JOURNAL OF
CHEMICAL INFORMATION
AND MODELING

Article

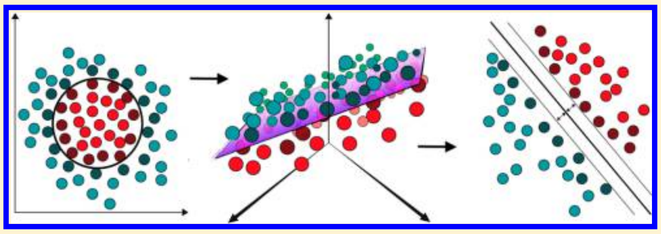
pubs.acs.org/jcim

Relevance Vector Machines: Sparse Classification Methods for QSAR

Frank R. Burden[†,‡] and David A. Winkler*[,†,‡,§,∥]

[†]CSIRO Manufacturing Flagship, Clayton South, Victoria 3169, Australia
[‡]Monash Institute of Pharmaceutical Sciences, Parkville, Victoria 3052, Australia
[§]Latrobe Institute for Molecular Science, Bundoora, Victoria 3083, Australia
[∥]School of Chemical and Physical Sciences, Flinders University, Bedford Park, South Australia 5042, Australia

ABSTRACT: Sparse machine learning methods have provided substantial benefits to quantitative structure property modeling, as they make model interpretation simpler and generate models with improved predictivity. Sparsity is usually induced via Bayesian regularization using sparsity-inducing priors and by the use of expectation maximization algorithms with sparse priors. The focus to date has been on using sparse methods to model continuous data and to carry out sparse feature selection. We describe the relevance vector machine (RVM), a sparse version of the support vector machine (SVM) that is one of the most widely used classification machine learning methods in QSAR and QSPR. We illustrate the superior properties of the RVM by modeling eight data sets using SVM, RVM, and another sparse Bayesian machine learning method, the Bayesian regularized artificial neural network with Laplacian prior (BRANNLP). We show that RVM models are substantially sparser than the SVM models and have similar or superior performance to them.



INTRODUCTION

The types of data that are modeled using quantitative structure−activity relationships (QSAR) or quantitative structure−property relationship (QSPR) methods are becoming increasingly diverse as the materials and nanotechnology fields adopt these methods.[1−5] Data sets can be broadly classified into two types: continuous (where the dependent variable can take any value) and categorical (where the dependent variable adopts a finite, usually small number of classes).

There are a large number of methods for generating mathematical models relating molecular properties to biological, physicochemical, or other useful properties of interest. Finding the optimum complexity for a QSAR model requires balancing bias (model being too simple to capture the underlying relationships in the data) and variance (where the model is too complex and over fits the data). Optimally sparse models have better predictive power than those that are less sparse (or too sparse). We, and others have shown that methods such as Bayesian regularization can automatically control the model complexity in an optimal way.[6−9] While the problem of finding robust, optimally sparse models relating independent model parameters (molecular descriptors) to continuous dependent variables has been essentially solved, there is a need for improvement with classification models. The most widely used classification method is the Support Vector Machine (SVM) first reported by Cortes and Vapnick.[10] This type of algorithm improves the prediction of class membership of data by transforming it into a higher dimensional space in which the classification problem is linearly separable.[11] SVM has been successfully applied to many QSAR classification problems, reviewed recently by Doucet et al.[12] The SVM algorithm is also applicable to continuous data, but the majority of QSAR applications have applied it to classification problems. However, SVM has been shown to generate models that are not optimally sparse, potentially compromising the ability of the models to effectively generalize to new data and make the most accurate predictions of their properties.

There is a related but sparser classification and regression method, the Relevance Vector machine (RVM), based in Bayesian statistics, which has significantly better properties than SVM. Sparser models generated by such methods have an improved ability to generalize to new data compared to less sparse models as we demonstrate here. We show the accuracy and parsimony advantages of this new classification and regression method relative to the widely used SVM method, using eight diverse data sets taken from the literature.

MACHINE LEARNING METHODS EMPLOYING EM LEARNING AND SPARSE PRIORS

Like most regression methods, artificial neural networks are prone to over training and over fitting. These drawbacks can be overcome by using Bayesian criteria to control model complexity and to provide an objective criterion for stopping the training. The complexity of the neural network is controlled by a penalty on the magnitude of the network weights w_j. These methods commonly employ a Gaussian prior $\sum_{j=1}^{N_w} w_i^2$ to control the complexity of models (denoted as Bayesian regularized artificial neural networks with a Gaussian prior, BRANNGP)[6,13,14] and use an expectation maximization

Received: May 6, 2015
Published: July 9, 2015

DOI: 10.1021/acs.jcim.5b00261

algorithm to achieve optimum sparsity in these and linear models.[9] These machine-learning methods have been successfully applied to QSAR modeling of diverse data sets.[1,2,5,15−22] They can be further improved by using a sparsity-inducing Laplacian prior $\sum_{j=1}^{N_w}|w_j|$ (denoted as Bayesian regularized artificial neural networks with a Laplacian prior, BRANNLP)[8,9], which enables the irrelevant weights in feature space to be set to zero, leaving the remainder to define the model. In practical terms this means that both the less relevant descriptors and the number of effective weights in the neural network model are pruned to the optimal level. In both cases the loss function to be minimized is $L = {}^1/_2 N_D \log E_D + N_W \log E_W$, where E_D and E_W are the errors in the data and weights, respectively. L can be minimized using a local search method such as the BFGS algorithm. Since $M(w) = \beta E_D + \alpha E_W$ the minimum with respect to the weights, $(\partial M/\partial w_j) = 0$ yields $|\delta E_D/\delta w_j| = (\alpha/\beta)|w_j|$ for a Laplacian prior. However, this function has a singularity at $w = 0$ so the derivative must be calculated for two cases $|\partial E_D/\partial w_j| = \alpha/\beta$ if $|w_j| > 0$ and $|\partial E_D/\partial w_j| < \alpha/\beta$ if $|w_j| = 0$. The first equation shows that the sensitivity toward fitting error is the same for all weights, while the second equation shows that, if the first condition does not hold, then the weights must vanish, generating a sparser solution.

The SVM is a supervised machine learning technique derived from a formulation in statistical learning theory.[10] The method implements a classifier with adjustable flexibility that is automatically optimized on the training data, providing good generalization performance on unknown data. The method and properties have been described in general terms recently by Noble.[11]

SVMs derive the class decision from selected subsets of samples, called support vectors, in which the characteristic information on class distinction is compressed. The training of the classifier involves finding the optimal separating hyperplane such that the largest margin is formed between two classes of vectors while minimizing the effects of classification errors. The maximization of the margin between the two classes is an optimization problem that results in maximizing $L_D = \sum_i \alpha_i - {}^1/_2 \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j$, where x is a generic sample data vector and y its label (e.g., ± 1) and the function L_D has to be maximized with respect to the coefficients α. Remarkably, unlike optimization functions in other learning algorithms such as neural networks, L_D has only one minimum, thus generating a completely reproducible solution. The α coefficients will directly determine the disposition of the class boundary and will be nonzero only for the support vectors, namely those samples lying at the very border between the two classes. The larger the coefficient, the bigger the influence of the sample on the determination of the class boundary. Those training samples distant from the boundary, which in many applications form the majority of cases, will have $\alpha = 0$, and their removal will not influence the final solution at all. Hence, the number of support vectors is a relevant diagnostic parameter because it represents an upper bound to the cross-validated error in a leave one out procedure. These coefficients are also subject to the constraint $0 \le \alpha_i \le C$, where C is an adjustable parameter to be chosen by the user and whose value is inversely proportional to the error tolerance for the training samples. Including no error (no upper bound to the α multipliers) corresponds to finding a hyperplane that minimizes the training error. However, this may result in a very narrow margin that indicates poor generalization ability. Generally, we want to trade-off between these two aspects, so it is sensible to include some error tolerance. In the graphical output, this will be proportional to the sum of the distances between the samples that lie on the wrong sides of the class margin and the margin itself.

The separating hyperplane (class boundary) is defined by $x \cdot w + b = 0$, where the characteristic parameters w and b are determined by $w = \sum_i^{N_s} \alpha_i y_i s_i$, $b = 1/y_i - \sum_{j=1}^{N_s} \alpha_j y_j x_j s_i$. Here, N_s is the number of support vectors, s is a generic support vector, and s_i in the equation for b refers to any support vector for which $\alpha < C$. The rule for classifying a new instance x is explicitly expressed as a function of the support vectors, $y = \mathrm{sgn}(\sum_{i=1}^{N_s} \alpha_i y_i s_i x + b)$, where the class label y depends on the sign of the function within the parentheses. The higher its value, the stronger is the SVM algorithm probability that a sample belong to one class.

The reason why SVMs are able to perform well when dealing with cases of different complexity in terms of class distribution is the addition of kernel functions to the above algorithm. The kernel function is defined as $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$, where the function $\phi(x)$ maps the input vector x to some higher dimension space such that a more suitable hyperplane can be found with minimal classification errors. The function to optimize becomes $L_D = \sum_i \alpha_i - {}^1/_2 \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$.

Because it is possible to replace the dot product $x_i \cdot x_j$ with $K(x_i, x_j)$ in the algorithm, there would never be a need to explicitly define what ϕ is. The algorithm searches for the optimal separating hyperplane that maximizes the margin within the embedded space of the kernel function. This linear boundary will then generate in a nonlinear boundary of variable complexity when returning to the feature space of the training vectors.

However, SVM methods have a number of disadvantages:[23]—

- Although relatively sparse, SVMs are not optimally so.
- Predictions are not probabilistic. In regression the SVM provides a single value, and in classification, a deterministic binary decision. Ideally, it would be useful to capture the uncertainty of predictions.
- It is necessary to estimate SVM parameters using a cross-validation procedure that is wasteful of time, data, and computation.

There are restrictions on the types of kernel functions that can be used—they must use a continuous symmetric kernel of a positive integral operator (Mercer's condition). Some of the most common kernels that meet this criterion are polynomial of degree p, $K(x_i, x_j) = (x_i \cdot x_j + 1)^p$; radial basis function $K(x_i, x_j) = \exp((-\|x_i - x_j\|^2)/2\sigma^2)$; and two-layer sigmoidal function $K(x_i, x_j) = \tanh(k \cdot x_i \cdot x_j - \delta)$. Changing the kernel results in dealing with SVMs classifiers of different nature. For example, the simple dot product $K(x_i, x_j) = (x_i \cdot x_j)$ of the original optimization problem generates a linear classifier, while other nonlinear functions could correspond to adopting a radial basis or feed-forward neural network, for example. Each kernel also requires the determination of optimal values for their related parameters, such as the polynomial degree p, or radial basis function width σ, for example.

The Relevance Vector Machine (RVM), introduced by Tipping,[23] was designed to overcome these disadvantages. It is a machine learning technique that uses Bayesian inference to obtain parsimonious solutions for regression and probabilistic classification. It has an identical form to the SVM support vector machine, but it provides probabilistic classification. It is equivalent to a Gaussian process[24] model with covariance

Table 1. Performance of Three Classification or Regression Methods against Diverse Data Sets

								Accuracy	
Data Set	Data size	Method	$N_{indices}$	$N_{Weights}$	Sparsity	N_{eff}	$\%N_{eff/SVM}$	Training	Test
Boston Housing[27]	506	RVM				27	7	0.94	0.96
		SVM				405	100	0.97	0.83
		BRANNLP	13	46	7	18	4	0.92	0.93
Log vapor pressure[29]	651	RVM				92	93	0.98	0.94
		SVM				99	100	0.97	0.92
		BRANNLP	45	142	4	85	86	0.93	0.92
Toxicity toward *T. pyriformis*[30]	278	RVM				23	19	0.87	0.65
		SVM				120	100	0.94	0.70
		BRANNLP	118	361	4	23	19	0.88	0.60
Transport through BBB[14]	106	RVM				11	18	0.82	0.72
		SVM				62	100	0.88	0.57
		BRANNLP	64	199	0	15	24	0.88	0.90
Inhibition of GABA receptor by benzo-diazepines[25]	245	RVM				68	42	0.87	0.61
		SVM				163	100	0.84	0.73
		BRANNLP	42	133	7	14	9	0.73	0.65
Breast cancer diagnosis[32]	369	RVM				68	88	0.94	0.89
		SVM				77	100	0.95	0.94
		BRANNLP	9	34	4	33	43	0.96	0.99
Biodegradability of chemicals[33]	1055	RVM				48	11	0.91	0.88
		SVM				454	100	0.99	0.87
		BRANNLP	41	130	7	32	7	0.88	0.85
Pima Indians[28]	768	RVM				16	5	0.81	0.75
		SVM				309	100	0.89	0.74
		BRANNLP	16	55	5	11	4	0.58	0.70

$N_{indices}$, number of indices/descriptors before sparsity; $N_{weights}$, number of weights in the BRANNLP model; sparsity (0−7) used in BRANNLP to control model sparsity; N_{eff}, number of effective weights or support vectors after sparsity; $\%N_{eff/SVM}$, percentage of final weights/support vectors/those for the SVM model, a measure of model sparsity; accuracy, number correctly predicted in all classes/total.

function $k(x,x') = \sum_j (1/\alpha_j)\varphi(x,x_j)\varphi(x',x_j)$ and is similar to the Automatic Relevance Determination (ARD)[25] method we have previously implemented. Like the Bayesian regularized neural networks (BRANNGP and BRANNLP), RVM uses an expectation maximization (EM) learning method, employing a sparse prior to reducing the number of support vectors required to model the decision boundary in hyperparameter space. SVM represents the instances of the data into space and tries to separate the distinct classes by the maximum possible gap width (hyperplane) that separates the classes. RVM, however, uses probabilistic measures to define this separation space. RVM uses Bayesian inference to obtain a succinct solution, so RVM uses significantly fewer basis functions.[26]

The advantages of the RVM method over SVM are as follows:

- Generalization is typically very good, and results comparable with the state-of-the-art can be obtained.
- Models are typically very sparse, approaching optimal compactness.
- Importantly, in classification, the model gives estimates of the posterior probability of class membership.
- There are no parameters to evaluate, as the type-II maximum likelihood (empirical Bayes) procedure automatically sets the "regularization parameters", while the noise variance can be similarly estimated in the regression.
- There is no constraint over the number or type of basis functions that may be used.
- The method can optimize "global" parameters, such as those that moderate the input variable scales. This is a very powerful feature, as it is impossible to set such scale parameters by cross-validation.

METHODS

Data sets. The data sets are quite diverse and are a mixture of standard SVM benchmarks (Boston Housing (factors influencing willingness to pay for "clean air" housing),[27] Pima Indians (factors influencing the diabetes rate)[28]), physicochemical properties (organic compound vapor pressure),[29] and biological end points (toxicity of chemicals toward *Tetrahymenas pyriformis*,[30] drug transport across the blood-brain barrier (BBB),[14] ability of benzodiazepines to inhibit the GABA receptor,[31] breast cancer diagnosis,[32] and biodegradability of chemicals.[33] Chemicals with a biochemical oxygen demand (BOD) value higher than 60% were considered as readily biodegradable whereas those with a BOD lower than 60% are regarded as not readily biodegradable. The eight data sets used either the descriptors given in the data source (class data) or those described in the cited references.

Some of the data sets were chosen specifically because they are difficult to model. This could be overcome to some extent by finding better descriptors, but we have not pursued this as our aim was to compare the methods rather than find the best model for the properties. To provide a common basis for comparison of the classification methods, the continuous data sets (vapor pressure, toxicity, drug transport, and benzodiazepines) were converted to classes defined according to which side of the (max - min)/2 decision threshold they occupied.

The data sets were split into 80% for the training set used to generate the QSAR model and 20% held back as a test set to assess the model prediction power.

Machine learning methods. The SVM and RVM employed radial basis function (Gaussian) kernels centered on the data. The Bayesian regularized neutral network using the sparsity-inducing Laplacian prior (BRANNLP) was a three layer network with linear transfer function for the input and output modes and sigmoidal transfer functions for the three hidden layer nodes.

The theory of the RVM has been very well described by Tipping,[23] and it follows very closely that of the SVM above. A working algorithm in MATLAB code has been given by Tipping (http://www.vectoranomaly.com/downloads/SB2_Release_200.zip). The RVM process is an iterative one and involves repeatedly re-estimating α and β until a stopping condition is met. The steps are as follows:

1. Select a suitable kernel function for the data set and relevant parameters. Use this kernel function to create the design matrix Φ.

2. Establish suitable convergence criteria for α and β, e.g. a threshold value for change δ_{Thresh} between one iteration's estimation of α and the next $\delta = \sum_{i=1} \alpha_1^{n+1} - \alpha_1^{n}$. Re-estimation stops when $\delta < \delta_{\mathrm{Thresh}}$.

3. Establish a threshold value α_{Thresh} above which it is assumed an α_{i} is increasing toward infinity (weight tending to zero).

4. Choose starting values for α and β.

5. Calculate $\mathrm{m} = \beta \sum \Phi^{\mathrm{T}}$ t and $\sum = (\mathrm{A} + \beta \Phi^{\mathrm{T}} \Phi)^{-1}$, where $\mathrm{A} = \mathrm{diag}(\alpha_0, \alpha_1, ..., \alpha_{\mathrm{N}})$ and $\mathrm{t} = (\mathrm{t}_1, \mathrm{t}_2, ..., \mathrm{t}_{\mathrm{N}})^{\mathrm{T}}$ are the target (y) values.

6. Update $\alpha_{\mathrm{i}} = \gamma\mathrm{i}/m_i^2$ and $\beta = (N - \sum_{\mathrm{i}} \gamma_{\mathrm{i}})/(\|\boldsymbol{t} - \boldsymbol{\phi m}\|^2)$, where $\gamma_i \equiv 1 - \alpha_i \sum_{ii}$ and $\sum_{ii}$ is the i-th diagonal element of the posterior weight covariance matrix, and N is the number of data points.

7. Prune the α_{i} and corresponding basis functions where $\alpha_{\mathrm{i}} > \alpha_{\mathrm{Thresh}}$.

8. Repeat steps (5) to (7) until the convergence criterion is met.

In statistical or machine learning several criteria can be used to judge the effectiveness of modeling methods, e.g. R^2 and q^2 values, standard errors of predictions of the training and test sets, the sparsity of the models, or the accuracy of prediction of data classes. In the present study we wanted common benchmarks for comparing the three methods, and we chose the accuracy of the two-class prediction and model sparsity. The accuracy was calculated as the total number of correct classifications as a fraction of the total number of data points $A = (TP + TN)/(TP + TN + FP + FN)$, where TP and TN are the number of correct positive and negative predictions and FP and FN are the numbers of false positive and false negative predictions, respectively.

Some models are most useful for prediction of properties of new candidates while others provide information on the relevance of each index in the models. The SVM and RVM models are commonly used for class assignment whereas the BRANNLP is also useful for understanding the contributions of indices to the model.

RESULTS AND DISCUSSION

Table 1 summarizes the results of modeling the data sets using SVM, RVM, and, for comparison, BRANNLP. The results show that all three methods can model the data sets and give high accuracy predictions for class membership for test set data that the models have not seen. The superior performance of the RVM compared with the SVM is clear. In some data sets, the RVM model is sparser than the SVM model and has a superior ability to predict the properties of the test set (Boston housing, vapor pressure, BBB); in others, the RVM model is sparser and has similar predictive performance on the test set to the SVM models (biodegradability and Pima Indians), and in some cases, the RVM models are sparser but perform slightly below the SVM (toxicity to *T. pyriformis*, breast cancer diagnosis, and bioactivity of benzodiazepines).

Interestingly, the Bayesian regularized neural network employing the sparsity-inducing Laplacian prior (BRANNLP) also performed well for continuous and classification data sets. The number of effective weights in the BRANNLP models is similar to the number of support vectors in the SVM and RVM models. Table 1 shows that, by this measure, the BRANNLP models are of similar sparseness to those generated by the RVM algorithm and have generally similar predictive power. The number of SVM weights is generally greater (sometimes much greater) than the number of BRANNLP effective weights or the RVM support vectors. The RVM (and similarly BRANNLP) models are much sparser than the SVM models, generally employing <20% and for some data sets <5% of the support vectors or weights than the SVM models for the same data sets. All algorithms could model the training data and predict the test set properties well for the eight diverse data sets studied.

The biodegradability model generated accuracy very similar to that of the literature SVM model.[33] Mansouri et al. analyzed this data set using k-nearest neighbors (kNN) clustering, partial least-squares discriminant analysis (PLSDA), and SVM using Gaussian kernels. They obtained an error rate (very similar to the complement of accuracy) of 15% for the first two methods and 14% for SVM, very similar to the 14% error rate from our RVM classification model. They did not state the number of support vectors, used but we expect this would be similar to the number we used for the SVM model for the same data set and same kernel. The breast cancer diagnosis and staging data had been previously analyzed[32] by Wolberg and Mangasarian using linear programming methods, not SVM. Their method, also quite sparse, generated test set errors between 4.1 and 6.5%, depending on the number of hyperplanes used, similar to the 94% accuracy for SVM and lower than the 99% accuracy of BRANNLP obtained in this study. Figueiredo[34] also reported SVM and RVM models for a smaller (532 data points) Pima Indians diabetes data set and obtained error rates of 12%, similar to the 15% error rates we found with a larger data set. The other data sets used have not been analyzed by SVM, so we could not run comparisons of other published SVM models of these properties. However, the ability of RVM (and BRANNLP) to generate much sparser models than SVM in most cases is clear from Table 1.

CONCLUSIONS

We have shown that relevance vectors machines are a superior classification method compared to SVM for a range of QSAR data sets modeled here. The RVM theory identifies how the algorithm overcomes the disadvantages of SVM, and our studies have shown that for relatively diverse data sets RVM models are usually sparser, more predictive, or both compared to SVM models of the same data using the same descriptors. Although RVM is an iterative method, training times are minimal for the data set sizes employed here, and the increased sparsity, which generates benefits in terms of predictive power and, arguably, interpretability, makes the small increase in computational effort worthwhile.

Interestingly, the Bayesian regularized neural network, which may be considered a kind of RVM with a sigmoidal kernel, also generated classification models that were also quite sparse when a Laplacian prior is used. These machine-learning algorithms are robust and predictive and have been validated in other areas of science and engineering[3−5,18,19,35−37] and also merit being similarly employed in QSAR and QSPR studies in place of the widely used SVM methods.

AUTHOR INFORMATION

Corresponding Author

*Phone: +61 3 95452477; E-mail: dave.winkler@csiro.au.

Notes

The authors declare no competing financial interest.

REFERENCES

(1) Epa, V. C.; Burden, F. R.; Tassa, C.; Weissleder, R.; Shaw, S.; Winkler, D. A. Modeling Biological Activities of Nanoparticles. *Nano Lett.* **2012**, *12*, 5808−5812.

(2) Epa, V. C.; Hook, A. L.; Chang, C.; Yang, J.; Langer, R.; Anderson, D. G.; Williams, P.; Davies, M. C.; Alexander, M. R.; Winkler, D. A. Modelling and Prediction of Bacterial Attachment to Polymers. *Adv. Funct. Mater.* **2014**, *24*, 2085−2093.

(3) Epa, V. C.; Yang, J.; Mei, Y.; Hook, A. L.; Langer, R.; Anderson, D. G.; Davies, M. C.; Alexander, M. R.; Winkler, D. A. Modelling Human Embryoid Body Cell Adhesion to a Combinatorial Library Of Polymer Surfaces. *J. Mater. Chem.* **2012**, *22*, 20902−20906.

(4) Le, T.; Epa, V. C.; Burden, F. R.; Winkler, D. A. Quantitative Structure-Property Relationship Modeling of Diverse Materials Properties. *Chem. Rev.* **2012**, *112*, 2889−2919.

(5) Winkler, D. A.; Burden, F. R. Robust, Quantitative Tools for Modelling Ex-Vivo Expansion of Haematopoietic Stem Cells and Progenitors. *Mol. BioSyst.* **2012**, *8*, 913−920.

(6) Burden, F. R.; Winkler, D. A. Robust QSAR Models using Bayesian Regularized Neural Networks. *J. Med. Chem.* **1999**, *42*, 3183−3187.

(7) Burden, F. R.; Winkler, D. A. New QSAR Methods Applied to Structure-Activity Mapping and Combinatorial Chemistry. *J. Chem. Inf. Model.* **1999**, *39*, 236−242.

(8) Burden, F. R.; Winkler, D. A. An Optimal Self-Pruning Neural Network and Nonlinear Descriptor Selection in QSAR. *QSAR Comb. Sci.* **2009**, *28*, 1092−1097.

(9) Burden, F. R.; Winkler, D. A. Optimal Sparse Descriptor Selection for QSAR Using Bayesian Methods. *QSAR Comb. Sci.* **2009**, *28*, 645−653.

(10) Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273−297.

(11) Noble, W. S. What is a Support Vector Machine? *Nat. Biotechnol.* **2006**, *24*, 1565−1567.

(12) Doucet, J.-P.; Barbault, F.; Xia, H.; Panaye, A.; Fan, B. Nonlinear SVM Approaches to QSPR/QSAR Studies and Drug Design. *Curr. Comput.-Aid. Drug. Des.* **2007**, *3*, 263−289.

(13) Winkler, D. A.; Burden, F. R. Robust QSAR Models from Novel Descriptors and Bayesian Regularised Neural Networks. *Mol. Simul.* **2000**, *24*, 243−+.

(14) Winkler, D. A.; Burden, F. R. Modelling Blood-Brain Barrier Partitioning using Bayesian Neural Nets. *J. Mol. Graphics Modell.* **2004**, *22*, 499−505.

(15) Autefage, H.; Gentleman, E.; Littmann, E.; Hedegaard, M. A.; Von Erlach, T.; O'Donnell, M.; Burden, F. R.; Winkler, D. A.; Stevens, M. M. Sparse Feature Selection Methods Identify Unexpected Global Cellular Response to Strontium-Containing Materials. *Proc. Natl. Acad. Sci. U. S. A.* **2015**, *112*, 4280−5.

(16) Celiz, A. D.; Smith, J. G. W.; Langer, R.; Anderson, D. G.; Winkler, D. A.; Barrett, D. A.; Davies, M. C.; Young, L. E.; Denning, C.; Alexander, M. R. Materials for Stem Cell Factories of the Future. *Nat. Mater.* **2014**, *13*, 570−579.

(17) Huh, Y. H.; Noh, M.; Burden, F. R.; Chen, J. C.; Winkler, D. A.; Sherley, J. L. Sparse Feature Selection Identifies H2A.Z as a Novel, Pattern-Specific Biomarker for Asymmetrically Self-Renewing Distributed Stem Cells. *Stem Cell Res.* **2015**, *14*, 144−54.

(18) Le, T. C.; Ballard, M.; Casey, P.; Liu, M. S.; Winkler, D. A. Illuminating Flash Point: Comprehensive Prediction Models. *Mol. Inf.* **2015**, *34*, 18−27.

(19) Le, T. C.; Mulet, X.; Burden, F. R.; Winkler, D. A. Predicting the Complex Phase Behavior of Self-Assembling Drug Delivery Nanoparticles. *Mol. Pharmaceutics* **2013**, *10*, 1368−1377.

(20) Salahinejad, M.; Le, T. C.; Winkler, D. A. Aqueous Solubility Prediction: Do Crystal Lattice Interactions Help? *Mol. Pharmaceutics* **2013**, *10*, 2757−2766.

(21) Winkler, D. A.; Breedon, M.; Hughes, A. E.; Burden, F. R.; Barnard, A. S.; Harvey, T. G.; Cole, I. Towards Chromate-Free Corrosion Inhibitors: Structure-Property Models for Organic Alternatives. *Green Chem.* **2014**, *16*, 3349−3357.

(22) Winkler, D. A.; Burden, F. R.; Yan, B.; Weissleder, R.; Tassa, C.; Shaw, S.; Epa, V. C. Modelling and Predicting the Biological Effects of Nanomaterials. *SAR QSAR Environ. Res.* **2014**, *25*, 161−172.

(23) Tipping, M. E. Sparse Bayesian Learning and the Relevance Vector Machine. *J. Mach. Learn. Res.* **2001**, *1*, 211−244.

(24) Burden, F. R. Quantitative structure - Activity Relationship Studies using Gaussian Processes. *J. Chem. Inf. Model.* **2001**, *41*, 830−835.

(25) Burden, F. R.; Ford, M. G.; Whitley, D. C.; Winkler, D. A. Use of Automatic Relevance Determination in QSAR Studies using Bayesian Neural Networks. *J. Chem. Inf. Model.* **2000**, *40*, 1423−1430.

(26) Rafi, M.; Shaikh, M. S. A Comparison of SVM and RVM for Document Classification. *5*, 2013; http://arxiv.org/abs/1301.2785 (accessed 24 December 2014).

(27) Harrison, D.; Rubinfeld, D. L. Hedonic Housing Prices and Demand for Clean-Air. *J. Environ. Econ. Manag.* **1978**, *5*, 81−102.

(28) Smith, J. W.; Everhart, J. E.; Dickson, W. C.; Knowler, W. C.; Johannes, R. S. Using the ADAP Learning Algorithm to Forecast the Onset of Diabetes Mellitus. In *Symposium on Computer Applications and Medical Care*; 1988; IEEE Computer Science Press: 1988; pp 261−265.

(29) Katritzky, A. R.; Slavov, S. H.; Dobchev, D. A.; Karelson, M. Rapid QSPR Model Development Technique for Prediction of Vapor Pressure of Organic Compounds. *Comput. Chem. Eng.* **2007**, *31*, 1123−1130.

(30) Burden, F. R.; Winkler, D. A. A Quantitative Structure-Activity Relationships Model for the Acute Toxicity of Substituted Benzenes to Tetrahymena Pyriformis using Bayesian-Regularized Neural Networks. *Chem. Res. Toxicol.* **2000**, *13*, 436−440.

(31) Winkler, D. A.; Burden, F. R.; Watkins, A. J. R. Atomistic Topological Indices Applied to Benzodiazepines using Various Regression Methods. *Quant. Struct.-Act. Relat.* **1998**, *17*, 14−19.

(32) Wolberg, W. H.; Mangasarian, O. L. Multisurface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology. *Proc. Natl. Acad. Sci. U. S. A.* **1990**, *87*, 9193−9196.

(33) Mansouri, K.; Ringsted, T.; Ballabio, D.; Todeschini, R.; Consonni, V. Quantitative Structure-Activity Relationship Models for Ready Biodegradability of Chemicals. *J. Chem. Inf. Model.* **2013**, *53*, 867−878.

(34) Figueiredo, M. A. T. Adaptive Sparseness for Supervised Learning. *IEEE Trans. Patt. Anal.* **2003**, *25*, 1150−1159.

(35) Le, T. C.; Conn, C. E.; Burden, F. R.; Winkler, D. A. Computational Modeling and Prediction of the Complex Time-Dependent Phase Behavior of Lyotropic Liquid Crystals under in Meso Crystallization Conditions. *Cryst. Growth Des.* **2013**, *13*, 1267−1276.

(36) Liu, Y. Y.; Winkler, D. A.; Epa, V. C.; Zhang, B.; Yan, B. Probing enzyme−nanoparticle interactions using combinatorial gold nanoparticle libraries. *Nano Res.* **2015**, *8*, 1293−1308.

(37) Salahinejad, M.; Le, T. C.; Winkler, D. A. Capturing the Crystal: Prediction of Enthalpy of Sublimation, Crystal Lattice Energy, and Melting Points of Organic Compounds. *J. Chem. Inf. Model.* **2013**, *53*, 223−229.