rodgers et al. [27]. They have postulated that chemical mechanism alone could be insufficient to account for the toxic potential which has resulted in the lack of performance of their models in classifying structurally similar compounds. They have proposed the use of toxicity pathway-based biological data with chemical descriptors to improve prediction performance and coverage of model.

The method

More than one type of learning algorithms was used in the process. This is because, no single learning algorithm is optimum for all modelling problems as it may not represent a large enough solution space. The ensemble method is robust and semi-automated because the user do not need to decide on the learning algorithm prior to training. Driven by the results and the training data, the ensemble will select the required base classifiers. Users may then select the desired model from the many ensemble models generated by the process through ranking or other methods. Referring to the breakdown of base classifiers ($n_{\text{Base}} = 617$) in the result section (66.1% kNN, 31.6% SVM, 2.3% NB) and those selected for the ensemble of minus-1 (results not shown) (93.8% kNN, 4.2% SVM, and 2.1% NB), it clearly shows that different algorithms in the base classifiers performed differently on the different training data sets. That is, by proportion of the total number of models used in the respective ensemble, SVM models were less influential in minus-1 ensemble with 4.2% share compared with 31.6% share in the ensemble with 617 base classifiers. Furthermore, only models of a certain quality were selected to form an ensemble. Although the base classifiers were selected by the process without direct human intervention, the minimum performance of the ensemble is expected to be at least as good as the base classifiers.

Various cutoffs, the stacking method, and ensemble trimming were introduced to reduce the ensemble size and increase the model sensitivity for toxic compounds. For this study, the cutoff for short listing base classifiers was set at $\text{AUC}(\text{pes}) \geq 0.6$, $\text{SEN} \geq 0.6$ and $\text{SPE} \geq 0.6$.

First, from observation the AUC cutoff should not be too far off from the maximum achievable AUC, although it should be low enough to include sufficient base classifiers for the ensemble model. If not, no model will be generated like the case of 25 rounds of y -randomization. Theoretically, it is sufficient to use AUC as the sole determinant for the selection of base classifiers. However, there was a large of pool of models to select from. Furthermore, it was observed that a model may have high AUC but unbalanced sensitivity and specificity score, for example 90% versus 10%. Besides, by random chance a large amount of unbalanced models might be selected and the ensemble may run the risk of high false positive or high false negative. Therefore, the cutoff for sensitivity and specificity was added to control the quality of the selected models.

In addition, on the hypothesis that better base models could produce better ensemble model, other cutoffs such as $0.5(\text{AUC}_{\text{pes}}) - 0.5(\text{SEN, SPE})$, 0.55–0.55, or 0.61–0.61 to 0.68–0.68 were tested. An ensemble of all base classifiers fulfilling each cutoff were generated and compared. From the results (not shown), the higher is the cutoff, the lesser is the number of base classifiers available for ensemble, and eventually no models were available at the cutoff of 0.69–0.69. It was observed that lower cutoff gave worse performances, probably due to the inclusion of low quality base models in the ensemble. Besides, the higher number of base models made it computationally intensive to build and to apply the ensemble method in this study. Conversely for higher cutoff values, the training performance is better and the model construction was computationally more manageable. However, the corresponding performance for external validation decreased for the cutoffs at the higher end. This suggests that the ensemble models with higher cutoff levels may be overfitted or the lesser number of base models have reduced its generalization power. Although efforts were made to prevent overtraining of the base models through the use of five-fold cross-validation, it is known that cross-validation results do not correlate well with actual prediction performance [88]. A lower threshold may be used as high training set accuracy does not guarantee high external validation performance [27, 91]. Consequently, we have arrived at the cutoff of 0.6–0.6 for further exploration (ensemble trimming). Furthermore, the lowering of the cutoff to 0.6–0.6 and the use of a large number of base models appears to somewhat compensates for overtraining of the NB in stacking as discussed above.

Second, in ensemble trimming, we have tested the reduction of base models of ensemble models. First, by removing base classifiers built from duplicated descriptor set or those built with one descriptor only. Second, by selecting the best performing ensemble built from a random combination of base classifiers. Figure 4 shows that

when n_{Base} increases, all indicators increased except sensitivity values. The increase starts to plateau off around n_{Base} of 500 (more obvious in MCC), but slight improvements in specificity were still observed. This indicates that this hepatotoxicity data set required a high n_{Base} for its ensemble to perform acceptably. The ensemble with 617 base classifiers were chosen as the best by applying the ranking of $\text{AUC}(\text{pes})$ to the training (not validation) performance. As a result, the ensemble trimming has managed to reduce the number of base classifiers from 794 to 617. Nevertheless, carefully trained base models may result in much smaller ensemble size; hence, future studies can look into this.

Third, in experiments comparing average consensus and stacking for this study (results not shown), the stacked model with naive Bayes has better sensitivity compared to the average consensus model. In average consensus, each base classifier will contribute equally and the prediction is made based on the class with the most votes. In contrast, the stacking method makes a prediction based on a meta-model; the predictions from the base classifiers together with the molecular descriptors were taken as features in stacking to build a meta-model for the final prediction. Hence, the decision mechanism is assumed to be less naive, thus, the information provided by the training set may help enhance the prediction accuracy. As a result, a great improvement in sensitivity value was observed in the final ensemble performance. However, the use of NB for stacking in this study was likely affected by the class imbalance (in the training set) as seen in the decrease in specificity value. Consequently, we acknowledged that the choice of learning algorithm for the meta-model plays an important role, therefore, a variety of learning algorithms should be explored and the class imbalance should be adjusted for future works.

Drawbacks of the $T_{\text{fix}}A_{\text{var}}F_{\text{var}}$ ensemble method include long computational hours and large disk space requirement especially when kNN was used. Depending on the type of learning algorithms employed, a huge number of models may be generated from various combinations in the modelling parameters, for example, one may permute the k and distance measures in kNN or the complexity, C , and gamma (or sigma) in the kernel of SVM to generate a plethora of base classifiers. Therefore, the full potential of $T_{\text{fix}}A_{\text{var}}F_{\text{var}}$ ensemble may be restricted by the demands on computational resources, depending on the combinations used.

In summary, although one of the major limitations is computational resources, the results driven ensemble method required minimal human intervention in its construction. Various performance cutoffs, stacking and ensemble trimming were introduced to the ensemble to

improve some aspects of the ensemble model. The cutoff for sensitivity and specificity was needed to ensure the quality of the base models. The AUC cutoff should not be too far off from the maximum achievable AUC, and if possible, the performance of various cutoffs should be tested before finalizing on one model. The NB was found to be useful as a meta-model, but it can be affected by training set class imbalance. It was observed that the ensemble required several hundreds of base classifiers to perform optimally. Hence, future work should look into the class imbalance and the stacking method so to reduce ensemble size and improve sensitivity without affecting the specificity significantly.

Other hepatotoxicity prediction methods

In silico as well as *in vitro* methods are useful complementary testing methods to animal model for toxicity predictions [11, 18]. The non-exhaustive list of studies on hepatotoxicity is available in Table 4. Note that all studies mentioned are not directly comparable due to the nature of the modelling methods, data and validation sets used, and specific endpoints examined. Some of the studies did not focus on hepatotoxicity; hence, some performance indicators were not available for compilation into Table 4. Nevertheless, these previous studies can give an insight to the difficulties and challenges faced for liver toxicity predictions. For clarity, the discussions will be grouped according to the five points of the Organisation for Economic Co-operation and Development (OECD) principles for the validation of QSAR for regulatory purposes: (1) defined endpoints, (2) unambiguous algorithm, (3) defined domain of applicability, (4) appropriate measures of goodness-of-fit, robustness and predictivity and (5) mechanistic interpretation if possible [92]. Note that the first three entries in Table 4 belonged to *in vitro* methods of which the OECD principles are not applicable. The information for some *in vitro* methods was added for a quick overview of their predictivities.

All QSAR models in the list have fulfilled the principles of a defined endpoint and the report of its predictivity. From Table 4, studies with smaller data set (74 compounds in Cruz-Monteagudo et al. and ~158 compounds in Rodgers et al.) tend to have better validation sensitivity at 75–87.9 and 60–87.5%, compared to studies with larger data set (~877 training compounds in Huang et al. and ~425 training compounds in Fourches et al.) which have sensitivity of 63% and accuracies of 55.7–72.6% respectively. This study which used a training size of 1,087 compounds has sensitivity of 80–95.7% for three of the validation sets.

In general, it can be observed that a majority of *in silico* hepatotoxicity models have acceptable performances and a

few have exceptional results. This suggests that the liver toxicity data set in most studies were “noisy”, hence, clean and exceptional prediction results were hard to achieve. This was expected as complex mechanisms are involved in liver toxicity. Nevertheless, good results can still be achieved with models made of small data sets. However, a drawback of these models is that the smaller data size might have limited representation of the vast chemical space. Therefore, applicability domain of these models may be limited, whereas models developed using larger data size are expected to have greater applicability. On the other hand, even models developed using larger data sets may not be able to solve inherently tough problems such as the resolution of structurally similar pairs. In the study by Rodgers et al., the model, developed using a relatively small data set of 158 compounds, was not able to resolve any similar pairs. Although this study had the largest data set, it was able to resolve only one similar pair. This highlights the challenges in resolution of structurally similar but toxicity dissimilar pairs. In this study, kNN was one of the three algorithms that were used to develop the base models. kNN works on the basis of structural similarity and thus is likely to fail on a compounds purposefully selected to be highly similar. Hence, the poor results of our ensemble model on valPAIR are not surprising and suggest further studies using other algorithms are needed.

All QSAR models (except two) were shown with their applicability domain which is needed to prevent extrapolation of the model within the chemistry space, which can result in unreliable predictions. Although desirable but not compulsory, a few models (linear discriminant analysis and weighted feature significance) passed the criterion on mechanistic interpretation. This is because the ensemble method, which is usually a conglomeration of many models, was applied in most studies. Thus, it makes mechanistic interpretation complex, although not impossible.

One general problem with existing QSAR hepatotoxicity models is the lack of a readily available and working model. For example, the three studies [4, 24, 25] which were proprietary in nature have large data sets (e.g. 1,266–1,608 compounds) but they were not disclosed and their models are unavailable or under licensing that restrict their distribution. Hence, validation of these models will be inconvenient for other parties. The other parties will need to redevelop the models using the same modelling methods and the same compounds (if available). Nonetheless, it may be impossible to reproduce the models exactly as most methods have a degree of inherent random variations. In order to prevent such problems and to aid in independent validation and use, we have made available the data set and a software based on our ensemble model for public use (<http://padel.nus.edu.sg/software/padelddpredictor/>).

Table 4 Information on other studies conducted for hepatotoxicity prediction

Author	Endpoints	Data size train (test)	Availability of data	Learning algorithm (features)	Applicability domain	Availability of model	Validation performance
Xu et al. [12]	(in vitro) 8 cytotoxicity assays, 1 animal test	Test = 611	No	Cell-based	NA	NA	Cytotoxicity: SEN = 1–25% animal: SEN = 52% SEN = 50–60%
Xu et al. [17]	(in vitro) Human hepatocyte imaging assay	Test = 344	Yes	Cell-based	NA	NA	SEN = 65–70%
Reese et al. [21]	(in vitro) GSH adduct formation, covalent binding, CYP metabolism-dependent inhibition	Test = 225	No	Cell-based	NA	NA	SEN = 61% SEN = 46%
Marchant et al. [24]	Intrinsic and idiosyncratic hepatic effects for human and animal	Test = 731	No	Structural alerts	Yes	Proprietary	SEN = 52.8–88.9%
Greene et al. [4]	Intrinsic and idiosyncratic hepatic effects for human and animal	1,266 (626)	Some	Structural alerts	No	Proprietary	
Matthews et al. [25]	5 hepatobiliary disorders	1,044–1,608 (18)	No	Consensus of 2 programs from any MC4PC, MDL-QSAR, BioEpisteme, or Predictive Data Miner	Yes	No	
Cruz-Monteagudo et al. [22]	Idiosyncratic drug hepatotoxicity	74 (13 toxic, 3 similar pairs)	Yes	LDA, ANN, and OneR (radial distribution function)	Yes	No	SEN = 75–87.9%
Huang et al. [23]	Hepatotoxicity (general)	Total = 1,755 50%(50%)	No	Weighted feature significance (WFS), SVM, NB (fragment based)	No	No	SEN = 63% (WFS only)
Fourches et al. [26]	Liver effects in rodents and human	Total = 531 80%(20%)	Yes	Ensemble of SVM (substructural molecular fragments, DRAGON)	Yes	No	ACC = 55.7–72.6% (no sensitivity values)
Rodgers et al. [27]	Effects in AST, ALT or composite	152–168 (36–42)	Yes	Ensemble of kNN (Molconn-Z [93], DRAGON [94])	Yes	No	SEN = 60–87.5%
This study	Hepatotoxicity (general)	1,087 (120, 47, 10 pairs)	Yes	Ensemble of kNN, SVM, NB (PaDEL-Descriptors)	Yes	Yes	SEN = 80–95.7%

LDA linear discriminant analysis, ANN artificial neural network, OneR one level decision tree, SVM support vector machine, kNN k-nearest neighbor, NB naive Bayes, NA not applicable for non-QSAR models

Conclusion

Hepatotoxicity prediction is not an easy problem as most *in vitro* and *in silico* studies gave average prediction performances and few exceptional performances. Although this study has achieved similar or slightly better results, the model produced by this study is advantageous compared with the other studies as the model was built from the largest data set and it was made available for public use. We have reported a list of substructures that may predispose compounds to cause hepatotoxicity. The $T_{fix}Al_{var}F_{var}$ ensemble method was shown to be robust and produces stable results. But, it has high computational and disk space requirement. The model was not suitable to distinguish structurally similar pairs of opposing hepatotoxicity as kNN is a major contributor to the ensemble model.

Acknowledgments This study was supported by the NUS start-up grant R-148-000-105-133.

References

- Björnsson E (2006) Clin Pharmacol Ther 79:521–528
- Gunawan BK, Kaplowitz N (2007) Clin Liver Dis 11:459–475
- Li AP (2002) Chem Biol Interact 142:7–23
- Greene N, Fisk L, Naven RT, Note RR, Patel ML, Pelletier DJ (2010) Chem Res Toxicol 23:1215–1222
- Dearden JC (2003) J Comput Aided Mol Des 17:119–127
- Richard AM (2006) Chem Res Toxicol 19:1257–1262
- Schultz TW, Cronin MTD, Netzeva TI (2003) J Mol Struct 622:23–38
- Veith GD (2004) SAR QSAR Environ Res 15:323–330
- Greene N, Judson PN, Langowski JJ, Marchant CA (1999) SAR QSAR Environ Res 10:299–314
- MetabolExpert¹ www.compudrug.com. <http://www.compudrug.com/?q=node/36>. Accessed 3 May 2011
- Muster W, Breidenbach A, Fischer H, Kirchner S, Müller L, Pähler A (2008) Drug Discov Today 13:303–310
- Xu JJ, Diaz D, O'Brien PJ (2004) Chem Biol Interact 150:115–128
- Subramanian K, Raghavan S, Rajan Bhat A, Das S, Bajpai Dikshit J, Kumar R, Narasimha MK, Nalini R, Radhakrishnan R, Raghunathan S (2008) Expert Opin Drug Saf 7:647–662
- Hultin-Rosenberg L, Jagannathan S, Nilsson KC, Matis SA, Sjogren N, Huby RD, Salter AH, Tugwood JD (2006) Xenobiotica 36:1122–1139
- Zidek N, Hellmann J, Kramer PJ, Hewitt PG (2007) Toxicol Sci 99:289–302
- Ebbels TM, Keun HC, Beckonert OP, Bollard ME, Lindon JC, Holmes E, Nicholson JK (2007) J Proteome Res 6:4407–4422
- Xu JJ, Henstock PV, Dunn MC, Smith AR, Chabot JR, de Graaf D (2008) Toxicol Sci 105:97–105
- Greer ML, Barber J, Eakins J, Kenna JG (2010) Toxicology 268:125–131
- Martinez SM, Bradford BU, Soldatow VY, Kosyk O, Sandot A, Witek R, Kaiser R, Stewart T, Amaral K, Freeman K, Black C, LeCluyse EL, Ferguson SS, Rusyn I (2010) Toxicol Appl Pharmacol 249:208–216
- Meng Q (2010) Exp Opin Drug Metab Toxicol 6:733–746
- Reese M, Sakatis M, Ambroso J, Harrell A, Yang E, Chen L, Taylor M, Baines I, Zhu L, Ayrton A, Clarke S (2011) Chem Biol Interact 192:60–64
- Cruz-Monteagudo M, Cordeiro MN, Borges F (2008) J Comput Chem 29:533–549
- Huang R, Southall N, Xia M, Cho MH, Jadhav A, Nguyen DT, Ingles J, Tice RR, Austin CP (2009) Toxicol Sci 112:385–393
- Marchant CA, Fisk L, Note RR, Patel ML, Suarez D (2009) Chem Biodivers 6:2107–2114
- Matthews EJ, Ursem CJ, Kruhlik NL, Benz RD, Sabaté DA, Yang C, Klopman G, Contrera JF (2009) Regul Toxicol Pharmacol 54:23–42
- Fourches D, Barnes JC, Day NC, Bradley P, Reed JZ, Tropsha A (2010) Chem Res Toxicol 23:171–183
- Rodgers AD, Zhu H, Fourches D, Rusyn I, Tropsha A (2010) Chem Res Toxicol 23:724–732
- Zheng W, Tropsha A (2000) J Chem Inf Comput Sci 40:185–194
- Tropsha A, Golbraikh A (2007) Curr Pharm Des 13:3494–3504
- Arodz T, Yuen DA, Dudek AZ (2005) J Chem Inf Model 46:416–423
- Bostrom H (2007) 10th International conference on information fusion, pp 1–7
- Li J, Lei B, Liu H, Li S, Yao X, Liu M, Gramatica P (2008) J Comput Chem 29:2636–2647
- Lei B, Xi L, Li J, Liu H, Yao X (2009) Anal Chim Acta 644:17–24
- Liew CY, Ma XH, Yap CW (2010) J Comput Aided Mol Des 24:131–141
- Asikainen AH, Ruuskanen J, Tuppurainen KA (2004) SAR QSAR Environ Res 15:19–32
- Votano JR, Parham M, Hall LH, Kier LB, Oloff S, Tropsha A, Xie Q, Tong W (2004) Mutagenesis 19:365–377
- Norinder U, Liden P, Bostrom H (2006) Mol Divers 10:207–212
- Tropsha A (2010) Mol Inform 29:476–488
- Tan P-N, Steinbach M, Kumar V (2005) Introduction to data mining. Addison Wesley, Reading
- Gramatica P, Pilutti P, Papa E (2004) J Chem Inf Comput Sci 44:1794–1802
- Gramatica P, Giani E, Papa E (2007) J Mol Graphics Model 25:755–766
- Agrafiotis DK, Cedeno W, Lobanov VS (2002) J Chem Inf Comput Sci 42:903–911
- Hong H, Tong W, Xie Q, Fang H, Perkins R (2005) SAR QSAR Environ Res 16:339–347
- Sushko I, Novotarskyi S, Körner R, Pandey AK, Cherkasov A, Li J, Gramatica P, Hansen K, Schroeter T, Müller KR, Xi L, Liu H, Yao X, Öberg T, Hormozdiari F, Dao P, Sahinalp C, Todeschini R, Polishchuk P, Artemenko A, Kuz'min V, Martin TM, Young DM, Fourches D, Muratov E, Tropsha A, Baskin I, Horvath D, Marcou G, Muller C, Varnek A, Prokopenko VV, Tetko IV (2010) J Chem Inf Model 50:2094–2111
- Kuz'min VE, Muratov EN, Artemenko AG, Varlamova EV, Gorb L, Wang J, Leszczynski J (2009) QSAR Comb Sci 28:664–677
- Shen M, Beguin C, Golbraikh A, Stables JP, Kohn H, Tropsha A (2004) J Med Chem 47:2356–2364
- Yap CW, Chen YZ (2005) J Chem Inf Model 45:982–992
- Zhang S, Wei L, Bastow K, Zheng W, Brossi A, Lee KH, Tropsha A (2007) J Comput Aided Mol Des 21:97–112
- Breiman L (2001) MLearn 45:5–32
- Sutherland JJ, O'Brien LA, Weaver DF (2003) J Chem Inf Comput Sci 43:1906–1915
- Oloff S, Mailman RB, Tropsha A (2005) J Med Chem 48:7322–7332
- Katritzky AR, Kuanar M, Slavov S, Dobchev DA, Fara DC, Karelson M, Acree WE Jr, Solov'ev VP, Varnek A (2006) Bioorg Med Chem 14:4888–4917

53. Zhang L, Zhu H, Oprea TI, Golbraikh A, Tropsha A (2008) *Pharm Res* 25:1902–1914
54. Gini G, Garg T, Stefanelli M (2009) *ApAI* 23:261–281
55. Roy K, Paul S (2009) *QSAR Comb Sci* 28:406–425
56. Dahlgren MK, Zetterstrom CE, Gylfe S, Linusson A, Elofsson M (2010) *Bioorg Med Chem* 18:2686–2703
57. Orange book: approved drug products with therapeutic equivalence evaluations. <http://www.accessdata.fda.gov/scripts/cder/ob/default.cfm>. Accessed 25 November 2010
58. Micromedex® Healthcare Series [Internet database]. Accessed 25 November 2010
59. Budavari S, O'Neil MJ, Smith A (1989) *The Merck index: an encyclopedia of chemicals, drugs, and biologicals*. Merck Publishing Group
60. Kaplowitz N (2003) *Drug-induced liver disease*. Marcel Dekker, Inc., New York
61. Bolton EE, Wang Y, Thiessen PA, Bryant SH, Ralph AW, David CS (2008) *Annual reports in computational chemistry*. Elsevier, Amsterdam, pp 217–241
62. CambridgeSoft Desktop Software—ChemDraw (Windows/Mac). <http://www.cambridgesoft.com/>. Accessed 3 Jun 2010
63. Pipeline Pilot Student Edition. <http://accelrys.com/solutions/industry/academic/student-edition.html>. Accessed 10 January 2011
64. CORINA: Generation of 3D coordinates. <http://www.molecular-networks.com/software/corina/index.html>. Accessed 3 Jun 2010
65. Walgren JL, Mitchell MD, Thompson DC (2005) *Crit Rev Toxicol* 35:325–361
66. Drug Safety and Availability. FDA Drug Safety Communication: Severe liver injury associated with the use of dronedarone (marketed as Multaq). <http://www.fda.gov/Drugs/DrugSafety/ucm240011.htm>. Accessed 17 January 2011
67. PaDEL-Descriptor. <http://padel.nus.edu.sg/software/padeldescriptor/index.html>. Accessed 3 Jun 2010
68. Yap CW (2011) *J Comput Chem* 32:1466–1474
69. Vapnik V (1995) *The nature of statistical learning theory*. Springer, London
70. Czermiński R, Yasri A, Hartsough D (2001) *Quant Struct Act Relat* 20:227–240
71. Trotter M, Buxton B, Holden SB (2001) *Measure Control* 34:235–239
72. Brent RP (2002) *Algorithms for minimization without derivatives*. Dover Publications, New York
73. Baldi P, Brunak S, Chauvin Y, Andersen CA, Nielsen H (2000) *Bioinformatics* 16:412–424
74. Matthews BW (1975) *Biochim Biophys Acta* 405:442–451
75. Fawcett T (2006) *Pattern Recog Lett* 27:861–874
76. Nicholls A (2008) *J Comput Aided Mol Des* 22:239–255
77. Mierswa I, Wurst M, Klinkenberg R, Scholz M and Euler T (2006) KDD '06: proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 935–940
78. Yu L, Liu H (2004) In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, Seattle, pp 737–742
79. Kuncheva L (2003) *Pattern recognition and image analysis*. Springer Berlin, pp 1126–1138
80. Fan W, Wang H, Yu PS and Ma S (2003) *ICDM 2003 Third IEEE international conference on data mining*, pp 51–58
81. Wolpert DH (1992) *Neural Netw* 5:241–259
82. Rücker C, Rücker G, Meringer M (2007) *J Chem Inf Model* 47:2345–2357
83. Jaworska J, Nikolova-Jeliazkova N, Aldenberg T (2005) *ATLA Altern Lab Anim* 33:445–459
84. Guengerich FP, MacDonald JS (2007) *Chem Res Toxicol* 20:344–369
85. Dearden JC, Cronin MT, Kaiser KL (2009) *SAR QSAR Environ Res* 20:241–266
86. Dragos H, Gilles M, Alexandre V (2009) *J Chem Inf Model* 49:1762–1776
87. Sazonovas A, Japertas P, Didziapetris R (2010) *SAR QSAR Environ Res* 21:127–148
88. Golbraikh A, Tropsha A (2002) *J Mol Graphics Model* 20:269–276
89. Yen MH, Ko HC, Tang FI, Lu RB, Hong JS (2006) *Alcohol* 38:117–120
90. Garbutt JC (2010) *Curr Pharm Des* 16:2091–2097
91. Golbraikh A, Shen M, Xiao Z, Xiao YD, Lee KH, Tropsha A (2003) *J Comput Aided Mol Des* 17:241–253
92. Validation of (Q)SAR Models. http://www.oecd.org/document/4/0,3746,en_2649_34379_42926724_1_1_1_1,00.html. Accessed 23 May 2011
93. Molconn Z. <http://www.edusoft-lc.com/molconn/>. Accessed 3 Jun 2010
94. Talete—Dragon. http://www.talete.mi.it/products/dragon_description.htm. Accessed 3 Jun 2010