JOURNAL OF CHEMICAL INFORMATION AND MODELING

Perspective

pubs.acs.org/jcim

Machine Learning Methods for Property Prediction in Chemoinformatics: *Quo Vadis*?

Alexandre Varnek*[,†] and Igor Baskin[†,‡]

[†]Laboratoire d'Infochimie, UMR 7177 CNRS, Université de Strasbourg, 4, rue B. Pascal, Strasbourg 67000, France

[‡]Department of Chemistry, Moscow State University, Moscow 119991, Russia

ABSTRACT: This paper is focused on modern approaches to machine learning, most of which are as yet used infrequently or not at all in chemoinformatics. Machine learning methods are characterized in terms of the "modes of statistical inference" and "modeling levels" nomenclature and by considering different facets of the modeling with respect to input/ouput matching, data types, models duality, and models inference. Particular attention is paid to new approaches and concepts that may provide efficient solutions of common problems in chemoinformatics: improvement of predictive performance of structure−property (activity) models, generation of structures possessing desirable properties, model applicability domain, modeling of properties with functional endpoints (e.g., phase diagrams and dose−response curves), and accounting for multiple molecular species (e.g., conformers or tautomers).

1. INTRODUCTION

Over the last 30 years, the area of machine learning (statistical learning or data mining) has undergone significant changes comparable with the revolution in physics at the beginning of the 20th century. The main problem in classical mathematical statistics concerns the inability to answer the "simple" question: *Why does a model that perfectly fits the training data lead sometimes to incorrect predictions for the independent test set*? Classical statistics in fact guarantees correct predictions only asymptotically, i.e., for infinitely large training sets. Fischer's *parametric* statistics requires the identification in advance of both relationships between the input and output data and the probability distributions of data. It specifies a few free parameters of those relationships and distributions to be found in the statistical study. More recent *nonparametric* statistics does not require exact model specification, but it is restricted to data of low dimensionality because of the "curse of dimensionality".[1] These limitations are too restrictive to allow solution of most real-world problems. Nowadays, the fundamental paradigm of statistical analysis has changed from "system identification" (in which the aim is to reconstruct true probability distributions as the necessary step to achieve good predictive performance) to "predictive modeling" (in which simple, although not necessarily correct, probability distributions or/and decision functions are used to build models with the highest predictive performance in the area occupied by actual data).[2] The new paradigm first employed with artificial neural networks[3,4] received theoretical backing through the development of new statistical theories capable of dealing with small data sets and oriented toward predictions: the statistical learning theory of Vapnik,[5,6] PAC (Probably Approximately Correct) theory of Valiant,[7] minimum description length concept of Rissanen,[8] and some others.

Chemoinformatics, an area at the interface of chemistry and informatics,[9−14] is constantly exposed to the evolution in statistics and machine learning. The penetration of new data mining approaches into chemoinformatics has sometimes been the result of short-lived enthusiasm for novel methods, as with neural networks and support vector machines. A reflection in chemoinformatics of the last crisis in statistics was the appearance of publications expressing disappointment in the capacity of QSAR/QSPR and similarity search methods to provide reliable predictions.[15] This is not unexpected given that instead of treating congeneric data sets one should be able to base models on very small (issuing from costly experiments) or very large (issuing from screening campaigns) structurally diverse data sets. The models developed on the limited size training sets should be applicable in virtual screening or for annotation of large databases. Thus, a subset of compounds should be identified to which the model can be applied with good predictive performance, i.e., by defining the model's applicability domain (AD). Despite the large number of publications devoted to AD, this problem is still far from being resolved.

The development of predictive tools for drug design is a major stimulus for the generation of experimental data, specifically for model development. The question is how to construct the "optimal" training set (size, composition) to build predictive models.

In fact, predictive performance of the models is not the only problem to solve (Figure 1); there are others where the absence of appropriate machine learning methods represents a real bottleneck. This concerns the modeling of properties with functional endpoints (e.g., phase diagrams and dose−response curves), accounting for multiple molecular species (e.g., conformers or tautomers[16]) and direct generation of chemical structures ("inverse QSAR"[17−26]).

There is also a fundamental problem of descriptors derived from molecular structures. There is in general a loss of information resulting from the representation of a molecular structure by a fixed number of descriptors. Therefore, the

Received: September 1, 2011
Published: May 14, 2012

   dx.doi.org/10.1021/ci200409x | *J. Chem. Inf. Model.* 2012, 52, 1413−1437

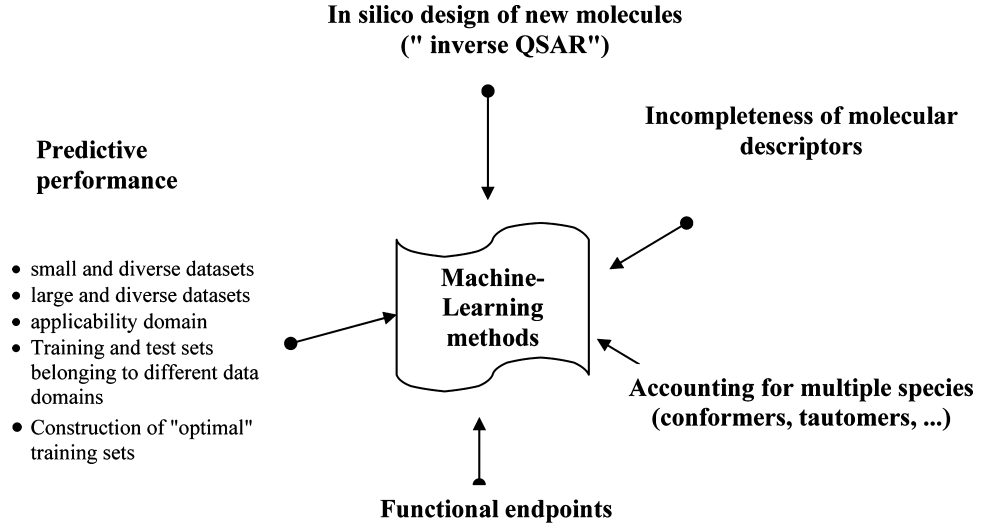

Figure 1. Main challenges of machine learning methods in chemoinformatics.

methods allowing one to build QSAR models directly from the structural formulas could become an interesting alternative to conventional descriptors-based modeling (e.g., special types of neural networks[27−32] or graph kernels[33−42]).

Analysis of the data mining literature reveals a growing number of new potentially useful machine learning methods that have been successfully used in different areas but are still not actively used in chemistry. For this reason, unlike numerous publications describing machine learning methods commonly used in QSAR[11,43−47] and in ligand-based virtual screening,[48,49] here, we focus mostly on methods that have been rarely or never used in this area. Some of these methods are listed in Table 1 in connection with the chemoinformatics tasks indicated in Figure 1. Certainly, we are not able to analyze all of them. Instead, we describe general characteristics of the methods using the "modes of statistical inference" and "modeling levels" nomenclature and considering different facets of the modeling with respect to input/output matching, data types, models duality, and models inference. Particular attention is paid to ensemble learning approaches and graph mining techniques particularly useful in chemoinformatics. Finally, we discuss some new machine learning methods that may provide efficient solutions to common problems in chemoinformatics: virtual screening performances, improvement of predictive power of structure−property (activity) models, generation of structures possessing desirable properties, model applicability domain, and some others.

It should be noted that this paper concerns machine-learning methods dedicated to structure−property modeling. Those concerning other tasks (data visualization, dimensionality reduction, etc.) are outside the scope of this review.

2. LEARNING APPROACHES: MODES OF STATISTICAL INFERENCE AND MODELING LEVELS

In this section, we discuss two main modes of statistical inference and three levels of statistical modeling.[50] Statistical inference is the process of drawing conclusions from observable data that are subject to random variations, for example, observational errors or sampling variation.[51] Two main modes of statistical inference, frequentist and Bayesian, are closely related to two different interpretations of probability in machine-learning. The term "Levels" indicate the main target of the modeling of property (activity) Y as a function of attributes (descriptors) X: (i) point estimation of Y, (ii) distribution function for Y, and, (iii) joint distribution function for Y and X. Thus, any particular machine learning method could be attributed to a given "mode/level" combination (Figure 2). Detailed analysis of modes and levels is given below.

2.1. Main Modes of Statistical Inference. *2.1.1. Frequentist Mode.* In the frequentist approach, the statistical model corresponds to a function $Y = F(X, A)$, where F is a class of parametric functions, and A is a set of used parameters. In classical statistics, the model parameters A are found by maximizing the likelihood function $L(A) = P(Y|X,A)$, where $P(Y|X,A)$ stands for a conditional probability of Y given X and A. This means that the "optimal" parameters A reproduce the values of Y with the highest probability for objects (chemical compounds) taken from the training set. Multiple linear regression (MLR) is an example of such approach.

The question arises: Does this maximum likelihood criterion ensure the best predictions on some test sets drawn from the same probability distribution? The theory answers "yes" in the case of very large training sets (asymptotical solution) and/or a small number of independent adjustable parameters A, and "no" for small training sets and/or a large number of A. In the latter case, the model parameters A are to a large extent influenced by data noise. This results in *overfitting*, in which a model accurately predicts the training set but does poorly on independent test sets. Overfitting is a big problem for all classical statistics methodologies, and it is still very common in chemoinformatics.

Unlike classical statistics methods, in modern frequentist approaches, the optimal parameter set A is usually obtained by minimizing the functional Φ

$$\Phi = L(A) + \lambda \cdot \Omega(A) \quad (1)$$

where $L(A)$ is the negative logarithm of the likelihood, $\Omega(A)$ is a *regularizer*, and λ is a mixing coefficient. It should, however, be noted that in certain cases (for example, in training backpropagation neural networks) regularization can be introduced without the explicit use of formula 1. In most modern machine learning methods, the regularizer $\Omega(A)$ contains quadratic forms $\|A\|^2$ and linear $\|A\|$ terms of the

Table 1. Chemoinformatics Tasks and the Appropriate Machine Learning Concepts and Methods

	Chemoinformatics task or problem	Machine Learning Concept	Machine Learning method	Implementation in freely available software
1	Increase of the predictive performance of models built on small and diverse data sets	Ensemble learning[291]	Different methods of combining classifiers[292]	meta/Vote (*W*)
			Bagging[79]	meta/Bagging (*W*), adabag (*R*)
			Boosting (classification)[88]	meta/AdaBoostM1 (*W*), ada, adabag (*R*)
			Boosting (regression)[91]	meta/AdditiveRegression (*W*) GAMBoost, mboost (*R*)
			Stacking[86]	meta/Stacking (*W*)
			Random subspace[85]	meta/RandomSubSpace (*W*)
			Random forest[80]	trees/RandomForest (*W*) randomForest (*R*)
		Semisupervised and transductive learning[96,293]	TSVM (transductive SVM)[97,294,295]	SVMlight[296]
			SGT (Spectral Graph Transducer)	SGTlight[297]
			SemiL (Semisupervised Learning)[250]	SemiL[298]
			LapSVM (Laplacian SVM),[299] Semisupervised learning based on one-class classification[300] and ensemble learning[301]	
		Inductive knowledge transfer,[281] multitask learning,[153,154] collaborative filtering[224]	Multitask learning using backpropagation neural networks[154]	RSNNS, AMORE, neuralnet, nnet (*R*); SNNS[302]
			Multitask learning using multitask kernels,[155] Bayesian multitask learning,[303,304] multitask learning using Partial Least Squares (PLS) method,[305] online multitask learning,[306] multitask learning with data editing,[307] semisupervised multitask learning using Dirichlet process,[308] conic programming for multitask learning,[309] multitask learning by multiple kernel learning[310]	
		L_2-Regularized methods	Support Vector Machines(SVM)[6,45,311,312]	functions/SMO (*W*); kernlab (*R*); LibSVM,[313] SVMlight[296]
			Ridge regression[314]	functions/LinearRegression (*W*); Penalized, RXshrink (*R*)
			Gaussian processes[59] (with Gaussian prior)	functions/GaussianProcesses (*W*) kernlab (*R*)
		L_1-Regularized methods	Least angle and lasso regression[315−318]	lars, biglars, lasso2, penalizedn, relaxo (*R*)
			Regularized least absolute deviation regression[319]	
			Sparse PCA[320,321] and CCA[320]	PMA (*R*)
			Linear programming boosting via column generation[93]	
2	Reliable estimation of the precision of predictions	Bootstrap,[68] Probabilistic discriminative level[50]	Gaussian processes[59]	gptk (*R*), kernlab (*R*), functions/GaussianProcesses (*W*)
3	Large data sets	Online methods	Online SVM (LASVM)[109,110]	
			Online Kernel-based Learning algorithms for classification, novelty detection, and regression[254]	kernlab (*R*)

dx.doi.org/10.1021/ci200409x | J. Chem. Inf. Model. 2012, 52, 1413−1437

Table 1. continued

	Chemoinformatics task or problem	Machine Learning Concept	Machine Learning method	Implementation in freely available software
		Efficient implementations of kernel algorithms for huge data sets	ISDA (Iterative Single Data Algorithm)[250,251]	ISDA[322]
			Stochastic variant of the PEGASOs (Primal Estimated sub-GrAdient SOlver for SVM)[252]	functions/SPegasos (*W*)
			Large scale multiple kernel learning[323]	Shogun[324]
		Ultrafast linear SVM approaches	LIBLINEAR[253]	functions/LibLINEAR (*W*); LibliniearR (*R*); LIBLINEAR[325]
			Linear SVM	LinearSVM[326]
		Ensemble learning	Dagging[84]	meta/Dagging (*W*)
4	Applicability domain of QSAR models	Internal[180] applicability domain: probabilistic (e.g., Bayesian methods), ensemble-based (e.g., bagging-based) methods	Associative Neural Networks (ASNN)[70,73]	Online Chemical Modeling Environment (OCHEM)[327]
			Gaussian processes[59]	gptk (*R*), kernlab (*R*), functions/GaussianProcesses (*W*)
		External[180] applicability domain: novelty detection,[156,157,159] one-class classification,[177,271,282,283] data domain description[158]	Wrapper using 2-class classifiers as 1-class classifiers	meta/oneClassClassifier (*W*)
			1-SVM[177,178]	functions/LibSVM (*W*) LibSVM[313]
			Support Vector Domain Description (SVDD)[158]	SVDD[328]
			SOM-based novelty detection[168]	Kohonen (*R*)
			Kernel PCA for novelty detection[169]	Kernlab (*R*)
			Fast Support Vector Domain Description (F-SVDD),[163] Structured one-class classification (TOCC),[164] One-class Very Fast Decision Tree (OcVFDT) algorithm,[165] Condensed Nearest Neighbor Data Description (CNNDD) algoritm,[166,329] semisupervised support vector domain description.[167] Novelty detection can also be performed with autoassociative neural network,[157] SOM,[168] kernel PCA,[169] single-class minimax probability machines,[170] evolving fuzzy classifier,[330] one-class Parzen density estimator,[171] Gaussian Mixture Models (GMM) in Gabor space,[172] multivariate extreme value statistics[173]	
5	Training and test sets belong to different data domains	Data set shift,[256,257] covariate shift,[256] domain adaptation,[182,258−260] transfer learning[281]	Covariate shift adaptation by importance weighted cross validation,[181] feature subsetting,[183] conditional random fields,[184] cross-domain generalizable features,[185,331] semisupervised domain adaptation via structural frequency features[186] and some others[187]	
6	Suggestions of molecules for "optimal" training sets	Active learning[98,284−289]	Implementations of active learning using neural networks,[103,104] neural network ensembles,[100] logistic regression,[99] SVM,[105−110] adaptive resampling,[111] maximizing information gain,[112] Naïve Bayes classifier,[113] Bayesian active learning[114]	
7	In silico design of new molecules, inverse QSAR, generation of structure possessing desirable properties	Generative models for graphs and chemical structures[26]	Linear generative model for graphs,[332] Spectral generative models for graphs,[333,334] parts based generative model for graphs,[335] White and Wilson generative model for chemical structures[26]	
8	Incompleteness of molecular descriptors	Subgraph (fragment) mining[95,213,216,219]	Frequent subgraph mining algorithms: AGM (Apriori-based Graph Mining),[336] the chemical substructure discovery,[337] the gSpan (graph-based Substructure pattern mining),[220] the TreeMiner,[338] and the CMTreeMiner algorithms,[339] etc.; weighted substructure mining in conjunction with linear programming boosting[93]	Subgraph (fragment) mining[95,213,216,219]
		Graph kernels[33,34,37,40]	Marginalized kernels,[36] graph kernels for chemical structures be Mahé et al,[34] pharmacophore kernel,[35] graph kernels for small molecules by Baldi et al,[33,37] kernel functions for attributed molecular graphs by Fröhlich et al.,[38] convolution kernel for additive inductive learning,[223] molecule kernel,[204] ligand-protein kernels[224−227]	Graph kernels[33,34,37,40]
		Neural network graph machines	BPZ neural device,[27,340] ChemNet,[28] MolNet,[29] recurrent cascade correlation neural networks,[30,31,341] graph learning machine[32,342]	Neural network graph machines
		Inductive Logic Programming (ILP)[141,142] and its applica-		

dx.doi.org/10.1021/ci200409x | J. Chem. Inf. Model. 2012, 52, 1413−1437

Table 1. continued

	Chemoinformatics task or problem	Machine Learning Concept	Machine Learning method	Implementation in freely available software
		tion to statistical relational learning[143]		
9	Accounting for multiple species (conformers, tautomers, ...)	Multi-instance learning[134]	Citation kNN[135]	mi/CitationKNN (*W*)
			Modified Diverse Density Method[136]	mi/MDD (*W*)
			Multi-instance SVM[137]	mi/MISMO, mi/MISVM (*W*)
			Wrapper for applying standard classifiers to multi-instance data[343]	mi/MIWrapper (*W*)
10	Functional input and output (phase diagrams, dose−response curves)	Functional data analysis[246]	Principal Derivative Analysis[246]	Fda,[344] refund (*R*)

norm of the parameter vector. Regularization is an important tool to fight overfitting.

The Vapnik−Chervonenkis theory of statistical learning[5,6] proves that predictive performance of the regularized methods does not directly depend on the number of descriptors. This completely destroys the dogma dominant earlier in the QSAR area concerning the necessity to limit the number of descriptors as much as possible. There exists a variety of methods applying regularizers, such as ridge regression (RR), regularized logistic regression (RLR), regularized neural networks with weight decay, and different types of support vector machines (SVM). The last has become very popular in chemoinformatics,[52] but RR and RLR approaches have been used in very few QSAR studies. Thus, RR has been used by Farkas and Heberger to model retention indices for aliphatic alcohols[53] and Hawkins and Basak[54] and by Merkwirth et al[55] in methodological studies for building predictive QSARs. The RLR method has been successfully used by Spycher et al.[56] to discriminate the modes of toxic action of phenols.

2.1.2. Bayesian Mode. In the Bayesian mode[57] model parameters are considered as random variables, for which the corresponding probability distribution functions can be learned from data by applying Bayes' theorem

$$P(A_i|D) = \frac{P(D|A_i)\cdot P(A_i)}{P(D)} = \frac{P(D|A_i)\cdot P(A_i)}{\sum_i P(D|A_i)\cdot P(A_i)} \quad (2)$$

where D denotes the data (both X and Y), and A_i stands for the i-th value of discrete parameter A. For real-valued A, summation is replaced by integration. The use of random variables for model parameters reflects the fact that there always exists some degree of uncertainty in their values. Predictions produced by such models on new data X' are also considered as random variables characterized by *predictive distributions*

$$P(Y|X') = \sum_i P(Y|X', A_i)\cdot P(A_i|D) \quad (3)$$

In this formula, the predictive probability distribution $P(Y|X')$ is computed as a linear combination of related distributions $P(Y|X', A_i)$ issuing from all possible models with fixed values of parameters A_i and weighted by the probabilities of these models $P(A_i|D)$.

According to the Bayesian approach, $P(A_i)$ is called a *prior* (or prior distribution), whereas $P(A_i|D)$ is called a *posterior* (or posterior distribution). Priors reflect initial beliefs concerning model parameters before seeing training data. In sharp distinction from the frequentist approach, model parameters are not derived entirely from training data. In the Bayesian mode, distributions of model parameters gradually evolve from priors to posteriors under the influence of training data in accordance with formula 2. So, explicit use of priors is a hallmark of Bayesian methods, maybe the most important distinctive feature. This opens additional ways of injecting domain-specific knowledge into models by constructing priors in accordance with the principle of maximum entropy.[58]

It should, however, be pointed out that there exists a strong relationship between both modes in machine learning. In particular, priors in the Bayesian approach can be viewed as counterparts of regularizers in the frequentist one. This can clearly be seen by comparing formulas 1 and 2. This enables to better understand the meaning of regularizers and the use of priors to interpret them. As a result, Bayesian machine learning methods reduce to frequentist ones; whenever a single, the

dx.doi.org/10.1021/ci200409x | J. Chem. Inf. Model. 2012, 52, 1413−1437

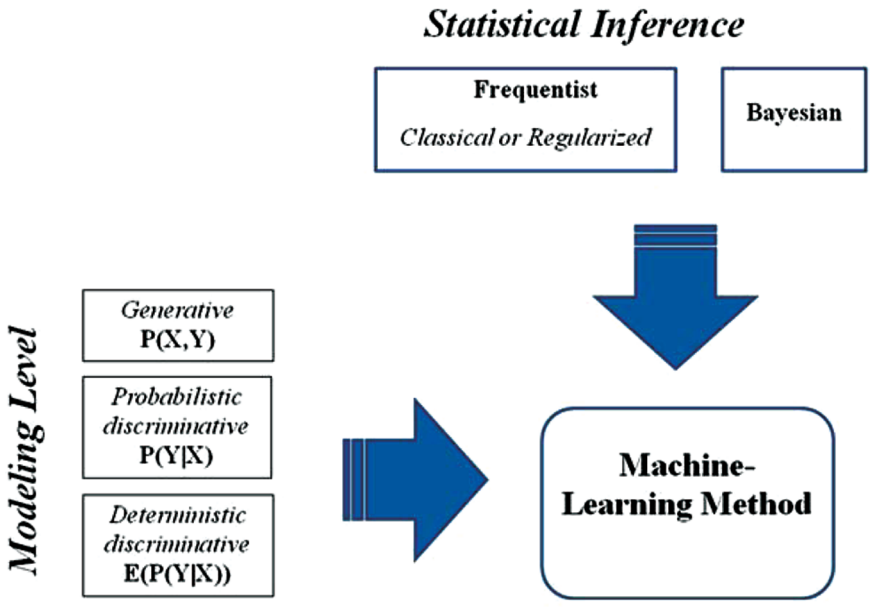

Figure 2. Modes of statistical inference and modeling levels: Two different ways to characterize machine learning methods. Ensemble learning could be positioned between the frequentist and Bayesian inference approaches.

most probable set of model parameters is considered instead of their posterior distribution. As an example, the mean predictor of the Gaussian processes regression[59] (Bayesian) exactly coincides with the solution provided by kernel ridge regression (frequentist).

The most popular machine learning methods involving Bayesian learning are Bayesian regression,[50] Bayesian neural networks,[50,60] and Gaussian processes.[59] The advantages of Bayesian learning algorithms have been demonstrated in recent QSAR studies involving Bayesian neural networks[61−64] and Gaussian processes.[65−67]

2.2. Three Levels of Modeling. In the monograph of Bishop,[50] three levels of modeling in machine learning are considered: deterministic discriminative, probabilistic discriminative, and generative. The simplest, *deterministic discriminative* (DD) level, encompasses all methods in which a model is represented as a function F that maps the input variables X to output variable(s) Y: $Y = F(X, A)$, where A, as before, is a set of model parameters. Regression models operate with real-valued Y, whereas classification models use discrete and especially binary values of Y. An absolute majority of models considered in chemoinformatics belong to this category. The main disadvantage of such modeling lies in the difficulty of assessing the reliability of prediction on new data in the general case without resorting to additional modeling using, for example, the bootstrap procedure. In fact, such an assessment is usually made only for linear models and input data X strictly following the Gaussian distribution and some other conditions.

2.2.1. Probabilistic Models. The *probabilistic discriminative* (PD) level covers approaches based on predictive distributions $P(Y|X, A)$ for regression and posterior probability for classification. In this case, the predicted value of Y for new data X can be extracted as an expectation of this distribution, whereas the prediction errors are assessed by its variance. Thus, both the predicted values and accuracy of predictions can be simultaneously assessed. If some machine learning method does not provide direct assessment of the probability distribution function $P(Y|X, A)$, this can, however, be estimated using resampling techniques such as bootstrap.[68]

Modeling at the probabilistic discriminative level offers some interesting opportunities. Tetko et al.[69,70] have demonstrated for the particular case of the Associative Neural Networks (ASNN) method that models applicability domains could be associated with a threshold of the estimated variance of prediction. The probability distribution $P(Y \mid X, A)$ resulting from PD-level modeling could directly be used for this purpose, as demonstrated for regularized logistic regression and SVM with Platt probability estimation[71] in the modeling of mutagenicity.[70] In principle, Gaussian processes[59] can also be used for this purpose.

2.2.2. Generative Models. The *generative* (G) level covers the methods in which a model is specified by means of either joint distribution of inputs and outputs $P(X, Y|A)$ or by the conditional distribution $P(X|Y, A)$related as

$$P(X, Y|A) = P(X|Y, A) \times P(Y) \quad (4)$$

In *generative* models, the input data X are generated by sampling from the $P(X|Y, A)$ distribution. The latter can be regarded as a description of a stochastic generator of inputs X possessing given values of outputs Y.

To summarize, modeling at the G-level, on one hand, leads to a more sophisticated model, but on the other hand, it requires more data and computational resources compared to the DD and PD-levels. Existing machine learning methods can be associated with one or several levels. For instance, in the frequentist (classical) inference, Multiple Linear Regression (MLR), Partial Least Squares (PLS), and Neural Network modeling are typically performed at the DD level. Support Vector Machine (SVM) is usually applied at the lowest DD-level, but the application of the Platt technique[71] raised it to the PD-level. Modeling completely at the G-level usually requires very sophisticated combined approaches, although some intermediate G-level steps are present in several well-known statistical methods, such as Naïve Bayes and Linear Discriminant Analysis procedures.

In chemoinformatics, the application of generative models can lead to the generation of chemical structures possessing given property values (the "inverse problem" in QSAR[17−23,25] that is an alternative solution to de novo design[25,26]). Early works in this direction used some heuristic approaches involving either stochastic or exhaustive (under some constraints) structures generation. One can mention studies by Zefirov's group,[17−19,21] Kier et al.,[20] Rücker et al.,[22] and Miyao et al.[25] in which molecular graphs have been reconstructed from simple topological indices correlating with certain physicochemical properties (typically, the boiling points) of alkanes, aliphatic alcohols and their derivatives, or a publication by Churchwell et al.[23] devoted to design of novel ICAM-1 peptide inhibitors using signature descriptors. In kernel-based methods (Section 4.5) the "inverse QSAR" is related to the *pre-image* problem: Given a point in the feature space, one should find a related point in the input space.[72] The points in the feature space can be either generated from some distribution or found, using the model's derivatives, from the required property value. Accordingly, Wong and Burkowski[24] suggested the "constructive approach", in which a new point generated in the feature space is directed back using a pre-image approximation algorithm to the initial descriptor space followed by reconstruction of the corresponding molecular graph using a suggested recovery algorithm. The probability distribution function is not considered explicitly in this approach.

To our knowledge, the only original algorithm to build real generative models $P(X)$ and to generate new chemical structures belonging to $P(X)$ distribution was suggested by White and Wilson.[26] They transformed connection tables of the molecular graphs from the initial set S into vectors, whose dimensionality was reduced using principal component analysis. Then, the distribution of the resulting vectors $P(X)$, defining the probability of the graph X belonging to the set S, was approximated by the ensemble of Gaussian functions. Sampling of $P(X)$ led to generation of new structures. In order to avoid generation of structures containing a noninteger number of atoms, several rules were suggested. The feasibility of this approach has been demonstrated in the case of ligands against the COX2 and EGFR biotargets and proved in docking experiments.[26]

3. ENSEMBLE LEARNING

The recently arisen *ensemble learning* concept can be positioned between the frequentist and Bayesian inference modes (Figure 2). This approach considers an ensemble of models each of which is obtained in the model selection procedure. A consensus model combines selected individual models in order to perform predictive calculations on the independent test set.

In the QSAR/QSPR area, consensus prediction by simple averaging of outputs of individual models has empirically been found as an efficient way to enhance predictive performances.[55,73−78] Nonetheless, chemoinformatics still benefits little from recent advances in the machine learning field, which could transform "bad" individual models into one "good" consensus.

In the data mining area, ensemble learning provides some meta-methods that combine several "weak learners" (very simple methods producing models with poor predictive performance) in order to produce "strong learners". There exist two main strategies to generate individual (or base) models issued from weak learners: *parallel*, in which all models are built independently, and *sequential*, in which they are built one-by-one, taking into account the model's performance at the previous stage.

There are three basic approaches to parallel model generation: (1) resampling at random the training set (*bagging and dagging methods*), (2) using random sets of descriptors (*random subspace*), and (3) introducing random modifications to learning algorithms or taking different learning methods (*stacking*).

The first approach was implemented by Breiman in the *bagging* (bootstrap + aggregation) method.[79] Here, each sample is obtained from the initial data set using the bootstrap procedure,[68] i.e., by drawing at random with replacement instances from the initial training set, the probability of each instance to be drawn being the same. After that, a set of samples formed in this way is used to build an ensemble of *base models*, which are combined into one consensus model by majority voting (for classification) or by averaging (for regression). Bagging allows one to achieve significant improvements of predictive performance of so-called "unstable" methods, which produce very different models for slightly perturbed training sets. For example, the Random Forest[80] algorithms based on an ensemble of random tree models (weak learners) is one of the most successful classification methods in chemoinformatics applications.[81−83]

In the *dagging* technique, the base models are built on different parts of the initial training set.[84] Evidently, this could be applied to process huge data sets. Diversity of the base models can also be reached by manipulating with attribute (descriptor) sets. Thus, in the *random subspace* meta method,[85] base models are built on random subsets of attributes. A similar approach involving the use of nonrandom sets of attributes (descriptors) has been applied in various QSAR studies.[78]

In the *stacking* approach, the base QSAR models with different machine learning methods are combined by means of a separate model (usually MLR or PLS), parameters of which are learned by data fitting.[86,87] The output of each individual model is considered as descriptor in the "consensus" MLR or PLS model.

A sequential strategy of base model generation is involved in the *boosting* approach in which each next learner focuses on mistakes of the previous learner. In the *AdaBoost* algorithm,[88] the most prominent implementation of boosting for classification tasks, the probability of an instance (chemical compound) depends on the predictive performance of previous base learners on the same instance, as applied to QSAR in ref 89. Another popular boosting algorithm is the *gradient boosting machine*[90] and its stochastic modification, the *stochastic gradient boosting*,[91] which can be applied to solve both classification and regression tasks in structure−activity studies.[92] A *linear programming boosting* algorithm[93] and gBoost method[94] in the model-building procedure perform extraction of the most useful fragment descriptors from molecular graphs.[94,95]

In most of cases, ensemble learning leads to considerable improvements in prediction performance (Figure 3). Its main advantages are (1) very easy implementation of basic algorithms and (2) surprisingly good prediction performance of consensus models.

4. MODELS DESCRIPTION

Here, we consider different facets of the modeling regarding input/output matching, model types, model interference, tasks types, and duality of models (Figure 4).

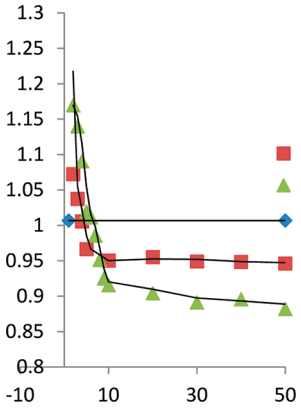

Figure 3. Prediction performance (RMSE) of QSAR modeling of aqueous solubility as a function of the numbers of individual MLR models involved in ensemble for (a) bagging and (b) the random subspace method.[291] The horizontal line at RMSE = 1.01 corresponds to the performance of one individual MLR model built on the whole set of compounds and descriptors.

4.1. Input/Output Matching. Most machine learning methods deal with either *unsupervised* or *supervised* learning. In unsupervised learning, the data is used without distinctions between "input" or "output" variables. Its goal is to analyze the data distribution, reduce data dimensionality, or reveal the patterns hidden in the data. In *supervised* learning, each training example contains both inputs (X) and outputs (Y) labels, and the task is to predict outputs for given inputs. There are also some other types of learning—Semi-Supervised, Transductive, Active, and Multi-Instant Learning—that cannot be assigned to these two categories and could be of particular interest for chemoinformatics.

4.1.1. Semi-Supervised and Transductive Learning. In the semisupervised learning,[96] the outputs are specified only for some examples (such examples are called *labeled*, while examples without outputs are called *unlabeled*), and the task is the same as in supervised learning. In QSAR, the labeled data denote those compounds for which a property is specified. In certain cases, the unlabeled part of training data improves a model built on the labeled part. As an example, one can consider "transductive" SVM (TSVM)[97] in which the separating hyperplane is directed through the region of low data density. This means that TSVM enforces unlabeled instances to be far from the separating hyperplane but does not take account of which side of the hyperplane they are situated. Addition of unlabeled data to the training set facilitates the density assessment and, hence, helps to define an "optimal" position of the hyperplane. This allows one to reduce the number of misclassified examples which in conventional SVM are mostly located near the separating hyperplane (Figure 5).

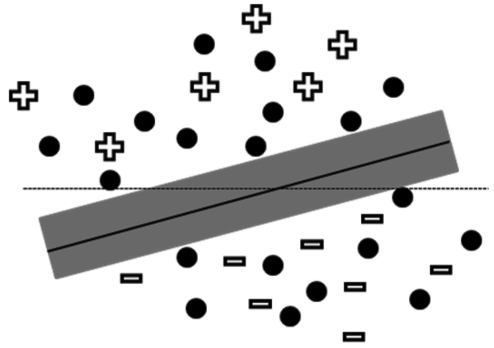

Figure 5. Object separation in SVM and TSVM. Labeled training set examples are depicted as "−" and "+", whereas unlabeled examples are shown as bold dots. Dashed and solid lines indicate the hyperplane found in conventional SVM and TSVM, respectively. One can see that the TSVM hyperplane passes through the low-density area.

A special case of semisupervised learning, when unlabeled data coincides with the test set and no other data is to be predicted, is called *transductive learning*. Unlike an ordinary inductive learning (conventional QSAR modeling) where the training data are used to build a "universal" model supposed to be used on any test set, in transductive learning, the model is specifically built to predict the objects in one particular test set. This may lead to significant improvement of the model's performance because a specificity of the test set is taken into account in the learning process.

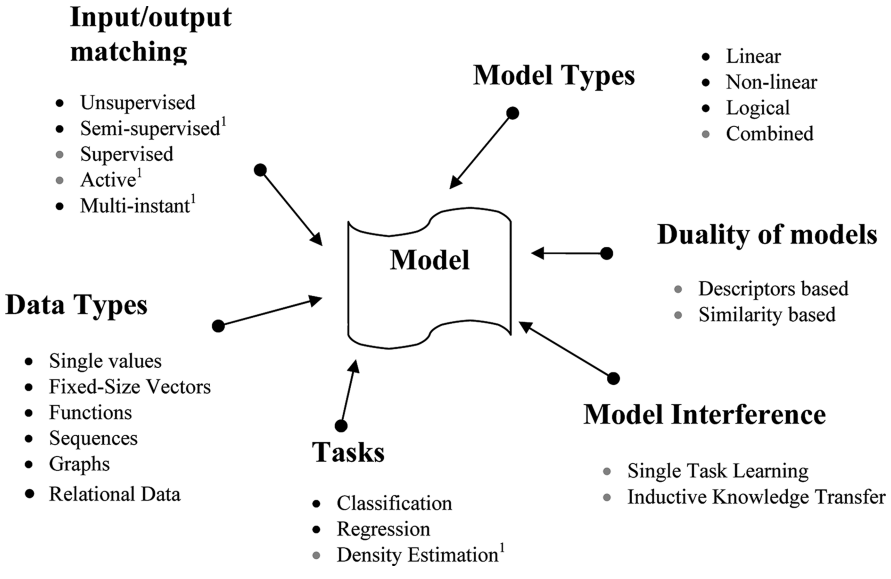

Figure 4. Different aspects of model description.

dx.doi.org/10.1021/ci200409x | J. Chem. Inf. Model. 2012, 52, 1413−1437

4.1.2. Active Learning. In active learning,[98−101] a given statistical model is used to suggest new data to be added to the training set in order to develop a new model with better predictive performance. Thus, the modeling could be started with relatively small training sets that then iteratively grow up in the learning process. This approach could be very useful in taking decisions as to which molecules should be acquired in order to build the most suitable new data for the QSAR modeling training set. Successful application of two virtual screening strategies, "query by bagging" and "query by bagging with descriptor-sampling", based on active learning to discover ligands for several G-protein coupled receptors has been reported in ref 102. There are many different implementations of this approach in neural networks[103,104] or their ensembles,[100] logistic regression,[99] SVM,[105−110] adaptive resampling,[111] maximizing information gain,[112] Naïve Bayes classifier,[113] Bayesian active learning,[114] and some other methods.[115]

4.1.3. Multi-Instance Learning. The question of the relative importance of different forms of the molecule (conformers, tautomers, protonated/deprotonated forms) is a permanent focus of chemoinformatics. Different approaches tackling this problem have been developed. Hopfinger et al.[116] invented a 4D QSAR method including sampling of conformation and different types of alignment. The composite information coming from each of these sampled property sets is embedded in the resulting QSAR model. This is an elegant way to assess a "bioactive" conformation that is not necessarily associated with the local minima of the free molecule and can also account for stereoisomeres and different protonation states of ionizable groups.[117] Work by the Hopfinger's group[116,118−123] inspired development of 5D QSAR accounting for a multiple representation of induced-fit hypotheses[124,125] and 6D QSAR that evaluates different solvation models.[126]

Topological Pharmacophore Triplets by Horvath et al.[127] and ISIDA Property-Labeled Fragment Descriptors by Ruggiu et al.[128] account for photolytic equilibrium assessed by ChemAxon tools.[129] These descriptors have successfully been used in QSAR[130] and in similarity search-based virtual screening.[78]

Balaz et al.[131−133] have developed the "multi-mode" CoMFA methodology based on explicit consideration of thermodynamic equilibrium between different binding modes. The prevalence of each mode is assessed by comparing binding energies predicted by QSAR models involving special descriptors, probe interaction energies weighted by contributions of individual binding modes. This requires iterative procedure for reaching self-consistency because the descriptors' values themselves depend on these binding energies.

In parallel to the above-mentioned chemoinformatics techniques, the multi-instance learning[134] approach has been developed in the data-mining area. In this method, every training object represents an ensemble (so-called *bag*) of instances, each of which is described by a fixed-sized vector of descriptors. Only one of the instances inside a bag is labeled, and just this label defines the label of the whole object (see example below).The goal of the training is to build a model that predicts both the label of a test object and the labeled instance in its bag. In chemoinformatics, the bag could be associated with the ensemble of conformers (instances), only one of which is able to bind a protein (which is labeled). In this case, the model is expected both to predict biological activity and to identify a biologically active conformer for each test set molecule.[16] Multi-instant learning is a general purpose mathematical method that is not constrained by some "chemical" assumptions, and therefore, it opens interesting perspectives for chemoinformatics. As an example, one could mention the publication by Dietterich et al.[16] in which a classification model of the strength of musk odor has been built accounting for the conformational sampling of the molecules. Multi-instant learning is involved in different machine learning methods, such as Citation kNN,[135] Modified Diverse Density Method,[136] and Multi-instance SVM[137] (Table 1).

4.2. Types of Models. Three main types of models are considered in machine learning: linear, nonlinear, and logical. One should make a clear distinction between linearity in X (attributes, features, molecular descriptors) and in A (model parameters). Although in most textbooks on machine learning the linearity of a model is defined with respect to A,[2,50,138] some publications consider also a linearity in X.[139,140] For instance, the model $y = a_1x + a_2x^2$ is considered to be linear according to the former convention and nonlinear according to the latter one. In classical statistics, the models with real-valued inputs were naturally classified as linear or nonlinear. Introduction of kernels makes this division conditional. Indeed, depending on the nature of the corresponding kernel, any linear model in the feature space can be either liner or nonlinear in the input space.

Logical models are becoming more and more popular in chemoinformatics. This area has received a powerful impetus due to developments of the Inductive Logical Programming (ILP)[141] and especially of its probabilistic variant[142] incorporating many basic ideas of modern machine learning, such as Bayesian learning, kernels, structured input, etc. The main advantage of ILP stems from its ability to provide relational learning[143] and therefore to treat structured input data of any complexity, including molecular graphs. ILP has successfully been applied to mutagenicity[144,145] and toxicity[146] prediction, pharmacophore discovery,[147] classification of bioactive chemical compounds,[148] scaffold hopping in drug discovery,[149] building ordinary[150,151] and field-based 3D QSAR models,[152] etc.

4.3. Model Interference: Inductive Transfer of Knowledge. Humans are known to be able to learn from a small number of training examples, whereas current machine learning approaches require a larger number of training examples to solve even relatively simple problems. An apparent explanation to this fact lies in the ability of humans to reuse the knowledge previously learned from related tasks. This strategy is taken into account in the *inductive knowledge transfer* approaches (see review in ref 153), Multitask and Feature Net learning. *Multitask Learning* (MTL)[154] takes several tasks in parallel and uses a shared representation of data. This can be carried out using machine learning methods yielding models with several outputs, such as neural networks, PLS, or SVM with special kernels.[155] *Feature Nets* (FN),[153] another type of inductive transfer, uses extra tasks to build the models, predictions of which are further used as extra inputs for the main task (Figure 6).

In chemoinformatics, the inductive knowledge transfer approach could become an efficient solution to build QSAR models on small and structurally diverse data sets. Most of these data sets are under-sampled in the sense that additional data could significantly improve performance of models built on them. However, the cost of obtaining new data could be rather high, especially for in vivo experiments, e.g., ADMET properties. In such cases, the integration of already available experimental data on other properties somehow related to the

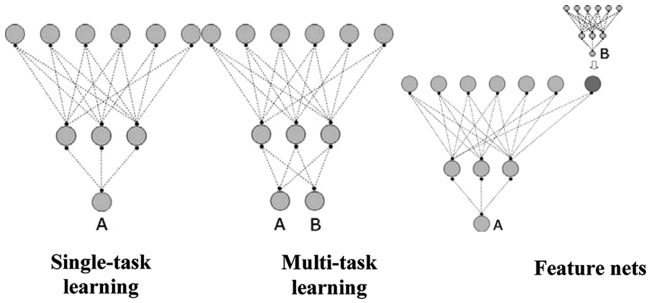

Figure 6. Single-task learning, multitask learning, and feature nets modeling performed with artificial neural networks. In single-task learning, a target property (A) is learned without taking into account a supplementary property (B). In multitask learning, both A and B are learned simultaneously, whereas in feature nets, the property B is used as an additional descriptor to build a model for A. Note that supplementary property B could be either experimentally measured or theoretically calculated.

target one could become a good alternative to costly and time-consuming acquisition of new experimental data. It should be noted that theoretically calculated values could also play a role of supplementary properties. In such a way, molecular descriptors could be used both as input and as output in the neural network realizing multitask learning (Figure 7).

Higher performance of MTL and FN approaches over conventional Single Task Learning (STL) has been demonstrated by Varnek et al.[153] in QSAR modeling of tissue−air partition coefficients (log *K*) using the neural networks method. The initial data set contained 11 different individual data sets for different types of log *K* for human (*H*) and rat (*R*); only four of which were of reasonable size (about 100 compounds), whereas the others contained from 27 to 38 compounds. The output layer of the 3-layers neuron network contained 1 (for STL and FN) or 11 (MTL) neurons, corresponding to the number of simultaneously treated properties. In STL and MTL calculations, only fragment descriptors were used as an input, while in FN calculations, the models built only for one target property used the other 10 properties as complementary descriptors. Figure 7 shows conventional STL modeling results in predictive models only for four properties corresponding to relatively large (about 100 compounds) data sets, whereas with MTL and FN approaches significantly improve the reliability of the calculations for predicting nine types of log *K*.

4.4. Tasks: Regression, Classification, and Density Estimation. Vapnik[5] considered three main tasks in statistical learning: regression, classification (pattern recognition), and density estimation. The last relates to the assessment of the probability density $P(X)$ (Section 2.2) and forms a basis of various popular unsupervised methods: clustering, dimensionality reduction, and novelty detection. Thus, clusters correspond to high density "clumps" of data. Dimensionality reduction methods detect subspaces (or *manifolds*) containing the greater part of the data density, while novelty detection[156−158] methods define regions with high data density.

The novelty detection (or one-class classification[156−158]) approach considers two types of instances: "object class" formed by the training objects and the others ("outliers"). A new instance is considered as belonging to the object class if it lies in the dense area of point clouds formed by the training set and as an outlier if outside. Thus, any "object class" instance is viewed as being similar to all instances in the training set.

The fundamental difference between the one-class classification and the conventional similarity search is an ability to use the whole training set instead of a single query instance and to learn implicitly the optimal metric for similarity measure. Another important feature of this approach is that only instances belonging to the "object class" take part in the learning. If the "object class" includes only active compounds, reliable models could be built on highly unbalanced data in which actives are rare. This makes the novelty detection (one-class classification) very promising tool for similarity-based virtual screening. Very few applications of this approach have been reported in chemoinformatics literature. Thus, Hristozov et al.[159] used a Kohonen self-organizing map as a model applied to the virtual screening of ligands. Karpov et al.[160−162] used the autoassociative neural networks and the one-class Support Vector Machines (1-SVM) in virtual screening against numerous biological targets.

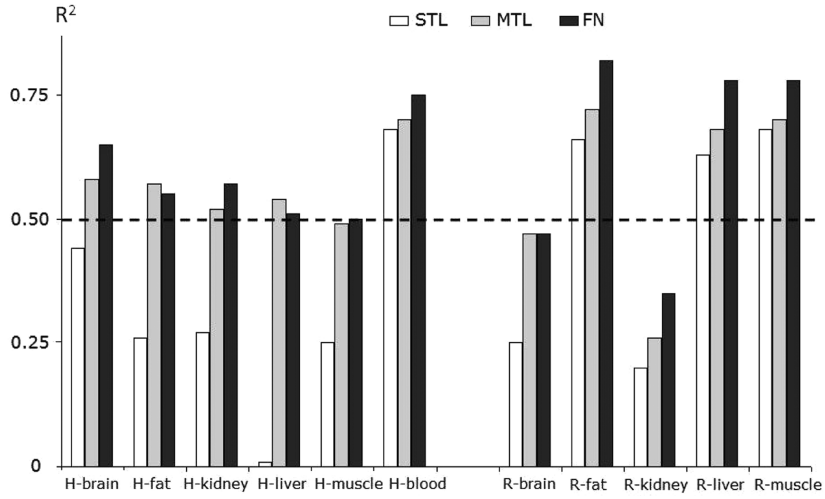

Figure 7. Performance of different learning strategies to predict human or rat air−tissue partition coefficients. MTL and FN calculations are involved all 11 studied properties. The determination coefficient R^2 was obtained in external 5-fold cross-validation. The horizontal line at $R^2 > 0.5$ corresponds to the model acceptance threshold (see details in ref 153).

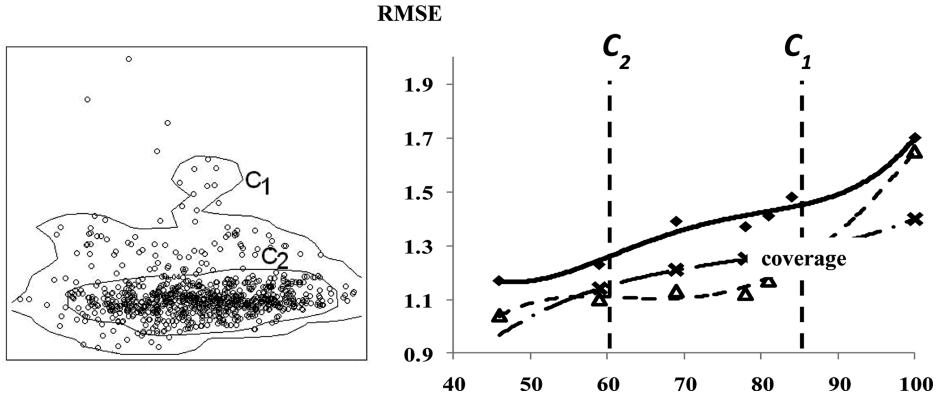

Figure 8. (left) Density-based approach to the applicability domain of any model build on the given data set. In chemical space defined by two variables, the data set includes both "target class" objects situated in a high density region inside the iso-density contour C_1 and the "outliers" located outside C_1 that should be excluded. In 1-SVM, target class objects and outliers are separated in the feature space by a hyperplane that corresponds to C_1. (right) QSPR modeling of stability constants for complexes of Ca^{2+}(rhombs), Sr^{2+} (crosses), and Ba^{2+} (triangles) cations with organic ligands: RMSE for the prediction of as a function of the data set coverage. Moving the iso-density contour to a higher density region (from C_1 to C_2) leads, on one hand, to the increase the model's performance, but on the other hand, to a decrease in the test set coverage (see details in ref 177).

Nowadays, many different machine learning methods performing novelty detection have been reported in the data mining literature: Fast Support Vector Domain Description (F-SVDD),[163] Structured one-class classification (TOCC),[164] One-class Very Fast Decision Tree (OcVFDT) algorithm,[165] Condensed Nearest Neighbor Data Description (CNNDD) algorithm,[166] and Semi-Supervised Support Vector Domain Description.[167] Novelty detection can also be performed with autoassociative neural network,[157] SOM,[168] kernel PCA,[169] single-class minimax probability machines,[170] one-class Parzen density estimator,[171] Gaussian Mixture Models (GMM) in Gabor space,[172] Multivariate Extreme Value Statistics,[173] and some others[174,175] (Table 1).

The one-class classification models can also be used to describe data domains[158] in which a statistical model provides reliable predictions ("models applicability domain"[176]). Baskin et al.[177] applied the 1-SVM approach[178] to build one-class models approximating bounds of high density levels of data points. The resulting models were used to define the applicability domain of regression QSPR models built on the same training set[177] (Figure 8). A similar approach was suggested by Fechner et al[179] to define the applicability domain of kernel-based models for virtual screening. This method is similar to the concept of an "external applicability domain" by Soto et al,[180] which is based on thresholding data density levels. This differs from the "internal applicability domain"[180] based on thresholding posterior class probabilities and which is conceptually close to the "distance to model" approach.[69,70]

An extrapolation of the models to the data outside of this domain may be possible in the framework of the *domain adaptation* concept. Several strategies have been applied in papers[181−184] to simulated data and text mining. Thus, if a data domain corresponding to a given training set partially overlaps with another data domain (test set), the model should specially be trained by ascribing different weights to training data points.[181] Various methods for domain adaptation have been reported: covariate shift adaptation by importance weighted cross validation,[181] feature subsetting,[183] conditional random fields,[184] cross-domain generalizable features,[185] semisupervised domain adaptation via structural frequency features,[186] and some others.[187]

4.5. Duality of Models: Primal and Dual Representations. For historical reasons, in chemoinformatics, there is still a distinction between similarity-based (such as similarity search, *k* nearest neighbors, etc.) and nonsimilarity-based (multiple linear regression, PLS, neural networks, etc.) methods, although in several approaches, such as the Influence Relevance Voter,[188] these two approaches are bridged. The former approach is directly linked to the Johnson−Maggiora postulate: "similar chemical compounds have similar properties".[189] In machine learning, these two approaches could be related via the "representer theorem" by Kimelford and Wahba,[190] according to which any statistical model obtained by minimization of the functional Φ in formula 1 has the following representation

$$F(X, A) = \sum_i \alpha_i K(X, X_i) \tag{5}$$

Here, $K(X, X_i)$ is a similarity measure between a test object X and the i-th object X_i from the training set, also called the *kernel*. The left side of formula 5 represents the primal representation, while the right side stands for the dual representation of a statistical model. Thus, in machine learning, similarity-based and nonsimilarity-based methods are considered as equivalent, and any model can be represented in the both the forms. In machine learning, the "chemical" Johnson−Maggiora similarity principle relates to the smoothness of the function $F(X, A)$, whereby the best model corresponds to the minimal squared norm of the $\|A\|^2$ regularizer and, as a consequence, to the smoothest function. It should be noted that the lack of smoothness of the function $F(X, A)$ may be interpreted as "activity cliffs".[15]

The most important property of kernels concerns their ability to define implicitly the vector space (so-called *feature space* or, in strict mathematical terminology, *Reproducing Kernel Hilbert Space*, RKHS) when the value of kernel $K(X_i, X_j)$ for a pair of objects X_i and X_j in the input space is equal to the inner product (also called the scalar or dot product) of their images $\Phi(\cdot)$ in the feature space

$$K(X_i, X_j) = <\Phi(X_i), \Phi(X_j)>_H \tag{6}$$

where $<\cdot,\cdot>_H$ denotes the inner product taken in the feature space. It is important to point out that feature spaces of this

dx.doi.org/10.1021/ci200409x | J. Chem. Inf. Model. 2012, 52, 1413−1437

kind can only be induced by so-called positive definite kernels, for which a kernel matrix contains only positive eigenvalues.

It can be shown that any "reasonable" regression model of any complexity, linear or nonlinear, using objects X of any complexity (not necessarily vectors of fixed length) can be represented as a linear model in the feature space. The dimensionality of the latter is usually higher than that of the initial space. Numerous kernel-based machine learning methods (SVM, Gaussian processes, kernel ridge regression, etc.) building models in the feature space have been developed.[178,191,192] In chemoinformatics, kernels accounting for the similarity between molecules are usually calculated from fingerprints or descriptor vectors using either some standard functions (linear, polynomial, Gaussian) or some other popular similarity measures such as Euclidean distance or Tanimoto coefficient. Positive definite kernels for complex systems (e.g., protein−ligand complexes) can be constructed using a set of simple rules (so-called "kernel engineering").[178,193] This led the "kernel revolution" in many fields, including chemoinformatics. In a short period, the kernel-based support vector machines (SVM) approach has become, perhaps, the most popular method to build classification and regression structure−activity models.[45,52]

In chemoinformatics, the choice of "optimal" kernel proceeds usually in empirical way. On other hand, *kernel learning* approaches reported in the data-mining literature offer a systematic means of kernels selection. Two kinds of approaches are considered. Parametric methods optimize the fixed-sized sets of parameters of kernel functions, whereas nonparametric ones optimize directly the elements of the kernel matrix. One can also distinguish between single kernel learning, in which a single kernel matrix is optimized, and *multiple kernel learning*,[194] in which several basis kernels are combined in an optimal way. These kernels may correspond to either different notions of similarity or different representations of objects (e.g., different types and subsets of descriptors in chemoinformatics). Resulting kernels are represented as linear combination of basis kernels with certain requirements imposed on the corresponding mixing coefficients

$$K(X_i, X_j) = \sum_m \eta_m K_m(X_i, X_j) \tag{7}$$

Numerous approaches addressing the problem of finding the optimal values of the mixing coefficients have been suggested, including the convex optimization by means of semidefinite programming,[195] methods based on the concept of kernel-target alignment,[196] the use of hyperkernels,[197] etc. Because in chemoinformatics different kernels $K_m(\cdot,\cdot)$ can be based on different ways to describe molecules, multiple kernel learning offers a unique possibility to find the best way to combine different types of molecular description. A similar approach has been taken recently in Baskin's group in the framework of the Continuous Molecular Fields approach,[162,198] in which "base kernels" correspond to different types of molecular fields. The mixing coefficients in such combinations correspond to the relative contributions of different types of intermolecular interactions and can be learned from data.

If the required good set of base kernels cannot be constructed, the entries in the kernel matrix can be learned directly from data with the help of nonparametric methods.[199] Although one can easily find an "optimal" kernel matrix by optimizing some performance measure of any kernel-based machine learning method on the training set with regard to the values of its entries, such a naïve approach does not solve the problem of how to obtain kernel matrix entries for the test set. A general approach to address this problem is to conduct a study in the transductive (semisupervised) setting (Section 4.1.1), in which all labeled (training set) and unlabeled (test set) data are combined into a single all-data set, which can be described by means of a joint similarity matrix. Information contained in such empirically chosen similarity matrices can effectively be used to regularize the above-mentioned optimization process and provide necessary connection between the training and the test sets.[199−201]

An alternative approach to learning kernels concerns only those of them that are functions of distances between objects, such as the Gaussian kernel. In the latter case, the *metric learning* algorithms[202,203] can be applied to learn the optimal metric for computing distances.

It should also be noted that some chemo- and bioinformatics studies involve *indefinite* kernels, the kernel matrix of which contains both positive and negative eigenvalues. In chemoinformatics, one can mention the "Molecular kernel" method of Mohr et al.[204] for estimating pairwise alignment of molecules. In bioinformatics, some similarity measures for protein sequences, e.g., Smith−Waterman and BLAST scores[205] and the similarity measure of Hoffmann et al.[206] for comparing protein binding pockets are also indefinite kernels.

Strictly speaking, indefinite kernels can result from nonconvex mixing of positive definite kernels. In principle, indefinite kernels should never be used with machine learning methods designed for positive definite kernels, such as SVM. However, in the so-called Reproducing Kernel Krein Spaces (RKKS), it has been shown that the "representer theorem" can be generalized for these kernels, and they also implicitly define feature spaces but with different mathematical structure.[207] In contrast to RKHS feature spaces, which are characterized by Euclidean geometry, RKKS are pseudo-Euclidean spaces, in which coordinates can be complex-valued numbers, and the square of distance can be negative. Recently, several machine learning methods specifically working with indefinite kernels, such as Indefinite Kernel Fisher Discriminant analysis (IKFD),[208] least-squares regression with indefinite kernels, and coefficient regularization,[209] etc. have been developed. To our knowledge, neither of these promising methods has been used in chemoinformatics.

4.6. Data Types. The main distinction of chemoinformatics from other fields in which machine learning is applied concerns the data types used. Chemistry mainly deals with chemical structures and their transformations. Therefore, molecular graphs describing chemical structures and molecular descriptors−some binary, integer, and real-valued parameters derived from these graphs−are basic data types to handle the information flow in chemistry.

Table 2 presents data types used as inputs (X) or outputs (Y) in statistical models. Classical statistics works with only fixed-size input vectors. To meet this requirement, molecular descriptors are used to map molecular graphs to vectors.[210] Three types of vectors are used in chemoinformatics: (i) vectors of bits (bitstrings) corresponding to the screens or fingerprints, (ii) vectors of integer values forming by fragment descriptors (counts of substructures),[211] and (iii) vectors of real-valued numbers involving other types of descriptors. Functions describing molecular fields[198] can be considered as an extreme case of real-valued vectors. Representation of chemical structures by descriptor vectors leads to at least two

Table 2. Different Types of Data Used in the Modeling: Input *X* and Output *Y*[d]

Y			
Data types / *Data templates*	**Binary**	**Integer**	**Real**
Single value	Binary Classification	Multi-class Classification	Regression
Fixed- size vector	MTL [c] Classification	MTL Ranking	MTL Regression
Sequence	*a*		
Graph	*b*		

X			
Data types / *Data templates*	**Binary**	**Integer**	**Real**
Fixed-size vector	Bitstrings	Counts	Real value descriptors
Functions			Continuous fields
Sequence	Sequence kernels		
Graph	Graph kernels		
Relational data [e]	Ensemble of Relational Tables		

[a]Structured output: sequences (e.g., aligned sequences in proteins or in nucleic acids) or [b]graphs (e.g., chemical structures). [c]Multi-Task Learning (MTL). [d]Color code: data types (approaches) widely used in chemoinformatics are given in blue, rarely used; in yellow, never used; in brown, not pertinent for chemoinformatics; colorless. [e]Data represented as a set of logical predicates kept in tables of relational databases.

problems: (1) a huge number of known molecular descriptors and, hence, the inability to guarantee an optimality of any their subset and (2) difficulties in establishing bijection (one-to-one correspondence) between descriptor vectors and graphs. A solution can be found in the framework of neural networks based molecular graph mining (Section 4.6.1), molecular subgraph mining (Section 4.6.2), and molecular graph kernels (Section 4.6.3) approaches. In Table 2, the "*X*"-part describes structured inputs, some of which are rarely (graphs and sequences) used in chemoinformatics. The "sequence" data type is used in bioinformatics to represent primary structures of biopolymers; another area of its application is in text processing. Both graph and relational data inputs can be used for direct processing of chemical structures. However, their application in chemoinformatics is still very limited. The "*Y*"-part (Table 2) describes the output part of supervised models. Most of the machine learning methods produce models with a single value as output. In this case, the binary values correspond to the binary classification and the integer values stand for the multiclass classification (or clustering), whereas the real-valued numbers correspond to the regression task. Vector outputs correspond, in particular, to multi-task learning (Section 4.3). Methods using graph input/output are of particular interest with respect to the modeling of molecules and reactions. They are considered below.

Structured data mining[212] approaches and *graph mining*[213] are particularly important in describing chemical entities. Unlike traditional data mining methods dealing only with fixed-sized vectors, they build statistical models for data of any complexity such as variable-sized vectors, sequences, sets, multisets, trees, graphs, relational data (i.e., organized into relational databases[214]), functions,[198] etc. Notice that the structured data can be used not only as inputs *X*, but also as outputs *Y* in supervised models $Y = F(X)$.[215] Realization of these possibilities could open new exciting perspectives for chemoinformatics (Section 4.6.4).

4.6.1. Molecular Graph Mining with Neural Networks. One of the earliest applications of graph mining in chemoinformatics concerns neural networks with special architecture that allows one to assess some molecular properties directly from molecular graphs, avoiding computation of molecular descriptors. Thus, this approach has been used in QSPR studies by Baskin et al.,[27] Kireev with ChemNet,[28] Ivanciuc with MolNet,[29] Bianucci et al. with recurrent cascade correlation neural networks,[30,31] and Dreyfus et al with "graph learning machine".[32]

4.6.2. Molecular(Sub)Graph Mining. Another very promising graph mining approach involves extracting from molecular graphs only those fragments (substructures) that could be useful to predict a given property of chemical compounds.[94,95,213,216−220] Conventional fragmental approaches imply generation of some particular types of fragment descriptors (e.g., sequences of atoms and bonds, atoms with their closest environment, etc.) followed by the variables selection procedure. Clearly, they are not able to enumerate all possible fragments because of their enormous variety. In contrast, (sub)graph mining methods extract from molecular graphs only the task-oriented fragment descriptors of any complexity.[221] The practical implementation of such methodology in the QSAR area has been demonstrated by Saigo et al.[94,95,222] who embedded a graph mining algorithm in some mother machine learning method. Figure 9 shows the list of the most important substructures extracted in ref 95 while building a QSAR model for the activity of endocrine disruptors using the DFS trees graph mining algorithm[220] and the linear programming boosting machine learning method. This list contains not only the linear ones but also branched and cyclic

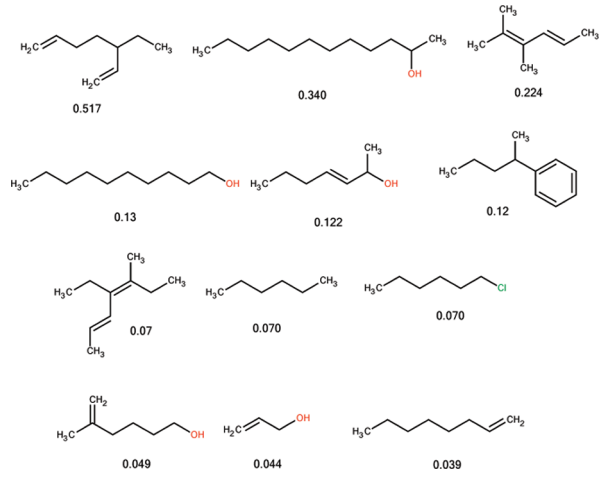

Figure 9. Substructures extracted by Saigo et al.[95]as fragment descriptors in linear QSAR model for predicting activity of endocrine disruptors (*Y*) using the linear programming boosting method. The numbers represent regression coefficients a_i in equation $Y = \sum a_i X_i$, where X_i is the occurrence of the *i-th* fragment. Notice that the selected fragments do not belong to any particular set of preselected fragments but are directly extracted from molecular graphs of the training set.

dx.doi.org/10.1021/ci200409x | *J. Chem. Inf. Model.* 2012, 52, 1413−1437

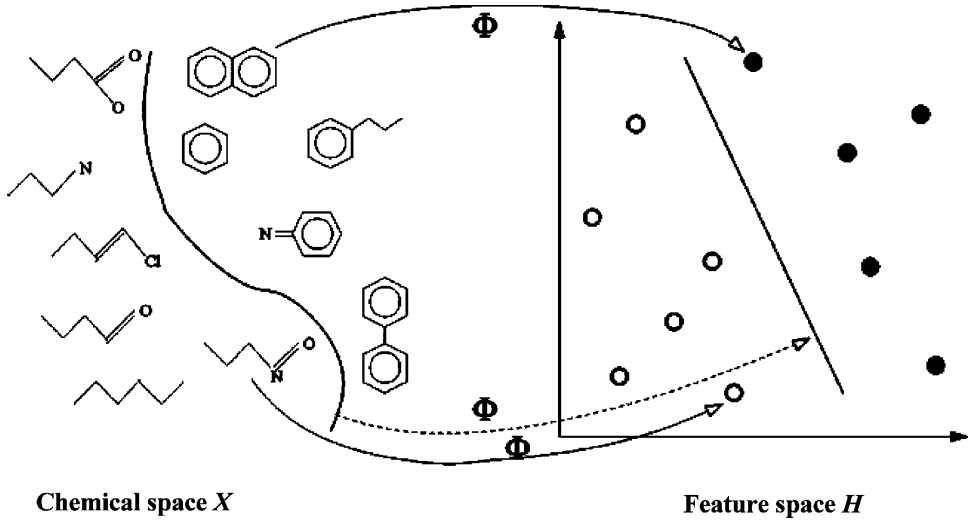

Figure 10. The graph kernel K $K(X_i, X_j) = < \Phi(X_i), \Phi(X_j) >_H$ implicitly defines the mapping from the initial (graph-based) chemical space X to the feature (vector-based) space H. The dimensionality of H may be huge.

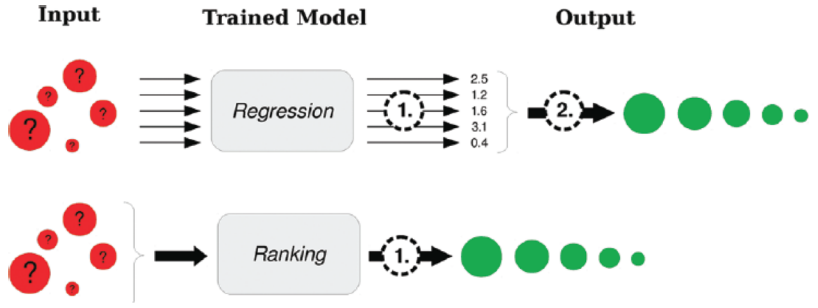

Figure 11. Two different approaches to perform ranking-based virtual screening. Top: Conventional two-step approach implying quantitative activity assessment using previously built QSAR regression model followed by sorting of the molecules and selection of top k actives. Bottom: Ranking is a structured output of the StructRank one step algorithm.[234].

fragments, which can only with difficulty be generated using commonly used software for conventional fragment descriptors (see ref 211). In ref 222, Saigo et al. successfully implemented the graph mining algorithms with support vector machines, partial least-squares, and least angle regression (LARS) as the mother machine learning method. Thus, a combination of modern machine learning methods with graph mining techniques could be regarded as a very promising direction in the future development of chemoinformatics.

4.6.3. Molecular Graph Kernels. Graph kernels[33,40] implicitly map molecular graphs into vectors in linear feature space without any need to compute molecular descriptors (Figure 10). They have successfully been used in several SAR/QSAR studies by Vert et al.,[34,35]Kashima,[36] Baldi et al.,[33,37] Fröhlich et al.,[38,39] Baskin et al.,[223] Rupp et al.[41], and some others. "Molecule kernels" invented by Mohr et al.[204] and measuring similarity of 3D structures can also be attributed to this category. The principles of the "kernel engineering" were used by Erhan et al.,[224] Faulon et al.,[225] Jacob and Vert,[226] and Bajorath et al.[227] to design combined chemical and biological kernels used to study ligand–biotarget interactions.

The main advantage of graph kernels over kernels built on the descriptors vectors (such as linear, Gaussian, Tanimoto kernels, etc.) is the ability to implicitly handle a huge number of fragment descriptors without any need to compute them. Therefore, the graph kernels provide an alternative solution to the problem of enumeration of all potentially useful fragments. Unlike (sub)graph mining methods that solve this problem by extraction of useful subgraphs, the graph kernels approaches perform their implicit weighting. Consequently, they are potentially capable of revealing some very complex substructures or hidden patterns of local properties, which cannot be found using commonly used descriptors. Nonetheless, in order to take advantage of graph kernels, efficient algorithms for their computation need to be developed.

4.6.4. Models with Structured Outputs. Obtaining a *structured output* of any complexity can be achieved in different ways.[228] Cortes et al.[229] suggested defining separate feature spaces for inputs and outputs, then establishing relationships between the corresponding vectors in the both spaces and, finally, performing predictive calculations in the original output space by solving the preimage problem. Another approach proposed by Joachims et al.[215] is based on joint features for input–output pairs. In the feature space implicitly defined by joint input–output kernels, classification models can be built as in SVM by constructing a separating hyperplane with a maximum margin.[215] In the approach by Geurts et al.,[230,231] the kernels are used to define output feature space, whereas inputs are treated in the framework of the ensemble learning (Section 3) with regression boosting methods based on regression trees. Some alternative approaches dealing with structured outputs have also been considered.[228] Statistical models with structured output are already used in text processing, in bioinformatics for

discerning the alignment of primary structures of biomolecules[232] and inferring biological networks.[233]

The first application of machine learning methods with structured outputs in chemoinformatics, the StructRank algorithm, was reported recently by Rathke et al.[234] This approach is an implementation of the *selective* inference invented by V. Vapnik to solve the following task: "given a training set of bioactive and non-bioactive drugs, select the k representatives with the highest probability of belonging to the bioactive group".[235] This one step algorithm contrasts with the conventional two-step procedure to rank the compounds using the previously developed QSAR regression model (Figure 11). The efficiency of StructRank for ranking the actives within the top k has been demonstrated on three examples including ligands of the benzodiazepine receptor and inhibitors of cyclooxygenase-2 and dihydrofolate reductase.[234]

4.6.5. Functional Data. Functional data represent functional dependence (usually continuous and smooth) on some factors, such as time, spatial coordinates, frequencies, concentrations, etc. For instance, molecules can be described in terms of electronic density or molecular fields. These functions themselves are not used as an *input* of QSAR models but usually are transformed into descriptor vectors that then can be treated by traditional methods of multivariate statistical analysis (e.g., PLS) as in 3D-QSAR methods. This can be achieved using either fixed grid methods (CoMFA,[236] CoMSIA,[237] and GRID[238]) or alignment-free techniques (CoMMA,[239] 3D WHIM,[240] GRIND,[241] and VolSurf[242]). In the Carbo-Dorca approach, a descriptor vector is formed from the quantum similarity indices (in most cases, overlap integral between electron density functions) between the given compound and training set molecules.[243] The common drawback of the above methods is an information loss upon transformation of spatial functions into vectors on the basis of discrete values. This problem has been overcome in the Method of Continuous Molecular Field (MCMF) developed by Baskin's group[162,198] that uses the fields directly as an input of SAR/QSAR models. In this method, comparison of molecular fields of two different molecules is performed by use of a special kernel. The resulting models also have a functional form that allows their simple interpretation in terms of molecular fields.

Although functional endpoints are very common in chemistry (dose−response curves, different types of spectra, kinetic curves, phase diagrams, titration curves), only a few related QSAR studies have been reported. Thus, Halberstam et al.[244] used molecular descriptors in combination with some physical parameters (temperature and pressure) to build models of viscosity and boiling points as a function of these parameters. Oprisiu et al.[245] used special "mixture" descriptors to predict phase diagrams for binary liquid mixtures. In these works, the output curves were represented by ensembles of predicted discrete values; this can certainly effect the curves' smoothness.

Functional Data Analysis (FDA),[246] a novel approach to process functional data, has been reported recently. In contrast to commonly used multivariate data analysis, FDA is designed to operate with functions instead of data vectors. A great advantage of FDA consists in the ability to use the derivatives of functions in order to achieve better predictive performance of models and to gain deeper insight into data. Although no chemoinformatics applications of FDA have been reported so far, we believe that this approach could be beneficial in the modeling of any "chemical" functional endpoints.

5. LITTLE KNOWN FEATURES OF COMMONLY USED MACHINE LEARNING METHODS

In this section, we briefly characterize several popular modern machine learning methods mostly focusing on some useful but still little-known in chemoinformatics functionalities.

5.1. Neural Networks. Artificial neural networks (NN) are one of the most popular machine learning methods in chemoinformatics.[44,46,247] At the same time, most of their applications in chemoinformatics are still confined to DD level targeting to predict property values rather than to make probabilistic predictions, data density approximations, or to build generative models. In principle, neural networks could be applied for these purposes, and therefore, one can expect many interesting developments in this direction. In particular, the ability of neural networks to solve the "inverse problems" by means of "mixed density" neural networks[50] or "deep learning" architectures[248] could be particularly useful in de novo design.

5.2. Classical Linear Regression and Classification. As mentioned in Section 2, the main drawback of "classical" regression and classification methods arises from the lack of proper regularization, which results in severe overfitting phenomena. Nowadays, regularization is widely used in "modern" approaches, such as neural networks or SVM. However, the mere addition of a regularization term to simple and well-known statistical methods makes them competitive with the most advanced ones. Thus, predictive performance of ridge regression (regularized multiple linear regression) and regularized logistic regression is not too far from that of the most modern methods. Taking into account the simplicity of their implementation, one can expect a growing interest to these approaches. It should, however, be taken into account that these methods are not efficient for very large numbers of descriptors.

Another drawback of classical regression and classification methods is related to their strict linear character and nonapplicability to complex data structures. This could be easily overcome using kernels, e.g., in kernel ridge regression and kernel PLS.

5.3. Support Vector Machines. The Support Vector Machines (SVM) approach has recently become one of the most popular machine learning methods in chemoinformatics.[45,52] In the 1990s, it led to a sort of revolution in machine learning by consistently introducing the ideas of regularization, dualism of models, and kernels. This led to an extraordinary popularity for SVM in all areas of informatics, including chemoinformatics. However, the shortcomings of this approach should not be forgotten. Thus, SVM is not a probabilistic method, and it cannot assess the accuracy of its predictions. For classification models, this problem could be solved by means of the Platt[71] and Wu et al.[249] approaches, which heuristically introduce probabilities into predictions.

Another important problem with SVM concerns its limitations in processing large databases. Indeed, in its original form, the method operates with a squared kernel matrix in which size is proportional to square of the number of examples. Several methods to solve this problem have been suggested in ref 250. The simplest of them are based on the dagging[84] ensemble learning approach including ISDA (Iterative Single Data Algorithm)[250,251] and a stochastic variant of the PEGASOs (Primal Estimated sub-GrAdient SOlver for SVM).[252] Another possibility is to use ultrafast linear SVM

approaches[253] or those based on online algorithms (Section 5.5).

5.4. Bayesian Regression and Gaussian Processes. Bayesian linear regression[50] is a kind of multiple linear regression in the Bayesian inference that affords predictive probability distributions for the predictions. The Gaussian processes[59] can be seen as a kernelized variant of Bayesian regression and classification. Despite the clear advantages of these methods over the frequentist ones, they are still rarely used in chemoinformatics.

5.5. Online Machine Learning. All machine learning algorithms can be assigned to one of two categories: "batch" learning and "on-line" learning. In batch algorithms, the whole training set is loaded into CPU memory, whereas the learning process in online algorithms is organized in a stepwise manner. In the latter case, training examples are introduced to the learning system one-by-one, so that only a single training example should reside in the CPU memory. Although batch algorithms are usually more efficient for processing small and medium-size databases, the online algorithms can handle huge databases of any size because they do not need be retrained from scratch for each additional portion of training examples. As an example, one can mention recently developed online SVM (LASVM)[109,110] and online kernel-based learning algorithms for classification, novelty detection, and regression[254] implemented in the *kernlab* package in *R*.

6. QUO VADIS?

6.1. Specificity of Machine Learning Methods in Chemoinformatics. In this section, we discuss specificity of machine-learning methods applied to chemoinformatics tasks. This specificity stems both from the actual nature of chemical objects (molecules and reactions) and from data availability.

Nature of Chemical Objects. Most of machine learning methods deal with fixed-sized data vectors built on binary-, integer-, or real-values features (e.g., fingerprints or molecular descriptors). Drawbacks of this descriptor-based representation of chemical objects are well-known: heuristic nature and incompleteness of descriptor sets; difficult to reconstruct molecules from descriptors; problem to handle multiple species such as tautomers, conformers, ionization states; etc. Therefore, work is needed to develop approaches considering chemical objects as graphs (e.g., graph mining,[95,213,216,219] in particularly, graph kernels,[33−41] special neural network architectures,[27−32] and inductive logical programming[141]), methods accounting for multiple species (e.g., multi-instance learning), as well as generative modeling techniques. The latter should account for synthetic feasibility of theoretically generated molecules.[26,255]

Representativity Problem. Generic machine learning methods assume that the data included in training and test sets are drawn from the same statistical distribution. This can be ensured by applying well-known sampling techniques, such as random sampling. The latter corresponds to random generation of molecular graphs followed by the synthesis and experimental studies of synthesized molecules. This is, however, an unrealistic scenario. In reality, training sets include available retrospective data for chemicals with measured property/activities, whereas test sets are typically composed of compounds planned to be synthesized and screened. Hence, data samples in chemoinformatics, on one hand, can hardly be considered as representative, and on the other hand, the test set in most of cases is rather different from corresponding training set. Therefore, it is not surprising if the predictive performance of the models applied on such test sets is worse than that obtained in a cross-validation technique on the training set. In chemoinformatics, this problem is partially addressed by the concept of the applicability domain. As well, the data set shift[256,257] and domain adaptation[182,258−260] can be used to bridge the gap between the training and test sets. Another solution to this difficult problem is offered by transductive learning methods still little used in chemoinformatics (Section 4.1.1).

Data Heterogeneity and Heteroscedasticity. In many cases, a training set is compiled from experimental chemical or biological data taken from different sources (heterogeneity) and for which experimental error could vary from one data subset to another one (heteroscedasticity). This could be a problem for generic statistical machine learning methods that do not allow for data heterogeneity and heteroscedasticity. In particular, the presence of heteroscedasticity can invalidate statistical tests of significance and standard statistical modeling techniques that assume that errors are uncorrelated and normally distributed with the same variance. So, heterogenic and heteroscedastic data require special treatment.[261,262] As an example, Gaussian processes can be equipped with different noise levels.[263] Recently, Rabu et al.[264] introduced a probabilistic structure miner, which effectively mines structures from noisy data, where some molecules are labeled with their probability of being active.

Unbalanced Data Sets Problem. Quite often, chemical databases are highly unbalanced with respect to active and inactive compounds (the latter are much better represented). Some machine learning methods, like SVM, show poor performance on those unbalanced data sets. The multiple resampling technique[265,266] specifically designed to correct this problem is not always efficient. On the other hand, Relevance Voter (IRV)[188,267] and one-class classification[160,161] approaches usually demonstrate good predictive performance, and therefore, they could be recommended as a reasonable solution for the unbalanced data sets problem.

It should however be noted that in virtual screening, one may target enrichment by actives of the first portion of retrieved data rather than an overall success of the model. In that case, several characteristics have been suggested in order to measure the "early recognition" performance: robust initial enhancement (RIE),[268] Boltzmann-enhanced discrimination of receiver operating characteristic (BEDROC),[269] sum of log ranks (SLR), CROC,[270] etc.

Uncertainty of Labeling for Inactives. In some databases (e.g., Database of Useful Decoys, DUD), some compounds are designated as inactive, although no experimental proofs of that are available. This may seriously impact the performance of models built on these data sets with any binary classification machine learning method. This problem could, however, be treated within either the one-class classification approach[160,177,271] considering actives only or semisupervised methods like PU learning[272] and semisupervised novelty detection[273] considering both actives and unlabeled compounds; the latter are those for which experimental data are not available (e.g., DUD decoys).

Interpretability of Models. The issue of interpretability has always been of prime importance in chemoinformatics[274] and constitutes one of its major distinctions from generic machine learning. Classical Hansch−Fujita QSAR analysis[275] and Cramer's CoMFA method[236] are largely praised for good interpretability of models, while the models involving

topological indices are often criticized for their lack of interpretability. Some efforts have been made to interpret QSAR models obtained with "black box" machine learning methods, such as neural networks.[276−278] Thus, Baskin et al[276] suggested calculation of mean values of the first and second partial derivatives of modeled property with respect to molecular descriptors. This allows one to assess their relative importance and, depending on descriptors used, to give rational interpretation of the models. The Influence Relevance Voter (IRV) approach of Baldi et al.[188,267] interprets predictions by examining the active compounds that are the most similar to the query molecule. Interesting approaches to interpret classification models have recently been developed in Müller's group.[279,280] In particular, they suggested the use of local gradients indicating the motion of the data point that may lead to change of its label.[279] Another method of visual interpretation of kernel-based prediction models[280] is based on the detection of training examples contributing most to the query molecule. According to the authors of;[280] this approach "helps to assess the domain of applicability of a model, to judge the reliability of a prediction, and to determine relevant molecular features".

6.2. Guide to Appropriate Machine Learning Methods and Software Tools. Table 1 relates common chemoinformatics tasks with previously discussed machine learning methods and related freely available software. Here, these relations are briefly discussed in the context of the properties of the data used for the training of the models, i.e., their amount (small or large data sets), distribution in the chemical space, types (descriptors or graphs), and complexity (Figure 12).

Amount of Data. Methods listed in entry 1 of Table 1 could be efficiently applied to increase the performance of models built on small and diverse data sets. Different strategies can be used, such as ensemble learning, semisupervised and transductive learning,[96] inductive knowledge transfer,[281] and L_1- and L_2- regularized methods. Bootstrap[68] and additional techniques could be used to assess the accuracy of predictions (entry 2). The above techniques are less useful for the large databases, for which classical statistics approaches are sufficient to build predictive models. On the other hand, a huge amount of data creates many technical problems that could be solved by the online techniques and some other methods grouped in entry 3.

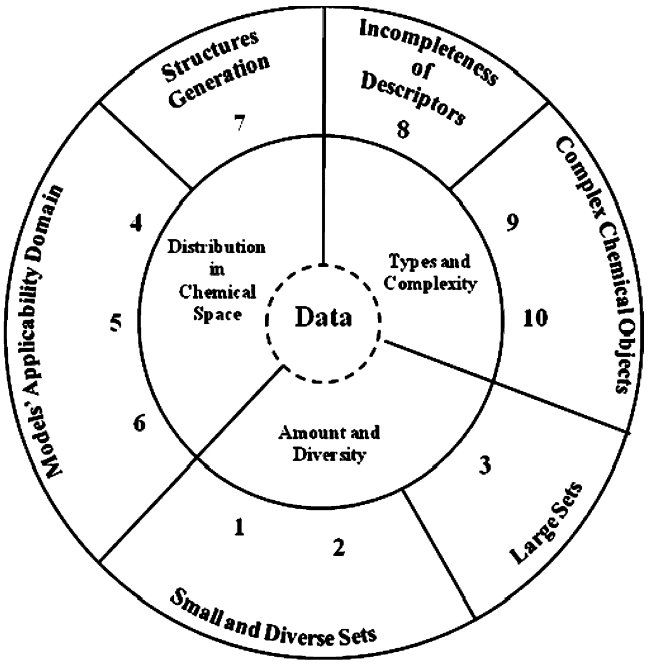

Figure 12. Different features of the data (inner circle) and their links to main chemoinformatics tasks (outer circle). The numbers enumerate groups of machine learning methods corresponding to the entries in Table 1.

Distribution of Data in Chemical Space. As follows from statistical learning theory,[6] conventional SAR/QSAR models lead to reliable predictions if both training and test sets belong to the same single data domain. The part of the chemical space occupied by the training set is traditionally associated with the applicability domain of the models built on this training set. In this context, novelty detection,[156,157,159] one-class classification,[177,271,282,283] and data domain description[158] approaches (entry 4) help to delineate this area. Note that the novelty detection approach can be efficiently used for virtual screening assuming that the target class in one-class classification includes only actives.

If training and test sets belong to different data domains, the data set shift[256,257] and domain adaptation[182,258−260] approach as well as other related methods listed in entry 5 should be used. This could allow one to obtain reliable predictions where there is a low density of training data.

Nowadays, more and more projects are concerned with the generation of new experimental data for creation of an "optimal" training set for QSAR modeling. The active learning approach[98,284−289] (entry 6) is an efficient way to suggest the most suitable candidates.

Inverse QSAR[19−26] leading to generation of new chemical structures possessing desirable properties is a dream of any chemoinformatician. Recent developments listed in entry 7 are well suited to treat this problem.

Types and Complexity of the Data. Most of machine learning methods are based on fixed sized vectors, and therefore, in conventional SAR/QSAR studies, chemical structures are represented as an ensemble of molecular descriptors. This causes several obvious problems. Thus, descriptors can be easily generated from a molecular graph, but reverse graph reconstruction from descriptors is an extremely difficult task. Modern machine learning methods offer a unique opportunity to work directly with the connectivity matrix (entry 8). One can also mention different graph mining approaches,[95,213,216,219] the use of graph kernels,[33,34,37,40] the Inductive Logic Programming (ILP),[141,142] and its application to chemoinformatics.[145−150,152]

A chemical structure is a complex object which, sometimes, must be represented by several graphs (tautomers) or 3D structures (conformers). The multi-instance learning approach (entry 9) could be very helpful in taking this into account in structure−property modeling.

Finally, entry 10 combines functional data analysis methods that are particularly useful in modeling systems with functional input (e.g., continuous molecular fields[198]) or output (e.g., phase diagrams or dose−response curves).

The analysis of recent developments clearly demonstrates the following trends in the machine learning area: (1) gradual transition to Bayesian inference that can be achieved either by application of the advanced Bayesian learning methods, e.g., Gaussian processes or Bayesian neural networks, or by further developments of ensemble modeling (although at least in the nearest future both Bayesian and frequentist approaches will

likely coexist[290]), (2) the use of regularized versions of commonly used statistical methods (e.g., ridge and lasso regression, regularized logistic regression, etc.), (3) transition toward kernel-based methods, (4) the use of predictive distributions instead of point predictions, and (5) application of generative models to de novo design.

7. CONCLUSIONS

In this article, we have discussed the most promising ideas, achievements, and approaches in machine learning that could potentially be useful in chemoinformatics in order to improve the accuracy of predictions and efficiency of virtual screening. Most of these methods are implemented in freely available software (Table 1) but still are little known in the chemoinformatics community. We hope that some recommendations given here would enrich the "modeling kit" used in computer-aided molecular design.

AUTHOR INFORMATION

Corresponding Author

*E-mail: varnek@unistra.fr.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors thank Prof. A. Tropsha, Dr. G. Marcou, and Dr. D. Horvath for stimulating discussion and Prof. J. Harrowfield for his help and advice. I.B. thanks GDRI SupraChem and the program "ARCUS-Alsace-Russia/Ukraine" for support.

REFERENCES

(1) Bellman, R. E. *Dynamic Programming*; Princeton University Press: Princeton, NJ, 1957.
(2) Cherkassky, V.; Mulier, F. *Learning from Data: Concept, Theory and Methods*, 2nd ed.; John Wiley & Sons, Inc.: Hoboken, NJ, 2007.
(3) Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.* **1982**, *79* (8), 2554−2558.
(4) Rumelhart, D. E.; McClelland, J. L. *Parallel Distributed Processing*; MIT Press: Cambridge, MA, 1986; Vol. *1,2*.
(5) Vapnik, V. N. *The Nature of Statistical Learning Theory*; Springer: Berlin, 1995.
(6) Vapnik, V. *Statistical Learning Theory*; Wiley-Interscience: New York, 1998.
(7) Valiant, L. G. A Theory of the Learnable. *Commun. ACM* **1984**, *27* (11), 1134−1142.
(8) Rissanen, J. A universal prior for the integers and estimation by minimum description length. *Ann. Stat.* **1983**, *11* (2), 416−431.
(9) Gasteiger, J. Chemoinformatics: A new field with a long tradition. *Anal. Bioanal. Chem.* **2006**, *384* (1), 57−64.
(10) Gasteiger, J.; Engel, T. *Chemoinformatics: A Textbook*; Wiley-VCH: Weinheim, 2003.
(11) Gasteiger, J. *Handbook of Chemoinformatics: From Data to Knowledge*; Wiley-VCH: Weinheim, 2003.
(12) Engel, T. Basic overview of chemoinformatics. *J. Chem. Inf. Model.* **2006**, *46* (6), 2267−2277.
(13) Varnek, A.; Baskin, I. I. Chemoinformatics as a theoretical chemistry discipline. *Mol. Inf.* **2011**, *30* (1), 20−32.
(14) Brown, N. Chemoinformatics: An introduction for computer scientists. *ACM Comput. Surv.* **2009**, *41* (2), 1−38.
(15) Maggiora, G. M. On outliers and activity cliffs why QSAR often disappoints. *J. Chem. Inf. Model.* **2006**, *46* (4), 1535−1535.
(16) Dietterich, T. G.; Lathrop, R. H.; Lozano-Pérez, T. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.* **1997**, *89* (1−2), 31−71.
(17) Baskin, I. I.; Gordeeva, E. V.; Devdariani, R. O.; Zefirov, N. S.; Palyulin, V. A.; Stankevich, M. I. Solving the inverse problem of structure−property relations for the case of topological indexes. *Dokl. Akad. Nauk SSSR* **1989**, *307* (3), 613−17.
(18) Gordeeva, E. V.; Molchanova, M. S.; Zefirov, N. S. General methodology and computer program for the exhaustive restoring of chemical structures by molecular connectivity indexes. Solution of the inverse problem in QSAR/QSPR. *Tetrahedron Comput. Methodol.* **1990**, *3* (6), 389−415.
(19) Skvortsova, M. I.; Baskin, I. I.; Slovokhotova, O. L.; Palyulin, V. A.; Zefirov, N. S. Inverse problem in QSAR/QSPR studies for the case of topological indexes characterizing molecular shape (Kier indices). *J. Chem. Inf. Comput. Sci.* **1993**, *33* (4), 630−634.
(20) Kier, L. B.; Hall, L. H.; Frazer, J. W. Design of molecules from quantitative structure−activity relationship models. 1. Information transfer between path and vertex degree counts. *J. Chem. Inf. Comput. Sci.* **1993**, *33* (1), 143−147.
(21) Skvortsova, M. I.; Baskin, I. I.; Palyulin, V. A.; Slovokhotova, O. L.; Zefirov, N. S. Structural design. Inverse Problems for Topological Indices in QSAR/QSPR Studies. In *AIP Conf. Proc. 330. E.C.C.C.1 Comput. Chem. F.E.C.S. Conf.*, Nancy, France, Bernardi, F., Rivail, J.-L., Eds. AIP Press: Woodbury, NY, 1995; pp 486−499.
(22) Rücker, C.; Meringer, M.; Kerber, A. QSPR using MOLGEN-QSPR: The example of haloalkane boiling points. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (6), 2070−2076.
(23) Churchwell, C. J.; Rintoul, M. D.; Martin, S.; Visco, D. P., Jr.; Kotu, A.; Larson, R. S.; Sillerud, L. O.; Brown, D. C.; Faulon, J. L. The signature molecular descriptor. 3. Inverse-quantitative structure−activity relationship of ICAM-1 inhibitory peptides. *J. Mol. Graphics Modell.* **2004**, *22* (4), 263−73.
(24) Wong, W.; Burkowski, F. A constructive approach for discovering new drug leads: Using a kernel methodology for the inverse-QSAR problem. *J. Cheminf.* **2009**, *1* (1), 4.
(25) Miyao, T.; Arakawa, M.; Funatsu, K. Exhaustive structure generation for inverse-QSPR/QSAR. *Mol. Inf.* **2010**, *29* (1−2), 111−125.
(26) White, D.; Wilson, R. C. Generative models for chemical structures. *J. Chem. Inf. Model.* **2010**, *50* (7), 1257−1274.
(27) Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. A neural device for searching direct correlations between structures and properties of chemical compounds. *J. Chem. Inf. Comput. Sci.* **1997**, *37* (4), 715−721.
(28) Kireev, D. B. ChemNet: A novel neural network based method for graph/property mapping. *J. Chem. Inf. Comput. Sci.* **1995**, *35* (2), 175−80.
(29) Ivanciuc, O. Molecular structure encoding into artificial neural networks topology. *Rom. Chem. Q. Rev.* **2001**, *8*, 197−220.
(30) Bianucci, A. M.; Micheli, A.; Sperduti, A.; Starita, A. Application of cascade correlation networks for structures to chemistry. *Appl. Intell.* **2000**, *12* (1−2), 117−146.
(31) Micheli, A.; Sperduti, A.; Starita, A.; Bianucci, A. M. Analysis of the internal representations developed by neural networks for structures applied to quantitative structure−activity relationship studies of benzodiazepines. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (1), 202−218.
(32) Goulon, A.; Picot, T.; Duprat, A.; Dreyfus, G. Predicting activities without computing descriptors: Graph machines for QSAR. *SAR QSAR Environ. Res.* **2007**, *18* (1−2), 141−153.
(33) Ralaivola, L.; Swamidass, S. J.; Saigo, H.; Baldi, P. Graph kernels for chemical informatics. *Neural Netw.* **2005**, *18* (8), 1093−1110.
(34) Mahé, P.; Ueda, N.; Akutsu, T.; Perret, J.-L.; Vert, J.-P. Graph kernels for molecular structure−activity relationship analysis with support vector machines. *J. Chem. Inf. Model.* **2005**, *45* (4), 939−951.
(35) Mahe, P.; Ralaivola, L.; Stoven, V.; Vert, J.-P. The pharmacophore kernel for virtual screening with support vector machines. *J. Chem. Inf. Model.* **2006**, *46* (5), 2003−2014.
(36) Kashima, H.; Tsuda, K.; Inokuchi, A. Marginalized Kernels Between Labeled Graphs. In *Proceedings, Twentieth International*

Conference on Machine Learning, AAAI Press: Washington, DC, 2003; Vol. *1*, pp 321−328.
(37) Swamidass, S. J.; Chen, J.; Phung, P.; Ralaivola, L.; Baldi, P. Kernels for small molecules and the prediction of mutagenicity, toxicity and anti-cancer activity. *Bioinformatics* **2005**, *21*, I359−I368.
(38) Fröhlich, H.; Wegner, J.; Sieker, F.; Zell, Z. Kernel functions for attributed molecular graphs − A new similarity based approach to ADME prediction inclassification and regression. *QSAR Comb. Sci.* **2006**, *25* (4), 317−326.
(39) Fröhlich, H. Optimal Assignment Kernels for ADME in Silico Prediction. In *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques*; Lodhi, H., Yamanishi, Y., Eds.; IGI Global: Hershey, PA, 2011; pp 16−34.
(40) Rupp, M.; Schneider, G. Graph kernels for molecular similarity. *Mol. Inf.* **2010**, *29* (4), 266−273.
(41) Rupp, M.; Körner, R.; Tetko, I. V. Predicting the pKa of small molecules. *Comb. Chem. High T. Scr.* **2011**, *14* (5), 307−327.
(42) Vishwanathan, S. V. N.; Schraudolph, N. N.; Kondor, R.; Borgwardt, K. M. Graph kernels. *J. Mach. Learn. Res.* **2010**, *11*, 1201−1242.
(43) Varmuza, K., Multivariate Data Analysis in Chemistry. In *Handbook of Chemoinformatics. From Data to Knowledge*; Gasteiger, J., Ed.; Wiley-VCH: Weinheim, 2003; pp 1098−1133.
(44) Zupan, J.; Gasteiger, J. *Neural Networks in Chemistry*; Wiley-VCH: Weinheim, 1999.
(45) Ivanciuc, O. Applications of Support Vector Machines in Chemistry. In *Reviews in Computational Chemistry*; Lipkowitz, K. B., Cundari, T. R., Eds.; John Wiley & Sons, Inc.: Hoboken, NJ, 2007; Vol. *23*, pp 291−400.
(46) Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Neural networks in building QSAR models. *Methods Mol. Biol.* **2008**, *458*, 137−158.
(47) Halberstam, N. M.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Neural networks as a method for elucidating structure−property relationships for organic compounds. *Russ. Chem. Rev.* **2003**, *72* (7), 629−649.
(48) Geppert, H.; Vogt, M.; Bajorath, J. Current trends in ligand-based virtual screening: Molecular representations, data mining methods, new application areas, and performance evaluation. *J. Chem. Inf. Mod* **2010**, *50* (2), 205−216.
(49) Eckert, H.; Bajorath, J. Molecular similarity analysis in virtual screening: Foundations, limitations and novel approaches. *Drug Discovery Today* **2007**, *12* (5−6), 225−33.
(50) Bishop, C. M. *Pattern Recognition and Machine Learning*; Springer: New York, 2006.
(51) Upton, G.; Cook, I. *Oxford Dictionary of Statistics*, 2nd ed. (revised); Oxford University Press: Oxford, 2008.
(52) Chen, N.; Lu, W.; Yang, J.; Li, G. *Support Vector Machine in Chemistry*; World Scientific: Singapore, 2004.
(53) Farkas, O.; Heberger, K. Comparison of ridge regression, partial least-squares, pairwise correlation, forward- and best subset selection methods for prediction of retention indices for aliphatic alcohols. *J. Chem. Inf. Model.* **2005**, *45* (2), 339−346.
(54) Hawkins, D. M.; Basak, S. C.; Shi, X. QSAR with few compounds and many features. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (3), 663−670.
(55) Merkwirth, C.; Mauser, H.; Schulz-Gasch, T.; Roche, O.; Stahl, M.; Lengauer, T. Ensemble methods for classification in chem-informatics. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (6), 1971−1978.
(56) Spycher, S.; Pellegrini, E.; Gasteiger, J. Use of structure descriptors to discriminate between modes of toxic action of phenols. *J. Chem. Inf. Model.* **2005**, *45* (1), 200−8.
(57) Jaynes, E. T. *Probability Theory. The Logic of Science*; Cambridge University Press: Cambridge, 2003.
(58) Jaynes, E. T. Prior probabilities. *IEEE Trans. Syst. Sci. Cyb.* **1968**, *4* (3), 227−241.
(59) Rasmussen, C. E.; Williams, C. K. I. *Gaussian Processes in Machine Learning*; The MIT Press: Cambridge, MA, 2006.
(60) Bishop, C. *Neural Networks for Pattern Recognition*; Oxford University Press: Oxford, 1995.
(61) Burden, F. R.; Winkler, D. A. Robust QSAR models using Bayesian regularized neural networks. *J. Med. Chem.* **1999**, *42* (16), 3183−3187.
(62) Bruneau, P. Search for predictive generic model of aqueous solubility using Bayesian neural nets. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (6), 1605−1616.
(63) Burden, F. R.; Ford, M. G.; Whitley, D. C.; Winkler, D. A. Use of automatic relevance determination in QSAR studies using Bayesian neural networks. *J. Chem. Inf. Comput. Sci.* **2000**, *40* (6), 1423−1430.
(64) Klocker, J.; Wailzer, B.; Buchbauer, G.; Wolschann, P. Bayesian neural networks for aroma classification. *J. Chem. Inf. Comput. Sci.* **2002**, *42* (6), 1443−1449.
(65) Burden, F. R. Quantitative structure−activity relationship studies using Gaussian processes. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (3), 830−835.
(66) Tino, P.; Nabney, I. T.; Williams, B. S.; Losel, J.; Sun, Y. Nonlinear prediction of quantitative structure−activity relationships. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (5), 1647−1653.
(67) Obrezanova, O.; Csanyi, G.; Gola, J. M. R.; Segall, M. D. Gaussian processes: A method for automatic QSAR modeling of ADME properties. *J. Chem. Inf. Model.* **2007**, *47* (5), 1847−1857.
(68) Efron, B. Bootstrap methods: Another look at the jackknife. *Ann. Stat.* **1979**, *7*, 1−26.
(69) Tetko, I. V.; Bruneau, P.; Mewes, H.-W.; Rohrer, D. C.; Poda, G. I. Can we estimate the accuracy of ADMET predictions? *Drug Discovery Today* **2006**, *11* (15/16), 700−707.
(70) Sushko, I.; Novotarskyi, S.; Korner, R.; Pandey, A. K.; Cherkasov, A.; Li, J.; Gramatica, P.; Hansen, K.; Schroeter, T.; Muller, K. R.; Xi, L.; Liu, H.; Yao, X.; Oberg, T.; Hormozdiari, F.; Dao, P.; Sahinalp, C.; Todeschini, R.; Polishchuk, P.; Artemenko, A.; Kuz'min, V.; Martin, T. M.; Young, D. M.; Fourches, D.; Muratov, E.; Tropsha, A.; Baskin, I.; Horvath, D.; Marcou, G.; Muller, C.; Varnek, A.; Prokopenko, V. V.; Tetko, I. V. Applicability domains for classification problems: Benchmarking of distance to models for Ames mutagenicity set. *J. Chem. Inf. Model.* **2010**, *50* (12), 2094−2111.
(71) Platt, J. Probabilities for SV Machines. In *Advances in Large Margin Classifiers*; Smola, A. J., Bartlett, P. L., Schölkopf, B., Schuurmans, D., Eds.; MIT Press: Cambridge, MA, 2000; pp 61−74.
(72) Kwok, J. T. Y.; Tsang, I. W. H. The pre-image problem in kernel methods. *IEEE Trans. Neural Netw.* **2004**, *15* (6), 1517−1525.
(73) Tetko, I. V. Neural network studies. 4. Introduction to associative neural networks. *J. Chem. Inf. Comput. Sci.* **2002**, *42* (3), 717−728.
(74) Tetko, I. V.; Tanchuk, V.; Chentsova, N. P.; Antonenko, S. V.; Poda, G. I.; Kukhar, V. P.; Luik, A. I. HIV-1 reverse transcriptase inhibitor design using artificial neural networks. *J. Med. Chem.* **1994**, *37* (16), 2520−6.
(75) Artemenko, N. V.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Artificial neural network and fragmental approach in prediction of physicochemical properties of organic compounds. *Russ. Chem. Bull.* **2003**, *52* (1), 20−29.
(76) Zhokhova, N. I.; Baskin, I. I.; Palyulin, V. A.; Zefirov, A. N.; Zefirov, N. S. Fragmental descriptors with labeled atoms and their application in QSAR/QSPR studies. *Dokl. Chem.* **2007**, *417* (2), 282−284.
(77) Zhu, H.; Tropsha, A.; Fourches, D.; Varnek, A.; Papa, E.; Gramatical, P.; Öberg, T.; Dao, P.; Cherkasov, A.; Tetko, I. V. Combinatorial QSAR modeling of chemical toxicants tested against *Tetrahymena pyriformis*. *J. Chem. Inf. Model.* **2008**, *48* (4), 766−784.
(78) Varnek, A.; Fourches, D.; Horvath, D.; Klimchuk, O.; Gaudin, C.; Vayer, P.; Solov'ev, V.; Hoonakker, F.; Tetko, I. V.; Marcou, G. ISIDA: Platform for virtual screening based on fragment and pharmacophoric descriptors. *Curr. Comput.-Aided Drug Des.* **2008**, *4* (3), 191−198.
(79) Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24* (2), 123−140.
(80) Breiman, L. Random forests. *Mach. Learn.* **2001**, *45* (1), 5−32.

(81) Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J. C.; Sheridan, R. P.; Feuston, B. P. Random forest: A classification and regression tool for compound classification and QSAR modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (6), 1947−1958.
(82) Guha, R.; Jurs, P. C. Development of linear, ensemble, and nonlinear models for the prediction and interpretation of the biological activity of a set of PDGFR inhibitors. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (6), 2179−2189.
(83) Li, S.; Fedorowicz, A.; Singh, H.; Soderholm, S. C. Application of the random forest method in studies of local lymph node assay based skin sensitization data. *J. Chem. Inf. Model.* **2005**, *45* (4), 952−964.
(84) Ting, K. M.; Witten, I. H. Stacking Bagged and Dagged Models. In *Fourteenth International Conference on Machine Learning*; Fisher, D. H., Ed.; Morgan Kaufmann Publishers: San Francisco, CA, 1997; pp 367−375.
(85) Ho, T. K. The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal.* **1998**, *20* (8), 832−844.
(86) Wolpert, D. H. Stacked generalization. *Neural Netw.* **1992**, *5* (2), 241−259.
(87) Breiman, L. Stacked regressions. *Mach. Learn.* **1996**, *24* (1), 49−64.
(88) Freund, Y.; Schapire, R. E., Experiments with a New Boosting Algorithm. In *Thirteenth International Conference on Machine Learning*; Morgan Kaufmann: San Francisco, 1996; pp 148−156.
(89) Wegner, J. K.; Froehlich, H.; Zell, A. Feature selection for descriptor based classification models. 2. Human intestinal absorption (HIA). *J. Chem. Inf. Comput. Sci.* **2004**, *44* (3), 931−939.
(90) Friedman, J. H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29* (5), 1189−1232.
(91) Friedman, J. H. Stochastic gradient boosting. *Comput. Stat. Data An.* **2002**, *38* (4), 367−378.
(92) Svetnik, V.; Wang, T.; Tong, C.; Liaw, A.; Sheridan, R. P.; Song, Q. Boosting: An ensemble learning tool for compound classification and QSAR modeling. *J. Chem. Inf. Model.* **2005**, *45* (3), 786−799.
(93) Demiriz, A.; Bennett, K. P.; Shawe-Taylor, J. Linear Programming boosting via column generation. *Mach. Learn.* **2002**, *46* (1−3), 225−254.
(94) Saigo, H.; Nowozin, S.; Kadowaki, T.; Kudo, T.; Tsuda, K. GBoost: A mathematical programming approach to graph classification and regression. *Mach. Learn.* **2009**, *75* (1), 69−89.
(95) Saigo, H.; Kadowaki, T.; Tsuda, K., A Linear Programming Approach for Molecular QSAR Analysis. In *International Workshop on Mining and Learning with Graphs 2006*; Gaertner, T., Garriga, G. C., Meinl, T., Eds.; Berlin, 2006; pp 85−96.
(96) Chapelle, O.; Schoelkopf, B.; Zien, A. *Semi-Supervised Learning*; The MIT Press: Cambridge, MA, 2006.
(97) Joachims, T. Transductive Inference for Text Classification Using Support Vector Machines. In *International conference on Machine Learning (ICML)*; Kaufmann, M., Ed.; Bled, Slovenia, 1999; pp 200−209.
(98) Cohn, D. A.; Ghahramani, Z.; Jordan, M. I. Active learning with statistical models. *J. Artif. Intell. Res.* **1996**, *4*, 129−145.
(99) Schein, A. I.; Ungar, L. H. Active learning for logistic regression: An evaluation. *Mach. Learn.* **2007**, *68* (3), 235−265.
(100) Wang, Z.; Chen, S.; Chen, Z. An active learning approach for neural network ensemble. *Jisuanji Yanjiu yu Fazhan/Computer Research and Development* **2005**, *42* (3), 375−380.
(101) Danziger, S. A.; Zeng, J.; Wang, Y.; Brachmann, R. K.; Lathrop, R. H. Choosing where to look next in a mutation sequence space: Active Learning of informative p53 cancer rescue mutants. *Bioinformatics* **2007**, *23* (13), i104−i114.
(102) Fujiwara, Y.; Yamashita, Y.; Osoda, T.; Asogawa, M.; Fukushima, C.; Asao, M.; Shimadzu, H.; Nakao, K.; Shimizu, R. Virtual screening system for finding structurally diverse hits by active learning. *J. Chem. Inf. Model.* **2008**, *48* (4), 930−940.
(103) Fukumizu, K. Statistical active learning in multilayer perceptrons. *IEEE Trans. Neural Netw.* **2000**, *11* (1), 17−26.
(104) Vijayakumart, S.; Ogawa, H. Improving generalization ability through active learning. *IEICE Trans. Inf. Syst.* **1999**, *E82-D* (2), 480−487.
(105) Warmuth, M. K.; Liao, J.; Rätsch, G.; Mathieson, M.; Putta, S.; Lemmen, C. Active learning with support vector machines in the drug discovery process. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (2), 667−673.
(106) Zomer, S.; Del Nogal Sánchez, M.; Brereton, R. G.; Pérez Pavón, J. L. Active learning support vector machines for optimal sample selection in classification. *J. Chemom.* **2004**, *18* (6), 294−305.
(107) Cheng, J.; Wang, K. Active learning for image retrieval with Co-SVM. *Pattern Recogn.* **2007**, *40* (1), 330−334.
(108) Gu, P.; Zhu, Q.; Zhang, C. A novel active learning approach for SVM in the presence of multi-views. *J. Chem. Inf. Comput. Sci.* **2010**, *7* (2), 317−324.
(109) Bordes, A.; Ertekin, S.; Weston, J.; Bottou, L. Fast kernel classifiers with online and active learning. *J. Mach. Learn. Res.* **2005**, *6*, 1579−1619.
(110) Glasmachers, T.; Igel, C. Second-order SMO improves SVM online and active learning. *Neural Comput.* **2008**, *20* (2), 374−382.
(111) Iyengar, V. S.; Apte, C.; Zhang, T., Active Learning Using Adaptive Resampling. In *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; Ramakrishnan, R., Stolfo, S., Bayardo, R., Parsa, I., Ramakrishnan, R., Stolfo, S., Bayardo, R., Parsa, I., Eds.; ACM: Boston, MA, 2000; pp 91−98.
(112) Xu, J.; Shi, P. Active learning based on maximizing information gain for content-based image retrieval. *J. Southeast Univ. (Engl. Ed.)* **2004**, *20* (4), 431−435.
(113) Kim, H. J.; Kim, J. U. Combining active learning and boosting for Naïve Bayes text classifiers. *Lect. Notes Comput. Sci.* **2004**, *3129*, 519−527.
(114) Yang, L.; Hanneke, S.; Carbonell, J. Bayesian active learning using arbitrary binary valued queries. *Lect. Notes Comput. Sci.* **2010**, *6331*, 50−58.
(115) Henrich, F. F.; Obermayer, K. Active learning by spherical subdivision. *J. Mach. Learn. Res.* **2008**, *9*, 105−130.
(116) Hopfinger, A. J.; Wang, S.; Tokarski, J. S.; Jin, B.; Albuquerque, M.; Madhav, P. J.; Duraiswami, C. Construction of 3D-QSAR models using the 4D-QSAR analysis formalism. *J. Am. Chem. Soc.* **1997**, *119* (43), 10509−10524.
(117) Müller, G. nD QSAR: A medicinal chemist's point of view. *Quant. Struct.-Act. Relat.* **2002**, *21* (4), 391−396.
(118) Albuquerque, M. G.; Hopfinger, A. J.; Barreiro, E. J.; De Alencastro, R. B. Four-dimensional quantitative structure: Activity relationship analysis of a series of interphenylene 7-oxabicycloheptane oxazole thromboxane A2 receptor antagonists. *J. Chem. Inf. Comput. Sci.* **1998**, *38* (5), 925−938.
(119) Klein, C. D. P.; Hopfinger, A. J. Pharmacological activity and membrane interactions of antiarrhythmics: 4D-QSAR/QSPR analysis. *Pharm. Res.* **1998**, *15* (2), 303−311.
(120) Hopfinger, A. J.; Reaka, A.; Venkatarangan, P.; Duca, J. S.; Wang, S. Construction of a virtual high throughput screen by 4D-QSAR analysis: Application to a combinatorial library of glucose inhibitors of glycogen phosphorylase b. *J. Chem. Inf. Comput. Sci.* **1999**, *39* (6), 1151−1160.
(121) Duca, J. S.; Hopfinger, A. J. Estimation of molecular similarity based on 4D-QSAR analysis: Formalism and validation. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (3−6), 1367−1387.
(122) Ravi, M.; Hopfinger, A. J.; Hormann, R. E.; Dinan, L. 4D-QSAR analysis of a set of ecdysteroids and a comparison to CoMFA modeling. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (6), 1587−1604.
(123) Santos-Filho, O. A.; Hopfinger, A. J. A search for sources of drug resistance by the 4D-QSAR analysis of a set of antimalarial dihydrofolate reductase inhibitors. *J. Comput.-Aided Mol. Des.* **2001**, *15* (1), 1−12.
(124) Vedani, A.; Dobler, M. Multidimensional QSAR: Moving from three- to five-dimensional concepts. *Quant. Struct.-Act. Relat.* **2002**, *21* (4), 382−390.
(125) Vedani, A.; Dobler, M. 5D-QSAR: the key for simulating induced fit? *J. Med. Chem.* **2002**, *45* (11), 2139−49.

(126) Vedani, A.; Dobler, M.; Lill, M. A. Combining protein modeling and 6D-QSAR. Simulating the binding of structurally diverse ligands to the estrogen receptor. *J. Med. Chem.* **2005**, *48* (11), 3700−3703.
(127) Bonachera, F.; Parent, B.; Barbosa, F.; Froloff, N.; Horvath, D. Fuzzy tricentric pharmacophore fingerprints. 1. Topological fuzzy pharmacophore triplets and adapted molecular similarity scoring schemes. *J. Chem. Inf. Model.* **2006**, *46* (6), 2457−2477.
(128) Ruggiu, F.; Marcou, G.; Varnek, A.; Horvath, D. I. Property-labelled fragment descriptors. *Mol. Inf.* **2010**, *29* (12), 855−868.
(129) *JChem*, version 5.9; ChemAxon: Budapest, Hungary, 2012. http://www.chemaxon.com/jchem/intro/index.html (accessed April 04, 2012).
(130) Horvath, D.; Bonachera, F.; Solov'ev, V.; Gaudin, C.; Varnek, A. Stochastic versus stepwise strategies for quantitative structure−activity relationship generation: How much effort may the mining for successful QSAR models take? *J. Chem. Inf. Mod* **2007**, *47* (3), 927−939.
(131) Lukacova, V.; Balaz, S. Multimode ligand binding in receptor site modeling: Implementation in CoMFA. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (6), 2093−2105.
(132) Zhang, Y.; Lukacova, V.; Bartus, V.; Balaz, S. Structural determinants of binding of aromates to extracellular matrix: A multi-species multi-mode CoMFA study. *Chem. Res. Toxicol.* **2007**, *20* (1), 11−19.
(133) Zhang, Y.; Lukacova, V.; Bartus, V.; Nie, X.; Sun, G.; Manivannan, E.; Ghorpade, S. R.; Jin, X.; Manyem, S.; Sibi, M. P.; Cook, G. R.; Balaz, S. Binding of matrix metalloproteinase inhibitors to extracellular matrix: 3D-QSAR analysis. *Chem. Biol. Drug. Des.* **2008**, *72* (4), 237−248.
(134) Zhou, Z. H. Multi-instance learning from supervised view. *J. Comput. Sci. Tech.* **2006**, *21* (5), 800−809.
(135) Dooly, D. R.; Zhang, Q.; Goldman, S. A.; Amar, R. A. Multiple-instance learning of real-valued data. *J. Mach. Learn. Res.* **2003**, *3* (4−5), 651−678.
(136) Maron, O.; Lozano-Perez, T. A Framework for Multiple-Instance Learning. In *Advances in Neural Information Processing Systems 10*; Jordan, M. I., Kearns, M. J., Solla, S. A., Eds.; MIT Press: Cambridge, 1998; Vol. *10*, pp 570−576.
(137) Andrews, S.; Hofmann, T.; Tsochantaridis, I. Multiple Instance Learning with Generalized Support Vector Machines. In *Eighteenth National Conference on Artificial Intelligence*; MIT Press: Cambridge, MA, 2002; pp 943−944.
(138) Rencher, A. C.; Schaalje, G. B. *Linear Models in Statistics*; John Wiley & Sons: Hoboken, NJ, 2008.
(139) Alpaydin, E. *Introduction to Machine Learning*, 2nd ed.; The MIT Press: Cambridge, MA, 2010.
(140) Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer: New York, 2001.
(141) Muggleton, S. H.; De Raedt, L. Inductive logic programming: Theory and methods. *J. Logic Program.* **1994**, *19* (20), 629−679.
(142) De Raedt, L.; Frasconi, P.; Kersting, K.; Muggleton, S. *Probabilistic Inductive Logoc Programming. Theory and Applications*;- Springer: Berlin, Heidelberg, 2008.
(143) Kersting, K. *An Inductive Logic Programming Approach to Statistical Relational Learning*; IOS Press: Amsterdam, 2006.
(144) King, R. D.; Muggleton, S. H.; Srinivasan, A.; Sternberg, M. J. E. Structure-activity relationships derived by machine learning: The use of atoms and their bond connectivities to predict mutagenicity by inductive logic programming. *Proc. Natl. Acad. Sci. U.S.A.* **1996**, *93* (1), 438−442.
(145) Srinivasana, A.; Muggleton, S. H.; Sternberg, M. J. E.; King, R. D. Theories for mutagenicity: A study in first-order and feature-based induction. *Artif. Intell.* **1996**, *85* (1−2), 277−299.
(146) Amini, A.; Muggleton, S. H.; Lodhi, H.; Sternberg, M. J. E. A novel logic-based approach for quantitative toxicology prediction. *J. Chem. Inf. Model.* **2007**, *47* (3), 998−1006.
(147) Sternberg, M. J. E.; Muggleton, S. H. Structure activity relationships (SAR) and pharmacophore discovery using Inductive Logic Programming (ILP). *QSAR Comb. Sci.* **2003**, *22* (5), 527−532.
(148) Cannon, E. O.; Amini, A.; Bender, A.; Sternberg, M. J. E.; Muggleton, S. H.; Glen, R. C.; Mitchell, J. B. O. Support vector inductive logic programming outperforms the Naive Bayes Classifier and inductive logic programming for the classification of bioactive chemical compounds. *J. Comput.-Aided Mol. Des.* **2007**, *21* (5), 269−280.
(149) Tsunoyama, K.; Amini, A.; Sternberg, M. J. E.; Muggleton, S. H. Scaffold hopping in drug discovery using inductive logic programming. *J. Chem. Inf. Model.* **2008**, *48* (5), 949−957.
(150) King, R. D.; Srinivasan, A. The discovery of indicator variables for QSAR using inductive logic programming. *J. Comput.-Aided Mol. Des.* **1997**, *11* (6), 571−580.
(151) Marchand-Geneste, N.; Watson, K. A.; Alsberg, B. K.; King, R. D. New approach to pharmacophore mapping and QSAR analysis using inductive logic programming. Application to thermolysin inhibitors and glycogen phosphorylase b inhibitors. *J. Med. Chem.* **2002**, *45* (2), 399−409.
(152) Buttingsrud, B.; King, R. D.; Alsberg, B. K. An alignment-free methodology for modelling field-based 3D-structure activity relationships using inductive logic programming. *J. Chemom.* **2007**, *21* (12), 509−519.
(153) Varnek, A.; Gaudin, C.; Marcou, G.; Baskin, I.; Pandey, A. K.; Tetko, I. V. Inductive transfer of knowledge: Application of multi-task learning and feature net approaches to model tissue−air partition coefficients. *J. Chem. Inf. Model.* **2009**, *49* (1), 133−144.
(154) Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28* (1), 41−75.
(155) Evgeniou, T.; Micchelli, C. A.; Pontil, M. Learning multiple tasks with kernel methods. *J. Mach. Learn. Res.* **2005**, 6.
(156) Markou, M.; Singh, S. Novelty detection: A review - Part 1: Statistical approaches. *Signal Process.* **2003**, *83* (12), 2481−2497.
(157) Markou, M.; Singh, S. Novelty detection: A review - Part 2:: Neural network based approaches. *Signal Process.* **2003**, *83* (12), 2499−2521.
(158) Tax, D. M. J.; Duin, R. P. W. Support vector data description. *Mach. Learn.* **2004**, *54* (1), 45−66.
(159) Hristozov, D.; Oprea, T. I.; Gasteiger, J. Ligand-based virtual screening by novelty detection with self-organizing maps. *J. Chem. Inf. Model.* **2007**, *47* (6), 2044−2062.
(160) Karpov, P. V.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Virtual screening based on one-class classification. *Dokl. Chem.* **2011**, *437* (2), 107−111.
(161) Karpov, P. V.; Osolodkin, D. I.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. One-class classification as a novel method of ligand-based virtual screening: The case of glycogen synthase kinase 3OI inhibitors. *Bioorg. Med. Chem. Lett.* **2011**, *21* (22), 6728−6731.
(162) Karpov, P. V.; Baskin, I. I.; Zhokhova, N. I.; Zefirov, N. S. Method of continuous molecular fields in the one-class classification task. *Dokl. Chem.* **2011**, *440* (2), 263−265.
(163) Liu, Y. H.; Liu, Y. C.; Chen, Y. J. Fast support vector data descriptions for novelty detection. *IEEE Trans. Neural Networks* **2010**, *21* (8), 1296−1313.
(164) Wang, D.; Yeung, D. S.; Tsang, E. C. C. Structured one-class classification. *IEEE Trans. Syst., Man, Cyber., Part B* **2006**, *36* (6), 1283−1294.
(165) Li, C.; Zhang, Y.; Li, X. OcVFDT: One-class very fast decision tree for one-class classification of data streams. In *3rd International Workshop on Knowledge Discovery from Sensor Data, SensorKDD'09 in Conjunction with the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD-09*; ACM: New York, NY, 2009; pp 79−86.
(166) Angiulli, F. Condensed nearest neighbor data domain description. *IEEE Trans. Pattern Anal.* **2007**, *29* (10), 1746−1758.
(167) Görnitz, N.; Kloft, M.; Brefeld, U. Active and semi-supervised data domain description. *Lect. Notes Comput. Sci.* **2009**, *5781*, 407−422.

(168) Lee, H. J.; Cho, S. SOM-based novelty detection using novel data. *Lect. Notes Comput. Sci.* **2005**, *3578*, 359−366.
(169) Hoffmann, H. Kernel PCA for novelty detection. *Pattern Recogn.* **2007**, *40* (3), 863−874.
(170) Kwok, J. T.; Tsang, I. W. H.; Zurada, J. M. A class of single-class minimax probability machines for novelty detection. *IEEE Trans. Neural Networks* **2007**, *18* (3), 778−785.
(171) Cohen, G.; Sax, H.; Geissbuhler, A. Novelty detection using one-class parzen density estimator. An application to surveillance of nosocomial infections. *Stud. Health Technol. Inform.* **2008**, *136*, 21−26.
(172) Savran, Y.; Gunsel, B. Novelty detection on metallic surfaces by GMM learning in Gabor space. *Lect. Notes Comput. Sci.* **2010**, *6112*, 325−334.
(173) Clifton, D. A.; Hugueny, S.; Tarassenko, L. Novelty detection with multivariate extreme value statistics. *J. Signal Process. Syst.* **2010**, 1−19.
(174) Rätsch, G.; Mika, S.; Schölkopf, B.; Müller, K. R. Constructing boosting algorithms from SVMs: An application to one-class classification. *IEEE Trans. Pattern Anal.* **2002**, *24* (9), 1184−1199.
(175) Kassab, R.; Alexandre, F. Incremental data-driven learning of a novelty detection model for one-class classification with application to high-dimensional noisy data. *Mach. Learn.* **2009**, *74* (2), 191−234.
(176) Jaworska, J.; Nikolova-Jeliazkova, N.; Aldenberg, T. QSAR applicability domain estimation by projection of the training set in descriptor space: A review. *Altern. Lab. Anim.* **2005**, *33* (5), 445−459.
(177) Baskin, I. I.; Kireeva, N.; Varnek, A. The one-class classification approach to data description and to models applicability domain. *Mol. Inf.* **2010**, *29* (8−9), 581−587.
(178) Schölkopf, B.; Smola, A. J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*; MIT Press: Cambridge, MA; London, 2002.
(179) Fechner, N.; Jahn, A.; Hinselmann, G.; Zell, A. Estimation of the applicability domain of kernel-based machine learning models for virtual screening. *J. Cheminf.* **2010**, *2*, 1.
(180) Soto, A. J.; Vazquez, G. E.; Strickert, M.; Ponzoni, I. Target-driven subspace mapping methods and their applicability domain estimation. *Mol. Inf.* **2011**, *30* (9), 779−789.
(181) Sugiyama, M.; Krauledat, M.; Mueller, K.-R. Covariate shift adaptation by importance weighted cross validation. *J. Mach. Learn. Res.* **2007**, *8*, 985−1005.
(182) Daume, H.; Marcu, D. Domain adaptation for statistical classifiers. *J. Artif. Intell. Res.* **2006**, *26*, 101−126.
(183) Satpal, S.; Sarawagi, S. Domain adaptation of conditional probability models via feature subsetting. *Lect. Notes Comput. Sci.* **2007**, *4702*, 224−235.
(184) Zhang, Q.; Qiu, X.; Huang, X.; Wu, L. Domain adaptation for conditional random fields. *Lect. Notes Comput. Sci.* **2008**, *4993*, 192−202.
(185) Jiang, J.; Zha, C. A Two-Stage Approach to Domain Adaptation for Statistical Classifiers. In *16th ACM Conference on Information and Knowledge Management*; ACM: New York, 2007; pp 401−410.
(186) Arnold, A.; Cohen, W. W. Intra-Document Structural Frequency Features for Semi-Supervised Domain Adaptation. In *17th ACM Conference on Information and Knowledge Management*; ACM: New York, 2008; pp 1291−1299.
(187) Gupta, R.; Sarawagi, S. Domain adaptation of information extraction models. *SIGMOD Record* **2008**, *37* (4), 35−40.
(188) Swamidass, S. J.; Azencott, C. A.; Lin, T. W.; Gramajo, H.; Tsai, S. C.; Baldi, P. Influence relevance voting: An accurate and interpretable virtual high throughput screening method. *J. Chem. Inf. Model.* **2009**, *49* (4), 756−766.
(189) Johnson, A. M.; Maggiora, G. M. *Concepts and Applications of Molecular Similarity*; John Willey & Sons: New York, 1990.
(190) Kimeldorf, G. S.; Wahba, G. Some results on Tchebycheffian spline functions. *J. Math. Anal. Appl.* **1971**, *33*, 82−95.
(191) Müller, K. R.; Mika, S.; Rätsch, G.; Tsuda, K.; Schölkopf, B. An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Networks* **2001**, *12* (2), 181−201.
(192) Hofmann, T.; Schölkopf, B.; Smola, A. J. Kernel methods in machine learning. *Ann. Stat.* **2008**, *36* (3), 1171−1220.
(193) Shawe-Taylor, J.; Cristianini, N. *Kernel Methods for Pattern Analysis*; Cambridge University Press: New York, 2004.
(194) Gönen, M.; Alpaydin, E. Multiple kernel learning algorithms. *J. Mach. Learn. Res.* **2011**, *12*, 2211−2268.
(195) Lanckriet, G. R. G.; Cristianini, N.; Bartlett, P.; El Ghaoui, L.; Jordan, M. I. Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.* **2004**, *5*, 27−72.
(196) Cristianini, N.; Kandola, J.; Elisseeff, A.; Shawe-Taylor, J. On kernel target alignment. *Stud. Fuzziness Soft. Comput.* **2006**, *194*, 205−256.
(197) Ong, C. S.; Smola, A. J.; Williamson, R. C. Learning the kernel with hyperkernels. *J. Mach. Learn. Res.* **2005**, *6*, 1043−4071.
(198) Zhokhova, N. I.; Baskin, I. I.; Bakhronov, D. K.; Palyulin, V. A.; Zefirov, N. S. Method of continuous molecular fields in the search for quantitative structure−activity relationships. *Dokl. Chem.* **2009**, *429* (1), 273−276.
(199) Zhuang, J.; Tsang, I. W.; Hoi, S. C. H. A family of simple non-parametric kernel learning algorithms. *J. Mach. Learn. Res.* **2011**, *12*, 1313−1347.
(200) Kulis, B.; Sustik, M. A.; Dhillon, I. S. Low-rank kernel learning with bregman matrix divergences. *J. Mach. Learn. Res.* **2009**, *10*, 341−376.
(201) Johnson, R.; Zhang, T. Graph-based semi-supervised learning and spectral kernel design. *IEEE Trans. Inf. Theory* **2008**, *54* (1), 275−288.
(202) Weinberger, K. Q.; Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.* **2009**, *10*, 207−244.
(203) Huang, K. Z.; Ying, Y. M.; Campbell, C. Generalized sparse metric learning with relative comparisons. *Knowl. Inf. Syst.* **2011**, *28* (1), 25−45.
(204) Mohr, J. A.; Jain, B. J.; Obermayer, K. Molecule kernels: A descriptor- and alignment-free quantitative structure−activity relationship approach. *J. Chem. Inf. Model.* **2008**, *48* (9), 1868−1881.
(205) Saigo, H.; Vert, J. P.; Ueda, N.; Akutsu, T. Protein homology detection using string alignment kernels. *Bioinformatics* **2004**, *20* (11), 1682−1689.
(206) Hoffmann, B.; Zaslavskiy, M.; Vert, J. P.; Stoven, V. A new protein binding pocket similarity measure based on comparison of clouds of atoms in 3D: Application to ligand prediction. *BMC Bioinf.* **2010**, *11*, Art. No. 99.
(207) Haasdonk, B. Feature space interpretation of SVMs with indefinite kernels. *IEEE Trans. Pattern Anal.* **2005**, *27* (4), 482−492.
(208) Pekalska, E.; Haasdonk, B. Kernel discriminant analysis for positive definite and indefinite kernels. *IEEE Trans. Pattern Anal.* **2009**, *31* (6), 1017−1031.
(209) Sun, H. W.; Wu, Q. A. Least square regression with indefinite kernels and coefficient regularization. *Appl. Comput. Harmon. Anal.* **2011**, *30* (1), 96−109.
(210) Todeschini, R.; Consonni, V. *Handbook of Molecular Descriptors*; Wiley-VCH Publishers: Weinheim, 2000.
(211) Baskin, I.; Varnek, A. Fragment Descriptors in SAR/QSAR/QSPR Studies, Molecular Similarity Analysis and in Virtual Screening. In *Chemoinformatics Approaches to Virtual Screening*; Varnek, A., Tropsha, A., Eds.; RSC Publishing : Cambridge, 2008; pp 1−43.
(212) Bakir, G.; Hofmann, T.; Schoelkopf, B.; Smola, A. J.; Taskar, B.; Vishwanathan, S. V. N. *Predicting Structured Data*; The MIT Press: Cambridge, MA, 2007.
(213) Cook, D. J.; Holder, L. B. *Mining Graph Data*; Wiley-Interscience: Hoboken, NJ, 2007.
(214) De Raedt, L. *Logical and Relational Learning*; Springer-Verlag: Berlin, Heidelberg, 2008.
(215) Tsochantaridis, I.; Joachims, T.; Hofmann, T.; Altun, Y. Large margin methods for structured and interdependent output variables. *J. Mach. Learn. Res.* **2005**, *6*, 1453−1484.
(216) Kramer, S.; De Raedt, L.; Helma, C. Molecular feature mining in HIV data. In *Proceedings of the Seventh ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining; ACM Press: New York, 2001; pp 136−143.
(217) De Raedt, L.; Kramer, S. The Levelwise Version Space Algorithm and its Application to Molecular Fragment Finding. In *The Seventeenth International Joint Conference on Articial Intelligence*; Morgan Kaufmann: San Francisco, 2001; pp 853−862.
(218) Kramer, S.; De Raedt, L. Feature Construction with Version Spaces for Biochemical Applications. In *The Eighteenth International Conference on Machine Learning*; Morgan Kaufmann: San Francisco, 2001; pp 258−265.
(219) Inokuchi, A., Mining Generalized Substructures from a Set of Labeled Graphs. In *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04)*; IEEE Computer Society: Brighton, UK, 2004; pp 415−418.
(220) Yan, X. F.; Han, J. W. gSpan: Graph-Based Substructure Pattern Mining. In *2002 IEEE International Conference on Data Mining, Proceedings*; Kumar, V.; Tsumoto, S., Zhong, N., Yu, P. S., Wu, X. D., Eds.; IEEE Computer Soc.: Los Alamitos, 2002; pp 721−724.
(221) Chi, Y.; Muntz, R. R.; Nijssen, S.; Kok, J. N. Frequent subtree mining -- an overview. *Fundam. Inform.* **2005**, *66* (1−2), 161−198.
(222) Saigo, H.; Tsuda, K. Graph Mining in Chemoinformatics. In *Chemoinformatics and Advanced Machine Learning Perspectives: Complex Computational Methods and Collaborative Techniques*; Lodhi, H.; Yamanishi, Y., Eds.; IGI Global: Hershey, PA, 2010; pp 95−128.
(223) Baskin, I. I.; Zhokhova, N. I.; Palyulin, V. A.; Zefirov, N. S. Additive inductive learning in QSAR/QSPR studies and molecular modeling. *Chem. Central J.* **2009**, *3*, 1−1.
(224) Erhan, D.; L'Heureux, P.-J.; Yue, S. Y.; Bengio, Y. Collaborative filtering on a family of biological targets. *J. Chem. Inf. Model.* **2006**, *46* (2), 626−635.
(225) Faulon, J. L.; Misra, M.; Martin, S.; Sale, K.; Sapra, R. Genome scale enzyme - Metabolite and drug - Target interaction predictions using the signature molecular descriptor. *Bioinformatics* **2008**, *24* (2), 225−233.
(226) Jacob, L.; Vert, J. P. Protein-ligand interaction prediction: An improved chemogenomics approach. *Bioinformatics* **2008**, *24* (19), 2149−2156.
(227) Geppert, H.; Humrich, J.; Stumpfe, D.; Gaertner, T.; Bajorath, J. Ligand prediction from protein sequence and small molecule information using support vector machines and fingerprint descriptors. *J. Chem. Inf. Model.* **2009**, *49* (4), 767−779.
(228) Bakır, G.; Hofmann, T.; Schölkopf, B.; Smola, A. J.; Taskar, B.; Vishwanathan, S. V. N. *Predicting Structured Data*; MIT Press: Cambridge, Massachusetts, London, 2007.
(229) Cortes, C.; Mohri, M.; Weston, J. A general regression technique for learning transductions.. In *ICML 2005 − Proceedings of the 22nd International Conference on Machine Learning*; ACM: New York, NY, USA, 2005; pp 153−160.
(230) Geurts, P.; Wehenkel, L.; D'Alché-Buc, F., Kernelizing the Output of Tree-Based Methods. In *ACM International Conference Proceeding Series*; 2006; Vol. *148*, pp 345−352.
(231) Geurts, P.; Wehenkel, L.; D'Alché-Buc, F., Gradient Boosting for Kernelized Output Spaces. In *ACM International Conference Proceeding Series*, 2007; Vol. *227*, pp 289−296.
(232) Yu, C. N. J.; Joachims, T.; Elber, R.; Pillardy, J. Support vector training of protein alignment models. *J. Comput. Biol.* **2008**, *15* (7), 867−880.
(233) Geurts, P.; Touleimat, N.; Dutreix, M.; d'Alché-Buc, F. Inferring biological networks with output kernel trees. *BMC Bioinf.* **2007**, *8* (Suppl.2), S4.
(234) Rathke, F.; Hansen, K.; Brefeld, U.; Muller, K.-R. StructRank: A new approach for ligand-based virtual screening. *J. Chem. Inf. Model.* **2010**, *51* (1), 83−92.
(235) Vapnik, V. Transductive Inference and Semi-Supervised Learning. In *Semi-Supervised Learning*; Chapelle, O., Schoelkopf, B., Zien, A., Eds.; MIT Press: Cambridge, MA, 2006; pp 453−472.
(236) Cramer, R. D.; Patterson, D. E.; Bunce, J. D. Comparative molecular field analysis (CoMFA). 1. Effect of shape on binding of steroids to carrier proteins. *J. Am. Chem. Soc.* **1988**, *110* (18), 5959−5967.
(237) Klebe, G.; Abraham, U.; Mietzner, T. Molecular similarity indices in a comparative analysis (CoMSIA) of drug molecules to correlate and predict their biological activity. *J. Med. Chem.* **1994**, *37* (24), 4130−46.
(238) Goodford, P. The Basic Principles of GRID. In *Molecular Interaction Fields. Applications in Drug Discovery and ADME Prediction*; Cruciani, G., Ed.; Wiley-VCH: Weinheim, 2006; pp 3−26.
(239) Silverman, B. D.; Platt, D. E. Comparative molecular moment analysis (CoMMA): 3D-QSAR without molecular superposition. *J. Med. Chem.* **1996**, *39* (11), 2129−40.
(240) Bravi, G.; Gancia, E.; Mascagni, P.; Pegna, M.; Todeschini, R.; Zaliani, A. MS-WHIM, new 3D theoretical descriptors derived from molecular surface properties: a comparative 3D QSAR study in a series of steroids. *J. Comput.-Aided Mol. Des.* **1997**, *11* (1), 79−92.
(241) Pastor, M.; Cruciani, G.; McLay, I.; Pickett, S.; Clementi, S. GRid-INdependent descriptors (GRIND): A novel class of alignment-independent three-dimensional molecular descriptors. *J. Med. Chem.* **2000**, *43* (17), 3233−43.
(242) Cruciani, G.; Pastor, M.; Guba, W. VolSurf: a new tool for the pharmacokinetic optimization of lead compounds. *Eur. J. Pharm. Sci.* **2000**, *11* (Suppl. 2), S29−S39.
(243) Carbo-Dorca, R.; Robert, D.; Amat, L.; Girones, X.; Besalu, E. *Molecular Quantum Similarity in QSAR and Drug Design*; Springer: Berlin, Heidelberg, New York, 2000.
(244) Halberstam, N. M.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Construction of neural-network structure-conditions-property relationships: Modeling of the physicochemical properties of hydrocarbons. *Dokl. Chem.* **2002**, *384* (1−3), 140−143.
(245) Oprisiu, I.; Varlamova, E.; Muratov, E.; Artemenko, A.; Marcou, G.; Polishchuk, P.; Kuz'min, V. QSPR approaches to predict non-additive properties of mixtures. Application to bubble point temperatures of binary mixtures of liquid. *Mol. Inf.*. **2012**, *31*, accepted for publication.
(246) Ramsay, J. O.; Silverman, B. W. *Functional Data Analysis*, 2nd ed.; Springer: New York, 2005.
(247) Devillers, J. *Neural Networks in QSAR and Drug Design*; Academic Press: London, 1996.
(248) Bengio, Y. Learning deep architectures for AI. *Found. Trends Mach. Learn.* **2009**, *2* (1), 1−127.
(249) Wu, T.-F.; Lin, C.-J.; Weng, R. C. Probability estimates for multi-class classification by pairwise coupling. *J. Mach. Learn. Res.* **2004**, *5*, 975−1005.
(250) Huang, T.-M.; Kecman, V.; Kopriva, I. *Kernel Based Algorithms for Mining Huge Data Sets. Supervised, Semi-Supervised, and Unsupervised Learning*; Springer: Berlin, Heidelberg, 2006.
(251) Huang, T. M.; Kecman, V.; Kopriva, I. Iterative single data algorithm for kernel machines from huge data sets: Theory and performance. *Stud. Comput. Intell.* **2006**, *17*, 61−95.
(252) Shalev-Shwartz, S.; Singer, Y.; Srebro, N.; Cotter, A. Pegasos: primal estimated sub-gradient solver for SVM. *Math. Program.* **2011**, *127* (1), 3−30.
(253) Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; Lin, C.-J. LIBLINEAR: A library for large linear classification. *J. Mach. Learn. Res.* **2008**, *9*, 1871−1874.
(254) Kivinen, J.; Smola, A. J.; Williamson, R. C. Online learning with kernels. *IEEE Trans. Signal Process.* **2004**, *52* (8), 2165−2176.
(255) Taniguchi, M.; Du, H.; Lindsey, J. S. Virtual libraries of tetrapyrrole macrocycles: Combinatorics, isomers, product distributions, and data mining. *J. Chem. Inf. Model.* **2011**, *51* (9), 2233−2247.
(256) Quinonero-Candela, J.; Sugiyama, M.; Schwaighofer, A.; Lawrence, N. D. *Dataset Shift in Machine Learning*; MIT Press: Cambridge, MA, 2009.
(257) Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; Vaughan, J. W. A theory of learning from different domains. *Mach. Learn.* **2010**, *79* (1−2), 151−175.
(258) Mansour, Y. Learning and domain adaptation. *Lect. Notes Comput. Sci.* **2009**, *5808*, 32−34.

(259) Mansour, Y. Learning and domain adaptation. *Lect. Notes Comput. Sci.* **2009**, *5809*, 4−6.
(260) Pathak, M. A.; Nyberg, E. H. Learning algorithms for domain adaptation. *Lect. Notes Comput. Sci.* **2009**, *5828*, 293−307.
(261) Woodward, A. M.; Alsberg, B. K.; Kell, D. B. The effect of heteroscedastic noise on the chemometric modelling of frequency domain data. *Chemom. Intell. Lab. Syst.* **1998**, *40* (1), 101−107.
(262) Lopera, L. G.; Cepeda-Cuervo, E.; Achcar, J. A. Heteroscedastic normal-exponential mixture models: Bayesian and classical approaches. *Appl. Math. Comput.* **2011**, *218* (7), 3635−3648.
(263) Munoz-Gonzalez, L.; Lazaro-Gredilla, M.; Figueiras-Vidal, A. R. Heteroscedastic Gaussian Process Regression Using Expectation Propagation. In *2011 IEEE International Workshop on Machine Learning for Signal Processing*; Tan, T., Katagiri, S., Tao, J., Nakamura, A.; Larsen, J., Eds.; IEEE: New York, 2011.
(264) Ranu, S.; Calhoun, B. T.; Singh, A. K.; Swamidass, S. J. Probabilistic substructure mining from small-molecule screens. *Mol. Inf* **2011**, *30* (9), 809−815.
(265) Estabrooks, A.; Jo, T.; Japkowicz, N. A multiple resampling method for learning from imbalanced data sets. *Comput. Intell.* **2004**, *20* (1), 18−36.
(266) Kondratovich, E. P.; Zhokhova, N. I.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Fragmental descriptors in (Q)SAR: Prediction of the assignment of organic compounds to pharmacological groups using the support vector machine approach. *Russ. Chem. Bull.* **2009**, *58* (4), 657−662.
(267) Baldi, P.; Azencott, C.; Swamidass, S. J. Bridging the gap between neural network and kernel methods: Applications to drug discovery. *Front. Artif. Intell. Appl.* **2011**, *226*, 3−13.
(268) Sheridan, R. P.; Singh, S. B.; Fluder, E. M.; Kearsley, S. K. Protocols for bridging the peptide to nonpeptide gap in topological similarity searches. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (5), 1395−1406.
(269) Truchon, J. F.; Bayly, C. I. Evaluating virtual screening methods: Good and bad metrics for the "early recognition" problem. *J. Chem. Inf. Model.* **2007**, *47* (2), 488−508.
(270) Swamidass, S. J.; Azencott, C. A.; Daily, K.; Baldi, P. A CROC stronger than ROC: Measuring, visualizing and optimizing early retrieval. *Bioinformatics* **2010**, *26* (10), 1348−1356.
(271) Tax, D. M. J. One-Class Classification. Concept-Learning in the Absence of Counter-Examples; Doctor Thesis, Technische Universiteit Delft, Delft, The Netherlands, 2001.
(272) Bhardwaj, N.; Gerstein, M.; Lu, H. Genome-wide sequence-based prediction of peripheral proteins using a novel semi-supervised learning technique. *BMC Bioinf.*. **2010**, *11* (SUPPL.1).
(273) Blanchard, G.; Lee, G.; Scott, C. Semi-supervised novelty detection. *J. Mach. Learn. Res.* **2010**, *11*, 2973−3009.
(274) Guha, R. On the interpretation and interpretability of quantitative structure−activity relationship models. *J. Comput.-Aided Mol. Des.* **2008**, *22* (12), 857−871.
(275) Hansch, C.; Muir, R. M.; Fujita, T.; Maloney, P. P.; Geiger, F.; Streich, M. The correlation of biological activity of plant growth regulators and chloromycetin derivatives with hammett constants and partition coefficients. *J. Am. Chem. Soc.* **1963**, *85* (18), 2817−2824.
(276) Baskin, I. I.; Ait, A. O.; Halberstam, N. M.; Palyulin, V. A.; Zefirov, N. S. An approach to the interpretation of backpropagation neural network models in QSAR studies. *SAR QSAR Environ. Res.* **2002**, *13* (1), 35−41.
(277) Guha, R.; Jurs, P. C. Interpreting computational neural network QSAR Models: A measure of descriptor importance. *J. Chem. Inf. Model.* **2005**, *45* (3), 800−806.
(278) Guha, R.; Stanton, D. T.; Jurs, P. C. Interpreting computational neural network quantitative structure−activity relationship models: A detailed interpretation of the weights and biases. *J. Chem. Inf. Model.* **2005**, *45* (4), 1109−1121.
(279) Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; Müller, K. R. How to explain individual classification decisions. *J. Mach. Learn. Res.* **2010**, *11*, 1803−1831.
(280) Hansen, K.; Baehrens, D.; Schroeter, T.; Rupp, M.; Müller, K. R. Visual interpretation of kernel-based prediction models. *Mol. Inf.* **2011**, *30* (9), 817−826.
(281) Pan, S. J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22* (10), 1345−1359.
(282) Ratle, F.; Kanevski, M.; Terrettaz-Zufferey, A. L.; Esseiva, P.; Ribaux, O. A comparison of one-class classifiers for novelty detection in forensic case data. *Lect. Notes Comput. Sci.* **2007**, *4881*, 67−76.
(283) Khan, S. S.; Madden, M. G. A survey of recent trends in one class classification. *Lect. Notes Comput. Sci.* **2010**, *6206*, 188−197.
(284) Cohn, D.; Atlas, L.; Ladner, R. Improving generalization with active learning. *Mach. Learn.* **1994**, *15* (2), 201−221.
(285) Kanamori, T. Statistical asymptotic theory of active learning. *Ann. I. Stat. Math.* **2002**, *54* (3), 459−475.
(286) Prince, M. Does active learning work? A review of the research. *J. Eng. Educ.* **2004**, *93* (3), 223−231.
(287) Asogawa, M.; Osoda, T.; Fujiwara, Y.; Yamashita, Y. Efficient drug screening using active learning. *NEC J. Adv. Technol.* **2005**, *2* (2), 145−148.
(288) Vogiatzis, D.; Tsapatsoulis, N. Active learning for microarray data. *Int. J. Approx. Reason.* **2008**, *47* (1), 85−96.
(289) Mohamed, T. P.; Carbonell, J. G.; Ganapathiraju, M. K. Active learning for human protein−protein interaction prediction. *BMC Bioinf.* **2010**, *11* (Suppl.1), Art. No. S57.
(290) Bayarri, M. J.; Berger, J. O. The interplay of Bayesian and frequentist analysis. *Stat. Sci.* **2004**, *19* (1), 58−80.
(291) Baskin, I.; Marcou, G.; Varnek, A. Tutorial on Ensemble Learning. http://infochim.u-strasbg.fr/new/CS3_2010/Tutorial/Ensemble/EnsembleModeling.pdf (accessed April 5, 2012).
(292) Kuncheva, L. I. *Combining Pattern Classifiers: Methods and Algorithms*; Wiley-Interscience: Hoboken, NJ, 2004.
(293) Huang, T. M.; Kecman, V.; Kopriva, I. Semi-supervised learning and applications. *Stud. Comput. Intell.* **2006**, *17*, 125−173.
(294) Joachims, T. Transductive Support Vector Machines. In *Semi-Supervised Learning*; Chapelle, O., Schoelkopf, B., Zien, A., Eds.; MIT Press: Cambridge, MA, 2006; pp 105−117.
(295) Liu, J.; Wang, H.; Zhao, T. Protein-protein interaction extraction based on combining TSVM and active learning. *Gaojishu Tongxin/Chinese High Technol. Lett.* **2009**, *19* (5), 480−486.
(296) Joachims, T. *SVMlight*, version 6.02; 2008. http://svmlight.joachims.org/ (accessed April 4, 2012).
(297) Joachims, T. *SGTlight*, version 1.00; 2003. http://sgt.joachims.org/ (accessed April 4, 2012).
(298) Huang, T.-M.; Kecman, V. *SemiL*, 2005. http://www.learning-from-data.com/te-ming/semil.htm (accessed April 4, 2012).
(299) Belkin, M.; Niyogi, P.; Sindhwani, V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.* **2006**, *7*, 2399−2434.
(300) Miao, Z. M.; Zhao, L. W.; Hu, G. Y.; Wang, Q. Semi-supervised learning based on one-class classification. *Moshi Shibie yu Rengong Zhineng/Pattern Recognit. Artif. Intell.* **2009**, *22* (6), 924−930.
(301) Pan, Z. S.; Yan, Y. S.; Miao, Z. M.; Ni, G. Q.; Zhang, H. Semi-supervised learning based on one-class classification and ensemble learning. *Jiefangjun Ligong Daxue Xuebao/J. PLA Univ. Sci. Technol. (Natural Science Ed.)* **2010**, *11* (4), 397−402.
(302) *Stuttgart Neural Network Simulator (SNNS)*, version 4.3. http://www.ra.cs.uni-tuebingen.de/SNNS/ (accessed April 4, 2012).
(303) Bakker, B.; Heskes, T. Task clustering and gating for Bayesian multitask learning. *J. Mach. Learn. Res.* **2004**, *4* (1), 83−99.
(304) Pillonetto, G.; Dinuzzo, F.; De Nicolao, G. Bayesian online multitask learning of Gaussian processes. *IEEE Trans. Pattern Anal.* **2010**, *32* (2), 193−205.
(305) Lu, W. C.; Chen, N. Y.; Li, G. Z.; Yang, J. Multitask Learning Using Partial Least Squares Method. In *Proceedings of the Seventh International Conference on Information Fusion*; Svensson, P.; Schubert, J., Ed.; International Society of Information Fusion: Stockholm, Sweden, 2004; Vol. *1*, pp 79−84.
(306) Dekel, O.; Long, P. M.; Singer, Y. Online multitask learning. *Lect. Notes Comput. Sci.* **2006**, *4005*, 453−467.

(307) Bueno-Crespo, A.; Sánchez-García, A.; Morales-Sánchez, J.; Sancho-Gómez, J. L. Multitask learning with data editing. *Lect. Notes Comput. Sci.* **2007**, *4527*, 320−326.
(308) Liu, Q.; Liao, X.; Carin, H. L.; Stack, J. R.; Carin, L. Semisupervised multitask learning. *IEEE Trans. Pattern Anal.* **2009**, *31* (6), 1074−1086.
(309) Kato, T.; Kashima, H.; Sugiyama, M.; Asai, K. Conic programming for multitask learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22* (7), 957−968.
(310) Widmer, C.; Toussaint, N. C.; Altun, Y.; Rätsch, G. Inferring latent task structure for Multitask Learning by Multiple Kernel Learning. *BMC Bioinf.* **2010**, *11* (Suppl. 8), Art. No. S5.
(311) Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20* (3), 273−297.
(312) Vapnik, V. N. An overview of statistical learning theory. *IEEE Trans. Neural Networks* **1999**, *10* (5), 988−999.
(313) Chang, C.-C.; Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Trans. Intel. Syst. Technol.* **2001**, *2* (3), 27:1−27:27.
(314) Hoerl, A. E.; Kennard, R. W. Risge regression: Application to nonorthogonal problems. *Technometrics* **1970**, *12* (1), 69−82.
(315) Efron, B.; Hastie, T.; Johnstone, I.; Tibshirani, R.; Ishwaran, H.; Knight, K.; Loubes, J. M.; Massart, P.; Madigan, D.; Ridgeway, G.; Rosset, S.; Zhu, J. I.; Stine, R. A.; Turlach, B. A.; Weisberg, S. Least angle regression. *Ann. Stat* **2004**, *32* (2), 407−499.
(316) Fraley, C.; Hesterberg, T. Least angle regression and LASSO for large datasets. *Stat. Anal. Data Mining* **2009**, *1* (4), 251−259.
(317) Tibshirani, R. Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B* **1996**, *58* (1), 267−288.
(318) Tibshirani, R. The lasso method for variable selection in the cox model. *Stat. Med.* **1997**, *16* (4), 385−395.
(319) Wang, L.; Gordon, M. D.; Zhu, J. Regularized Least Absolute Deviations Regression and an Efficient Algorithm for Parameter Tuning. In *ICDM 2006: Sixth International Conference on Data Mining, Proceedings*; Clifton, C. W.; Zhong, N., Liu, J. M., Wah, B. W., Wu, X. D., Eds.; IEEE Computer Soc.: Los Alamitos, 2006; pp 690−700.
(320) Witten, D. M.; Tibshirani, R.; Hastie, T. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics* **2009**, *10* (3), 515−534.
(321) Witten, D. M.; Tibshirani, R. J. Extensions of sparse canonical correlation analysis with applications to genomic data. *Stat. Appl. Genet. Mol*, **2009**, *8* (1), Art. No. 28.
(322) Huang, T.-M.; Kecman, V.; Kopriva, I. *ISDA*, 2006. http://www.learning-from-data.com/download.htm (accessed April 4, 2012).
(323) Sonnenburg, S.; Rätsch, G.; Henschel, S.; Widmer, C.; Behr, J.; Zien, A.; De Bona, F.; Binder, A.; Gehl, C.; Franc, V. The Shogun machine learning toolbox. *J. Mach. Learn. Res.* **2010**, *11*, 1799−1802.
(324) *SHOGUN*, version 1.1.0; 2011. http://www.shogun-toolbox.org/ (accessed April 4, 2012).
(325) *LIBLINEAR*, version 1.8; 2011. http://www.csie.ntu.edu.tw/~cjlin/liblinear/ (accessed April 4, 2012).
(326) Huang, T.-M.; Kecman, V. *LinearSVM*, version 3.0; 2009. http://www.linearsvm.com/ (accessed January 24, 2012).
(327) Sushko, I.; Novotarskyi, S.; Körner, R.; Pandey, A. K.; Rupp, M.; Teetz, W.; Brandmaier, S.; Abdelaziz, A.; Prokopenko, V. V.; Tanchuk, V. Y.; Todeschini, R.; Varnek, A.; Marcou, G.; Ertl, P.; Potemkin, V.; Grishina, M.; Gasteiger, J.; Schwab, C.; Baskin, I. I.; Palyulin, V. A.; Radchenko, E. V.; Welsh, W. J.; Kholodovych, V.; Chekmarev, D.; Cherkasov, A.; Aires-De-Sousa, J.; Zhang, Q. Y.; Bender, A.; Nigsch, F.; Patiny, L.; Williams, A.; Tkachenko, V.; Tetko, I. V. Online chemical modeling environment (OCHEM): Web platform for data storage, model development and publishing of chemical information. *J. Comput.-Aided Mol. Des.* **2011**, *25* (6), 533−554.
(328) Wang, L.; Froehlich, H.; Rieck, K.; Tsai, C.-T.; Lin, T.-J. *SVDD*. http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/#libsvm_for_svdd_and_finding_the_smallest_sphere_containing_all_data (accessed January 12, 2012).
(329) Angiulli, F. Condensed nearest neighbor data domain description. *Lect. Notes Comput. Sci.* **2005**, *3646*, 12−23.
(330) Angelov, P.; Zhou, X. Evolving fuzzy classifier for novelty detection and landmark recognition by mobile robots. *Stud. Comput. Intell.* **2007**, *50*, 89−118.
(331) Wu, Q.; Tan, S.; Duan, M.; Cheng, X. A two-stage algorithm for domain adaptation with application to sentiment transfer problems. *Lect. Notes Comput. Sci.* **2010**, *6458*, 443−453.
(332) Luo, B.; Wilson, R. C.; Hancock, E. R. A linear generative model for graph structure. *Lect. Notes Comput. Sci.* **2005**, *3434*, 54−62.
(333) Xiao, B.; Hancock, E. R. A spectral generative model for graph structure. *Lect. Notes Comput. Sci.* **2006**, *4109*, 173−181.
(334) White, D.; Wilson, R. C. Spectral Generative Models for Graphs. In *14th International Conference on Image Analysis and Processing, Proceedings*; IEEE Computer Soc.: Los Alamitos, 2007; pp 35−40.
(335) White, D.; Wilson, R. C. Parts Based Generative Models for Graphs. In *19th International Conference on Pattern Recognition*; Vols. *1*−*6*, IEEE: New York, 2008; pp 3318−3321.
(336) Inokuchi, A.; Washio, T.; Motoda, H. An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data. In *PKDD '00 Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*; Springer-Verlag: London, 2000; pp 13−23.
(337) Borgelt, C.; Meinl, T.; Berthold, M. MoSS: A Program for Molecular Substructure Mining. In *Proceedings of the 1st international Workshop on Open Source Data Mining: Frequent Pattern Mining Implementations*; ACM Press: New York, 2005; pp 6−15.
(338) Zaki, M. J. Efficiently mining frequent trees in a forest. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM Press: New York, 2002; pp 71−80.
(339) Chi, Y.; Yang, Y.; Xia, Y.; Muntz, R. R. CMTreeMiner: Mining both closed and maximal frequent subtrees. *Lect. Notes Comput. Sci.* **2004**, *3056*, 63−73.
(340) Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. A methodology for searching direct correlations between structures and properties of organic compounds by using computational neural networks. *Dokl. Akad. Nauk* **1993**, *333* (2), 176−179.
(341) Bianucci, A. M.; Micheli, A.; Sperduti, A.; Starita, A. A novel approach to QSPR/QSAR based on neural networks for structures. *Stud. Fuzziness Soft Comput.* **2003**, *120*, 265−296.
(342) Goulon, A.; Duprat, A.; Dreyfus, G. Graph machines and their applications to computer-aided drug design: A new approach to learning from structured data. *Lect. Notes Comput. Sci.* **2006**, *4135*, 1−19.
(343) Frank, E.; Xu, X. *Applying Propositional Learning Algorithms to Multi-Instance Data*; Working paper 06/03); University of Waikato, Department of Computer Science: Hamilton, New Zealand, 2003.
(344) Ramsay, J. O.; Hooker, G.; Graves, S. *Functional Data Analysis with R and MATLAB*; Springer: New York, 2009.