Chapter 5

Determining the Validity of a QSAR Model: A Classification Approach

5.1 Introduction

Quantitative structure-activity relationship (QSAR) modeling is based on the construction of predictive models using a set of known molecules and associated activity values. Such models can be generated using a wide variety of methods ranging from linear methods (e.g., linear regression and linear discriminant analysis) to non-linear methods (e.g., random forests and neural networks). As described in Chapter 3, an important step of the QSAR modeling process, irrespective of the nature of the modeling technique used, is validation. In all cases, the predictive ability of the models are tested with a set of molecules (the prediction set), which were not used during the model building process. Once a model has been built and validated it can be used on data for which no activity values are available. However, even though a model may have proved to exhibit good predictive ability based on the statistics for the prediction set, this is not always a guarantee that the model will perform well on a new set of data. The problem boils down to the fact that when a model is built and validated we can compare the predicted values to previously measured activity values. However, when the model is applied to new data, the predicted values of the activity cannot be compared with actual values. This leads to a problem: the training set and prediction set statistics may indicate that the model has good predictive ability. But when we use the model to predict values for molecules with unknown activity, how can we be sure that the predicted activity will be close to the actual activity? If the model were able to provide some measure of confidence for its prediction, this would be helpful. Such confidence measures (also known as scores) can be defined for various models. Examples include confidence bands for linear regression models and frequency based confidence measures for decision trees. However, such measures are specific to the modeling algorithm.

This work was published as Guha, R.; Jurs, P.C., "Determining the Validity of a QSAR Model - A Classification Approach", *J. Chem. Inf. Model.*, **2005**, *45*, 65–73.

This chapter describes a more general approach that should be applicable to any form of quantitative model. One possible approach is based on similarity. This is based on the assumption that a molecule that is structurally very similar (based on some sort of similarity metric such as atom pair[1] similarity or fingerprint similarity) to the training set molecules will be predicted well because the model has captured features that are common to the training set molecules and is able to find them in the new molecule. On the other hand, a new molecule with very little in common with the training set data should not be predicted very well; that is, the confidence in its prediction should be low.

An alternative approach to linking similarity measures and model quality (defined by residuals) is classification. In this method, the regression residuals for the training set are classified as good or bad, and a classification model is trained with the training set residuals. Once a trained model is obtained, we then predict the class of the prediction set residuals. However an important requirement for this process is that we be able to provide some measure of correctness for the predicted class assignments. Clearly this method does not fully solve the problem, as the classification algorithm would rarely be 100% correct. However the attractive feature of the approach discussed here is its generality. That is, it may be applied to any type of quantitative model, whether linear regression or a computational neural network. Furthermore, depending on how one defines a good residual or a bad residual, the classification model may be trained to detect unusual cases.

The fundamental decision that must be made when using the approach described in this chapter is the actual class assignments of the training set residuals. Since the fact that a compound is well predicted or poorly predicted is relatively subjective (except in extreme cases), the initial assignment of classes to the training set residuals is necessarily somewhat arbitrary. Furthermore, the nature of this class assignment defines the sizes of the two classes and hence plays a role in the choice of classification algorithm. These aspects are described in more detail in the following sections.

5.2 Datasets

Since one of the aims of this technique was generality, we attempted to test it on a variety of data. We considered three datasets covering both physical and biological properties. The first dataset was obtained from Goll et al.[2] and consisted of the boiling points of 277 molecules obtained from the Design Institute for Physical Property Data

(DIPPR) Project 801 database. This dataset is relatively diverse but contains several homologous series.

The second dataset consisted of a 179-compound subset of artemisinin analogues described by Avery et al.[3] We have previously developed and reported linear and CNN models based on this dataset.[4] The linear model from our previous study was used for the purposes of this work.

The third dataset consisted of 65 molecules. This dataset contained 56 molecules from a study carried out by Liu et al.[5] The remaining 9 molecules were selected from the literature, such that some were similar in structure to the molecules from Liu and some were distinctly different so as to be well-defined outliers in the final linear model. The molecules taken from Liu were all straight chain or branched hydrocarbons whereas the remaining molecules included polycyclic systems as well as molecules containing heteroatoms. The dependent variable in the original work was a transformation of the boiling point defined as

$$y = \log\left(266.7 - \mathrm{BP}\right) \tag{5.1}$$

where BP was the observed normal boiling points, in degrees Kelvin, of the molecules. Since the linear model we developed for this dataset did not use the all the molecules described by Liu, we did not use the logarithmic transformation and instead used boiling point values directly. The molecules and associated boiling points are shown in Table 5.1.

Each dataset was divided into a training and prediction set. The training set was used to build a linear model, and the prediction set was used to test the models themselves as well as the algorithms developed for this study. In the case of the artemisinin dataset, the same training and prediction sets that were used to develop the reported model were used in this study. Training and prediction sets for the DIPPR dataset were created using the activity binning method. In both cases, the training set contained approximately 80% of the whole dataset and the remainder was placed in the prediction set. The sets for the Liu dataset were created by hand. The training set contained 55 compounds selected from Liu, and the prediction set contained 10 compounds. Of these 10 compounds, one was taken from Liu and the remaining nine were selected from the literature. The reasoning for this specific construction was to allow the prediction set to contain molecules which were very dissimilar to the training set, so that the resultant linear model would exhibit distinct outliers.

5.3 Development of Linear Models

The first step of this study involved the development of a multiple linear regression model for each dataset. In the case of the artemisinin dataset, we used the linear model published by Guha et al.[4] This model contained four descriptors, and the statistics of the model are summarized in Table 5.2. Linear models for the DIPPR and Liu datasets were developed using the ADAPT[6,7] methodology described in Chapter 3. In all cases the final linear models were subjected to a PLS analysis to ensure that they were not overfitted. The statistics of the models for the DIPPR and Liu datasets selected by this procedure are summarized in Tables 5.3 and 5.4. Summary statistics for all three linear models are presented in Table 5.5.

5.4 The Classfication Approach

The aim of this study was to be able to decide whether a compound with unknown activity will be predicted well by a previously developed model. Though we focused only on linear regression models, the idea is general enough to be extended to other types of quantitative models (such as neural networks, support vector machines).

Initial attempts to develop a methodology to answer the above question focussed on evaluating a similarity measure between the new compound and the training set used to develop the existing model and then attempting to correlate the similarity measure with some measure of model quality. As we restricted ourselves to linear models we considered standard error of predictions and residuals. This line of attack did lead to the observation of some general trends. That is, compounds that were more similar to the training set generally exhibited smaller residuals and standard error of predictions. However, the observations were not conclusive, and the plots of the trends appeared to be too noisy to be able to draw any firm conclusions.

We then considered a classification approach. That is, can we classify a compound with no measured activity as *well predicted* or *poorly predicted* given a previously generated model and its associated training set? Our approach was to build a classification model using the original training set and the descriptors used in the original model and use this to predict the class of new compounds. The key word here is class. Before any model can be built we must decide how to classify the training set. We decided to consider regression residuals, as it would allow the technique to be generalized to other types of quantitative algorithms. The training set members were classified as bad or

good depending on whether their residuals were above or below a user specified cut-off value. This cut-off value plays a central role as it determines the size of the two classes. Our current strategy is to use a cut-off value obtained by visual inspection of a residual plot for the training set. The value of the cut-off was selected so that the minor class contained approximately 20% to 30% of the whole dataset. Clearly this leads to a highly imbalanced classification problem, but we felt that it would model a real world application of this technique more closely than allowing the classes to be of similar size. Alternative (non-arbitrary) methods include classifying training set members as good or bad depending on whether some regression diagnostic (Cooks distance, Mahalanobis distance) determined that it was an outlier. Fig. 5.1 shows the plots of residuals along with a line at the cut-off value for the three datasets studied here. Table 5.6 summarizes the cut-off values and associated class sizes for each dataset.

5.4.1 Classification Algorithms

Given the training set class structure, the choice of algorithm is guided by two requirements. First, the goal is to be able to test compounds for which we have no measured activity value. As a result, the classification algorithm must be able to produce some measure of confidence in its class predictions or else a probability of class membership (posterior probability). In the absence of such a quantity the final output of the classification model does not provide any more information than produced by simply processing the new compound through the original predictive model. The second requirement is that the algorithm must be able to handle unbalanced classes. In general, several schemes are available that can be used to modify the standard classification algorithms. These include over-sampling the minority class[8] and under-sampling the majority class.[9] Maloof[10] discusses the application of receiver operator characteristics (ROC) analysis in comparing how various sampling strategies and cost schemes affect classification of skewed datasets. Breiman[11] describes a simple method to increase the size of the dataset without simply repeating observations. The extra samples are termed convex pseudo-data and the generating algorithm requires a single parameter (as opposed to kernel density methods). We investigated the use of this method in attempt to improve classification accuracy.

We considered a wide variety of classification algorithms: logistic regression, partial least squares (PLS), discriminant analysis, neural networks and random forests. Random forests were first described by Breiman[12] and have been used in a variety of

QSAR applications. The original random forest algorithm was not suited for very unbalanced datasets, but current implementations[13] use a weighting scheme which overcomes this problem. We decided not to use this algorithm, due to the fact that it works with large descriptor pools owing to its ability to ignore irrelevant descriptors as well as the fact that the algorithm is resistant to overfitting. We did not want to build the classification models with more (and different) information than was available to the original regression models. Given that the good performance of random forest models is due to their ability to build trees based on good descriptor subsets, restricting the descriptor pool to four or five descriptors would probably result in lower quality random forest models.

In the case of discriminant analysis, we investigated the use of linear and quadratic discriminant methods. In each case, the algorithms employed were able to generate posterior probabilities via a cross-validation scheme. As the results were quite similar we only present the results of the linear discriminant analysis. All the algorithms mentioned above were obtained using the R software package.[14]

As mentioned in the previous section, we used the descriptors from the original regression model to build the classification model. We also investigated the use of a similarity measure as a source of extra information. Intuitively, one would expect that a new molecule that is similar to the molecules in the training set should be well predicted by the regression model. Thus, in addition to classification models built only with descriptors from the regression model we also built models that also contained a similarity value. We chose to use the atom pair similarity described by Carhart et al.[1] Atom pair similarities are calculated between pairs of molecules. To provide a single similarity value for each compound we calculated the average similarity value between each compound and all the compounds in the training set.

5.5 Results

Most of the algorithms exhibited good predictive ability considering the fact that the datasets used were not very large (especially the Liu dataset). As expected, the neural network performed very well, with a 90% correct prediction rate on the training set and 72% to 85% correct on the prediction set. The inclusion of the similarity values as a descriptor did not appear to improve the results significantly.

Table 5.7 shows the confusion matrices for the training and prediction sets generated using linear discriminant analysis for the artemisinin dataset. The implementation

used for this study allowed us to specify the prior probabilities for each class. We assumed that the priors could be approximated by the class proportions. Clearly, the very poor predictions for the minority class indicate the problem due to the imbalanced nature of the class distributions. To try and remove the bias due to the imbalanced nature of the problem, the model was regenerated with an over-sampled minority class. However the results did not improve significantly. To investigate whether extra information might improve the situation we also regenerated the model using the averaged atom pair similarity values as an extra independent variable. We felt that this was justified (as compared to using extra molecular descriptors) since this descriptor essentially compares the molecules amongst themselves. Table 5.7 displays the confusion matrices for the resultant model. The predictions for the good class are now 100% but members of the bad class are mispredicted in all cases. The results for this algorithm when applied to the DIPPR dataset give similar results. The classes assigned in this dataset are also quite unbalanced. The confusion matrices are presented in Table 5.8. The results for the Liu dataset (Table 5.9) are marginally better, more so for the prediction set than the training set. This is probably due to the slightly higher proportion of the minor class in the training set.

The results from the PLS classification scheme were not significantly better than those obtained with LDA and in some cases worse, and as a result we omit their presentation.

Table 5.10, 5.11 and 5.12 present the confusion matrices for the three datasets generated using a neural network. The network used entropy outputs[15] and thus provided the associated probabilities with each class assignment. In all cases, the inverse of the class proportions were used as example weights. Table 5.10 shows that the performance for the artemisinin dataset was not very impressive. However the imbalanced nature of the dataset does not affect the performance as much as in the case of LDA. In contrast, the DIPPR dataset showed very good performance using the neural network methodology as can be seen from Table 5.11. In this case, the bad class was very well predicted in both the training and prediction sets. Finally the Liu dataset also yielded good results (Table 5.12). In all cases, the use of average atom pair similarity as an extra independent variable did not appear to improve results.

Table 5.13 displays the weighted success rates for all the classification methods on all the datasets. This measure of classification success was described by Weston et

al.[16] and is defined as

$$w = \frac{1}{2}\left(\frac{\text{No. true positives}}{\text{Total positives}} + \frac{\text{No. true negatives}}{\text{No. negatives}}\right)$$

The above expression indicates that $0 \leq w \leq 1$. As mentioned by Weston, this measure is suitable for unbalanced classification problems. The values indicate the poor performance of the LDA (in fact, it appears to be not much better than random) and PLS methods and the much better performance of the neural network approach.

We also attempted to improve the classification results by using the convex pseudo-data method described by Breiman[11] to increase the size of the training sets. We considered two approaches. In the first method, we simply extended the whole dataset without regard to class. The new samples were placed in the training set and the extended training sets were used to build models. In the second approach we only extended the portion of the training set that was assigned to the bad class (essentially increasing the size of the bad class). Though the results in some cases (PLS and LDA) did improve to some extent, the increases in classification rates did not appear to be significant and hence we omit them in this study.

One way to consider the performance of the models is shown in Figs. 5.2, 5.3, and 5.4. The probability for membership in the good class is plotted against the residuals from the original linear regression model. The probabilities were obtained from the neural network classification models. Fig. 5.2 is the plot for the prediction set of the DIPPR dataset. Ideally one would expect that such a graph would have a cluster of points in the upper left quadrant and a cluster in the lower right quadrant. However, in practice such a perfect distribution is rare, although the graph does indicate the general trends. In the lower right there is a vertical set of points with probability 1.0 that exhibit low values of the absolute standardized residual. On the left hand side of the graph, we see a similar set of points with probability values equal to 0.0 (indicating that they belong to the bad class). In between these two extremes we see points that have probabilities indicating membership to the good class. However for points with probabilities lying in the range 0.5 to 0.7 such membership is probably not conclusive, and we see that their residuals are also midway between the two extremes. The points at the left and right edges of the graph indicate that when the class predictions of the CNN classifier are accompanied by high or low probabilities, the residuals from the linear regression model can be expected to be low or high, respectively. The two anomalous points marked by red triangles represent the misclassified cases. The one on the right was predicted as belonging to

the good class, whereas its true membership was to the bad class, and vice versa for the point on the right. It is not apparent why these points would be misclassified. But more importantly, it is not clear how one might consider them misclassified without having the actual residuals available, since in a real application we would be dealing with observations whose actual activities are not known.

Fig. 5.3 shows the corresponding plot for the Liu dataset. As before, observations predicted to be in the bad class and the good class (with high certainty) are located in the upper left and lower right quadrants respectively. In this case, there is only a single point whose membership is not absolutely certain.

Finally, Fig. 5.4 shows the plot for the artemisinin dataset. In this case the plot is not as tight as the previous ones, with the probability values of a number of observations indicating that membership in the good class is not very conclusive. The misclassified observations are interesting. The one misclassified point on the left hand edge would certainly be difficult to detect in the absence of residuals. However, the remaining two misclassified points are more or less on the border between the two lower quadrants. In addition they are also quite close to points that have been correctly classified. This is indicative of the fact that membership of observations when their probabilities lie around 0.5 can inconclusive and thus one should be wary of such points.

5.6 Further Work

The methodology described here appears to perform reasonably well on the three datasets we investigated. However, there are several features that require further study. First, the classification approach described here is a two-class problem. We restricted ourselves to the two-class problem for simplicity. Considering the scheme as a three-class problem might enable the user to draw more fine-grained conclusions regarding the validity of the results obtained from a regression model. However, increasing the number of classes will certainly require a large dataset and even if such a dataset is used, the unbalanced nature of the classes will require careful selection of a classification technique. We note that the results presented in this study are dependent on the nature of the datasets employed – specifically the distribution of residuals which is itself dependent on the distribution of the compounds in descriptor space. However, the datasets that we selected for testing include both physical properties for a number of congeneric series as well as biological properties for a set of molecules containing exhibiting varying structures and functionality. Furthermore the datasets we selected allowed us to test our techniques

with different types of linear models. For example, the DIPPR dataset was described by a linear model with very good statistics and very low residual values in general. On the other hand, the artemisinin dataset was characterized by lower values of R^2, high RMSE value and a number of observations with large residuals. As a result the DIPPR dataset presented our methodology with severely unbalanced classes whereas the class distribution was not as skewed in the case of the arteminsin dataset. Furthermore it is often the case that linear models for biological properties do not exhibit high quality statistics and contain a number of outliers. Thus the use of this dataset allowed us to test our technique in a real world scenario. Finally, the Liu dataset that was prepared by hand allowed us to have specific observations with large residuals and thus test the ability of the methodology to specifically detect these types of compounds. As has been shown, our methodology appears to perform well on these varied datasets. The only downside to the selection of our datasets is that the sizes are not as large as we would have liked them to be. Larger datasets would allow us to experiment with more than two classes as well as other classification schemes as discussed below. Clearly, one possible avenue of investigation is the validation of our methodology on different (and larger) datasets.

Modified sampling schemes like those described do not appear to improve the results significantly. The initial assignment of classes to the training set data is a step that could be modified, as the current approach employs an arbitrary assignment scheme. To remove this user defined task, class assignments can be automated by the use of regression diagnostics. However, such a scheme would then restrict the application of this methodology to linear models only. It appears that for full generality some form of cut-off value must be specified by the user. However, one advantage of a user- specified cut-off value is that it allows the user to focus on a range of residual values. Coupled with multiple (more than two) classes, this would allow the user to perform a fine-grained analysis of the residual classes.

Of the classification techniques investigated in this study it appears that neural networks performed the best with overall classification rates ranging from 79% to above 90% for the training set and 73% to 90% for the prediction set. The linear methods did not appear to perform significantly better than random. Furthermore, introduction of a similarity measure as an independent variable did not lead to improved classification results using any of the methods.

An alternative approach that may be considered is a Bayesian classification scheme whereby the training set class assignments are used to build up a prior probability distribution and the probability of new compounds belonging to a given class can be obtained by sampling from the simulated distribution. Associated with each class prediction is a probability for the membership to the predicted class. This requirement restricted our choice of classification technique somewhat but we feel that the lack of such a posterior probability would result in this method not being any more useful than simply recalculating the original regression model with some sort of scoring feature. The plots of posterior probability versus residuals are a good indicator of the performance of this methodology and also allows us to identify misclassifications in general. However, misclassified examples that are associated with posterior probabilities around 0.5 are, in general, not distinguishable from correctly predicted examples with similar posterior probabilities. In such cases one would probably be justified in ignoring compounds whose class predictions are borderline and rather concentrate on those compounds that are classified with high posterior probabilities of belonging to the good or bad class.

5.7 Conclusions

This chapter describes a novel and general scheme to provide a measure of confidence for the predictions from a regression model. The methodology described here attempts to answer the following question: how well will a regression model predict the property value for a compound that was not in the training or prediction set of the model? That is, we have attempted to extend and unify the characterization of generalizability for different types of QSAR models. Multiple approaches were investigated resulting in a classification scheme in which the training set residuals were assigned to one of two classes depending on whether they lay above or below a cut-off value. A classifier was then built with these assignments and used to predict the class of the residual for a new compound. The technique appears to be general enough to be applicable to any given regression model. We investigated several classification techniques and a neural network approach produced the best classification rates. The performance of the algorithm was visualized by considering plots of posterior probabilities versus residuals.

Though the performance of regression models may be judged via other scoring methods, such as confidence bands or frequency based scores, these methods are generally specific to the regression modeling technique employed. The method described here is quite general and thus can be applied to regression models developed using linear

regression, neural networks or random forests. Furthermore, the methodology is not dependent on the original dataset. All that is required is the availability of the original residuals (which is generally available in models developed with common statistical packages). Another attractive feature is that apart from the threshold residual value, the methodology does not require extra information such as similarity measures or new descriptors, since it restricts itself to using the descriptors that were used in the original quantitative model. We believe that such a parsimonious approach minimizes complexity as well as user intervention. The net result of our methodology is a probability of whether a compound (with an unknown property value) will have a high or low residual (relative to a user specified cut-off value) when processed by the regression model. Clearly, this does not replace the use of the original quantitative model. Rather, the methodology allows us to generate confidence measures for new compounds for any type of quantitative regression model in the absence of the original data and in a parsimonious manner. As a result methodology could be used as a component of a high throughput screening process in which different regression techniques are employed in a consensus based strategy.

Table 5.1: Molecules and experimental boiling point values comprising the toy dataset selected by hand from Liu et al.[5] and the literature

Name	BP (K)	Name	BP(K)
methane	-164.00	2,2,3-trimethylpentane	110.00
ethane	-88.60	2,2,4-trimethylpentane	99.20
propane	-42.10	2,3,3-trimethylpentane	114.70
butane	-0.50	2,3,4-trimethylpentane	113.40
2-methylpropane	-11.70	2-methyl-3-ethylpentane	115.60
pentane	36.10	3-methyl-3-ethylpentane	118.20
2-methylbutane	27.80	2,2,3,3-tetramethylbutane	106.50
2,2-dimethylpropane	9.50	nonane	150.77
hexane	69.00	2-methyloctane	142.80
2-methylpentane	60.30	3-methyloctane	143.80
3-methylpentane	63.30	4-methyloctane	142.40
2,2-dimethylbutane	49.70	2,2-dimethylheptane	132.70
2,3-dimethylbutane	58.00	2,3-dimethylheptane	140.50
heptane	98.40	2,4-dimethylheptane	133.50
2-methylhexane	90.00	2,5-dimethylheptane	136.00
3-methylhexane	92.00	2,6-dimethylheptane	135.20
2,2-dimethylpentane	79.20	3,3-dimethylheptane	137.30
2,3-dimethylpentane	89.80	3,4-dimethylheptane	140.10
2,4-dimethylpentane	80.50	3,5-dimethylheptane	136.00
3,3-dimethylpentane	86.10	4,4-dimethylheptane	135.20
3-ethylpentane	93.50	3-ethylheptane	143.00
2,2,3-trimethylbutane	80.90	4-ethylheptane	141.20
octane	125.70	benzene[a]	80.10
2-methylheptane	117.60	benzoic acid[a]	249.00
3-methylheptane	118.00	cyclohexane[a]	80.70
4-methylheptane	117.70	decane[a]	174.10
2,2-dimethylhexane	106.80	bromomethane[a]	3.50
2,3-dimethylhexane	115.60	propylamine[a]	48.00
2,4-dimethylhexane	109.40	2,3,3-trimethylhexane	131.70

Table 5.1: (continued)

Name	BP (K)	Name	BP(K)
2,5-dimethylhexane	109.00	pyrrole[a]	130.00
3,3-dimethylhexane	112.00	anthracene[a]	340.00
acetic acid[a]	117.90		

[a]Boiling point obtained from www.chemfinder.com

Table 5.2. Statistics for the linear regression model using the artemisinin dataset.

Description	β	Std. Error	t	P	VIF
Constant	-60.5625	5.2834	-11.5	2×10^{-16}	
N7CH	-0.2148	0.0134	-16.1	2×10^{-16}	1.6
NSB-12	0.2238	0.0238	9.4	2×10^{-16}	1.3
WTPT-2	27.9391	2.6136	10.7	2×10^{-16}	1.4
MDE-14	0.1118	0.0247	4.5	1.18×10^{-5}	1.5

N7CH - number of 7^{th} order chains;[17–19] NSB-12 - number of single bonds; WTPT-2 - the molecular ID number[20] considering only carbon atoms; MDE-14 - the molecular distance edge vector,[5] considering only primary and quaternary atoms.

Table 5.3. Statistics for the linear regression model using the DIPP dataset.

Description	β	Std. Error	t	P	VIF
Constant	179.15628	2.02828	88.329	$< 2 \times 10^{-16}$	
FPSA-3	-175.87824	2.88552	-60.952	$< 2 \times 10^{-16}$	1.6
FNSA-3	1.36298	0.01395	97.675	$< 2 \times 10^{-16}$	1.8
RNCG-1	-0.65982	0.11676	-5.651	4.70×10^{-8}	1.2
RPCS-1	-0.38502	0.07294	-5.279	3.00×10^{-7}	1.1

FPSA-3 - partial positive surface area divided by the total molecular surface area;[21] FNSA-3 - charge weighted partial surface area divided by the total molecular surface area;[21] RNCG-1 - the difference between the relative negative charge and the most negative charge divided by the total negative charge;[21] RPCS-1 - the positive charge analog of RNCG-1 multiplied by the difference between relative positively charged surface area and the most positively charged surface area.[21]

Table 5.4. Statistics for the linear regression model using the toy dataset.

Description	β	Std. Error	t	P	VIF
Constant	-381.6960	60.3677	-6.323	8.72×10^{-8}	
EMIN-1	-43.2189	9.1003	-4.749	1.95×10^{-5}	1.1
EMAX-1	88.8862	10.4446	8.510	4.46×10^{-11}	1.5
ECCN-1	1.2717	0.1052	12.089	4.99×10^{-16}	1.2
SHDW-6	501.1936	136.7371	3.665	6.27×10^{-4}	1.2

EMIN-1 - minimum atomic estate value;[22] EMAX-2 - maximum atomic estate value;[22] ECCN-1 - eccentric connectivity index;[23] SHDW-6 - the area of the molecule when projected onto the XY plane[24,25]

Table 5.5. Summary statistics for the three linear models used in this study

	Training Set		Prediction Set			
Dataset	R^2	RMSE	R^2	RMSE	F statistic	p value
Artemisinin	0.70	0.87	0.05	0.75	95.28 (4,156)	2.2×10^{-16}
DIPP	0.99	7.22	0.99	7.42	9521 (4,230)	2.2×10^{-16}
Toy	0.90	18.84	0.01	352.30	111.9 (4,47)	2.2×10^{-16}

Table 5.6. Cutoff values used for each dataset and the resultant size of each class

		Class Size	
Dataset	Cutoff	Good	Bad
artemisinin	1.0	133	46
DIPP	1.0	213	64
toy	1.0	44	21

Table 5.7. Confusion matrices for the linear discriminant analysis of the artemisinin dataset with and without atom pair similarity.

AP similarity excluded

Training Set

	Predicted	
Actual	bad	good
bad	2	40
good	2	117

Prediction Set

	Predicted	
Actual	bad	good
bad	0	4
good	0	14

AP similarity included

	Predicted	
Actual	bad	good
bad	0	42
good	0	119

	Predicted	
Actual	bad	good
bad	0	4
good	0	14

Table 5.8. Confusion matrices for the linear discriminant analysis of the DIPP dataset with and without average atom pair similarity .

AP similarity excluded

Training Set

	Predicted	
Actual	bad	good
bad	4	50
good	4	177

Prediction Set

	Predicted	
Actual	bad	good
bad	1	9
good	1	31

AP similarity included

	Predicted	
Actual	bad	good
bad	4	50
good	4	177

	Predicted	
Actual	bad	good
bad	1	9
good	1	31

Table 5.9. Confusion matrices for the linear discriminant analysis of the toy dataset with and without average atom pair similarity .

AP similarity excluded

Training Set

	Predicted	
Actual	bad	good
bad	7	11
good	4	30

Prediction Set

	Predicted	
Actual	bad	good
bad	2	1
good	3	7

AP similarity included

	Predicted	
Actual	bad	good
bad	4	14
good	4	30

	Predicted	
Actual	bad	good
bad	3	0
good	2	8

Table 5.10. Confusion matrices for the of the artemisinin dataset using a neural network with and without atom pair similarity.*

AP similarity excluded

Training Set

	Predicted	
Actual	bad	good
bad	38	4
good	27	92

Prediction Set

	Predicted	
Actual	bad	good
bad	4	0
good	3	11

AP similarity included

	Predicted	
Actual	bad	good
bad	34	8
good	46	73

	Predicted	
Actual	bad	good
bad	3	1
good	4	10

* The architecture for the CNN with atom pair similarity excluded was 4–9–1 and with the similarity included was 5–5–1

Table 5.11. Confusion matrices for the DIPP dataset using a neural network with and without average atom pair similarity .*

AP similarity excluded

Training Set

	Predicted	
Actual	bad	good
bad	54	0
good	5	176

Prediction Set

	Predicted	
Actual	bad	good
bad	9	1
good	1	31

AP similarity included

	Predicted	
Actual	bad	good
bad	54	0
good	5	176

	Predicted	
Actual	bad	good
bad	8	2
good	2	30

* The architecture for the CNN with atom pair similarity excluded was 4–5–1 and with the similarity included was 5–4–1

Table 5.12. Confusion matrices for the toy dataset using a neural network and without average atom pair similarity .*

AP similarity excluded

Training Set

	Predicted	
Actual	bad	good
bad	18	0
good	1	33

Prediction Set

	Predicted	
Actual	bad	good
bad	3	0
good	2	8

AP similarity included

	Predicted	
Actual	bad	good
bad	17	1
good	2	32

	Predicted	
Actual	bad	good
bad	3	0
good	2	8

* The architecture for the CNN with atom pair similarity excluded was 4–5–1 and with the similarity included was 5–5–1

Table 5.13. Weighted success rates for the various classification algorithms

Method	Dataset	Without Similarity		With Similarity	
		TSET	PSET	TSET	PSET
LDA	Artemisinin	0.51	0.50	0.50	0.50
	DIPP	0.52	0.53	0.52	0.53
	Toy	0.63	0.68	0.55	0.90
PLS	Artemisnin	0.51	0.46	0.49	0.5
	DIPP	0.36	0.53	0.36	0.53
	Toy	0.59	0.51	0.59	0.73
CNN	Artemisinin	0.79	0.80	0.71	0.73
	DIPP	0.98	0.93	0.98	0.86
	Toy	0.98	0.90	0.94	0.90

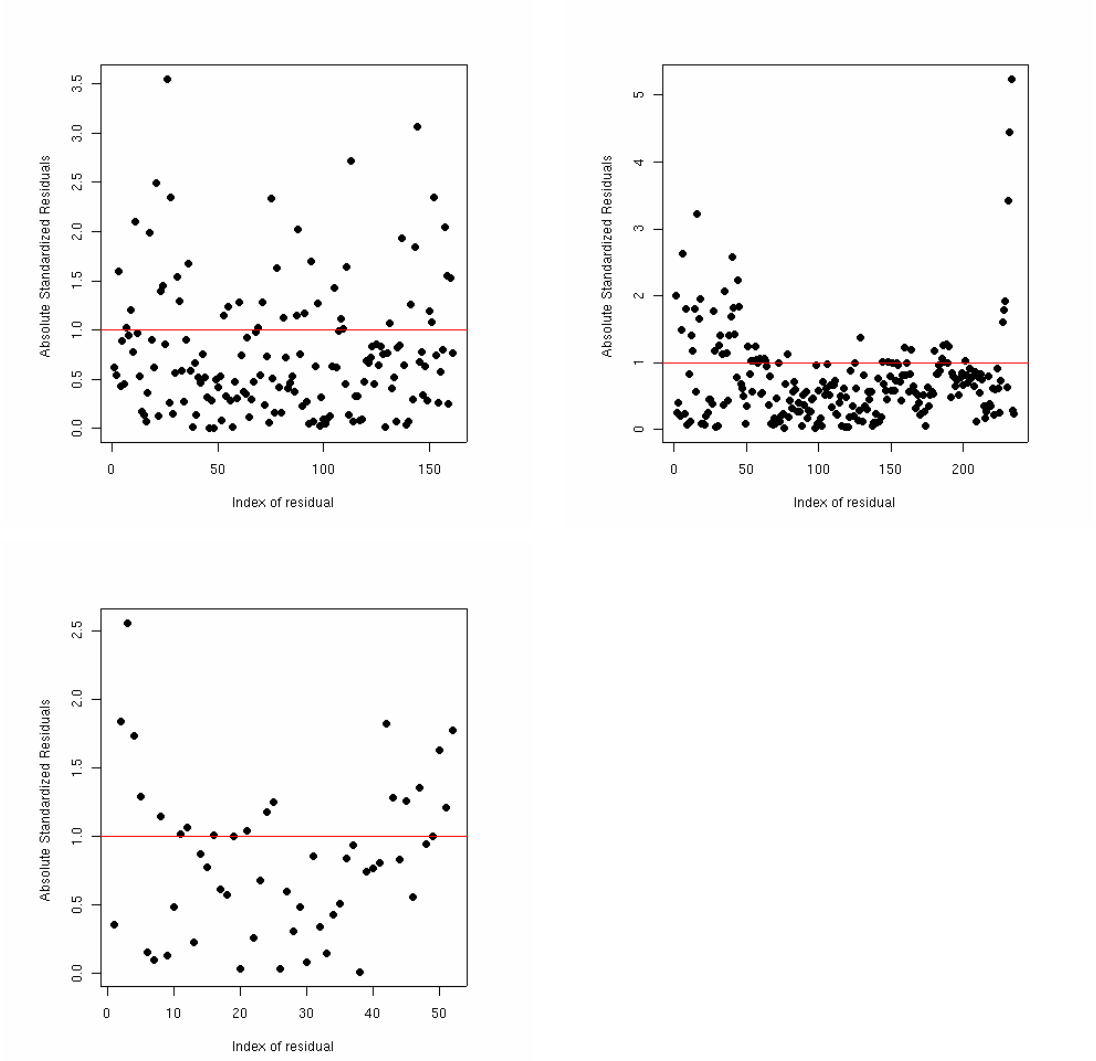

Fig. 5.1. Plots of absolute standardized residuals versus index of residual for the best linear models developed using the training sets for each dataset, with the cutoff value displayed. Residuals lying above the cutoff line are classfied as *bad* and those below as *good*.

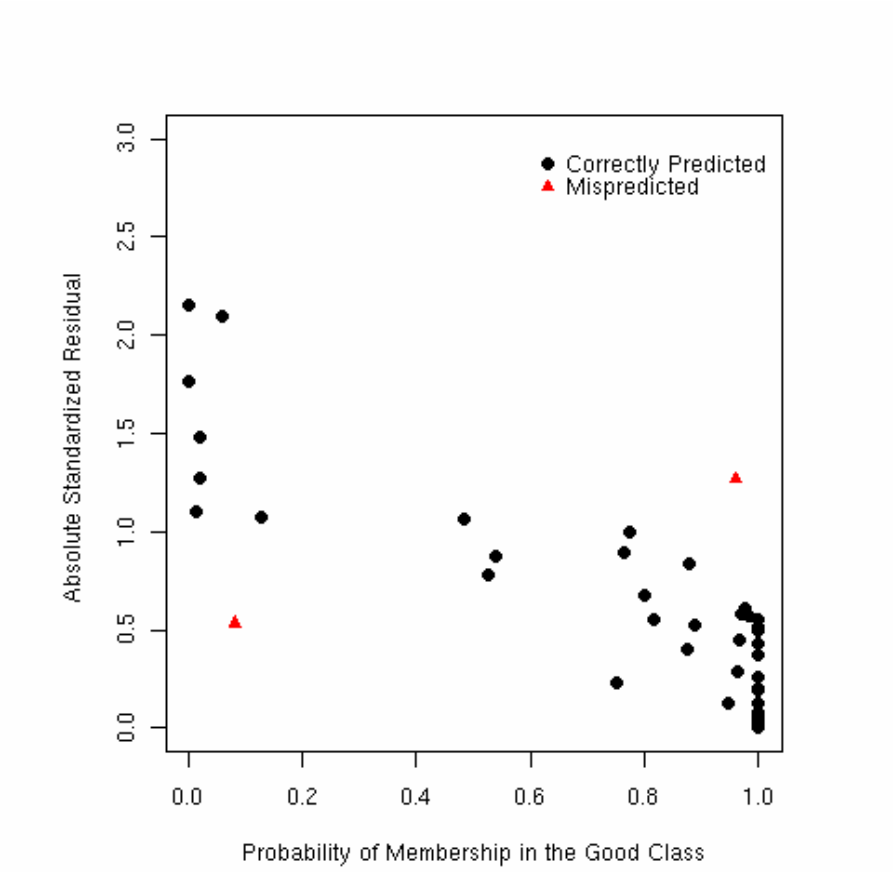

Fig. 5.2. Plot of probability of membership to the good class versus the absolute standardized residual for the DIPP dataset. The probabilities were obtained from the CNN model. The prediction set portion was used to generate the plot.

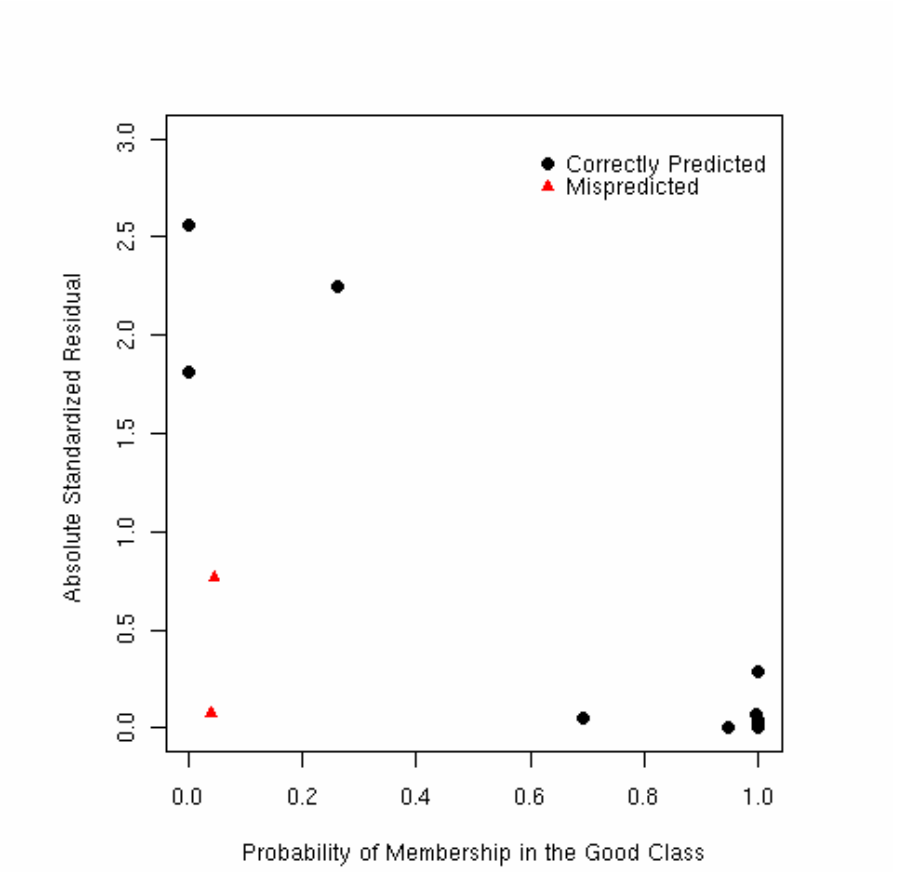

Fig. 5.3. Plot of probability of membership to the good class versus the absolute standardized residual for the toy dataset. The probabilities were obtained from the CNN model. The prediction set portion was used to generate the plot.

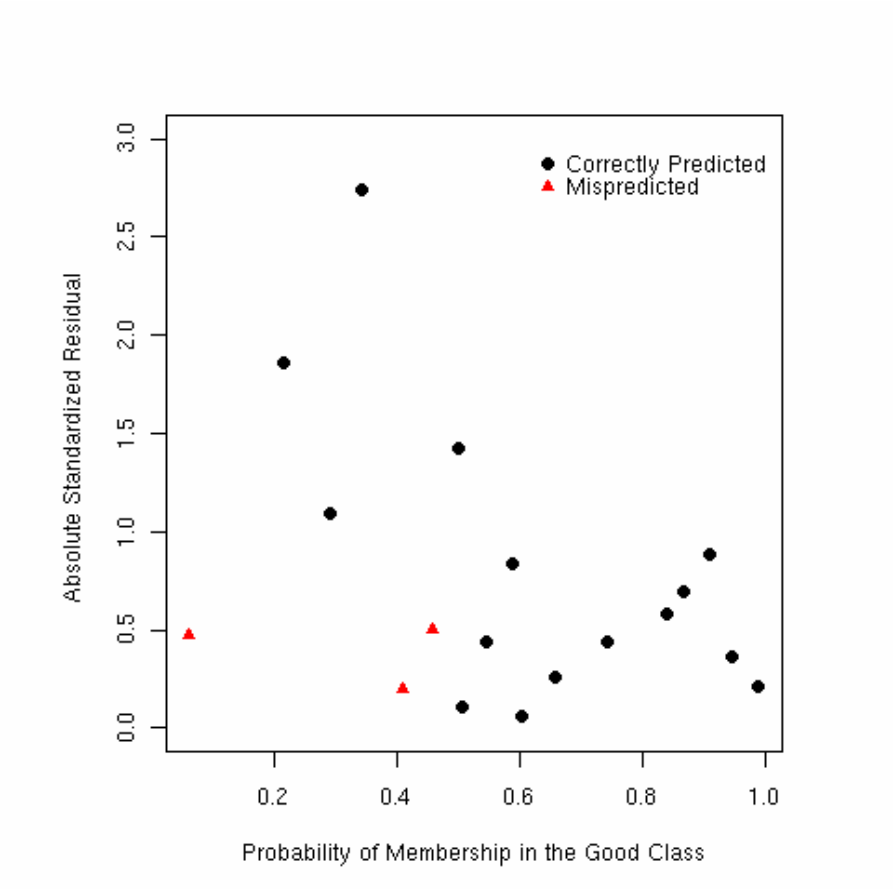

Fig. 5.4. Plot of probability of membership to the good class versus the absolute standardized residual for the artemisinin dataset. The probabilities were obtained from the CNN model. The prediction set portion was used to generate the plot.

References

[1] Carhart, R.; Smith, D.; Venkataraghavan, R. Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications. *J. Chem. Inf. Comput. Sci.* **1985,** *25,* 64–73.

[2] Goll, E.; Jurs, P. Prediction of the Normal Boiling Points of Organic Compounds From Molecular Structures with a Computational Neural Network Model. *J. Chem. Inf. Comput. Sci.* **1999,** *39,* 974–983.

[3] Avery, M. A.; Alvim-Gaston, M.; Rodrigues, C. R.; Barreiro, E. J.; Cohen, F. E.; Sabnis, Y. A.; Woolfrey, J. R. Structure-Activity Relationships of the Antimalarial Agent Artemisinin. The Development of Predictive In Vitro Potency Models Using CoMFA and HQSAR Methodologies. *J. Med. Chem.* **2002,** *45,* 292–303.

[4] Guha, R.; Jurs, P. The Development of QSAR Models to Predict and Interpret the Biological Activity of Artemisinin Analogues. *J. Chem. Inf. Comput. Sci.* **2004,** *44,* 1440–1449.

[5] Liu, S.; Cao, C.; Li, Z. Approach to Estimation and Prediction for Normal Boiling Point (NBP) of Alkanes Based on a Novel Molecular Distance Edge (MDE) Vector, λ. *J. Chem. Inf. Comput. Sci.* **1998,** *38,* 387–394.

[6] Jurs, P.; Chou, J.; Yuan, M. Studies of Chemical Structure and Biological Activity Relations Using Pattern Recognition. In *Computer assisted drug design*; Olsen, E.; Christoffersen, R., Eds.; American Chemical Society: Washington D.C., 1979.

[7] Stuper, A.; Brugger, W.; Jurs, P. *Computer Assisted Studies of Chemical Structure and Biological Function;* Wiley: New York, 1979.

[8] Japkowicz, N. Learning From Imbalanced Datasets: A Comparison of Various Strategies. In *Learning From Imbalanced Datasets: Papers From The AAAI Workshop*; AAAI Press: Menlo Park, CA, 2000.

[9] Kubat, M.; Matwin, S. Addressing The Curse Of Imbalanced Training Sets: One Sided Selection. In *Proceedings Of The 14^{th} International Conference On Machine Learning*; Morgan Kauffmnan: San Francisco, CA, 1997.

[10] Maloof, M. Learning When Data Sets are Imbalanced and When Costs are Unequal and Unknown. In *Workshop on Learning from Imbalanced Data Sets II*; ICML: Washington, D.C., 2003.

[11] Breiman, L. "Using Convex Pseudo-Data to Increase Prediction Accuracy", Technical Report, University of California, Berkeley, 1998.

[12] Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. *Classification and Regresion Trees;* CRC Press: Boca Raton, FL, 1984.

[13] Svetnik, V.; Liaw, A.; Tong, C.; Culberson, C.; Sheridan, R.; Feuston, B. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Comput. Sci.* **2003,** *42,* 1947–1958.

[14] R Development Core Team, "R: A Language and Environment for Statistical Computing", R Foundation for Statistical Computing, Vienna, Austria, 2004 ISBN 3-900051-07-0.

[15] Ripley, B. *Pattern Recognition and Neural Networks;* Cambridge University Press: Oxford, 1996.

[16] Weston, J.; Pérez-Cruz, F.; Bousquet, O.; Chapelle, O.; Elisseeff, A.; Schölkopf, Feature Selection and Transduction for Prediction Of Molecular Bioactivity for Drug Design. *Bioinformatics* **2003,** *19,* 764–771.

[17] Kier, L.; Hall, L.; Murray, W. Molecular Connectivity I: Relationship to Local Anesthesia.. *J. Pharm. Sci.* **1975,** *64,*.

[18] Kier, L.; Hall, L. Molecular Connectivity VII: Specific Treatment to Heteroatoms. *J. Pharm. Sci.* **1976,** *65,* 1806–1809.

[19] Kier, L.; Hall, L. *Molecular Connectivity in Structure Activity Analysis.;* John Wiley & Sons: Hertfordshire, England, 1986.

[20] Randic, M. On Molecular Idenitification Numbers. *J. Chem. Inf. Comput. Sci.* **1984,** *24,* 164–175.

[21] Stanton, D.; Jurs, P. Development and Use of Charged Partial Surface Area Structural Descriptors in Computer Assissted Quantitative Structure Property Relationship Studies. *Anal. Chem.* **1990,** *62,* 2323–2329.

[22] Kier, L.; Hall, L. An Electrotopological-State Index for Atoms in Molecules. *Pharm. Res.* **1990,** *7,* 801–807.

[23] Sharma, V.; Goswami, A.; Madan, A. Eccentric Connectivity Index: A Novel Highly Discriminating Topological Descriptor For Structure-Property and Structure-Activity Studies. *J. Chem. Inf. Comput. Sci.* **1998,** *37,* 273–282.

[24] Stouch, T.; Jurs, P. A Simple Method for the Representation, Quantification and Comparison of the Volumes and Shapes of Chemical Compounds. *J. Chem. Inf. Comput. Sci.* **1986,** *26,* 4–12.

[25] Rohrbaugh, R.; Jurs, P. Molecular shape and Prediction of High Performace Liquid Chromatographic Retention Indices of Polycyclic Aromatic Hydrocarbons. *Anal. Chem.* **1987,** *59,* 1048–1054.