Towards the Revival of Interpretable QSAR Models

Watshara Shoombuatong, Philip Prathipati, Wiwat Owasirikul, Apilak Worachartcheewan, Saw Simeon, Nuttapat Anuwongcharoen, Jarl E. S. Wikberg and Chanin Nantasenamat

Abstract Quantitative structure-activity relationship (QSAR) has been instrumental in aiding medicinal chemists and physical scientists in understanding how modification of substituents at different positions on a molecular structure exert its influence on the observed biological activity and physicochemical property, respectively. QSAR has received great attention owing to its predictive capability and as such efforts had been directed toward obtaining models with high prediction performance. However, to be useful QSAR models need to be informative and interpretable in which the underlying molecular features that contribute to the increase or decrease of the biological activity are revealed by the model. Thus, the aim of this chapter is to briefly review the general concepts of QSAR modeling, its development and discussions on key issues influencing and contributing to the interpretability of QSAR models.

W. Shoombuatong and P. Prathipati
These authors contributed equally to this work.

W. Shoombuatong · S. Simeon · N. Anuwongcharoen · C. Nantasenamat (✉)
Center of Data Mining and Biomedical Informatics, Faculty of Medical Technology, Mahidol University, Bangkok 10700, Thailand
e-mail: chanin.nan@mahidol.edu

P. Prathipati
National Institutes of Biomedical Innovation, Health and Nutrition, Osaka 567-0085, Japan

W. Owasirikul
Department of Radiological Technology, Faculty of Medical Technology, Mahidol University, Bangkok 10700, Thailand

A. Worachartcheewan
Department of Community Medical Technology, Faculty of Medical Technology, Mahidol University, Bangkok 10700, Thailand

J.E.S. Wikberg
Department of Pharmaceutical Biosciences, BMC, Uppsala University, SE-751 24 Uppsala, Sweden

K. Roy (ed.), *Advances in QSAR Modeling*, Challenges and Advances in Computational Chemistry and Physics 24, DOI 10.1007/978-3-319-56850-8_1

Keywords Quantitative structure-activity relationship · Quantitative structure-property relationship · Proteochemometrics · Data mining · Machine learning · Cheminformatics · Chemogenomics · QSAR · QSPR · Interpretable · Drug discovery · Drug design

1 Introduction

Quantitative structure-activity relationship (QSAR) can be considered to be one of the pillars for driving drug discovery efforts forward by enabling practitioners to make sense of the big data from bioactivity assays of chemical library (Nantasenamat et al. 2009, 2010; Cherkasov et al. 2014). Computer-aided drug design or simply computational drug design is essentially comprised of four major levels: (i) fragment, (ii) ligand, (iii) structure and (iv) systems based approaches (Nantasenamat and Prachayasittikul 2015). QSAR is a ligand-based approach meaning that it primarily makes use of information derived from ligands that does not require the need for details of the target protein. Thus, ligand-based approaches are particularly suited in situations where there is negligible information on the biological target. The reasons for using QSAR and quantitative structure-property relationship (QSPR) models are many: (i) to reduce time and cost; (ii) to rationally predict biological, pharmaceutical, physical and chemical activities/properties; (iii) to aid experimental scientists by providing the collective wisdom learned from previous big data; (vi) to shed light on the mechanism of action for biological activities of interest. QSAR/QSPR has found wide applications in the life sciences (Prachayasittikul et al. 2015) (e.g. biology, agriculture and medicine) as well as the physical sciences (Katritzky et al. 2010) (e.g. organic chemistry, physical chemistry, materials sciences). In drug discovery, QSAR has been successfully applied in the prediction of logP and pK_a values as well as absorption, distribution, metabolism, excretion and toxicity (ADMET) properties (Khan and Sylte 2007). It is indeed a difficult task to design a drug that exert activity toward the target protein(s) of interest while at the same time show proper uptake, metabolism, excretion and be devoid of toxicity. To aid medicinal chemists in understanding the origin of ADMET properties Gleeson proposed a set of simple and interpretable rules through the use of principal component analysis of simple descriptors (e.g. molecular weight, logP, ionization state, etc.) (Gleeson 2008).

The robustness of QSAR relies on its capability to predict the biological activities or chemical properties of interests by learning from retrospective experimental data sets. Particularly, each compound in a chemical library is quantitatively or qualitatively described by a set of molecular descriptors and such vector of descriptors (also known as independent variables in statistics) are mathematically correlated with the biological or chemical endpoint of interest (i.e. pIC_{50}, log$P$, etc.) via traditional multivariate analysis or machine learning algorithm. However, it is worthy to note that QSAR models is only as good as the data that was used to train it and in spite of its predictive capability it should not be viewed as a replacement of domain knowledge

of scientists but rather should be considered as a complementary tool for aiding the decision-making process.

In spite of its widespread usage, it seems that the full potential for QSAR models has not yet been achieved as current efforts are localized on generating models with good predictive performance at the cost of vague or uninterpretable models. Most robust machine learning algorithms are so-called *black box* since the underlying features contributing to the variation in the endpoint values are not accessible to practitioners. To be of benefit for the experimental biologist or chemist, models need to be transparent such that the underlying important features are revealed. Moreover, features describing the general or unique characteristics of compounds needs to be unambiguous, interpretable and easily comprehensible. Upstream to the issue of interpretability is the accessibility or the know-how on the development of robust QSAR models. Nowadays, the construction of QSAR models may seem to be a trivial and mainstream task in computational drug design. However, a robust, reliable and reproducible model can only be achieved through careful data curation and analysis, which certainly requires the expertise of trained practitioners. This is particularly true as not all starting data set is *modelable* or may not always yield promising results right out of the box owing to several inherent issues that will be discussed in this chapter.

2 Brief History of QSAR

More than a century ago, QSAR was developed by several research groups. The precursor to the birth of QSAR began in 1863 when Cros (Cros 1863) observed that there exists an inverse correlation between toxicity and water solubility. Particularly, the toxicity of alcohols toward mammals increased as the water solubility of alcohols decreased. Shortly after, Crum-Brown and Fraser (1868) reported that there was a correlation between chemical substituents and their physiological properties. Later in the 1890s, Hans Horst Meyer reported that the toxicity of organic compounds depended on their lipophilicity (Borman 1990; Lipnick 1991). Subsequently, the linear correlation between lipophilicity (e.g. oil-water partition coefficients) and biological properties was investigated. Louis Hammett (Hansch et al. 1991) investigated the relationship between electronic properties of organic acids and bases with their equilibrium constants and reactivity. These early studies form the basis for the development of modern QSAR by establishing the idea that molecular structures directly influenced the endpoint (i.e. biological activity and chemical property) of interest. In 1962, Hansch et al. (1962) formally coined the term QSAR and laid its initial foundations by investigating the structure-activity relationship (SAR) of plant growth regulators and pesticides and their dependency on Hammett constants (Hammett 1937) and hydrophobicity (Gallup et al. 1952).

The Free-Wilson model (Free and Wilson 1964) is a simple and efficient method for the quantitative description of SAR. It explains the variation in a series of congeneric compounds using the presence or absence of substituents or functional

groups as molecular descriptors. It is the only numerical method that directly relates structural features with biological properties, which is in contrast to Hansch analysis where physicochemical properties are correlated with biological activity values (Kubinyi 1988). Nevertheless, both approaches are closely interrelated, not only from a theoretical point of view but also in their practical applicability (Kubinyi 1988). In many cases both models were combined to a afford a mixed approach that includes Free-Wilson type parameters for describing the activity contributions of certain structural modifications and physicochemical parameters for describing the effect of substituents on the biological activity (Kubinyi 1988; Wei et al. 2001). Many successful applications, especially from the work of Hansch and his group (Verma and Hansch 2009; Hansch et al. 2002; Kurup et al. 2000; Gao et al. 1999; Selassie et al. 2002; Kurup et al. 2001; Hansch and Gao 1997; Kurup et al. 2001; Hansch et al. 1996; Hadjipavlou-Litina et al. 2004; Garg et al. 1999, 2003) on the SAR of enzyme inhibitors, demonstrated that this combined model affords stellar performance for classical QSAR (Hansch 2011). Several variations to Free-Wilson approach have been developed and recently found useful applications in fragment-based drug design (Eriksson et al. 2014; Chen et al. 2013; Radoux et al. 2016).

The field of QSAR modeling had evolved progressively and this encompasses two radical transformations as follows:

1. Paradigm shift from the *classical* to the *non-classical* QSAR approach (Fujita and Winkler 2016). The former is based on a small set of congeneric series of compounds that usually have a single mode of action while the latter is based on large, heterogeneous and non-congeneric data set that may contain several mode of actions.
2. Paradigm shift of QSAR models (Nantasenamat et al. 2009, 2010; Cherkasov et al. 2014) that considers the SAR of *several compounds against a single target protein* to the so-called proteochemometric model (Cortes-Ciriano et al. 2015; Qiu et al. 2016) (sometimes referred to as computational chemogenomics) that investigates the SAR of *several compounds against several target proteins*.

3 How Far Can QSAR Take Us: Can It Really Bring a Drug to Market?

QSAR modeling have evolved from concept to initial hype followed by skepticism thereby leading to the identification of their pitfalls and caveats to a moderation of their expectations (Doweyko 2008). QSAR models are routinely used in the prediction of physicochemical properties (e.g. log$P$, pK_a and solubility) as well as pharmacokinetic and toxicity endpoints (e.g. permeability, plasma protein binding, liver toxicity, carcinogenicity, seizure and off-target activities). However, their usage for actual lead identification and optimization phase has remained quite limited. The skepticism from medicinal chemists towards QSAR models stems from the inability of descriptor based QSAR models (constructed using fingerprints and various

topological descriptors) to rationalize activities in terms of simple, meaningful and constructive ways that can clearly provide details on what modifications should be made to the chemical structure that can afford activity enhancement. Furthermore, with better ability to assimilate data from human readable patents and publications of SAR data in concomitant with better understanding of the isosteric concept, medicinal chemists are better able to capture the underlying principles of SAR and make synthetically feasible and conservative predictions. However, many encouraging signs are beginning to appear as more robust machine learning algorithm and interpretable molecular descriptors are being developed. It is still early to predict the potential of QSAR modeling for bringing a drug to market since they are used in the early stages of a drug discovery project. With the ever increases in the availability of clinical and adverse effect data, the use of QSAR modeling together with complementary computational approaches (e.g. cheminformatics, computational chemistry, molecular docking, molecular dynamics, etc.) helps improve the odds of bringing a drug to market. QSAR modeling in combination with other computer-aided drug design techniques have already shown numerous success stories as summarized in an excellent report by Kubinyi (2006).

3.1 Why Does QSAR Fail?

QSAR modeling, like many other research disciplines, has had its fair share of ups and downs. Many predicted the eventual demise of QSAR due to the advances in synthetic chemistry techniques (e.g. combinatorial chemistry) and assay attributes (e.g. automation and miniaturization). Drug discovery researchers dissolution with QSARs is rooted in the fact that it has yet to demonstrate a robust ability to predict the desired biological activities. The disappointing results from QSAR models in certain situation can be attributed to features obtained by chance correlation, rough response surfaces, incorrect functional forms and overtraining (Johnson 2008; Doweyko 2008). Particularly, rough response surfaces are an inherent characteristic of SAR data sets that nevertheless significantly affect the QSAR model predictions. For instance, most aminergic GPCR ligands' agonistic activities correlate with their pK_a and in many instances an order of magnitude change in the pK_a results in a comparable or even an multi-fold change in the biological activity. Such conservative change in the chemical structure leading to a large change in the activity are often not captured by QSAR models which rely heavily on statistical approaches to capture the features that cause the biological responses. On the other hand, a chemist quickly grasps the trend using rational thought, controlled experiments and personal observation assisted by prior knowledge of the protein's structure-function relationships. This over-reliance on statistical procedures by QSAR researchers for feature selection and data modeling has led to the identification of features that may have no mechanistic role in modulating the activities but might have correlated by chance. The excessive emphasis on machine learning has also resulted in model overfitting, models that uses the incorrect functional forms and/or highly predictive

models with vague or little interpretability. Hence, the resulting QSAR models do not reflect the reality of the binding or modulation event, which causes the predictions to eventually fail. Thus, to derive meaningful hypothesis, practitioners should not blindly rely on results from computational models but should view the results as hints or guides for supporting their own decision-making process (Nantasenamat and Prachayasittikul 2015). Thus, it is recommended to implement some form of expert knowledge guided component in the QSAR workflows such that new solutions are built upon prior knowledge of targets and their modulation (Saxena and Prathipati 2003). In fact, such data-driven approach as implemented in the HADDOCK docking software (Vries et al. 2010) relies on prior biochemical and biophysical data to drive the docking simulations. Moreover, several recent blinded genomic challenges for phenotype prediction such as sbvImprover (Tarca et al. 2013) and DREAM (Costello et al. 2014) also suggests that the inclusion of prior knowledge can significantly enhance the predictive power while consuming minimal computational resources. In this context, the use of interpretable molecular descriptors aided by transparent machine learning models can greatly alleviate the existing problems of QSAR models.

4 Recommendations for Building Robust QSAR Models

In practice, the development of QSAR models can be carried out to reveal the relationship between the chemical structures and their respective endpoint through the use of various types of mathematical and statistical methods for constructing predictive models that can reveal the origin of bioactivity of interest. A typical $m \times n$ data matrix is comprised of m descriptors and n compounds. A closer look at the M descriptors revealed that it is typically comprised of a set of $\mathbf{X}_{ij}$ descriptors and an $\mathbf{y}_i$ endpoint. In a nutshell, a typical QSAR model is essentially described by an equation the form of $\mathbf{Y} = \mathrm{f}(\mathbf{X}) + error$ that can be used to predict the endpoint for new compounds in lieu of cost and time-consuming approaches. The classical QSAR modeling workflow can be broken down into five prime steps as demonstrated in Fig. 1.

Thus far, several thousands of QSAR models have been developed for various endpoints and these models are created using different model construction schemes (e.g. stringency of data pre-processing, descriptor types, learning methods and evaluation metrics) and published in the public domain (i.e. this is not including the thousands of QSAR models developed in pharmaceutical companies that are not ever published). The variability in the methods used for the QSAR models and their quality may obviously give rise to different outcome for the conclusions possible to draw from them. To further complicate the picture, the reproduction of QSAR models by following the often rather vague instructions in the Methodology sections of research articles do not always yield the same outcome as in the original article owing to the aforementioned factors.

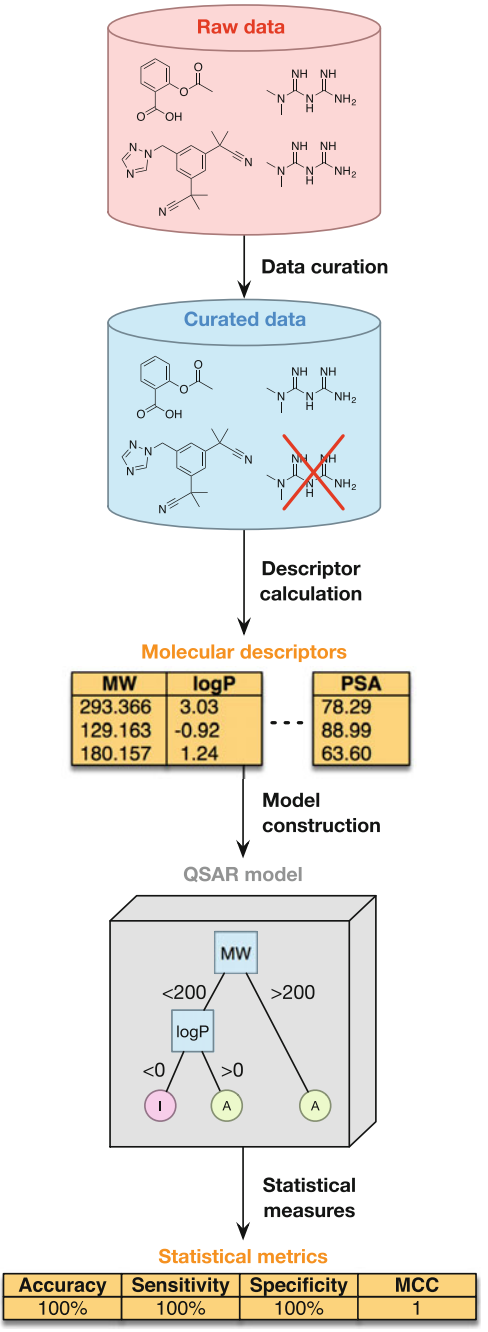

Fig. 1 General workflow of QSAR modeling. Raw data compiled from the literature or public databases are often noisy and dirty and therefore requires curation to clean the data. In this example, redundant chemical structure is removed followed by descriptor calculation, model building and model performance evaluation

Table 1 Summary of the OECD principles for QSAR modeling

No.	OECD principles	Description
1	Defined endpoint	To ensure that all endpoint values within a given data set are consistent
2	Unambiguous algorithm	To ensure transparency and reproducibility of the proposed QSAR model
3	Defined applicability domain	To determine the boundaries in which the model is robust for predicting query compounds
4	Measures of model's predictive potential	To evaluate the internal and external predictive power of the model
5	Mechanistic interpretation	To ensure that the underlying mechanism of action of compounds can be elucidated

Thus, owing to such lack of standards in QSAR/QSPR modeling, the OECD principles was established to address such issues. This first draft initially took place in Setubal, Portugal in 2002 and a revised version in Paris, France in 2004 at the *Workshop on Regulatory Acceptance of QSAR Modelling for Human Health and Environmental Endpoints* and *37th Joint Meeting of Chemicals Committee and Working Party on Chemicals, Pesticides & Biotechnology*, respectively (Worth and Cronin 2004). It has been mandated that to facilitate the consideration of a QSAR model for regulatory purposes, the model should conform to the five principles summarized in Table 1.

Moreover, the integrity of a QSAR model could be pursued by following suggested sets of standards and best practices (Dearden et al. 2009; Tropsha 2010; Tropsha et al. 2003; Dimova and Bajorath 2016; Spjuth et al. 2010) in the development of robust QSAR models. Particularly, Tropsha et al. stressed the importance of leave-many-out validation, bootstrapping, Y-scrambling test and external validation. Moreover, conflicting viewpoints exist on whether to evaluate the robustness of QSAR models on the basis of external validation in which Hawkins et al. (2003) is against this while Esbensen and Geladi (2010) is in support of this. Moreover, recent investigations clearly favor cross-validation over a single external one (Gütlein et al. 2013; Rácz et al. 2015).

In a nutshell, the development of robust QSAR models should address the following key issues:

1. *Data curation*—The curation or pre-processing of data sets prior to performing any form of data analysis is of utmost importance for QSAR modeling. Raw data sets are often *noisy* or *dirty* in the sense that they may inherently contain redundant compounds, redundant descriptors, incorrect representation of the chemical structure or molecular charge. Curation helps to *clean* and increases the reliability of the data set for subsequent analysis.

2. *Modelability*—Modelability is an a priori estimate of the feasibility to obtain externally predictive QSAR models. Modelability is based on the fact that QSAR models are influenced either by data set characteristics (i.e. size, chemical diversity, activity distribution, presence of activity cliffs, etc.), or by modeling workflow steps (e.g. data set curation, feature selection, external validation, consensus modeling, applicability domain, etc.). Particularly, influences arising from the composition of the modeling workflow can be quantified and can be varied given the wide range of molecular descriptors and machine learning methods that are available. However, effects of data set characteristics can be rather difficult to quantify. While size and chemical diversity are subjective attributes of a data set and are difficult to quantify, recent advances have provided methods for objective quantification of activity cliffs (Guha and Drie 2008; Seebeck et al. 2011; Bajorath 2014; Stumpfe et al. 2014; Hu et al. 2012). Building on the earlier proposed concept of the activity cliffs, Golbraikh et al. (2014) proposed a novel modelability index (MODI) that can be easily computed for any dataset at the onset of any QSAR investigation.
3. *Reproducibility*—This important issue is often overlooked by the QSAR community. This is particularly true as often times, QSAR models are built using proprietary software or code that are often restricted to a selected few and not accessible to the general public thereby precluding further attempts to make use of these models. Moreover, the reproduction of QSAR models is a very difficult task indeed as the construction of QSAR models employs different data sets (e.g. different version of the same bioactivity databases such as ChEMBL 19, 20 or 21; it is also highly likely that data sets focused on the same target protein and performed by different laboratory tend to contain different compounds as they may be compiled from different papers), descriptor types, learning methods and evaluation metrics. Spjuth et al. (2010) examines this issue by proposing an open XML format known as QSAR-ML to formalize QSAR data sets with meta-data, which will facilitate the exchange and reproducibility of the model.
4. *Model validation*—The robustness of QSAR models is reliant on stringent validation of QSAR models. Several validation strategies including (1) randomization of the modelled property also known as Y-scrambling, (2) *k*-fold cross-validations and (3) external validation using rational division of a data set into training and test sets are currently the de facto standard for ensuring the utility of a model for virtual screening (Tropsha et al. 2003).
5. *Outliers*—Outlying compounds are those molecules which have unexpected biological activity and do not fit in a QSAR model owing to the fact that such compounds may be acting in a different mechanism or interact with its respective target molecules in different modes (Nantasenamat et al. 2009; Verma and Hansch 2005). Similarly, conformational flexibility of target protein binding site (Kim 2007a) and unusual binding mode are attributed as the possible source of outliers (Kim 2007b). Mathematically speaking, an outlier is essentially a data point that has high standardized residual in absolute value when compared to the other samples of the data set. Furthermore, the building of robust and reliable QSAR models generally emphasizes two major aspects: (1) feature selection and

(2) outlier detection. The two problems are interrelated as outlier definitions are dependent on the selected features. In the realm of QSAR, outliers can be classified as belonging to the following two types: (1) those that fall outside the applicability domain or (2) activity cliffs as discussed in the next section. As the applicability domain considers both chemical and biological space, therefore outliers with respect to biological space can be safely eliminated from QSAR models. However outliers defined based on the chemical space needs further attention. Recent methods such as those from Cao et al. (2011) have argued in support for simultaneously performing variable subset selection and outlier detection using the idea of statistical distribution that can be simulated by the establishment of many cross-predictive linear models. Their approaches build on the concept that the distribution of linear model coefficients provides a mechanism for ranking and interpreting the effects of variables while the distribution of prediction errors provides a mechanism for differentiating the outliers from normal samples (Cao et al. 2011).

6. *Applicability domain*—The applicability domain (AD) (Sahigara et al. 2013) of a QSAR model defines the model limitations with respect to its structural subspace and response space. AD is an indication of the degree of generalization of a given predictive model. AD associated with an endpoint prediction is often well defined if the endpoint prediction for a chemical structure is within the scope of the model. The AD is thus critically reliant on the sampling of chemical subspace and the range of biological readouts that are used for the model development (Sheridan 2015). A commonly overlooked aspect in AD is also the influence of molecular descriptors, generally degenerate and transparent molecular descriptors such as logP, pK_a, etc. afford better degree of generalization to the model while lacking the superior predictive abilities of the more recent topological graph-based descriptors. The various approaches for AD determination are classified as range-based (e.g. bounding box, principal component analysis bounding box and convex hull) and geometric methods (e.g. *k*-nearest neighbours, DTs, probability density based methods) (Sahigara et al. 2012).
7. *Structure-activity cliffs*—Compounds within a congeneric series whose subtle differences in the chemical structure lead to striking differences in the observed bioactivity are called activity cliffs (Bajorath 2014). Although, the activity cliffs are appealing to medicinal chemists their presence may be detrimental to QSAR models. The inclusion should be carefully reviewed after analyzing for filters such as PAINS (Baell and Holloway 2010) as unusual activity could be due to a wide range of mechanisms such as outliers of different kinds or even the presence of reactive functional groups (Saxena and Prathipati 2006). However, these compounds belonging to the *activity cliffs* are currently categorized as outliers and frequently removed from QSAR models (Guha and Drie 2008). The MODI quantifies the extent of activity cliffs and serves as a guide to the modelability of a data set (Golbraikh et al. 2014).
8. *Feature selection*—The number of molecular descriptors that can capture various aspects of a chemical structure have proliferated in recent years (Todeschini and Consonni 2008). Hence, feature or variable selection is an important and hot

area of research (Guyon 2003; Eklund et al. 2014; Goodarzi et al. 2013). In the context of QSAR studies, feature selection improves interpretability by neglecting non-significant effects thereby reducing noise, enhancing generalization by reducing overfitting (also known as reduction of variance), increasing the models' predictive ability and speeds up the QSAR model building process (Saxena and Prathipati 2003). Some widely used and relevant approaches for QSAR studies includes: (1) all subset models (ASM), (2) sequential search (SS), (3) stepwise methods (SW), (4) genetic algorithm (GA), (5) particle swarm optimization (PSO), (6) ant colony optimization (ACO), (7) least absolute shrinkage and selection operator (LASSO), (8) elastic net and (9) variables importance on PLS projections (VIP) (Eklund et al. 2014), (10) correlation-based feature selection (CFS) (Hall 1999), (11) simulated annealing (Siedlecki and Sklansky 1988), (12) sequential feature backward selection (Pudil et al. 1994), (13) sequential feature forward selection (Pudil et al. 1994), (14) minimum-redundancy-maximum-relevance (mRMR) (Peng et al. 2005), (15) ReliefF (Liu and Motoda 2007), (16) Tikhonov regularization (Destrero et al. 2009), (17) recursive feature elimination (RFE) (Guyon et al. 2002), (18) random forest (RF) (Breiman 2001), (19) decision tree (DT) (Quinlan 1993), etc.

9. *Class imbalance*—Class imbalance in supervised machine learning is a major confounding problem for the construction of QSAR models (Li et al. 2009). In a classification setting, the size of the active and inactive sets of compounds may be significantly disproportional and may therefore lead to biased predictive models. Several solutions that include artificially undersampling the overrepresented class or oversampling the underrepresented class or using one class learning or cost-sensitive training have all been suggested as possible remedies to address this issue (Zakharov et al. 2014; Capuzzi et al. 2016).
10. *Chance correlation*—Objectivity is a critical component of any hypothesis generating workflow including QSAR. It has been stressed that causation and correlation are indeed two different things and that a model's performance may possibly arise by chance. A possible remedy is to apply Y-scrambling (Rucker et al. 2007) to evaluate model robustness.
11. *Confidence/reliability of the model*—QSAR models are not universally applicable as predictions may fail under certain conditions. QSAR models are based on mathematical formulations for modeling the bioactivity as well as to draw conclusions from. Their utilization in medicinal chemistry encompasses idea generation, virtual screening and knowledge discovery. Hence, the confidence in the predictions derived from QSAR model should be accessible. Substantial efforts have been devoted to research on this topic within the QSAR community over the last decade and a number of methods have been suggested for estimating the confidence of QSAR predictions. These confidence estimates are typically based on the very loosely defined concept of a QSAR models applicability domain (AD), which is described as the response and chemical structure space in which the model makes predictions with a given reliability. The assumption is that the further away a molecule is from a QSAR models AD,the less reliable the prediction becomes. This confidence measure can be afforded by an approach

known as conformal prediction (Shafer et al. 2008), which has been successfully applied in QSAR modeling (Eklund et al. 2012). The conformal prediction framework provides a unified view of the different approaches for estimating a QSAR models AD. Moreover, conformal prediction provides a natural and intuitive way of interpreting the AD estimates as prediction intervals with a given confidence.

12. *Interpretability of the model*—Perhaps, the most important contribution of QSAR modeling lies in their ability to propose a hypotheses to rationalize the binding/function modulation phenomenon via interpretation of the model's features. In view of its critical role in fulfilling the objectives of QSAR modeling, we focus our chapter on their interpretability. The hypothesis gleaned from QSAR models can benefit biologists and chemists by providing insights into the cause-effect relationships between molecular features and bioactivity measures. These insights can aid medicinal chemists to design future SAR studies objectively and comprehensively. They can also assist molecular and structural biologists in proposing candidates for site-directed mutagenesis and related structure-function experiments. This chapter proposes the use of interpretable molecular descriptors together with interpretable machine learning methods. Recent interest in the field had also shifted towards making the black box learning methods more transparent and amenable to interpretations, which will be covered in the forthcoming sections.

5 Trade-Offs Between Performance and Interpretability

Over the past decades, many QSAR studies had predominantly focused on enhancing and improving the predictive performance instead of the interpretability of the model (Fujita and Winkler 2016). The shift can be seen in QSAR model descriptors moving away from the physicochemical and indicator variables of Hansch-Fujita and Free-Wilson approaches towards highly non-degenerate and continuous molecular descriptors which offer high predictive power. However, improved understanding of the concepts of bioisosterism and the molecular recognition events, identification of problems associated with capturing molecular structures and errors in assay data of widely used SAR databases give credence to the use of moderately degenerate and interpretable 1D or fingerprint based molecular descriptors as expanded elsewhere in this chapter. Learning methods in QSAR modeling have evolved from simple interpretable methods such as linear regression as used by Hansch and Fujita to the complex black box approaches such as neural networks and deep learning. While many experts agree with the obvious improvements (i.e. approximately 10%) to the predictive power from these complex machine learning methods, they argue that the loss of interpretability of the feature contributions are not worth the gain in predictive power. Hence, in Sect. 8.1.4 we expand upon the recent advances in rule extraction techniques that help to provide enhanced interpretation of the complex black box approaches. This section also presents several recent enhancements that

significantly improve the predictive power of the white box learning approaches. Hence, this chapter presents and advance the case for interpretable QSAR models in drug discovery research. We argue that a simple and interpretable QSAR model with modest predictive performance would be more valuable to experimental scientists than a highly predictive but black box model since no or minimal insights can be gained from it.

6 Reverse Engineering of QSAR models

Designing new molecules corresponding to the given biological activity is invaluable to the chemical, material and pharmaceutical industries. The traditional approaches of computer-aided molecular design based on QSAR modeling can be used to solve two main problems: (i) *forward QSAR problem*, which identifies the compounds' structural and physicochemical features related to the experimental readout using machine learning (ii) *inverse QSAR problem* that seeks to reconstruct compounds' structures which correspond to the specific features related with the readout (Faulon et al. 2005; Brown et al. 2006).

The inverse problem is generally addressed as a subgraph construction. Previously, there were five types of approaches to solve the inverse problem: random search, heuristic enumeration, mathematical programming, knowledge-based system, and graphical reconstruction methods. The inverse QSAR analysis is quite challenging for various reasons: combinatorial complexity of the search space, design knowledge acquisition difficulties, nonlinear structure property correlations, and problems in incorporating higher level chemical and biological knowledge (Venkatasubramanian et al. 1995). Thus, it is not surprising that constructing new structural compound given a desired activity is a long-standing problem. In practice, the inverse QSAR method can be divided into the common four steps (Skvortsova et al. 1993; Wong and Burkowski 2009; Churchwell et al. 2004; Visco et al. 2002; Weis et al. 2005). Firstly, a QSAR equation is constructed to derive a forward QSAR model that essentially discerns the relationship between a set of descriptors and their activities. The second step is to generate the set of constraint equations with integer coefficients. The constraints are used for ensuring that the constructed compounds afford the desired activities. There are two types of constraint equations: graphical and consistent equations, which are then solved in the third step. Finally, the compound structures are enumerated and constructed to afford the desired activity while their activities are predicted using the forward QSAR model described in the first step.

Until now, there are relatively few studies providing computational-based models for solving this problem (Visco et al. 2002). Almost all of the proposed computational-based methods that are used are essentially a stochastic model in nature and use either genetic algorithm (GA) or Monte Carlo simulated annealing approach to construct new chemical compounds. In 1995, Venkatasubramanian et al. (1995) and Sheridan and Kearsley (1995) proposed a stochastic model based on Monte Carlo. GA is a general purpose approach based on the Darwinian

principle for natural selection and evolution, which are used for stochastic, evolutionary search, and optimization strategies. The main advantage of GA lies in its ability to allow a dynamically evolving population of molecules to gradually improve by competing for the best performance. However, the problem from these studies represent a combinatorial explosion (Kvasnicka and Pospichal 1996). In order to analyze a huge number of compounds, Kvasnicka and Pospichal (1996) developed a new approach based on a random search that not only afford all solutions but also provide users with a high probability of deriving the correct solution. In 2002, Visco et al. introduced the use of signature descriptors to represent compounds as molecular graphs. In this study, a set of 121 HIV-1 protease inhibitors were analyzed by comparing the proposed QSAR model with other descriptor types consisting of connectivity indices, KierHall shape indices, fragments, electrotopological states and information indices. This work also revealed that signature descriptors are particularly well suited for tackling the inverse problem (also see the work from Faulon 1994, 1996; Faulon et al. 2003; Churchwell et al. 2004; Faulon et al. 2004; Weis et al. 2005). Also from the same group, Churchwell et al. (2004) applied the inverse QSAR approach to a small set of peptide inhibitors that targets the leukocyte functional antigen-1 (LFA-1)/intercellular adhesion molecule-1 (ICAM-1) complex. Their prediction results showed that the predicted IC_{50} values were very close to that of the experimental IC_{50} values. Practically, the inverse QSAR problem is relatively difficult when compared to the forward QSAR problem because the molecular descriptors used for constructing the inverse QSAR model must adequately address the forward QSAR model for the activity or property of a given data, if the subsequent recovery phase is to be meaningful. Additionally, a major problem is to reconstruct and enumerate the chemical structures from its extracted descriptors. To solve such problem, Wong and Burkowski proposed (Wong and Burkowski 2009) a new workflow using a vector space model molecular descriptor (VSMMD) to represent the chemical structures. Their proposed inverse QSAR model consists of five key steps: (i) calculating the VSMMD for each compound from the training set; (ii) apply the kernel function (i.e. more detail is discussed in a subsequent section) to map each VSMMD from the input space (i.e. low dimension) to the feature space (i.e. high dimension); (iii) designing a new point in the feature space using a kernel function algorithm; (iv) map the new point from the feature space and trace back to the input space using a pre-image approximation algorithm and (v) building the chemical structures using the VSMMD recovery algorithm.

As can be seen, inverse QSAR models has great potential for obtaining desirable compounds directly from the trained QSAR model. Further work in this area is highly encouraged as to help steer towards the practical utility of QSAR models for building promising chemical structures aside from making predictions of their bioactivity values or class label.

7 Interpretable Molecular Descriptors

7.1 Role of Molecular Descriptors in Post-genomic Drug Discovery

Molecular descriptors encode the physical and chemical properties of molecules of interest and are central to QSAR/QSPR studies (Danishuddin 2016). The availability and the use of high quality, interpretable descriptors can greatly contribute to the formulation of an intuitive model for retrospective and prospective analysis of life or material sciences data (Cherkasov et al. 2014). As depicted in Fig. 2, molecular descriptors play a critical role in enabling mathematical and statistical analysis for relating chemical structure with biological data. While human intuitive molecular graphics depictions use the atom, bond, angle coordinates together with charge

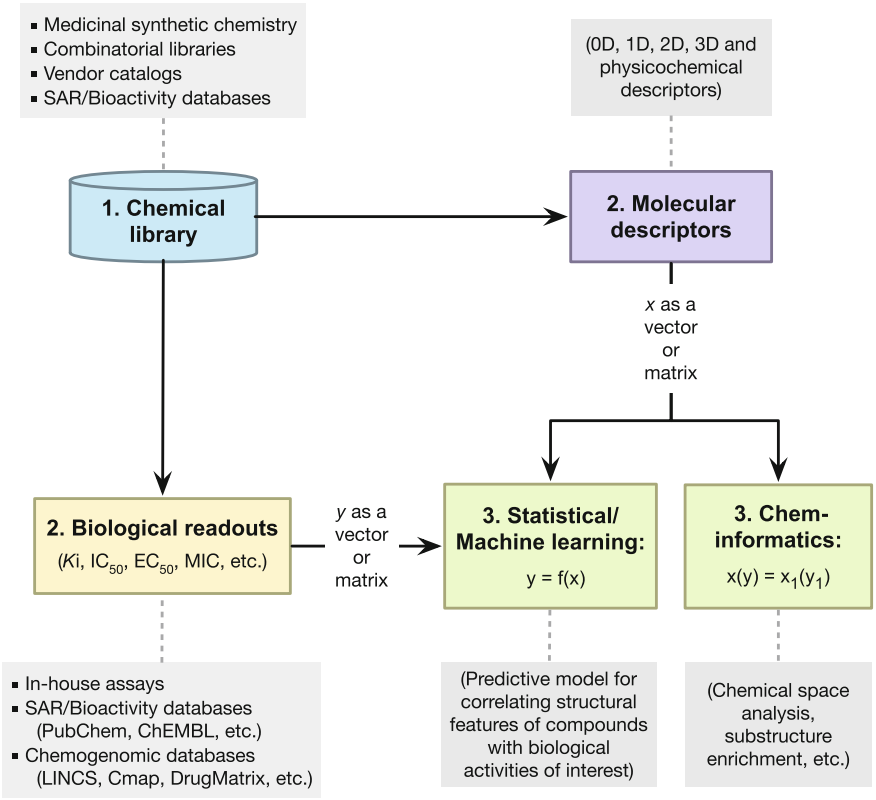

Fig. 2 General schematic diagram depicting the importance of molecular descriptors for capturing the details of chemical structures (from a chemical library) as vectors and matrices; hence enabling mathematical and statistical procedures for QSAR and other chemoinformatics analysis

information to reconstruct the chemical structures as 2D and 3D projections, encoding chemical structure as machine readable matrices and vectors is required for performing mathematical and statistical analysis. In this regard molecular descriptors play a key role in establishing QSARs and in performing chemo-informatics tasks such as chemical space mapping, substructure analysis, etc. In the pre-genomic era, biological readouts were available as single vectors, however advances in miniaturization, robotics and automation in the post-genomic era presented QSAR researchers with a complex array of biological data as matrices. The complex biological matrices include both the traditional target and phenotypic measurements and the recent clinical chemistry and histopathology findings and microarray and proteomics data (Prathipati and Mizuguchi 2016a). These data were generated in standardized high-throughput format and are available in databases such as LINCS, Open TG-Gates, CEBS, DrugMatrix and CMap (Prathipati and Mizuguchi 2016a). Several advanced multi-label statistical techniques (such as network-based inference) and complex molecular descriptors (such as proteo-chemometric) are presently under development which can capture both the biological data's relationship with the chemical structure together with complex relationships among the biological readouts and the chemical structures (Prathipati and Mizuguchi 2016a). Thus a range of machine learning methods are under consideration for multi-label QSAR models depending on the data types such as support vector machines (SVMs), neural networks (NN), *k*-nearest neighbors (*k*NN), boosting methods for unrelated multi-label datasets and similarity based approaches such as DT-hybrid, kernel regression methods such as lasso or elastic nets or pairwise kernel method (PKM) for related multi-label datasets (Prathipati and Mizuguchi 2016a). While some of these machine learning methods are discussed in Sect. 8, in the following subsections we expand upon the range of molecular descriptors and their attributes and their utility for modelling the wide array of biological readouts.

7.2 Interpretability of Molecular Descriptors Advances Ligand-Based Approaches

The continuing appeal of QSAR models as part of ligand-based approaches in the face of the ever increasing structural data of target proteins and advancements in structure-based approaches is an interesting conundrum (Prathipati and Mizuguchi 2016a). Although structure-based approaches are highly interpretable and intuitive to drug researchers, their efficiency and effectiveness is limited by several factors including ambiguity in pose prediction, limitations of scoring functions at capturing the molecular recognition event, limitations of existing methods in considering bridging water molecules and induced fit phenomenon (Prathipati et al. 2007; Prathipati and Mizuguchi 2016b). Furthermore, drug targets such as nuclear receptors, G protein-coupled receptors (GPCRs) and kinases are known to have multiple conformational states that exists in equilibrium in the absence of their cognate

ligands (Spyrakis and Cavasotto 2015; Zhao et al. 2014; Rueda et al. 2009, 2010). Most often the X-ray structures of one or the other of these conformational states are difficult to obtain. For instance, several kinases are known to exist in at least 4 different conformational states (e.g. DFG-in, DFG-out, A-loop-out and A-loop-in) in recognizing type -I, -II and -III inhibitors (Chiu et al. 2013). The DGF-out inactive conformational state of a kinase is quite flexible and is quite difficult to crystallize where the catalytically important p-loop is most often difficult to resolve (Kufareva and Abagyan 2008). Similarly, GPCRs too exist in the active, inactive and apo conformational states. While the inactive GPCR conformational states are easy to crystallize owing to its rigidity as conferred by the strong salt-bridge interactions between the helices (e.g. helices 3 and 6 and helices 2 and 5), the active conformational state stabilized in the presence of an agonist disrupts these interactions through charge neutralization, hence becomes flexible and is difficult to crystallize and resolve (Standfuss et al. 2011). Conversely, ligand-based QSAR models are quick and can be dynamically adapted to model both target and phenotypic endpoints as well as different types of chemotypes with relatively little effort (Prathipati and Saxena 2005). QSAR models derived using molecular descriptors were shown to provide high predictive power and were successfully used for hit identification (Krasavin 2015; Geronikaki et al. 2008; Poroikov et al. 2003). The disadvantages of this approach is their comparatively low intuitiveness and their difficulty for interpretation (Saxena and Prathipati 2006). Hence, we shall attempt to discuss the pros and cons of various descriptors in terms of their quality and interpretability.

7.3 Assessing the Quality and Interpretability of a Molecular Descriptor

Historically, the Hammett equation (Hammett 1937) describes one of the earliest known mathematical formulations relating structures with the property of interest (i.e. reactivity in this instance) and remains the most widely used and understood mathematical equation to date. It describes a linear free-energy relationship relating rate or equilibrium of a reaction with a substituent's position and electronic property (i.e. withdrawing or donating) captured as 'Sigma' (Hammett 1937). The molecular descriptor 'Sigma' as proposed by Hammett (1937) to explain the acidity of substituted benzoic acids also serves as useful guidepost in evaluating the quality and interpretability of a molecular descriptor. 'Sigma', also called the substituent constant, has several features that makes it an excellent molecular descriptor, particularly it has (1) high structural interpretation, (2) good correlation with biological or physical property (i.e. pK_a in this case), (3) can be applied to local structure (substructures), (4) uses the familiar structural and electronic concepts (e.g. electronegativity and polarizability), (5) high sensitivity (i.e. varies with structures; even isomers) and (6) size dependence (i.e. changes with molecular weight). However, the original implementation of Hammett involves using experimental properties and makes the

Table 2 Summary of the strengths and weaknesses of the various dimensions of molecular descriptors. The number of stars denotes the strengths and weaknesses for each characteristics while the exclamation mark designate that caution should be taken

Characteristics	0D	1D	2D	3D	PC
Simplicity	★ ★ ★	★ ★ ★	★★	★	★ ★ ★
Calculation efficiency	★ ★ ★	★ ★ ★	★★	!	★
Structural interpretation	★	★★	★	★ ★ ★	★ ★ ★
Correlation with biological property	★	★★	★★	★ ★ ★	★ ★ ★★
Applicable to local structure (substructures)	★	★ ★ ★	★★	★	★★
Use familiar structural and electronic concepts	★	★	★★	★ ★ ★	★ ★ ★★
Sensitivity (discriminate different structures including isomers)	!	★	★★	★ ★ ★	★★
Size dependency (varies with MW)	★	★★	★★	★★	★★

0D: zero-dimensional descriptors, 1D: one-dimensional descriptors,
2D: two-dimensional descriptors, 3D: three-dimensional descriptors,
PC: physicochemical descriptors

computation of sigma highly inefficient and hence not practical for high-throughput virtual screening workflows. We shall discuss the importance of physicochemical properties descriptors and the 4 major class of structural descriptors in light of the features discussed above (Table 2). Furthermore, several novel applications of QSAR such as the modelling of peptides, nucleotides and nanostructures for biologics-based drug discovery research requires the availability of novel descriptors. Hence, Table 4 presents the list of free software along with availability of various descriptor types (Table 3).

7.4 Trade-Offs Between Descriptor Quality and Interpretability

Thus, it should be noted that a descriptor's quality and its interpretability, together with the use of an appropriate machine learning method can greatly produce a practical and interpretable QSAR model that scientists can use. The *sensitivity* or the *degeneracy* of a molecular descriptor is the measure of its ability to avoid equal values for different molecules. This is the most critical attribute of a descriptor's quality. Furthermore, a descriptor's interpretability can be defined as its ability to elucidate and rationalize the underlying structural and physicochemical properties responsible for the biological response.

3D descriptors which most accurately encode the structural and physicochemical properties that are responsible for the investigated endpoint are presently regarded to afford robust quantitative descriptions of molecular structures. They have high

Table 3 Summary of model techniques used in QSAR modeling and their advantages and disadvantages

Method[a]	Interpretable	Linear	Supervised learning?	Advantage(s)	Disadvantage(s)
MLR	Yes	Yes	Yes	Good interpretability	Problem of learning dichotomous variables
LR	Yes	Yes	Yes	Deal with dichotomous variables	Perform poor on complex data
ELM	Yes	Yes	Yes	Interpretability	Perform poor on multiclass data
PCA	Yes	Yes	No	Dimension reduction	Unsupervised learning
PLSR	Yes	Yes	Yes	Interpretable and reducted dimension	Linear model
DT	Yes	No	Yes	High interpretability	Overfitting
RF	Yes	No	Yes	High interpretability/tolerant to overfitting	Long training time
ANN	No	No	Yes	Perform well on complex data	Poor interpretability
DL	No	No	Yes	Hierarchical features learning	Poor interpretability
SVM	No	No	Yes	Good generalization performance	Poor interpretability

[a]MLR: multiple linear regression, LR: logistic regression, ELM: efficient learning method,
PCA: principal component analysis, PLSR: partial least squares regression,
DT: decision tree, RF: random forest, ANN: artificial neural network, DL: deep learning,
SVM: support vector machine

sensitivity and present different values of different isomers and other subtle structural variations. Some 3D descriptors such as those based on the GRID concept or obtained from quantum chemical computations provide causal insights while those based on the graph concept akin to the 2D graph-based descriptors present very little causal interpretation. Furthermore, 2D graph-based descriptors are equally as degenerate as a 3D descriptor and can also be regarded as a descriptor of high quality. However, most medicinal chemistry SAR data are not highly sensitive to small changes in the structure (i.e. the addition of substructures to non-pharmacophoric areas) and are shown to have moderate complexity (Schuffenhauer et al. 2006). Furthermore,the assay data too are prone to experimental artifacts (e.g. aggregation, reactive functional groups induced assay readouts) and errors (i.e. standard deviation of technical replicates) (Feng et al. 2005; Feng and Shoichet 2006; Feng et al. 2007; McGovern et al. 2002; Thorne et al. 2010). The moderate complexity of the chemical space can be attributed to the difficulties in their synthesis and purification as well as the characterization of stereo- and regioisomers.

In light of the moderately complex chemical space, 1D or fingerprint descriptors having moderate sensitivity (e.g. non-degenerativity) and interpretability, have become the de facto standard in chemoinformatics both for a prospective and retrospective QSAR analysis (Schuffenhauer et al. 2006). The compact nature of the bit-vector representation makes them amenable to not only QSAR modeling but also for a wide range of computations such as similarity searching (Prathipati et al. 2008), clustering (Prathipati et al. 2008), substructure searching and the inverse QSAR problems (Rosenbaum et al. 2011). As to address issues such as assay errors, artifacts and heterogeneity of assay methods, the use of classification models has been proposed as a promising solution and as such its usage has steadily increased in recent years.

7.5 Dimensions of Molecular Descriptors

7.5.1 0D Descriptors

The 0D descriptors (Todeschini and Consonni 2008) capture the counts of atoms (e.g. number of carbon atoms, number of nitrogen atoms, etc.) and bonds as well as their constitution (e.g. hybridization states and bond orders). In addition, 0D descriptors also encode the sum or average of the atomic properties such as weight, volume, polarizability, electronegativity, etc. These descriptors are easily calculated and naturally interpreted but they may not be very sensitive to subtle changes in molecular structures (e.g. isoforms). However, this class of descriptors have successfully been used in explaining the variation effect of structures on activity/property of several data sets as has extensively been shown by the research group of Andrey Toropov and Alla Toropova (Toropov and Benfenati 2007a, b; Toropov et al. 2010).

Particularly, the research group of Toropov and Toropova proposed the SMILES-based descriptors for the easy computation and interpretation of the importance of

features followed by QSAR modeling using the Monte Carlo approach. This computational methodology has been produced as a free software called the CORrelation And Logic (CORAL) (http://www.insilico.eu/coral) (Toropov and Benfenati 2007a, b; Toropov et al. 2010). The SMILES notation is used to directly extract 1D molecular features (e.g. atom, bond and other elements) from the chemical structures without the need for external software for descriptor calculation. It can be used for the development of regression and classification based predictive models using the Monte Carlo technique for biological activities (Worachartcheewan et al. 2015; Masand et al. 2014), chemical properties (Toropova and Toropov 2014; Gobbi et al. 2016) and nanomaterial properties (Toropov et al. 2013). CORAL requires an input file consisting of the compound name, SMILES notation and the bioactivity values or class labels. Compounds from the data set are separated into training, invisible training, calibration sets (i.e. used as visible data set) and validation set (i.e. used as invisible data set that is not used during the model construction). Moreover, such data subsets are generated for three or more independent data splits as to evaluate variability from the prediction models. The performance of such models can be derived from statistical parameters such as R^2, Q^2, $R^2 - Q^2$ (Worachartcheewan et al. 2014).

A set of local and global molecular features can be derived from the SMILES notations as follows:

$$\begin{aligned} abcdef &\rightarrow a + b + c + d + e + f(S_k) \\ abcdef &\rightarrow ab + bc + cd + de + ef(SS_k) \\ abcdef &\rightarrow abc + bcd + cde + def(SSS_k) \end{aligned} \tag{1}$$

These are the examples of local descriptors that represents the elements in the SMILES notation. In addition, global descriptors are also encoded designated as *BOND*, *PAIR*, *NOSP* and *HALO* as follows:

- *BOND* is presence/absence of bond in the SMILES input such as double bond (=), triple bond (#) and stereo chemical bond (@)
- *PAIR* is the co-incidence of two elements of the following: F, Cl, Br, I, N, O, S, P, #, = and @
- *NOSP* is presence/absence of N, O, S and P
- *HALO* is presence/absence of halogens

In the software, optimized parameters include threshold and correlation weights (CW). An example of equation of SMILES-based optimal attributes, was calculated by the following equation:

$$\begin{aligned} DCW(Threshold, N_{epoch}) = \sum CW(S_k) + \sum CW(SS_k) + \sum CW(SSS_k) + \\ \sum CW(BOND) + \sum CW(NOSP) + \sum CW(HALO) + \sum CW(PAIR) \end{aligned} \tag{2}$$

The biological/chemical endpoint can be calculated as follows:

$$Endpoint = C_0 + C_1 \times DCW(Threshold, N_{epoch}) \tag{3}$$

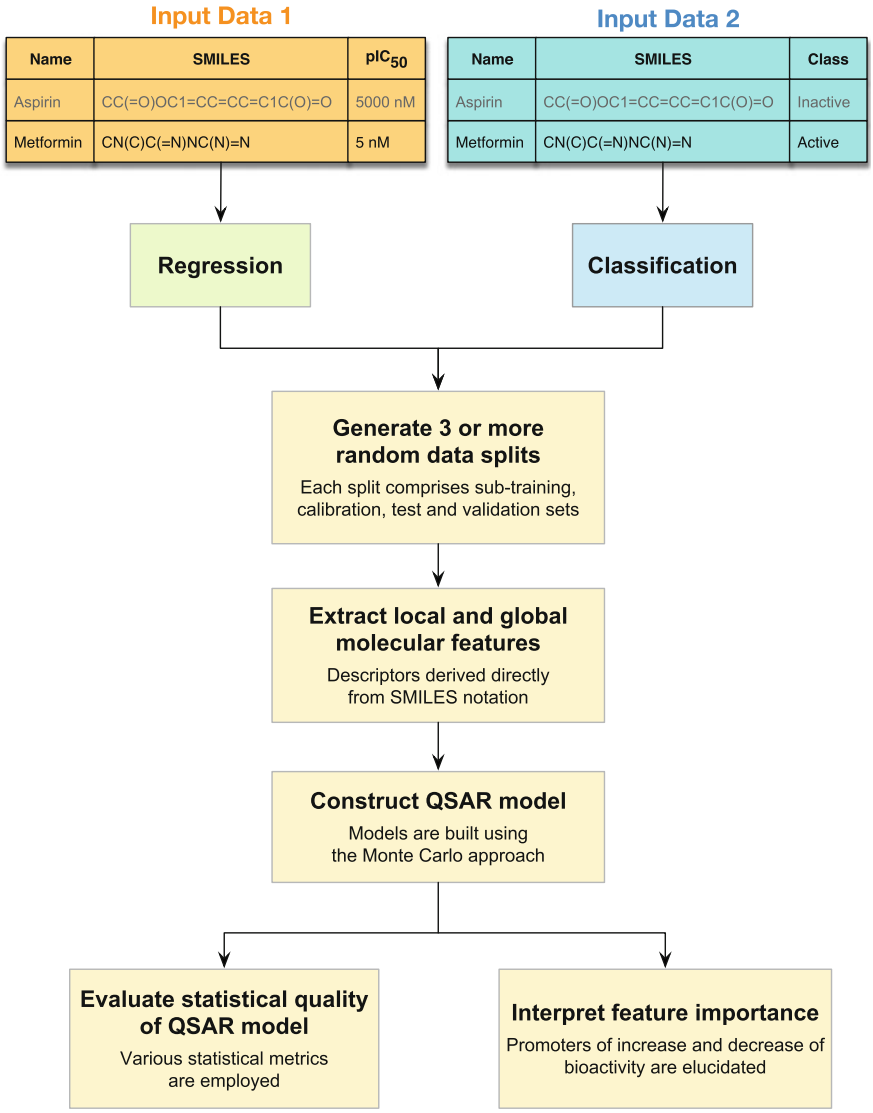

Fig. 3 Workflow of the CORAL software for constructing QSAR modelings using SMILES-based descriptors

where C_0 is the intercept and C_1 is the slope or correlation coefficient.

Furthermore, the molecular fragments obtained from the software can give knowledge of important chemical feature influencing their activities as promoters for increasing or decreasing biological activity. The summary of development of predictive models using SMILES-based descriptors by CORAL software are outlined in Fig. 3.

Recently, Filimonov et al. (2009) proposed a novel QNA-based Star Track QSAR approach in which any molecule is represented as a set of points in 2D space of QNA descriptors. The Star Track approach is in contrast with the classical QSAR method and does not require the use of feature selection. This approach is implemented in the GUSAR software package and is based on a self-consistent regression, QNA descriptors and the topological length and volume of a molecule. This approach predicts quantitative values of biological activity of compounds on the basis of their structural formula and does not require the use of information about the 3D structures of ligands and/or target proteins. The Star Track QSAR approach compares favorably with different 3D and 2D QSAR methods on various gold standard data sets and does not select models based on Q^2 values. Thus, the Star Track QSAR approach as implemented in the GUSAR software package is a potentially useful approach for the derivation of statistically robust, interpretable and fast QSAR models.

7.5.2 1D Descriptors

1D descriptors, also referred to as fingerprints, essentially capture the counts and properties of functional groups and substructural fragments (Todeschini and Consonni 2008). A fundamental difference between 1D descriptors and fingerprints is that the former uses a predefined set of keys (i.e. functional groups and substructures) to generate the descriptors while the latter uses either a predefined set or a set of keys generated on the fly. The older generation of fingerprints consisting of MACCS (Durant et al. 2002), PubChem, and SMARTS still uses a predefined set of keys (Hinselmann et al. 2011) for generating fingerprints and are critically limited at capturing the domain (target- and ligand-) specific structural features responsible for variation in activities. For instance, predefined fingerprints may capture too few or too many correlating features which may have moderate value in QSAR studies. However, recent advances in computer science led to the concept of hashed fingerprints where a set of patterns are generated by gathering atom environment information or subgraph information or both. The generated context dependent patterns are then transformed into hash codes (i.e. a fixed size vector) using hashing algorithm. These hash codes can then be transformed into bit strings using a random number generation of a defined length (i.e. size of the fingerprint). The presence and absence of a pattern is marked as being either *1* and *0*, respectively. Extended connectivity fingerprint (ECFP) (Rogers and Hahn 2010) is a prototypical example of a hashed fingerprint. A major advantage of 1D descriptors or hashed fingerprints is their ability to capture complex structural patterns in uniform fixed bit vectors, which can be quickly computed (Rogers and Hahn 2010). These bit vectors are amenable for molecular similarity/substructure analysis problems, show little degeneracy, are naturally interpreted and are widely used in chemoinformatics (Prathipati et al. 2008). In view of the intuitive concepts of substructures' and functional groups' contributions to drug design and their efficient computation, the 1D descriptors or fingerprints were primarily used for the inverse QSAR problems (Rosenbaum et al. 2011) as discussed in the Introduction.

7.5.3 2D Descriptors

2D or topological descriptors (Gozalbes et al. 2002) are computed by encoding the atoms and their connectivity as a graph. Several variations to the graph-theoretic representation of atoms and their connectivities led to the wide plethora of methods for the generation of 'graph-theoretic' descriptors such as Kier and Hall (1976), Broto et al. (1984), Balaban (1982), Randic (1975), MEDV etc. Although they lack in interpretability, 2D descriptors can be considered good descriptors in many aspects (as listed in Table 2). However, the poor interpretability of this class of descriptor critically limits its usage in retrospective QSAR analysis (Gozalbes et al. 2002). Furthermore, since correlation does not always imply causality, models derived using these class of descriptors are difficult to prioritize from a pool of models that offer very similar statistical significance (Saxena and Prathipati 2006). There are two excellent techniques to mitigate this problem and discriminate seemingly equivalent models via the generalized pairwise correlation method (GPCM) (Héberger and Rajkó 2002) and the sum of ranking differences (Heberger and Skrbic 2012). However, QSAR models derived from these descriptors are ideally suited for a prospective virtual screening analysis as they can be efficiently computed and generally have very low levels of degeneracy (Saxena and Prathipati 2006).

Among the various topological indices, the molecular electronegativity distance vector based on 13 atomic types called the MEDV-13, is a fast, easy to use, reproducible and predictable descriptor for QSAR studies. The studies by Liu et al. (2001) show the performance of MEDV-13 models were comparable to 3D QSAR studies and are also applicable to QSARs of peptides. MEDV-13 descriptor in addition employs information about an element atom type, valence electronic state, and chemical bond type from 2D molecular topology and requires no information related to 3D structures or physicochemical properties or molecular alignments.

7.5.4 3D Descriptors

3D descriptors characterize the 3D structure of a molecule in terms of their shape, steric and electronic features (Kubinyi 1993). While shape-based 3D descriptors (e.g. volume, *RDF* (Gonzlez et al. 2005), *autocorrelation3D* (Sliwoski et al. 2016), etc.) are highly relevant in explaining SAR data, they remain difficult to interpret. Furthermore, the 3D descriptors comprising of *RDF* (Gonzlez et al. 2005), *3D-MoRSE* (Devinyak et al. 2014), *WHIM* (Bravi et al. 1997) and *GETAWAY* (Consonni et al. 2002) descriptors share many similarities with 2D descriptors as described above. While the latter encodes atoms and their connectivity as simple graphs, the 3D shape-based descriptors capture these features together with their distances and angles as part of a complex graphs. On the other end, the 3D descriptor spectrum includes descriptors such as steric and electrostatic fields that are computed using semi-empirical quantum chemical methods as part of the GRID concept (Sippl 2006). The 3D QSAR paradigm asserts the importance of conformational preferences of compounds for molecular recognition to its target protein in addition to structural

and physicochemical features as described above. The CoMFA/CoMSIA methods (Cramer et al. 1988) to date remains the prototypical examples of this paradigm and several leading publications reported seemingly interpretable retrospective analysis of both target-based (Prathipati et al. 2005) and phenotype-based SAR data. However, in a seminal paper, Doweyko (2004) debunked the commonly asserted illusion and showed that the so-called significant regions are subject to the vagaries of alignment and that the nature of possible interactions heavily depends on the eye of the beholder. Furthermore, the arbitrary nature of both the alignment paradigm and atom description lends itself to capricious models, which in turn can lead to distorted conclusions (Doweyko 2004). In spite of limitations of the 3D QSAR approach, this class of descriptors demonstrates very low levels of degeneracy (i.e. extremely sensitive to changes in the structure) and is considered as the gold standard amongst the QSAR modelling techniques. Although, the 3D steric and electrostatic fields have been very intuitive both for explaining the SAR data and for guiding several novel designs, a potential limitation is their rationalization is limited to a congeneric series of compounds. Hence, 3D-QSAR models are not typically used for large-scale prospective virtual screening analysis (Doweyko 2004). Although, several variations of the Tripos CoMFA/CoMSIA (Cramer et al. 1988) have emerged in recent years, the only known freeware is Open3DQSAR (Tosco et al. 2011), which is potentially an interesting addition to the growing number of 3D QSAR software.

7.5.5 Physicochemical Properties

Physicochemical properties are considered to be one of the most relevant descriptors for drug design (Brustle et al. 2002; Taskinen and Yliruusi 2003). While they are mostly measured quantities, they are calculated based on parameterization with measured data. Thus, these descriptors differ from others in that they are not derived from first principles but are obtained from models trained using either 0D, 1D, 2D, 3D (e.g. 3D quantum chemical descriptors calculated using the GRID approach) to fit with experimentally obtained physical and chemical properties such as log$P$, pK_a and solubility measures (Taskinen and Yliruusi 2003). Hence, in contrast to some molecular descriptor software and reviews, which had categorized this class of molecular descriptors as 0D, 1D, 2D or 3D. Thus, in this chapter we have placed this class of descriptors separately. These descriptors (e.g. log$P$, log$D$, pK_a) play a major role in both pharmacodynamic and pharmacokinetic properties of compounds (Taskinen and Yliruusi 2003). Furthermore, they have now become a part of the standard checklist for assessing the drug-likeness (e.g. Lipinski's rule-of-five) and other pharmacokinetic liabilities. Moreover, they are also widely used in explaining the variation of target-based SAR data. Most proteins' structure-function modulation is mediated via salt-bridges and small molecules typically modulate the function of a protein via charge neutralization thereby leading to the disruption of salt-bridges followed by a consequent change in the structure and function of the protein (Prathipati and Saxena 2005). In this context, physicochemical properties like pK_a and other quantum chemically derived electronic properties are widely used (Manallack 2008). In

spite of their widespread usage, intuitive appeal and interpretability, these descriptors remain difficult to compute. Given the importance of modelling various electronic effects (e.g. inductive, mesomeric, polar) (Thornber 1979; Patani and LaVoie 1996; Jelfs et al. 2007; O'Boyle et al. 2017b; Harding et al. 2009; Morgenthaler et al. 2007; Xing et al. 2003; Manallack 2008), it should be noted that computationally-expensive quantum chemical descriptors are often used to train models that can predict the pK_a, polarizability, etc. Thus, the development of software for computing these descriptors is an area of active research. Improvements in GPU technology have greatly accelerated the utilization of quantum chemical simulations (Patani and LaVoie 1996) for the prediction of physicochemical properties and biological activities.

8 Interpretable Learning Algorithms

8.1 Black Box Learning Methods

Kurgan et al. (2009) used the term black box models to describe the fact that machine learning models do not identify the underlying associations of individual features with the specific outcome as well as not revealing which features provide essential contribution to the observed prediction accuracy. Black box have demonstrated success in modeling a wide range of bioactivities and properties (Charoenkwan et al. 2013; Shoombuatong et al. 2015; Simeon et al. 2016a, b; Shoombuatong et al. 2015; Nantasenamat et al. 2005, 2007a).

8.1.1 Support Vector Machine

Support vector machine (SVM) (Cortes and Vapnik 1995; Burges 1998; Barakat and Bradley 2010) is a statistical learning approach and a well-known maximum margin classifier that is based on the principles of structural risk minimization (SRM). The SRM principle is utilized to seek a hypothesis function with low capacity from a nested sequence of functions that can simultaneously minimize both the true error rate (i.e. prediction error on the external set) and the empirical error rate (i.e. prediction error on the training set) as illustrated in Fig. 4.

Given a training set $D_{Tr}^{m} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), ..., (\mathbf{x}_m, \mathbf{y}_m)\}$, where $\mathbf{x}_i \in \mathbb{R}^n$ and $\mathbf{y}_i \in -1, +1$, the SVM classifier finds the optimal separating hyperplane that has the largest margin and satisfies the following conditions:

$$\begin{aligned} \mathbf{w}^T \varphi(\mathbf{x}_i) + \mathbf{b} \geq +1, \quad for \quad \mathbf{y}_i = +1 \\ \mathbf{w}^T \varphi(\mathbf{x}_i) + \mathbf{b} \leq +1, \quad for \quad \mathbf{y}_i = -1 \end{aligned} \tag{4}$$

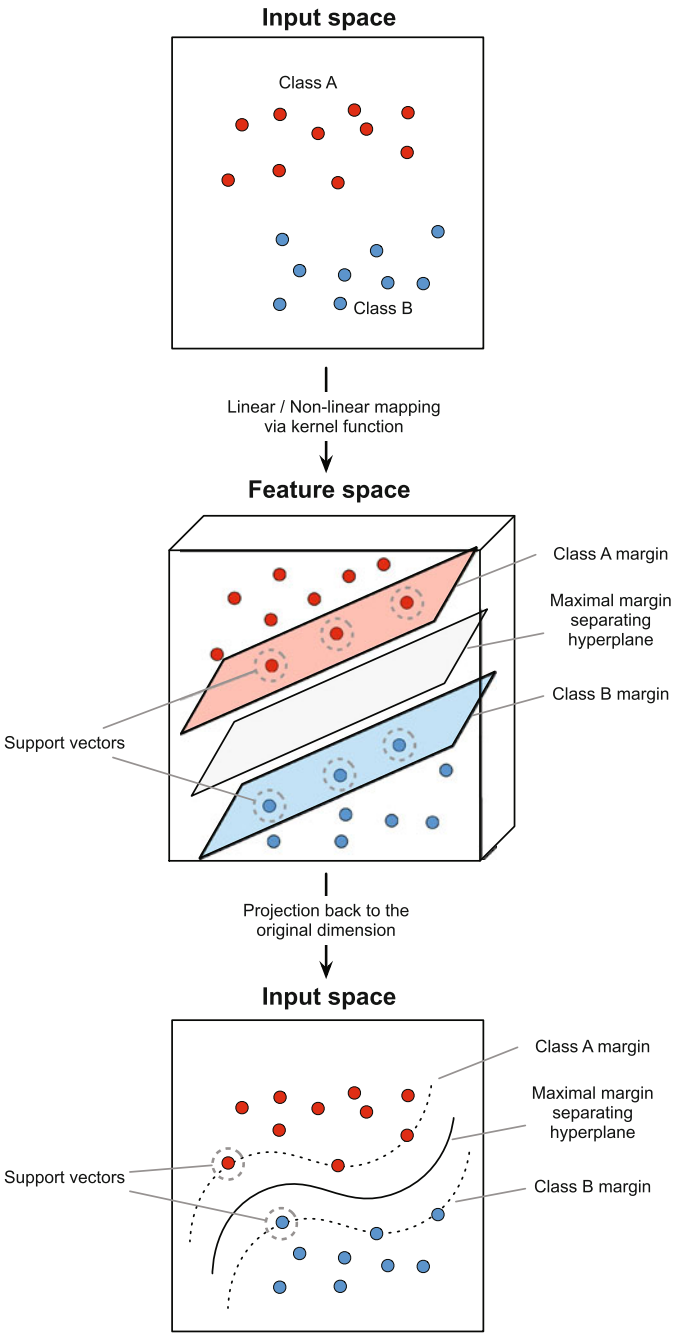

Fig. 4 Illustration of the SVM learning process. Initially, the input space is transformed to a higher dimensional feature space via the use of kernel functions whereby the maximal margin separating hyperplane is obtained after defining the margins of the two classes. It should be noted that compounds (denoted by *circles*) lying on the margin represents the support vectors

which is equivalent to:

$$\mathbf{y}_i[\mathbf{w}^T\varphi(\mathbf{x}_i)+\mathbf{b}] \geq +1, \quad i=1,2,...,m \tag{5}$$

The non-linear function maps the input space to a higher dimensional space called the feature space. The mapping function $\varphi(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^p$, where $n << p$, is performed by defining the inner product between two samples through kernel function $K(\mathbf{x},\mathbf{y})$. Practically, the kernel function $K(\mathbf{x},\mathbf{y})$ is expressed with a similarity measurement between two samples in the data set, which is defined as Burges (1998):

$$\begin{aligned} K(\mathbf{x},\mathbf{y}) &= \varphi(\mathbf{x})\cdot\varphi(\mathbf{y}) \\ &= \sum_i \varphi(\mathbf{x})_i\varphi(\mathbf{y})_i \end{aligned} \tag{6}$$

For the kernel function $K(\mathbf{x}_i,\mathbf{x}_j)$, the most popular kernel function includes: the linear kernel $\varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$; the polynomial kernel $(1+\varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j))^d$, where d = 2, 3, and 4 (i.e. it should be noted that $d = 1$ for linear kernel); and the radial basis function (RBF) kernel $\exp(-\gamma(\left\|\mathbf{x}_i-\mathbf{x}_j\right\|))$, where C (the penalty factor), γ (trading off error predictions against margin width) and ε (the percentage of support vectors in the SVM model) are parameters to be optimized. Kernel functions are often used in SVM because of the scalar product in the dual form. In fact, these approaches can also be used for other machine learning algorithms, but they are not tied to the SVM formalism. It should be noted that the RBF kernel has been widely used in SVM modelling. The decision function of the SVM classifier is given by:

$$y(x) = sign[\sum_{i=1}^{m}\alpha_i\mathbf{y}_iK(\mathbf{x}_i,\mathbf{x})+\mathbf{b}] \tag{7}$$

where α is the parameter solved by the Lagrangian algorithm and $\mathbf{x} = (x_1, x_2, ..., x_M)$.

This method was not originally developed as a tool for statistical prediction by Cortes and Vapnik (1995). However, Vapnik enabled the original SVM to solve regression problems also known as support vector regression (SVR), by choosing a suitable cost function (ε-insensitive loss function) that enables a sparse set of support vectors to be obtained. The standard regression procedures is to identify a function f(x) that provides the least square error between predicted and actual observed responses for all training data set. In contrast, SVR attempts to minimize the generalization error bound for achieving higher generalization performance. This generalization error bound is derived from the combination of the training error and a regularization term controlling the complexity of hypothesis space. The first term is calculated by the ε-insensitive losses. The ε-insensitive loss function for SVR method (Drucker et al. 1996; Song et al. 2002) is defined as follows:

$$L_\varepsilon(\mathbf{y},f(\mathbf{x},\beta)) = \begin{cases} |\mathbf{y}-f(\mathbf{x},\beta)|-\varepsilon, & |\mathbf{y}-f(\mathbf{x},\beta)| \geq \varepsilon \\ 0 \quad , & |\mathbf{y}-f(\mathbf{x},\beta)| < \varepsilon \end{cases} \tag{8}$$

where **y** is the actual value, $f(\mathbf{x}, \beta)$ is the predicted value (i.e. in which the simple form is f(**x**), where $\mathbf{x} = (x_1, x_2, ..., x_n)$) and ε is the insensitivity parameter.

8.1.2 Artificial Neural Network

Artificial neural network (ANN) is a well-established machine learning algorithm for establishing QSAR models (Nantasenamat et al. 2005, 2007a, b, 2008; Worachartcheewan et al. 2009). ANN represents biologically inspired prediction and classification methods whose original development was based on the structure and function of the network of neurons (Zurada 1992). A typical ANN is established with three major components, namely the transfer function, the learning rule and the connection formula (Simpson 1990) as illustrated in Fig. 5. Until now, the feed-forward ANN (FF-ANN) is the most popular ANN that has been used in real-life situation (Ebrahimi et al. 2016). Among many learning algorithm for estimating the parameter of FF-ANN, the back-propagation (BP) algorithm is the most extensively used for finding the optimal parameters, which is carried out by minimizing the error of the network through the derivatives of the error function. For a given training set D_{Tr}^{m} in a BP-ANN task, the input layer starts to propagate the signal through the connection

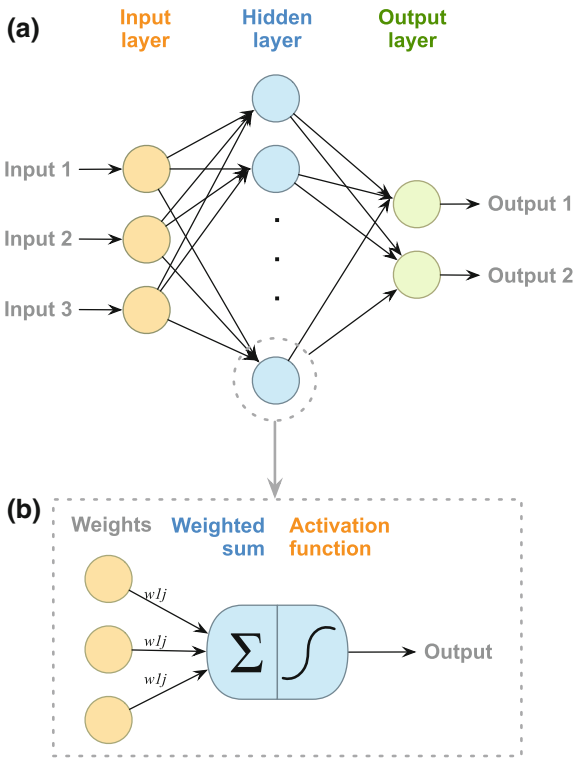

Fig. 5 Illustration of the architecture of artificial neural network (**a**) and inner working of neurons in a hidden layer (**b**)

weights and the transfer function to produce the output for each neuron. The output or predicted value is then compared to the actual value and the differences in the value between the predicted and actual values is minimized by the BP algorithm. Practically, the delta rule is used to optimize the weights via the BP algorithm:

$$W_{ij}^{new} = W_{ij}^{old} + \triangle W_{ij} \tag{9}$$

$$\triangle W_{ij} = -\mu \frac{\partial E_p}{\partial W_{ij}} out_j \tag{10}$$

where out_j is the output of the *jth* neuron, μ is the training rate and E_p is the error. The output layer of ANN can be represented mathematically as:

$$O = f(\sum_{i=1}^{M} \mathbf{w}_i \mathbf{x}_i + \mathbf{b}) \tag{11}$$

8.1.3 Deep Learning

Owing to the limitations of FF-ANN, a deep learning (DL) method was proposed by three separate groups (Hinton et al. 2006; Raiko 2012; Bengio 2009) for solving the process of training models in many layers. In 2006, DL also known as deep neural network has become increasingly popular for parameter approximation by allowing computational models to learn from representations of data using multiple levels of abstraction (Hinton et al. 2006, 2012). Many research groups reported that there are many different points between ANN and DL (Xing et al. 2003; Leung et al. 2014; Ma et al. 2015). Firstly, each layer of the neural network is constructed from a row of neurons while DL is built from several layers of neurons. Layers in a DL consist of three main layers: (i) the input layer (i.e. the bottom layer), where the descriptors of a molecule are entered; (ii) the output layer (i.e. the top layer), where prediction results are created; (iii) the hidden (middle) layers, where the word "deep" in DL implies that there is more than one hidden layer, as illustrated in Fig. 6. There are two popular choices of activation functions (f) that are used in the hidden (f_H) and output (f_O) layers, namely the sigmoid function and the rectified linear unit (ReLU) function. Secondly, the output layer of ANN basically has one or more neurons and each output neuron generates prediction for a separate endpoint while DL can naturally model multiple endpoints at the same time. Finally, DL employs ReLU instead of sigmoids (i.e. usually used in ANN) as activation functions in order to overcome the vanishing gradient problem. These activation functions have non-vanishing derivative.

Previously, many reports suggested that the predictive performance of DL has dramatically improved as compared to that of standard ANN. The strength of DL lies in its ability to manipulate the intricate structure in large training set by using the backpropagation algorithm. Presently, DL is being applied to many domains of science, business and government. For instance, in the domain of bioinformatics,

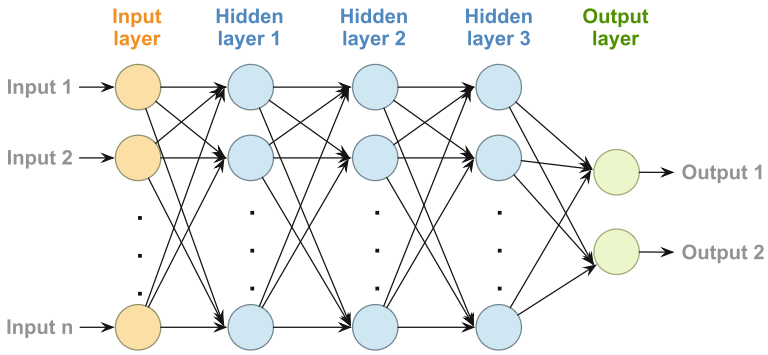

Fig. 6 Illustration of the architecture of deep learning algorithm

DL has been compared with other conventional machine learning algorithm for predicting the activity of potential drug molecules (Ma et al. 2015), analysing particle accelerator data (Ciodaro et al. 2012), reconstructing brain circuits (Helmstaedter et al. 2013) and predicting the effects of mutations in non-coding DNA on gene expression and disease (Xing et al. 2003; Leung et al. 2014). DL has also yielded promising results in natural language processing (NLP) (Collobert et al. 2011), especially for topic classification, sentiment analysis, question answering and language translation (Bordes 2014; Sutskever et al. 2014).

8.1.4 Towards Opening the Black Box

The classical QSAR approach developed by Hansch in the 1960s (Hansch et al. 1962) has a long history in predicting biological activities and physical properties. The original model used a simple, transparent and interpretable MLR model and provided excellent mechanistic interpretation of the biological activity. However, QSAR models are expected to provide both quick predictions (i.e. in a prospective manner) and mechanistic interpretation (i.e. through its features in a retrospective manner). The superior performance of SVM and ANN models vis-a-vis other computational-based models in a variety of application areas is widely known. The high accuracy and robustness of these methods can be attributed to their ability to build non-linear, black-box models that can account for the complexity of the input data. This inability to provide an explanation or comprehensible justification for the predicted solutions critically limits their application to several areas. In application areas such as medical diagnosis, it is highly desirable to give a clear mechanistic interpretation associated with the classification decisions in order to aid the compliance by both the physician and the patient. To mitigate this problem, methods that can aid the interpretation of significant features used by the model can be obtained via the use of rule extraction methods as had recently been shown for ANNs (Fung et al. 2005; Andrews et al. 1995; Setiono et al. 2002) and SVMs (Andrews et al. 1995; Barakat and Bradley

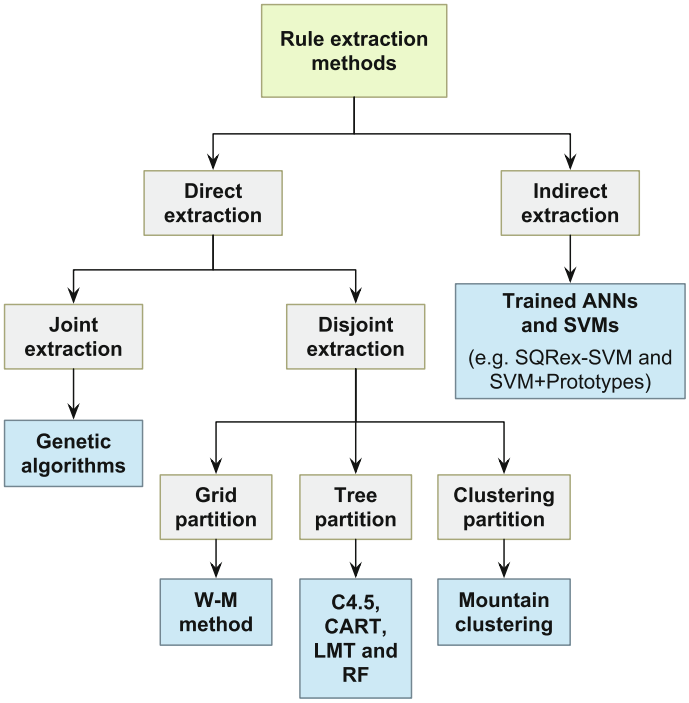

Fig. 7 Taxonomy of rule extraction techniques

2010; Núñez et al. 2002; Zhang et al. 2005; Fu et al. 2004; Barakat and Diederich 2004, 2005).

In recent years, many rule extraction techniques were developed to extract easy-to-understand regularities from data. Figure 7 illustrates the taxonomy of those methods that are derived from the data mining research community. Firstly, they are divided into direct and indirect methods according to the approach that rules are reasoned out. As mentioned, indirect rule extraction methods (e.g. SQRex-SVM and SVM+Prototypes) have been developed for providing explanations as well as affording prediction. Direct rule extraction methods are more widely studied in theory and applied in practice. The direct extraction of rules contains two critical tasks namely antecedent (i.e. representing the condition part of rules) and consequent (i.e. defining the behavior within each region) identifications. Based on the approach that these two tasks are carried out, methods to extract rules are further divided into two groups consisting of joint methods and disjoint methods. Joint methods, such as GA (Lawrence 1991), simultaneously identifies the antecedent and consequent by exceeding the capabilities of most optimization algorithms as they can afford the capability of finding global optimal solutions by mimicking biological evolution. As for disjoint methods, the divide and conquer approach is used as the strategy for optimizing the following two tasks: separating and identifying advantages over joint

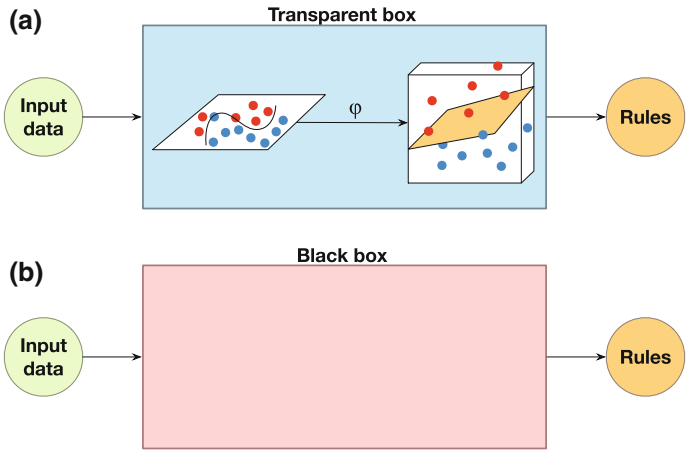

Fig. 8 System flowchart of decompositional and pedagogical rule extraction techniques

ones in computational efficiency. There are three methods that are widely used for partition namely grid (e.g. Wang-Mendel (WM) method (Wang and Mendel 1992)), tree partition (e.g. C4.5 (Quinlan 1993), classification and regression trees (CART), logistic model tree (LMT) and random forest (RF) (Breiman 2001)) as well as clustering (e.g. Mountain clustering and its extension, subtractive clustering (Yager and Filev 1994)).

The interpretability of ANNs and SVMs can be obtained by extracting symbolic rules from the trained model. The rule extraction techniques are used to open up the black box approach by generating symbolic, comprehensible descriptions while maintaining the same predictive power (Martens et al. 2007). Andrews et al. (Andrews 1974; Andrews et al. 1995) proposed an approach for the rule extraction from ANN that can be easily extended to SVMs. Two approaches exist to extract rules from the black-box ANN and SVM models (Martens et al. 2007) which are the decompositional and pedagogical approaches. The decompositional approach determines rules by utilizing information from the internal components of the constructed SVM model while the pedagogical approach considers SVM model as a black box and derives its rules by relating the inputs with the outputs of the SVM model. The difference between the decompositional and pedagogical rule extraction techniques is schematically illustrated in Fig. 8.

For the *decompositional approach*, Setiono and Liu (1995) firstly proposed an approach to understand the ANN's results. Understanding the ANN's results through rule extraction was obtained via the use of a three-phase algorithm as follows: (i), a weight-decay back-propagation network is built such that important connections are reflected by the larger weight values; (ii) the network is pruned by deleting non-informative connections while still maintaining its predictive accuracy; (iii) rules are extracted and produced. In 1997, the decompositional technique NeuroLinear (Setiono and Liu 1997) was developed to extract oblique classification

rules from neural networks comprising of one hidden layer. Kim and Lee (2000) have proposed an algorithm for feature extraction and feature combination by utilizing multilayer perceptron networks with sigmoid functions. A few years later, Gupta et al. (1999) had proposed an analytical framework for classifying existing rule extraction methods for FF-ANN. This method extracts rules by directly interpreting the strengths of the connection weights in a trained network. In the case of the decompositional method, a few research have been published for extracting rules from SVMs. For instance, Núñez et al. (2002) proposed the SVM+Prototypes method for extracting rules from SVMs. The basic idea of this approach consists of: (i) determining the decision function by means of SVM while a clustering algorithm is used to determine prototype vectors for each class; (ii) defining regions in the input space that can be transferred to if-then rules. In 2007, Barakat and Bradley (2007) proposed a novel algorithm for the rule extraction from SVMs known as SQRex-SVM. After training the SVM model, SQRex-SVM directly extracts rules from the support vectors (SVs) by using a modified sequential covering algorithm. Rules are then produced by using the rank of the most discriminative features as measured by the interclass separation.

For the *pedagogical approach*, there are a large number of studies focused on opening the black box nature of ANN as to improve their interpretability. In 1988, Saito and Nakano (1988) have proposed a workflow for medical diagnosis using rule extraction from a modified ANN. A few years later, the BRAINNE system was proposed (Sestito and Dillon 1992) for extracting rules from ANN using backpropagation algorithm. The major contribution of the BRAINNE system is that it can directly deal with continuous data as inputs without requiring discretization. Shortly afterwards, Thrun (1993) proposed the VIA method for extracting rules by mapping inputs directly to the output through the use of a generate-and-test procedure for extracting symbolic rules from ANN trained by the backpropagation algorithm. Furthermore, details on how to improve the interpretability of the black box ANN have been discussed previously (Zhou and Chen 2002; Andrews et al. 1995; Augasta and Kathirvalavakumar 2012). Similar to the case of the decompositional method, only a few studies have been reported for improving the interpretability of ANN via the pedagogical approach. For example, Trepan (Craven and Shavlik 1996) was the first to introduce the pedagogical tree extraction algorithm by extracting decision trees from trained neural networks having an arbitrary architecture. In constructing a tree, this method makes use of the best first expansion strategy to build a tree via recursive partitioning. Trepan allowed splits with at least M-of-N type of tests. At each step, a queue of leaves is further expanded into sub-trees until a stopping criterion is met. In 2007, Martens et al. (2007) proposed the use of an SVM model as an oracle to generate rules. For the convenience of the vast majority of scientists, a MATLAB toolbox for generating rules using any black box model as oracle has been implemented and made publicly available. Previously, many researchers reported that ANN and SVM rule extraction approaches had equal or higher performance when compared with the original ANN and SVM methods (Barakat and Bradley 2007; Augasta and Kathirvalavakumar 2012; Gong et al. 2008).

8.2 White Box Learning Methods

8.2.1 Multiple Linear Regression

MLR is one of the most basic method for performing regression in QSAR modeling. Given a matrix X of a compound of interest, the MLR model assumes that the expected value of Y could be expressed in the form of a linear equation as summarized below:

$$\mathbf{y}_i = \sum_{i=1}^{m} \mathbf{b}_i \mathbf{x}_i + \mathbf{b}_0 \tag{12}$$

Generally, this approach is favored for its simplicity and ease of interpretation as the model assumes that there exists a linear relationship between a set of molecular descriptors and the bioactivity. When using MLR, regression coefficients can be obtained via the use of the least squares method. The size of the coefficient may reveal the degree of influence that molecular descriptors has on the bioactivity. Moreover, a positive coefficient indicate that the respective molecular descriptors contributes positively to the bioactivity and vice versa for the negative coefficient. However, in the presence of collinear descriptors, these interpretations may be error prone. A general rule of thumb states that the sample size (i.e. number of compounds in the data set) should be at least five times the number of descriptors that are used.

8.2.2 Logistic Regression

The transformation of MLR to a logistic regression (LR), can be easily performed by representing the Y variable via the conditional probability of Y given X variables ($\pi(X)$) when the logistic distribution is used (Hosmer et al. 2013). The specific formula of LR is defined as follows:

$$\pi(X) = \frac{e^{b_0 + b_1 x_1 + + b_2 x_2 + \cdots + + b_M x_M}}{1 + e^{b_0 + b_1 x_1 + + b_2 x_2 + \cdots + + b_M x_M}} \tag{13}$$

where $\mathbf{b}_i$ represents the transformation of $\pi(X)$. Furthermore, the logit transformation is defined in terms of $\pi(X)$:

$$\begin{aligned} g(X) &= ln[\frac{\pi(X)}{1 - \pi(X)}] \\ &= b_0 + b_1 x_1 + + b_2 x_2 + \cdots + + b_M x_M \end{aligned} \tag{14}$$

For the MLR method, the least square approach is used to estimate unknown parameters $\mathbf{b}_i$. The basic idea of this method is to minimize the sum of square error between predicted Y and actual Y values. Unfortunately, the least square approach cannot be used to optimize $\mathbf{b}_i$ on a data having a dichotomous variable (i.e. variables

that have a value of 0 or 1). As for the LR method, the maximum likelihood estimator is used to alleviate the problem of dichotomous variables. A convenient way to represent the likelihood probability function for $(\mathbf{x}, \mathbf{y})$ where $\mathbf{x} = (x_1, x_2, ..., x_M)$ and $\mathbf{y} = (y_1, y_2, ..., y_M)$ can be defined as follows:

$$\pi(x_i)^{y_i}[1 - \pi(x_i)]^{1-y_i} \tag{15}$$

Since the data set (X, Y) is assumed to be independent variables, the likelihood probability function is used to estimate β_i in expressions summarized as follows:

$$l(b_i) = \prod_{i=1}^{M} \pi(x_i)^{y_i}[1 - \pi(x_i)]^{1-y_i} \tag{16}$$

In the binomial case, where the outputs of LR is close to 0 and 1, respectively, indicates low and high probability of occurrences.

8.2.3 Efficient Linear Method

Efficient linear method (ELM) is a general-purpose learning method proposed by Shoombuatong et al. (2015) that can be used for performing both classification and regression tasks. This approach was first applied in the QSAR study of the bioactivity of aromatase inhibitors (AIs) where it has been shown to afford an interpretable model in which significant features are transparent and can be used to provide insights pertaining to the origin of its bioactivity. The main procedures of the ELM method entails the following steps:

Step 1: Prepare a training data set D_{Tr}^{M} consisting of positive and negative samples.

Step 2: Formulate a predictive model with a weighted summation $f(C)$ in the form of a linear model as follows:

$$f(C) = \sum_{i=1}^{m} \mathbf{b}_i \mathbf{x}_i + \mathbf{b}_0 \tag{17}$$

Step 3: Select informative features using the fitness function of the Akaike information criterion (AIC). Finally, features affording high feature usage is selected for the construction of a predictive model.

Step 4: Estimate the optimal parameter **b** by using the genetic algorithms (GA) with the Andrews' sine function *fitness(x)* (Andrews 1974). To obtain a reliable parameter, the fitness function utilizes a 10-fold cross-validation (10-fold CV) scheme.

Step 5: Predict the unknown P with the scoring function (*Pred*(*C*)) using the weighted summation and subsequently discriminate it using only the threshold as obtained from:

$$Pred(C) = \begin{cases} positive, & f(C) > threshold \\ negative, & otherwise \end{cases} \quad (18)$$

8.2.4 Principal Component Analysis

The aforementioned learning approach are supervised (e.g. MLR, ANN, or SVM) in which the SAR is discerned from a list of compounds in the training set using the function in the form of $Y = f(X)$ (i.e. Y can be computed as a function of X descriptors). As a counterpart, unsupervised learning methods aim to characterize the underlying patterns of X variables without the need for Y variable. Principal component analysis (PCA) is one of the most commonly used unsupervised learning method for multivariate data analysis that can help reveal details from the high-dimensional information hidden inside the array of numerical descriptors (Jolliffe 2002). PCA analyzes the high-dimensional and intercorrelated X variables and compresses its information into a few dimensions without much loss of the core information while filtering out the noise. Briefly, the first principal component (PC) lies along the direction of maximal data variance capturing the most variability of all possible linear combinations. Because PCA seeks the linear combination of X variables that are uncorrelated with maximal variability, the assumption can be made that the first PC contains the most core information while much of the last PCs contain the noise. PCA focuses on identifying the data structures based on measurement scales and the resulting PC weights will be larger for X variables with higher variation. Two of the most useful features of PCA are the loadings and scores values.

8.2.5 Partial Least Squares Regression

Partial least squares regression (PLSR) is a commonly used learning method for the analysis of large data sets owing to its inherent ability to handle large redundant features and readily produce interpretable regression coefficients from the predictive model. In PCA, only the X variables are considered in the multivariate analysis as it does not take into account the biological properties of compounds (i.e. the Y variable). However, PLSR makes use of the information of Y variables to maximize inter-class variance (Helland 1988). PLSR is a widely used method for constructing predictive models in which features are compressed into orthogonal latent variable or PCs. The origins of PLSR can be traced back to the non-linear iterative partial least squares (NIPALS) algorithm as proposed by Herman Wold (Helland 2001). For the principle assumption of PLSR methods, a data set with intercorrelated variables is generated and then the latent structure are projected by means of PLSR. This learning method can be used for both regression and classification tasks where dimension reduction of the original feature space is an integral part of its modeling process.

8.2.6 Decision Tree

Decision trees (DT) are tree-like graphs that model a decision, which are commonly learned by recursively splitting the set of training instances into subsets based on the instances' values for the explanatory variables (Quinlan 1993). It uses the conditional statement consisting of if-then statement, which allows us to make a prediction. In short, DT constitutes a series of split points that are known as nodes. To make a prediction, we start at the top-most root node, which represents the most important feature. From this root node, a decision threshold value leads to divergence of two subsequent nodes in which the value of the feature of interest is greater than or less than the threshold value. This process is repeated at each subsequent inner nodes until we reach one of the terminal leaf nodes, which are the prediction class (i.e. whether the compound's bioactivity is classified as either being active or inactive).

8.2.7 Random Forest

Random Forest (RF) is an ensemble of unpruned classification and regression tree (Breiman et al. 1984; Breiman 2001). RF takes advantage of two efficient machine learning methods (e.g. bagging and random feature selection). RF is a further development of bagging. Instead of using all features, RF randomly selects two-third of a training data set to build the predictor and the other one-third of the training data set, known as the out-of-bag (OOB) data set, is utilized to evaluate the performance of the predictor. Predictions are derived from the majority vote or averaging the output of all trees for classification and regression problems, respectively. To evaluate the importance for each feature f_i, the values of features f_i in the OOB data set are randomly permutated and the feature importance for f_i can then be evaluated by measuring the decrease of prediction performance of the permutated OOB data set. The prediction performance can be measured by using accuracy or Gini index. The Gini index is calculated by using the impurity of each feature that is capable of separating samples of two (or more) classes. The size of the feature subsets used is a fixed number in which the number of different features tried at each split (m_{try}) are set at $p^{1/2}$ and $p/3$ for classification and regression problems.

9 Resources and Software for Performing QSAR Modeling

In this section, we present some of the software that can be used for the construction of QSAR models. This spans molecular descriptor software, multivariate analysis software and integrated software that typically lowers the steep learning curve that are usually required to get up and running in developing QSAR models.

Prior to the construction of QSAR models, the molecular features of compounds can be discerned via the use of software for computing molecular descriptors. Table 4

Table 4 List of open source descriptor calculation software

	0D	1D	2D	3D	PCP	Availability	Ref.
CDK	✓	✓	✓	✓	✓	Java, R and Python	Guha (2017), rcdk (2017), O'Boyle and Hutchison (2008)
RCPI	✓	✓	✓	✓	✓	R	Xiao et al. (2017)
ChemmineR	✓	✓	✓		✓	R	Girke (2017)
PaDEL	✓	✓	✓	✓	✓	Java, Standalone	Yap (2017), Yap (2011)
ChemDes	✓	✓	✓	✓	✓	Web server	Cao (2017a), Dong et al. (2015)
jCompoundMapper		✓	✓	✓		Java	Hinselmann et al. (2017), Hinselmann et al. (2011)
QuBiLs-MAS			✓			Standalone	Ponce (2017a), Medina Marrero et al. (2015)
QuBiLs-MIDAS				✓		Standalone	Ponce (2017b), Garcia-Jacas et al. (2014)
Chemical Descriptors Library (CDL)	✓	✓	✓	✓	✓	C++ library	Molplex Ltd. and Sykora (2017)
ChemoPy	✓	✓	✓	✓	✓	Web server	Cao (2017b), Cao et al. (2013)
Pybel	✓	✓	✓	✓	✓	Python	O'Boyle et al. (2017a), Oldham et al. (2008)
Babel	✓	✓	✓	✓	✓	Standalone	O'Boyle et al. (2017b, 2011)

Table 5 Summary of software for performing QSAR modeling

Software	Description	Standalone	Online	Ref.
AutoWeka	Automated data mining software based on Weka machine learning package	✓		Nantasenamat et al. (2015)
AZOrange	Open source high performance machine learning in a graphical environment	✓		Stalring et al. (2011)
CDK-Taverna	Platform independent workflow environment for cheminformatics	✓		Kuhn et al. (2010)
CHARMMing	Aside from ligand docking this suite of tools supports QSAR model building		✓	Miller et al. (2008)
ChemBench	Web platform for building QSAR models		✓	Walker et al. (2010)
ChemMine	Cheminformatics and data mining tools for small molecule data analysis		✓	Backman et al. (2011)
CORAL	Software for building QSAR models using SMILES-based descriptors via Monte Carlo	✓		Benfenati et al. (2011)
DMax Chemistry Assistant	Data mining tool for QSAR, compound data analysis and virtual screening	✓		DTAI Research Group (2017)
MOE Cheminformatics and QSAR	Module for performing cheminformatics and QSAR modeling	✓		Chemical Computing Group Inc. (2017)
OCHEM	Online platform for building QSAR models	✓	✓	Sushko et al. (2011)
OCED QSAR Toolbox	QSAR application toolbox for assessing hazards of chemicals	✓		Dimitrov et al. (2016)
PASS Online	Predicts the biological activity spectra of query compounds	✓	✓	Filimonov et al. (2014)
QSARINS	QSAR modeling tool in agreement with OECD principles	✓		Gramatica et al. (2013)
QSAR Workbench	QSAR workflow tool with numerical and graphical results	✓		Cox et al. (2013)
Toxtree	Toxicity estimation using decision tree	✓		Patlewicz et al. (2008)

Table 6 Summary of software for multivariate analysis

Software	Description	License	Ref.
Benchware	Data mining software for analyzing biological and chemical data	Commercial	Certara (2017)
ChemmineR	Cheminformatics package for analyzing drug-like small molecule data in R	Free	Cao et al. (2008)
IBM SPSS	Statistical and data mining software for multivariate data analysis	Commercial	IBM (2017)
KEEL	Java-based software for performing various	Free	Alcal-Fdez et al. (2011)
KNIME	Modular data exploration and mining platform that allow users to create data flows and extend functionality via modular API	Free	Mazanetz et al. (2012)
LIBSVM	Data mining software based on SVM algorithm	Free	Chang and Lin (2011)
Neuralware	Platform for developing and deploying empirical modeling based on neural networks	Commercial	NeuralWare (2017)
Neural Network Toolbox	MATLAB package providing algorithms, functions, and tools to create, train, visualize and simulate neural networks	Commercial	The MathWorks, Inc. (2017a)
MAPLE	Mathematical and computational engine with an intuitive user interface	Commercial	Maplesoft (2017)
MATLAB	Interactive environment and programming language for performing computationally intensive tasks visual programming on Python scripting	Commercial	The MathWorks, Inc. (2017b)
PyChem	Python package for chemometric for univariate and multivariate data analysis	Free	Jarvis et al. (2006)
R	Comprehensive statistical environment for data analysis and graphics visualization	Free	Ripley (2017)
RapidMiner	Open source system for data mining with an intuitive graphical user interface	Free	RapidMiner, Inc. (2017)

(continued)

Table 6 (continued)

Software	Description	License	Ref.
SAS Enterprise Miner	Reveal insights from data mining analysis	Commercial	SAS Institute Inc. (2017)
Scikit-learn	Python package for data mining analysis	Free	Pedregosa et al (2017)
SNNS	Software simulator for neural networks on Unix workstations	Free	Zell et al. (2017)
SOM Toolbox	MATLAB package for implementation the self-organizing map algorithm and more	Free	Kohonen (2017)
Spotfire S+	Statistical programming environment for analysis large scale data as well as an interactive graphics system for creation of statistical charts	Commercial	TIBCO Software Inc. (2017)
The Unscrambler	Chemometric software for data analysis and design of experiments	Commercial	CAMO Software AS (2017)
WEKA	Java-based software for data analysis via a wide range of machine learning algorithm	Free	Frank et al. (2017)

Table 7 Comparison of machine learning packages and modules from R, Python's scikit-learn and WEKA

Methods	R package	Python's scikit-learn	Weka
SVM	e1071	SVC, NuSVC and LinearSVC	LibSVM
ANN	neuralnet	MLPClassifier and MLPRegressor	MultilayerPerceptron
DL	deeplearning	–	–
MLR	car	LinearRegression	LinearRegression
LR	logistf	LogisticRegression	Logistic
ELM	*R script*[a]	–	–
PCA	princomp	PCA	PrincipalComponents
PLSR	pls	PLSRegression	PartialLeastSquares
DT	C50	DecisionTreeClassifier and DecisionTreeRegressor	J48graft
RF	randomForest	RandomForestClassifier and RandomForestRegressorr	RandomForest

[a]http://dx.doi.org/10.6084/m9.figshare.1274030

summarizes the available software along with the dimensional type of descriptor that can be computed.

A wide range of software and tools for performing QSAR modeling are available as either standalone desktop-based application or as web-based application as summarized in Table 5.

Table 6 lists some of the software for performing multivariate analysis for computer savvy scientists as the software may require a steeper learning curve than those listed in Table 5.

Table 7 summarizes the comparison between three popular machine learning packages in three popular languages namely R, Python and Java.

10 Conclusion

In spite of certain inherent flaws, the QSAR paradigms inevitably is one of the driving forces contributing to the advancements in drug discovery and design. As with all technologies, QSAR is not perfect, however, its weaknesses and flaws are continuously being identified, solved and reformed to help shape a more robust QSAR model. Particularly, the present chapter argues for the increased use of interpretable QSAR models in drug discovery research. QSAR models were originally intended to assist medicinal chemists with design ideas that are often overlooked as a useful approach; one reason is that chemists and biologists do not understand the underlying assumptions of the predictions. Hence, we have presented several concepts pertaining to inverse QSAR techniques that can reconstruct a chemical

structure with good synthetic feasibility based on features identified by QSAR models. We have also presented concepts on rule extraction methods that can unravel the black box and make interpretations of machine learning approaches. Furthermore, we reviewed the utility of various molecular descriptors in the post-genomic era of the biological data deluge. Moreover, the concept of conformal prediction have also been discussed as a novel and potentially powerful approach that can define the relative confidence or reliability of predictions made. The inherent heterogeneity and vagueness of details describing the construction of QSAR models in the literature may hinder further progress. Therefore, markup language such as QSAR-ML have been suggested as a means to solve the reproducibility of QSAR models by standardizing and demystifying the underlying details of QSAR models (i.e. addition of metadata on the source of the data set, the type of descriptors used, the machine learning employed, software names and version that are used, etc.) as well as making them exchangeable (i.e. in the context that they can be shared and readily be used by the scientific community). The availability of interpretable molecular descriptors and transparent machine learning methods presents a positive outlook for the utility of QSARs in drug discovery research. The application of several key sets of standards in QSAR modeling will further help to enhance their generalization and acceptance by the wider drug research community.

Acknowledgements This work is supported by a Research Career Development Grant (No. RSA5780031) to CN from the Thailand Research Fund; the New Scholar Research Grant (No. MRG5980220) to WS from the Thailand Research Fund; and the Swedish Research Links program (No. C0610701) to CN and JESW from the Swedish Research Council.

References

Alcal-Fdez, J., Fernandez, A., Luengo, J., Derrac, J., Garca, S., Snchez, L., et al. (2011). *Journal of Multiple-Valued Logic and Soft Computing*, *17*, 255.

Andrews, D. F. (1974). *Technometrics*, *16*(4), 523.

Andrews, R., Diederich, J., & Tickle, A. B. (1995). *Knowledge-Based Systems*, *8*(6), 373.

Augasta, M. G., & Kathirvalavakumar, T. (2012). *Proceedings of the International Conference on Pattern Recognition, Informatics and Medical Engineering, Salem, Tamilnadu* (pp. 21–23).

Backman, T. W., Cao, Y., & Girke, T. (2011). *Nucleic Acids Research*, *39*, W486.

Baell, J. B., & Holloway, G. A. (2010). *Journal of Medicinal Chemistry*, *53*(7), 2719.

Bajorath, J. (2014). *Molecular Informatics*, *33*(6–7), 438.

Balaban, A. T. (1982). *Chemical Physics Letters*, *89*(5), 399.

Barakat, N. H., & Bradley, A. P. (2007). *IEEE Transactions on Knowledge and Data Engineering*, *19*(6), 729.

Barakat, N., & Bradley, A. P. (2010). *Neurocomputing*, *74*(1), 178.

Barakat, N., & Diederich, J. (2004). *14th International Conference on Computer Theory and Applications (ICCTA'2004)*. Alexandria, Egypt.

Barakat, N., & Diederich, J. (2005). *International Journal of Computational Intelligence*, *2*(1), 59.

Benfenati, E., Toropov, A. A., Toropova, A. P., Manganaro, A., & Gonella, D. R. (2011). *Chemical Biology and Drug Design*, *77*(6), 471.

Bengio, Y. (2009). *Foundations and Trends in Machine Learning*, *2*(1), 1.

Borman, S. (1990). *Chemical and Engineering News*, *68*(8), 20.

Bordes, A., Chopra, S., & Weston, J. (2014). arXiv preprint: arXiv:1406.3676.
Bravi, G., Gancia, E., Mascagni, P., Pegna, M., Todeschini, R., & Zaliani, A. (1997). *Journal of Computer-Aided Molecular Design*, *11*(1), 79.
Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. New York, USA: CRC Press.
Breiman, L. (2001). *Machine Learning*, *45*(1), 5.
Broto, P., Moreau, G., & Vandycke, C. (1984). *European Journal of Medicinal Chemistry*, *19*(1), 66.
Brown, N., McKay, B., & Gasteiger, J. (2006). *Journal of Computer-Aided Molecular Design*, *20*(5), 333.
Brustle, M., Beck, B., Schindler, T., King, W., Mitchell, T., & Clark, T. (2002). *Journal of Medicinal Chemistry*, *45*(16), 3345.
Burges, C. J. (1998). *Data Mining and Knowledge Discovery*, *2*(2), 121.
Cao, D. S. (2017a). ChemDes: An integrated web-based platform for molecular descriptor and fingerprint computation. http://www.scbdd.com/chemdes.
Cao, D. S. (2017b). ChemoPy Descriptor Calculator. http://www.scbdd.com/chemopy_desc/index/.
Cao, Y., Charisi, A., Cheng, L. C., Jiang, T., & Girke, T. (2008). *Bioinformatics*, *24*(15), 1733.
Cao, D., Liang, Y., Xu, Q., Yun, Y., & Li, H. (2011). *Journal of Computer-Aided Molecular Design*, *25*(1), 67.
Cao, D. S., Xu, Q. S., Hu, Q. N., & Liang, Y. Z. (2013). *Bioinformatics*, *29*(8), 1092.
CAMO Software AS. (2017). The Unscrambler. http://www.camo.com/rt/Products/Unscrambler/unscrambler.html.
Capuzzi, S. J., Politi, R., Isayev, O., Farag, S., & Tropsha, A. (2016). *Frontiers of Environmental Science*, *4*, 3.
Certara. (2017). Benchware 3D Explorer. https://www.certara.com/software/molecular-modeling-and-simulation/benchware-3d-explorer/.
Chang, C. C., & Lin, C. J. (2011). *ACM Transactions on Intelligent Systems and Technology*, *2*(27), 1.
Charoenkwan, P., Shoombuatong, W., Lee, H. C., Chaijaruwanich, J., Huang, H. L., & Ho, S. Y. (2013). *PLoS One*, *8*(9), e72368.
Chemical Computing Group Inc. (2017). Molecular Operating Environment (MOE). https://www.chemcomp.com/MOE-Cheminformatics_and_QSAR.htm.
Chen, H., Carlsson, L., Eriksson, M., Varkonyi, P., Norinder, U., & Nilsson, I. (2013). *Journal of Chemical Information and Modeling*, *53*(6), 1324.
Cherkasov, A., Muratov, E. N., Fourches, D., Varnek, A., Baskin, I. I., Cronin, M., et al. (2014). *Journal of Medicinal Chemistry*, *57*(12), 4977.
Chiu, Y. Y., Lin, C. T., Huang, J. W., Hsu, K. C., Tseng, J. H., You, S. R., et al. (2013). *Nucleic Acids Research*, *41*(Database issue), D430.
Churchwell, C. J., Rintoul, M. D., Martin, S., Visco, D. P., Kotu, A., Larson, R. S., et al. (2004). *Journal of Molecular Graphics and Modelling*, *22*(4), 263.
Ciodaro, T., Deva, D., De Seixas, J., & Damazio, D. (2012). *Journal of Physics: Conference Series*, *368*, 012030. IOP Publishing.
Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). *Journal of Machine Learning Research*, *12*, 2493.
Consonni, V., Todeschini, R., & Pavan, M. (2002). *Journal of Chemical Information and Computer Sciences*, *42*(3), 682.
Cortes-Ciriano, I., Ain, Q. U., Subramanian, V., Lenselink, E. B., Mendez-Lucio, O., IJzerman, A. P., et al. (2015). *Medicinal Chemical Communications*, *6*, 24.
Cortes, C., & Vapnik, V. (1995). *Machine Learning*, *20*(3), 273.
Costello, J. C., Heiser, L. M., Georgii, E., Gonen, M., Menden, M. P., Wang, N. J., et al. (2014). *Nature Biotechnology*, *32*(12), 1202.
Cox, R., Green, D. V., Luscombe, C. N., Malcolm, N., & Pickett, S. D. (2013). *Journal of Computer-Aided Molecular Design*, *27*(4), 321.

Cramer, R. D., Patterson, D. E., & Bunce, J. D. (1988). *Journal of the American Chemical Society*, *110*(18), 5959.
Craven, M. W., & Shavlik, J. W. (1996). *Advances in neural information processing systems* (pp. 24–30). Cambridge, USA: MIT Press.
Cros, A. F. A. (1863). Action de lalcohol amylique sur lorganisme. Ph.D. thesis, University of Strasbourg.
Crum-Brown, A., & Fraser, T. (1868). *Transactions of the Royal Society of Edinburgh*, *25*, 151.
Danishuddin, A. U. K. (2016). *Drug Discovery Today*, *21*(8), 1291.
Dearden, J., Cronin, M., & Kaiser, K. (2009). *SAR and QSAR in Environmental Research*, *20*(3–4), 241.
de Vries, S. J., van Dijk, M., & Bonvin, A. M. (2010). *Nature Protocols*, *5*(5), 883.
Destrero, A., Mosci, S., De Mol, C., Verri, A., & Odone, F. (2009). *Computational Management Science*, *6*(1), 25.
Devinyak, O., Havrylyuk, D., & Lesyk, R. (2014). *Journal of Computer-Aided Molecular Design*, *54*, 194.
Dimova, D., & Bajorath, J. (2016). *Molecular Informatics*, *35*(5), 181.
Dimitrov, S. D., Diderici, R., Sobanski, T., Pavlov, T. S., Chapkov, G. V., Chapkonov, A. S., et al. (2016). *SAR and QSAR in Environmental Research*, 1–17.
Dong, J., Cao, D. S., Miao, H. Y., Liu, S., Deng, B. C., Yun, Y. H., et al. (2015). *Journal of Cheminformatics*, *7*, 60.
Doweyko, A. M. (2004). *Journal of Computer-Aided Molecular Design*, *18*(7), 587.
Doweyko, A. M. (2008). *Journal of Computer-Aided Molecular Design*, *22*(2), 81.
Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A., Vapnik, V. (1996). *Proceedings of the 9th International Conference on Neural Information Processing Systems, NIPS'96* (pp. 155–161). Cambridge, MA, USA: MIT Press.
DTAI Research Group (2017). DMax Chemistry Assistant. https://dtai.cs.kuleuven.be/software/dmax/.
Durant, J. L., Leland, B. A., Henry, D. R., & Nourse, J. G. (2002). *Journal of Chemical Information and Computer Sciences*, *42*(6), 1273.
Ebrahimi, E., Monjezi, M., Khalesi, M. R., & Armaghani, D. J. (2016). *Bulletin of Engineering Geology and the Environment*, *75*(1), 27.
Eklund, M., Norinder, U., Boyer, S., & Carlsson, L. (2012). In L. Iliadis, I. Maglogiannis, H. Papadopoulos, K. Karatzas, & S. Sioutas (Eds.), *Artificial Intelligence Applications and Innovations: AIAI 2012 International Workshops: AIAB, AIeIA, CISE, COPA, IIVC, ISQL, MHDW, and WADTMB, Halkidiki, Greece, September 27–30, 2012, Proceedings, Part II* (pp. 166–175). Berlin, Germany: Springer.
Eklund, M., Norinder, U., Boyer, S., & Carlsson, L. (2014). *Journal of Chemical Information and Modeling*, *54*(3), 837.
Eriksson, M., Chen, H., Carlsson, L., Nissink, J. W., Cumming, J. G., & Nilsson, I. (2014). *Journal of Chemical Information and Modeling*, *54*(4), 1117.
Esbensen, K. H., & Geladi, P. (2010). *Journal of Chemometrics*, *24*(3–4), 168.
Faulon, J. L. (1994). *Journal of Chemical Information and Computer Sciences*, *34*(5), 1204.
Faulon, J. L. (1996). *Journal of Chemical Information and Computer Sciences*, *36*(4), 731.
Faulon, J. L., Churchwell, C. J., & Visco, D. P. (2003). *Journal of Chemical Information and Computer Sciences*, *43*(3), 721.
Faulon, J. L., Collins, M. J., & Carr, R. D. (2004). *Journal of Chemical Information and Computer Sciences*, *44*(2), 427.
Faulon, J. L., Brown, W. M., & Martin, S. (2005). *Journal of Computer-Aided Molecular Design*, *19*(9–10), 637.
Feng, B. Y., Shelat, A., Doman, T. N., Guy, R. K., & Shoichet, B. K. (2005). *Nature Chemical Biology*, *1*(3), 146.
Feng, B. Y., Simeonov, A., Jadhav, A., Babaoglu, K., Inglese, J., Shoichet, B. K., et al. (2007). *Journal of Medicinal Chemistry*, *50*(10), 2385.

Feng, B. Y., & Shoichet, B. K. (2006). *Nature Protocols*, *1*(2), 550.
Filimonov, D. A., Zakharov, A. V., Lagunin, A. A., & Poroikov, V. V. (2009). *SAR and QSAR in Environmental Research*, *20*(7), 679.
Filimonov, D. A., Lagunin, A. A., Gloriozova, T. A., Rudik, A. V., Druzhilovskii, D. S., Pogodin, P. V., et al. (2014). *Chemistry of Heterocyclic Compounds*, *50*(3), 444.
Frank, E., Hall, M. & Trigg, L. Weka. http://www.cs.waikato.ac.nz/ml/weka/.
Free, S. M., & Wilson, J. W. (1964). *Journal of Medicinal Chemistry*, *7*(4), 395.
Fu, X., Ong, C., Keerthi, S., Hung, G. G., & Goh, L. (2004). In *Proceedings of IEEE International Joint Conference on Neural Networks* (pp. 291–296). Budapest, Hungary: IEEE.
Fujita, T., & Winkler, D. A. (2016). *Journal of Chemical Information and Modeling*, *56*(2), 269.
Fung, G., Sandilya, S., & Rao, R. B. (2005). *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (pp. 32–40). New York, USA: ACM.
Gallup, G. A., Gilkerson, W., & Jones, M. (1952). *Transactions of the Kansas Academy of Science*, *55*(2), 232.
Gao, H., Katzenellenbogen, J. A., Garg, R., & Hansch, C. (1999). *Chemical Reviews*, *99*(3), 723.
Garcia-Jacas, C. R., Marrero-Ponce, Y., Acevedo-Martinez, L., Barigye, S. J., Valdes-Martini, J. R., & Contreras-Torres, E. (2014). *Journal of Computational Chemistry*, *35*(18), 1395.
Garg, R., Gupta, S. P., Gao, H., Babu, M. S., Debnath, A. K., & Hansch, C. (1999). *Chemical Reviews*, *99*(12), 3525.
Garg, R., Kurup, A., Mekapati, S. B., & Hansch, C. (2003). *Chemical Reviews*, *103*(3), 703.
Geronikaki, A. A., Lagunin, A. A., Hadjipavlou-Litina, D. I., Eleftheriou, P. T., Filimonov, D. A., Poroikov, V. V., et al. (2008). *Journal of Medicinal Chemistry*, *51*(6), 1601.
Girke, T. (2017). ChemmineR: Cheminformatics toolkit for R. https://www.bioconductor.org/packages/release/bioc/html/ChemmineR.html.
Gleeson, M. P. (2008). *Journal of Medicinal Chemistry*, *51*(4), 817.
Gobbi, M., Beeg, M., Toropova, M. A., Toropov, A. A., & Salmona, M. (2016). *Toxicology Letters*, *250*, 42.
Golbraikh, A., Fourches, D., Sedykh, A., Muratov, E., Liepina, I., & Tropsha, A. (2014). *Practical aspects of computational chemistry III* (pp. 187–230). Boston, USA: Springer.
Gong, R., Huang, S. H., & Chen, T. (2008). *IEEE Transactions on Industrial Informatics*, *4*(3), 198.
Gonzlez, M. P., Tern, C., Fall, Y., Teijeira, M., & Besada, P. (2005). *Bioorganic and Medicinal Chemistry*, *13*(3), 601.
Goodarzi, M., Heyden, Y. V., & Funar-Timofei, S. (2013). *Trends in Analytical Chemistry*, *42*, 49.
Gozalbes, R., Doucet, J. P., & Derouin, F. (2002). *Current Drug Targets Infectious Disorders*, *2*(1), 93.
Gramatica, P., Chirico, N., Papa, E., Cassani, S., & Kovarich, S. (2013). *Journal of Computational Chemistry*, *34*(24), 2121.
Guha, R. (2017). CDK Descriptor Calculator GUI (version 1.4. 6). http://www.rguha.net/code/java/cdkdesc.html.
Guha, R., & Van Drie, J. H. (2008). *Journal of Chemical Information and Modeling*, *48*(8), 1716.
Gupta, A., Park, S., & Lam, S. M. (1999). *IEEE Transactions on Knowledge and Data Engineering*, *11*(6), 985.
Gütlein, M., Helma, C., Karwath, A., & Kramer, S. (2013). *Molecular Informatics*, *32*(5–6), 516.
Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). *Machine Learning*, *46*(1–3), 389.
Guyon, I. (2003). *Journal of Machine Learning Research*, *3*, 1157.
Hadjipavlou-Litina, D., Garg, R., & Hansch, C. (2004). *Chemical Reviews*, *104*(9), 3751.
Hall, M. A. (1999). Correlation-based feature selection for machine learning. Ph.D. thesis, The University of Waikato.
Hammett, L. P. (1937). *Journal of the American Chemical Society*, *59*(1), 96.
Hansch, C., Maloney, P. P., Fujita, T., & Muir, R. M. (1962). *Nature*, *194*, 178.
Hansch, C., Leo, A., & Taft, R. (1991). *Chemical Reviews*, *91*(2), 165.
Hansch, C., Hoekman, D., & Gao, H. (1996). *Chemical Reviews*, *96*(3), 1045.

Hansch, C., Hoekman, D., Leo, A., Weininger, D., & Selassie, C. D. (2002). *Chemical Reviews*, *102*(3), 783.
Hansch, C. (2011). *Journal of Computer-Aided Molecular Design*, *25*(6), 495.
Hansch, C., & Gao, H. (1997). *Chemical Reviews*, *97*(8), 2995.
Harding, A. P., Wedge, D. C., & Popelier, P. L. (2009). *Journal of Chemical Information and Modeling*, *49*(8), 1914.
Hawkins, D. M., Basak, S. C., & Mills, D. (2003). *Journal of Chemical Information and Computer Sciences*, *43*(2), 579.
Héberger, K., & Rajkó, R. (2002). *Journal of Chemometrics*, *16*(8), 436.
Heberger, K., & Skrbic, B. (2012). *Analytica Chimica Acta*, *716*, 92.
Helland, I. S. (1988). *Communication in Statistics: Simulation and Computation*, *17*(2), 581.
Helland, I. S. (2001). *Chemometrics and Intelligent Laboratory*, *58*(2), 97.
Helmstaedter, M., Briggman, K. L., Turaga, S. C., Jain, V., Seung, H. S., & Denk, W. (2013). *Nature*, *500*(7461), 168.
Hinselmann, G., Rosenbaum, L., Jahn, A., Fechner, N., & Zell, A. (2011). *Journal of Cheminformatics*, *3*(1), 3.
Hinselmann, G., Rosenbaum, L., Jahn, A., Fechner, N., & Zell, A. (2017). jCompoundMapper: An open source java library and command-line tool for chemical fingerprints. http://jcompoundmapper.sourceforge.net/.
Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). *Neural Computing*, *18*(7), 1527.
Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. R. (2012). arXiv preprint arXiv:1207.0580.
Hosmer, D. W, Jr., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (pp. 1–33). New Jersey, USA: Wiley.
Hu, X., Hu, Y., Vogt, M., Stumpfe, D., & Bajorath, J. (2012). *Journal of Chemical Information and Modeling*, *52*(5), 1138.
IBM. (2017). IBM SPSS Software. http://www.ibm.com/analytics/us/en/technology/spss/.
Jarvis, R. M., Broadhurst, D., Johnson, H., O'Boyle, N. M., & Goodacre, R. (2006). *Bioinformatics*, *22*(20), 2565.
Jelfs, S., Ertl, P., & Selzer, P. (2007). *Journal of Chemical Information and Modeling*, *47*(2), 450.
Johnson, S. R. (2008). *Journal of Chemical Information and Modeling*, *48*(1), 25.
Jolliffe, I. (2002). *Principal component analysis*. New York, USA: Springer.
Katritzky, A. R., Kuanar, M., Slavov, S., Hall, C. D., Karelson, M., Kahn, I., et al. (2010). *Chemical Reviews*, *110*(10), 5714.
Khan, M. T., & Sylte, I. (2007). *Current Drug Discovery Technologies*, *4*(3), 141.
Kier, L. B., & Hall, L. H. (1976). *Molecular connectivity in chemistry and drug research*. New York, USA: Academic Press.
Kim, K. H. (2007a). *Journal of Computer-Aided Molecular Design*, *21*(8), 421.
Kim, K. H. (2007b). *Journal of Computer-Aided Molecular Design*, *21*(1–3), 63.
Kim, D., & Lee, J. (2000). In López de Mántaras and Plaza (Eds.), *Proceedings of the 11th European conference on machine learning* (pp. 211–219). London, UK: Springer.
Kohonen, T. (2017). SOM: Self-Organization Map. http://www.cis.hut.fi/somtoolbox/.
Krasavin, M. (2015). *European Journal of Medicinal Chemistry*, *97*, 525.
Kubinyi, H. (1988). *Quantitative Structure-Activity Relationship*, *7*(3), 121.
Kubinyi, H. (1993). *3D QSAR in drug design: Volume 1: Theory methods and applications* (Vol. 1). Dordrecht, Netherlands: Springer Science & Business Media.
Kubinyi, H. (2006). In S. Ekins (Ed.) *Computer applications in pharmaceutical research and development* (pp. 377–424). New Jersey, USA: Wiley.
Kufareva, I., & Abagyan, R. (2008). *Journal of Medicinal Chemistry*, *51*(24), 7921.
Kuhn, T., Willighagen, E. L., Zielesny, A., & Steinbeck, C. (2010). *BMC Bioinformatics*, *11*, 159.
Kurgan, L., Razib, A. A., Aghakhani, S., Dick, S., Mizianty, M., & Jahandideh, S. (2009). *BMC Structural Biology*, *9*, 50.
Kurup, A., Garg, R., & Hansch, C. (2000). *Chemical Reviews*, *100*(3), 909.

Kurup, A., Garg, R., Carini, D. J., & Hansch, C. (2001). *Chemical Reviews*, *101*(9), 2727.
Kurup, A., Garg, R., & Hansch, C. (2001). *Chemical Reviews*, *101*(8), 2573.
Kvasnicka, V., & Pospichal, J. (1996). *Journal of Chemical Information and Computer Sciences*, *36*(3), 516.
Lawrence, D., et al. (1991). *Handbook of genetic algorithms*. New York, USA: Van No Strand Reinhold.
Leung, M. K., Xiong, H. Y., Lee, L. J., & Frey, B. J. (2014). *Bioinformatics*, *30*(12), i121.
Li, Q., Wang, Y., & Bryant, S. H. (2009). *Bioinformatics*, *25*(24), 3310.
Lipnick, R. L. (1991). *Studies of narcosis*. Dordrecht, Netherlands: Springer.
Liu, S. S., Yin, C. S., Li, Z. L., & Cai, S. X. (2001). *Journal of Chemical Information and Computer Sciences*, *41*(2), 321.
Liu, H., & Motoda, H. (2007). *Computational methods of feature selection*. Boca Raton, Florida: CRC Press.
Ma, J., Sheridan, R. P., Liaw, A., Dahl, G. E., & Svetnik, V. (2015). *Journal of Chemical Information and Modeling*, *55*(2), 263.
Manallack, D. T. (2008). *Perspectives in Medicinal Chemistry*, *1*, 25.
Maplesoft. (2017). Maple. https://www.maplesoft.com/products/Maple/.
Martens, D., Baesens, B., Van Gestel, T., & Vanthienen, J. (2007). *European Journal of Operational Research*, *183*(3), 1466.
Masand, V. H., Toropov, A. A., Toropova, A. P., & Mahajan, D. T. (2014). *Current Computer-Aided Drug Design*, *10*, 75.
Mazanetz, M. P., Marmon, R. J., Reisser, C. B., & Morao, I. (2012). *Current Topics in Medicinal Chemistry*, *12*(8), 1965.
McGovern, S. L., Caselli, E., Grigorieff, N., & Shoichet, B. K. (2002). *Journal of Medicinal Chemistry*, *45*(8), 1712.
Medina Marrero, R., Marrero-Ponce, Y., Barigye, S. J., Echeverria Diaz, Y., Acevedo-Barrios, R., Casanola-Martin, G. M., et al. (2015). *SAR and QSAR in Environmental Research*, *26*(11), 943.
Miller, B. T., Singh, R. P., Klauda, J. B., Hodoscek, M., Brooks, B. R., & Woodcock, H. L. (2008). *Journal of Chemical Information and Modeling*, *48*(9), 1920.
Molplex Ltd., & Sykora, V. (2017). Chemical Descriptors Library (CDL). https://sourceforge.net/projects/cdelib/.
Morgenthaler, M., Schweizer, E., Hoffmann-Roder, A., Benini, F., Martin, R. E., Jaeschke, G., et al. (2007). *ChemMedChem*, *2*(8), 1100.
Nantasenamat, C., Naenna, T., Isarankura-Na-Ayudhya, C., & Prachayasittikul, V. (2005). *Journal of Computer-Aided Molecular Design*, *19*(7), 509.
Nantasenamat, C., Isarankura-Na-Ayudhya, C., Naenna, T., & Prachayasittikul, V. (2007a). *Biosensors and Bioelectronics*, *22*(12), 3309.
Nantasenamat, C., Isarankura-Na-Ayudhya, C., Tansila, N., Naenna, T., & Prachayasittikul, V. (2007b). *Journal of Computational Chemistry*, *28*(7), 1275.
Nantasenamat, C., Piacham, T., Tantimongcolwat, T., Naenna, T., Isarankura-Na-Ayudhya, C., & Prachayasittikul, V. (2008). *Journal of Biological Systems*, *16*(02), 279.
Nantasenamat, C., Isarankura-Na-Ayudhya, C., Naenna, T., & Prachayasittikul, V. (2009). *EXCLI Journal*, *8*(7), 74.
Nantasenamat, C., Isarankura-Na-Ayudhya, C., & Prachayasittikul, V. (2010). *Expert Opinion on Drug Discovery*, *5*(7), 633.
Nantasenamat, C., Worachartcheewan, A., Jamsak, S., Preeyanon, L., Shoombuatong, W., Simeon, S., et al. (2015). In H. Cartwright (Ed.), *Artificial neural networks* (pp. 119–147). New York, NY, USA: Springer.
Nantasenamat, C., & Prachayasittikul, V. (2015). *Expert Opinion on Drug Discovery*, *10*(4), 321.
NeuralWare. (2017). NeuralWare. http://www.neuralware.com/.
Núñez, H., Angulo, C., & Català, A. (2002). *10th European Symposium on Artificial Neural Networks (ESANN)*, pp. 107–112.
O'Boyle, N. M., & Hutchison, G. R. (2008). *Chemistry Central Journal*, *2*, 24.

O'Boyle, N. M., Morley, C., & Hutchison, G. R. (2008). *Chemistry Central Journal*, *2*, 5.
O'Boyle, N. M., Morley, C. & Hutchison, G. R. (2017a). Pybel. https://openbabel.org/docs/dev/UseTheLibrary/Python_Pybel.html.
O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., & Hutchison, G. R. (2011). *Journal of Cheminformatics*, *3*, 33.
O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., & Hutchison, G. R. (2017b). Open Babel: The open source chemistry toolbox. http://openbabel.org/.
Oldham, M. C., Konopka, G., Iwamoto, K., Langfelder, P., Kato, T., Horvath, S., et al. (2008). *Nature Neuroscience*, *11*(11), 1271.
Patani, G. A., & LaVoie, E. J. (1996). *Chemical Reviews*, *96*(8), 3147.
Patlewicz, G., Jeliazkova, N., Safford, R. J., Worth, A. P., & Aleksiev, B. (2008). *SAR and QSAR in Environmental Research*, *19*(5–6), 495.
Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M. et al. (2017). Scikit-learn. http://scikit-learn.org/.
Peng, H., Long, F., & Ding, C. (2005). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(8), 1226.
Poroikov, V. V., Filimonov, D. A., Ihlenfeldt, W. D., Gloriozova, T. A., Lagunin, A. A., Borodina, Y. V., et al. (2003). *Journal of Chemical Information and Computer Sciences*, *43*(1), 228.
Prachayasittikul, V., Worachartcheewan, A., Shoombuatong, W., Songtawee, N., Simeon, S., Prachayasittikul, V., et al. (2015). *Current Topics in Medicinal Chemistry*, *15*(18), 1780.
Prathipati, P., Pandey, G., & Saxena, A. K. (2005). *Journal of Chemical Information and Modeling*, *45*(1), 136.
Prathipati, P., Dixit, A., & Saxena, A. K. (2007). *Journal of Computer-Aided Molecular Design*, *92*, 29.
Prathipati, P., Ma, N. L., & Keller, T. H. (2008). *Journal of Chemical Information and Modeling*, *48*(12), 2362.
Prathipati, P., & Mizuguchi, K. (2016a). *Current Topics in Medicinal Chemistry*, *16*(9), 1009.
Prathipati, P., & Mizuguchi, K. (2016b). *Journal of Chemical Information and Modeling*, *56*(6), 974.
Prathipati, P., & Saxena, A. K. (2005). *Journal of Computer-Aided Molecular Design*, *19*(2), 93.
Pudil, P., Novovičová, J., & Kittler, J. (1994). *Pattern Recognition Letters*, *15*(11), 1119.
Ponce, Y. M. (2017a). QuBiLs-MAS. http://tomocomd.com/qubils-mas.
Ponce, Y. M. (2017b). QuBiLs-MIDAS. http://tomocomd.com/qubils-midas.
Qiu, T., Qiu, J., Feng, J., Wu, D., Yang, Y., Tang, K., et al. (2016). *Briefings in Bioinformatics*, *18*(1), 125.
Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Francisco, USA: Morgan Kaufmann Publishers Inc.
RapidMiner, Inc. (2017). RapidMiner. https://rapidminer.com/.
rcdk: Interface to the CDK Libraries. https://cran.r-project.org/web/packages/rcdk/index.html.
Rácz, A., Bajusz, D., & Héberger, K. (2015). *SAR and QSAR in Environmental Research*, *26*(7–9), 683.
Radoux, C. J., Olsson, T. S., Pitt, W. R., Groom, C. R., & Blundell, T. L. (2016). *Journal of Medicinal Chemistry*, *59*(9), 4314.
Raiko, T., Valpola, H., & LeCun, Y. (2012). In *Proceedings of the Fifteenth Internation Conference on Artificial Intelligence and Statistics (AISTATS). JMLR Workshop and Conference Proceedings* (Vol. 22, pp. 924–932).
Randic, M. (1975). *Journal of the American Chemical Society*, *97*(23), 6609.
Ripley, B. D. (2017). The R project in statistical computing. https://www.stats.ox.ac.uk/pub/bdr/LTSN-R.pdf.
Rogers, D., & Hahn, M. (2010). *Journal of Chemical Information and Modeling*, *50*(5), 742.
Rosenbaum, L., Hinselmann, G., Jahn, A., & Zell, A. (2011). *Journal of Cheminformatics*, *3*(1), 11.

Rucker, C., Rucker, G., & Meringer, M. (2007). *Journal of Chemical Information and Modeling*, *47*(6), 2345.
Rueda, M., Bottegoni, G., & Abagyan, R. (2009). *Journal of Chemical Information and Modeling*, *49*(3), 716.
Rueda, M., Bottegoni, G., & Abagyan, R. (2010). *Journal of Chemical Information and Modeling*, *50*(1), 186.
Sahigara, F., Mansouri, K., Ballabio, D., Mauri, A., Consonni, V., & Todeschini, R. (2012). *Molecules*, *17*(5), 4791.
Sahigara, F., Ballabio, D., Todeschini, R., & Consonni, V. (2013). *Journal of Cheminformatics*, *5*(1), 27.
Saito, K., & Nakano, R. (1988). In *IEEE International Conference on Neural Networks, 1988* (pp. 255–262). IEEE.
SAS Institute Inc. (2017). SAS Enterprise Miner. http://www.sas.com/en_th/software/analytics/enterprise-miner.html.
Saxena, A. K., & Prathipati, P. (2003). *SAR and QSAR in Environmental Research*, *14*(5–6), 433.
Saxena, A. K., & Prathipati, P. (2006). *SAR and QSAR in Environmental Research*, *17*(4), 371.
Schuffenhauer, A., Brown, N., Selzer, P., Ertl, P., & Jacoby, E. (2006). *Journal of Chemical Information and Modeling*, *46*(2), 525.
Seebeck, B., Wagener, M., & Rarey, M. (2011). *ChemMedChem*, *6*(9), 1630.
Selassie, C. D., Garg, R., Kapur, S., Kurup, A., Verma, R. P., Mekapati, S. B., et al. (2002). *Chemical Reviews*, *102*(7), 2585.
Sestito, S., & Dillon, T. (1992). *Proceedings of the 12th International Conference on Expert Systems and their Applications (AVIGNON'92)* (pp. 645–656).
Setiono, R., & Liu, H. (1995). *Proceedings of the 14th International Joint Conference on Artificial Intelligence—Volume 1, IJCAI'95* (pp. 480–485). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
Setiono, R., & Liu, H. (1997). *Neurocomputing*, *17*(1), 1.
Setiono, R., Leow, W. K., & Zurada, J. M. (2002). *IEEE Transactions on Neural Networks*, *13*(3), 564.
Shafer, G., & Vovk, V. (2008). *Journal of Machine Learning Research*, *9*, 371.
Sheridan, R. P. (2015). *Journal of Chemical Information and Modeling*, *55*(6), 1098.
Sheridan, R. P., & Kearsley, S. K. (1995). *Journal of Chemical Information and Computer Sciences*, *35*(2), 310.
Shoombuatong, W., Prachayasittikul, V., Prachayasittikul, V., & Nantasenamat, C. (2015). *EXCLI Journal*, *14*, 452.
Shoombuatong, W., Prachayasittikul, V., Anuwongcharoen, N., Songtawee, N., Monnor, T., Prachayasittikul, S., et al. (2015). *Drug Design. Development and Therapy*, *9*, 4515.
Siedlecki, W., & Sklansky, J. (1988). *International Journal of Pattern Recognition and Artificial Intelligence*, *2*(02), 197.
Simeon, S., Möller, R., Almgren, D., Li, H., Phanus-umporn, C., Prachayasittikul, V., et al. (2016a). *Chemometrics and Intelligent Laboratory Systems*, *151*, 51.
Simeon, S., Spjuth, O., Lapins, M., Nabu, S., Anuwongcharoen, N., Prachayasittikul, V., et al. (2016b). *PeerJ*, *4*, e1979.
Simpson, P. K. (1990). *Artificial neural system: Foundation, paradigm, application and implementations*. Pennsylvania, USA: Windcrest/McGraw-Hill.
Sippl, W. (2006). *Molecular interaction fields* (pp. 145–170). KGaA: Wiley-VCH Verlag GmbH & Co.
Skvortsova, M. I., Baskin, I. I., Slovokhotova, O. L., Palyulin, V. A., & Zefirov, N. S. (1993). *Journal of Chemical Information and Computer Sciences*, *33*(4), 630.
Sliwoski, G., Mendenhall, J., & Meiler, J. (2016). *Journal of Computer-Aided Molecular Design*, *30*(3), 209.
Song, M., Breneman, C. M., Bi, J., Sukumar, N., Bennett, K. P., Cramer, S., et al. (2002). *Journal of Chemical Information and Computer Sciences*, *42*(6), 1347.

Spjuth, O., Willighagen, E. L., Guha, R., Eklund, M., & Wikberg, J. E. (2010). *Journal of Cheminformatics*, *2*, 5.
Spyrakis, F., & Cavasotto, C. N. (2015). *Archives of Biochemistry and Biophysics*, *583*, 105.
Stalring, J. C., Carlsson, L. A., Almeida, P., & Boyer, S. (2011). *Journal of Cheminformatics*, *3*, 28.
Standfuss, J., Edwards, P. C., D'Antona, A., Fransen, M., Xie, G., Oprian, D. D., et al. (2011). *Nature*, *471*(7340), 656.
Stumpfe, D., Hu, Y., Dimova, D., & Bajorath, J. (2014). *Journal of Medicinal Chemistry*, *57*(1), 18.
Sushko, I., Novotarskyi, S., Krner, R., Pandey, A. K., Rupp, M., et al. (2011). *Journal of Computer-Aided Molecular Design*, *25*(6), 533.
Sutskever, I., Vinyals, O., & Le, Q. V. (2014). In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, K., & Q. Weinberger (Ed.) *Advances in neural information processing systems 27* (pp. 3104–3112). Curran Associates, Inc.
The MathWorks, Inc. (2017a). Neural Network Toolbox. http://www.mathworks.com/products/neural-network/.
The MathWorks, Inc. (2017b). MATLAB. https://www.mathworks.com/products/matlab/.
TIBCO Software Inc. (2017). TIBCO Spotfire S+. http://spotfire.tibco.com/discover-spotfire/who-uses-spotfire/by-role/statisticians.
Tarca, A. L., Than, N. G., & Romero, R. (2013). *Systems Biomedicine*, *1*(4), 217.
Taskinen, J., & Yliruusi, J. (2003). *Advanced Drug Delivery Reviews*, *55*(9), 1163.
Thornber, C. W. (1979). *Chemical Society Reviews*, *8*(4), 563.
Thorne, N., Auld, D. S., & Inglese, J. (2010). *Current Opinion in Chemical Biology*, *14*(3), 315.
Thrun, S. (1993). *Extracting provably correct rules from artificial neural networks*. Bonn, Germany: University of Bonn.
Todeschini, R., & Consonni, V. (2008). *Handbook of molecular descriptors*. Weinheim, Germany: Wiley-VCH Verlag GmbH.
Toropov, A. A., Toropova, A. P., Benfenati, E., Leszczynska, D., & Leszczynski, J. (2010). *Journal of Computational Chemistry*, *31*(2), 381.
Toropov, A. A., Toropova, A. P., Puzyn, T., Benfenati, E., Gini, G., Leszczynska, D., et al. (2013). *Chemosphere*, *92*(1), 31.
Toropova, A. P., & Toropov, A. A. (2014). *European Journal of Pharmaceutical Sciences*, *52*, 21.
Toropov, A. A., & Benfenati, E. (2007a). *European Journal of Medicinal Chemistry*, *42*(5), 606.
Toropov, A. A., & Benfenati, E. (2007b). *Current Drug Discovery Technologies*, *4*(2), 77.
Tosco, P., Balle, T., & Shiri, F. (2011). *Journal of Computer-Aided Molecular Design*, *25*(8), 777.
Tropsha, A., Gramatica, P., & Gombar, V. K. (2003). *QSAR and Combinatorial Science*, *22*(1), 69.
Tropsha, A. (2010). *Molecular Informatics*, *29*(6–7), 476.
Venkatasubramanian, V., Chan, K., & Caruthers, J. M. (1995). *Journal of Chemical Information and Computer Sciences*, *35*(2), 188.
Verma, R. P., & Hansch, C. (2005). *Bioorganic and Medicinal Chemistry*, *13*(15), 4597.
Verma, R. P., & Hansch, C. (2009). *Chemical Reviews*, *109*(1), 213.
Visco, D. P., Pophale, R. S., Rintoul, M. D., & Faulon, J. L. (2002). *Journal of Molecular Graphics and Modelling*, *20*(6), 429.
Walker, T., Grulke, C. M., Pozefsky, D., & Tropsha, A. (2010). *Bioinformatics*, *26*(23), 3000.
Wang, L. X., & Mendel, J. M. (1992). IEEE Transactions on Systems. *Man and Cybernetics: Systems*, *22*(6), 1414.
Wei, D. B., Zhang, A. Q., Han, S. K., & Wang, L. S. (2001). *SAR and QSAR in Environmental Research*, *12*(5), 471.
Weis, D. C., Faulon, J. L., LeBorne, R. C., & Visco, D. P. (2005). *Industrial and Engineering Chemistry*, *44*(23), 8883.
Wong, W. W., & Burkowski, F. J. (2009). *Journal of Cheminformatics*, *1*, 4.
Worachartcheewan, A., Nantasenamat, C., Naenna, T., Isarankura-Na-Ayudhya, C., & Prachayasittikul, V. (2009). *European Journal of Medicinal Chemistry*, *44*(4), 1664.

Worachartcheewan, A., Mandi, P., Prachayasittikul, V., Toropova, A. P., Toropov, A. A., & Nantasenamat, C. (2014). *Chemometrics and Intelligent Laboratory Systems*, *138*, 120.
Worachartcheewan, A., Prachayasittikul, V., Toropova, A. P., Toropov, A. A., & Nantasenamat, C. (2015). *Molecular Diversity*, *19*(4), 955.
Worth, A. P., & Cronin, M. T. (2004). *Alternatives to Laboratory Animals*, *32*, 703.
Xiao, N., Cao D. S., & Xu, Q. (2017). Rcpi: Toolkit for compound-protein interaction in drug discovery. http://bioconductor.org/packages/release/bioc/html/Rcpi.html.
Xing, L., Glen, R. C., & Clark, R. D. (2003). *Journal of Chemical Information and Computer Sciences*, *43*(3), 870.
Yager, R. R., & Filev, D. P. (1994). *Journal of Intelligent & Fuzzy Systems*, *2*(3), 209.
Yap, C. W. (2011). *Journal of Computational Chemistry*, *32*(7), 1466.
Yap, C. W. (2017). PaDEL-Descriptor. http://www.yapcwsoft.com/dd/padeldescriptor.
Zakharov, A. V., Peach, M. L., Sitzmann, M., & Nicklaus, M. C. (2014). *Journal of Chemical Information and Modeling*, *54*(3), 705.
Zell, A., Mache, N., Hubner, R., Mamier, G., Vogt, M., Döring, S., et al. (2017). SNNS: Stuttgast neural network simulator. http://www.ra.cs.uni-tuebingen.de/SNNS/.
Zhao, Z., Wu, H., Wang, L., Liu, Y., Knapp, S., Liu, Q., et al. (2014). *ACS Chemical Biology*, *9*(6), 1230.
Zhang, Y., Su, H., Jia, T., & Chu, J. (2005). In Ho T. B., Cheung D., Liu H. (Eds.), *Advances in knowledge discovery and data mining: 9th Pacific-Asia conference on knowledge discovery and data mining* (pp. 61–70). Berlin/Heidelberg, Germany: Springer.
Zhou, Z. H., & Chen, S. F. (2002). *Journal of Research and Development*, *39*(4), 398.
Zurada, J. M. (1992). *Introduction to artificial neural systems* (Vol. 8). Minnesota, USA: West Publishing Co.