Optimization of Functional Group Prediction from Infrared Spectra Using Neural Networks

Christoph Klawun[‡] and Charles L. Wilkins*

Department of Chemistry, University of California, Riverside, California 92521-0403

Received August 31, 1995[⊗]

In a large-scale effort, numerous parameters influencing the neural network interpretation of gas phase infrared spectra have been investigated. Predictions of the presence or absence of 26 different substructural entities were optimized by systematically observing the impact on functional group prediction accuracy for the following parameters: training duration, learning rate, momentum, sigmoidal discrimination and bias, spectral data reduction with four different methods, number of hidden nodes, individual instead of multioutput networks, size of the training set, noise level, and 12 different spectral preprocessing functions. The most promising approaches included constant monitoring of training progress with a 500 spectra cross-validation set, increasing the number of spectral examples in the training set from 511 to 2588, employing variance scaling, and using specialized instead of multioutput networks. An overall recognition accuracy of 93.8% for the presence and 95.7% for the absence of functionalities was achieved, while perfect prediction was reached for several present functional groups.

INTRODUCTION

Looking back at the literature on computer-assisted structural elucidation, some underlying currents can be observed in the coming and going of algorithmic schemes. Promising new methods often spark hope and enthusiasm among the researchers in the field. This is frequently followed by sobering skepticism and disappointed abandonment, while a few investigators cling onto the residual advances, possibly invent new approaches, and start the cycle all over again. This has been the case with the development of perceptrons,[1] linear learning machines in analytical chemistry,[2] and expert systems for chemical applications.[3] The stream of neural network papers in analytical chemistry[4−9] is currently beginning to be tempered by realistic reservations, but with the variety of applications, programs, and the simultaneous appearance of powerful computers on almost every chemist's lab bench, neural networks are unlikely to be phased out soon. Particularly in feature recognition applications from infrared spectra,[10−15] there are numerous research projects, dominated by the back-propagation algorithm[16,17] in various implementations, whose details have been discussed elsewhere.[18,19] However, accuracy of prediction of *present* functional groups from the spectra not used in training neural networks has exceeded 90% only for a few selected substructures. In fact, until now an overall prediction accuracy of about 80% apparently was the limit for general purpose approaches. However, the results of the present study, with overall accuracy of 94−95% suggest there is room for guarded optimism regarding the utility of neural networks for the purposes described here.

OVERVIEW OF OPTIMIZATION APPROACHES

Neural Network Training. One of the easiest methods to obtain a neural network with good generalization capabilities encompasses frequent tests of the network training progress. These tests are carried out with an independent set of data not used in the training procedure.[13,20] As soon as a new prediction accuracy maximum or a new network error minimum has been encountered, the current network state is stored for later application. Training is completed when no new improvements have been detected for a while. After producing several of these well trained networks, the one with the best prediction accuracy can be chosen for solving the actual problem.[21] Ubiquitous local minima in the multidimensional error surface often threaten to thwart successful training with the back-propagation algorithm, which led to the development of the "flashcard algorithm" for escape from these minima.[18,19] This algorithm, whose name is an analogy to learning vocabulary words with a set of index cards, incorporates "difficult" examples or statistical outliers into the neural network training by overrepresenting them according to their degree of difficulty. Periodic[18,22,23] or continuous[18,19,24−26] addition of random noise to the training examples also assists in avoiding undesired local minima and in achieving a better generalization by circumventing the association of spectral noise spikes with functional groups.[27] In back-propagation training with a sigmoidal transfer function (eq 1), the degree of discrimination and nonlinearity can be controlled[18,28] by varying the "discrimination parameter" β, sometimes also called "temperature".

$$\mathrm{Sig}(x) = \frac{1}{1 + e^{-\beta(x+\theta)}} \qquad (1)$$

Shifts along the x-axis using $\theta \neq 0$ result in a bias toward higher or lower input values, so that some investigators call θ the "bias node". A single bias node value is generally sufficient for most problems.[29] Successful and speedy neural network training largely depends on the magnitude of the learning rate η, which governs how much a given output error will influence the change of the network state. Increasing learning rates during training[30] or use of different learning rates for transitions to the hidden or input layer[31]

[‡] Current address: Applied Automation, P.O. Box 9999, Bartlesville, OK 74005-9999.

[⊗] Abstract published in *Advance ACS Abstracts,* December 15, 1995.

0095-2338/96/1636-0069$12.00/0

have been reported, and it has been suggested that a value for η which is inversely proportional to the number of hidden or input nodes should be chosen.[32] The need for locally different learning in a multidimensional error surface has been met by assigning individual learning rates to every adjustable weight in the neural network.[33] From an initial value η, training progresses by changing η locally after every epoch, allowing large steps in shallow gradients and small steps in steep minima. To the authors' knowledge, no systematic study has been conducted on the influence of α, the momentum term in back-propagation training. It helps maintain the general training direction in the error surface to speed up the neural network convergence but has surprisingly little influence on the prediction quality, as will be discussed subsequently.

Much has been said and written about the optimal number of hidden nodes, *H*, necessary to complete a neural network training session in a satisfactory fashion, and a variety of mathematical solutions including proofs have been offered.[32,34,35] But it seems that for large, real-world data sets, no simple formula can be applied. With too few hidden nodes, the network is "too stupid" to derive a generalized model from the training data, and with too many nodes, a "grandmother network" results, which can remember everything it has seen before but is unable to handle new situations properly. In the extreme, a network with far too many adjustable weights constitutes an underdetermined problem with no defined solution. This scenario is dangerously close in two papers where the authors suggested setting *H* equal[36] to the number of spectral examples, *N*, or proportional to that number[37] according to $H = 0.7N$. In a previous paper,[19] we reported that when the number of spectral examples is increased by a factor *k*, the number of hidden nodes for satisfactory training may be increased by $k^{0.5}$. Pruning hidden nodes as needed during training avoids inefficient networks but favors prominent features in the data set.[38] The opposite tactic—growing hidden nodes—presents some overtraining danger due to too many weights.[39] To deal with this dilemma, fuzzy logic has been used to determine the necessary number of hidden nodes.[40] Another study[32] revealed that, for completely random data, $H \approx N$, but highly ordered cases such as spectral data necessitate $H \ll N$. It also has been shown[29] that a single hidden layer is sufficient to approximate any continuous function, *i.e.*, a single layer is appropriate for most nonlinear problems in chemistry.

One of the better approaches to improved neural network prediction accuracy involves the use of several networks with single outputs instead of a single network with several outputs. In a quantitative analysis of pyrolysis mass spectra,[41] three networks with one output node each outperformed a single network with three output nodes by a factor of ~2. When interpreting infrared spectra with back-propagation networks, specialized networks dedicated to a few similar functional groups[42,43] or single-output networks[15] yielded better prediction accuracies, as well. The network specialization was driven even further—perhaps too far—by training one network to recognize the presence and another to predict the absence of a feature. The network for features present was trained with a set consisting of many present, but few absent, examples, while the training set composition was reversed for the other network.[44]

Data Sets. Proper selection of a training set with the right type of data preprocessing and an appropriate number of data points may outrank neural network parameters in importance. For interpretation of infrared spectra, most researchers scale the spectral range to absorbances between 0.0 and 1.0. As one example of data preprocessing strategies, a large variety of statistical preprocessing functions was applied to near-infrared data for recognition of recyclable plastic types.[45] Data point reduction methods for infrared spectra range from simple resolution reduction in the higher wavenumber region,[20] to nonlinear approaches[10,18] as well as more sophisticated methods, *e.g.*, removal of less important coefficients after Fourier or Hadamard transformation,[46] followed by inverse transformation into the original data domain. However, it was found that Hadamard compression and boxcar averaging give almost the same prediction results.[47] Wavelet compression of infrared data[48] was shown to give the best results when the number of data points was reduced from 460 to ~50. (A wavelet transformation works similar to a Fourier transformation but uses functions different from sine/cosine and produces a two-dimensional matrix with frequency and temporal information instead of two vectors with frequency and phase information. Compression is achieved by zeroing the smallest coefficients in the matrix prior to inverse transformation.) The effect of spectral resolution on prediction quality was studied,[13] and, for the prediction of O−H, N−H, and C−H groups from infrared spectra, no difference between 2 and 16 cm^{-1} resolution could be detected.[49] These authors also found that including or excluding specific regions of the IR spectrum produced worse results than utilizing the entire spectral range. Principal component analysis (PCA) represents another important method for reducing the dimensionality of a data set, although the nature of this mathematical manipulation may impose a linear structure on the data set, possibly obscuring nonlinear features. In a study on highly compressed infrared data for feature recognition, Meyer *et al.*[50] found an overall prediction accuracy of 75.0% for present functional groups including eight data points from PCA compression, and with 10 data points, the accuracy went up to 78.9%. Using PCA to preprocess ultraviolet visible (UV−vis) spectra for quantitative analysis, a neural network was trained with the orthogonalized data set from PCA. Unnecessary input data were determined by "pruning" input nodes, with very small contributions to the overall error. This resulted in faster training and a lower calibration error.[51] Possibly, similar results could have been obtained by using available methods which calculate the number of "important" factors after PCA orthogonalization.[52,53]

Training set composition plays an influential role in the achievable prediction quality with neural networks, especially with multioutput neural networks, where it is impossible to avoid imbalances of present and absent classifiers. An algorithm to compose a compromise between data set size and balanced classifier distribution has been devised recently,[20] and *a priori* knowledge about the training set composition may facilitate better learning by weighting the neural network outputs according to class occurrence.[19,54] Doubling the number of Raman spectral examples from 22 to 44 reduced the RMS output error by ~80% for a quantitative study,[36] and more examples yielded a more flexible system for quantitative modeling of Rhodamine 6G and B mixtures.[55] For qualitative predictions, it was proposed that the hard-encoded 0/1 class membership of functional groups should be abandoned and that, instead,

fuzziness should be introduced into the classification procedure,[56] *e.g.*, different membership magnitudes in a C═O class for esters (higher) and aldehydes (lower).

This paper will show that, through combination of carefully conducted optimizations, it is possible to achieve an overall success rate well above 90% for present functionalities *without* neglecting the successful recognition of the *absence* of groups. Many of the findings in this paper can also be applied to neural network training of other data sets related to other types of problems in chemistry.

EXPERIMENTAL SECTION

Infrared spectra from the NIST/EPA gas phase infrared database (database 35, JCAMP-DX format, National Institute of Standards and Technology, Gaithersburg, MD) form the basis of all neural network data manipulations. This database comes in two spectral sets, both with one data point every 4 cm^{-1}: (a) NIST file, 2120 spectra, 825 data points between 550 and 3846 cm^{-1}; (b) EPA file, 3108 spectra, 880 data points between 450 and 3966 cm^{-1}. No duplicates exist in this collection of 5228 spectra, but only 5191 spectra have structures assigned to them. All structures were checked for consistency with the associated molecular formula, and a program was written in Borland C++ 4.0 (Borland, Scotts Valley, CA) to inspect each structure for suspicious atom valencies, illegal bonds, and incorrect double bond equivalents. Twenty-four structures/molecular formulas were determined to be faulty and corrected subsequently. Excluding seven structures with bond uncertainties and three with more than one molecule, the available pool of spectra shrank to 5181 different compounds, encompassing 19 different types of atoms and 66 different bond types. No further pruning of this list was carried out so that the results of this paper retain as much generality as possible.

Prior to any neural network training, it was necessary to generate a uniform data set from the two spectral files. First, all EPA spectra were truncated to the NIST range of 825 data points, followed by a reduction of the entire set of spectra to 512 points each to facilitate data compression methods such as fast Hadamard and Fourier transformation: The first 356 data points between 550 and 1970 cm^{-1} were left unchanged in their resolution of 4 cm^{-1}, but the next 468 data points were compressed via boxcar averaging to 156 points between 1982 and 3842 cm^{-1} with a resolution of 12 cm^{-1}, similar to the procedure used by Novič and Zupan.[46] The last data point at 3846 cm^{-1} was omitted, and the resulting 512 data points were scaled between 0.0 and 1.0 absorbance.

Fast Fourier and Hadamard transformations and data reductions were carried out in standard C following algorithms outlined in Zupan's text,[57] and each resulting spectrum was scaled again between 0.0 and 1.0 absorbance. Simple matrix algebra routines were developed in C adhering to standard mathematical equations,[58] but more complex procedures for eigenvalue and eigenvector extraction (Jacobi transformation for simplicity) were taken from a collection of numerical recipes in C,[59,60] which in turn had been adapted from older FORTRAN[61,62] and PL/1[63] code. The covariance matrix for PCA was generated from 511 spectra (512 data points each) with equations from Zupan's book.[57] The backpropagation neural network code was written in standard C to run on a variety of platforms.[20] It utilized the entire range of the flashcard algorithm training extensions[18,19] for circumventing local minima. Most computation experiments executed on a Digital AlphaServer 2100 4/275 running OpenVMS 6.1, but some of the initial experiments were conducted with SISAL[64−67] core code (remaining code in C) using a Cray C90 under UNICOS 7.0 with up to 16 vector processors. Most training runs on the AlphaServer were completed within 5−30 min, which translates into a training time of ∼440 spectral presentations per second (256-25-26 neural network architecture = 7050 adjustable network weights) for the standard back-propagation algorithm. In comparison, the Cray times were approximately five times faster and about four times slower for a PC with an Intel Pentium-100 CPU running the Unix freeware operating system Linux 1.2.3.

Table 1. Classifier Abundance in the Database: Test and Training Sets

functional group	database	test set	training set
O−H	1264	214	221
N−H	599	111	101
C−N	1089	200	201
X≡Y, X═Y═Z	326	44	44
C═O	1664	203	201
C−O	2403	276	299
C═C	952	67	73
C⋯C	2352	259	264
halogen	1193	84	77
N−O, N═O	260	44	44
CH_3, CH_2	4392	410	420
primary alcohol	297	44	44
secondary alcohol	305	44	44
tertiary alcohol	143	44	44
Ar−OH	302	44	44
COOH	228	44	44
primary amine	286	44	45
secondary amine	187	44	44
tertiary amine	157	44	44
(CO)NH	249	44	44
(CO)OR	671	44	49
(CO)H	125	44	44
C(CO)C	460	44	44
aromatic C_6-ring	2231	230	238
pyridine ring	184	44	44
C−Cl	680	50	61
total spectra	5181	500	511

Unless specifically noted otherwise, the standard parameters for neural network training were as follows: 128 input nodes (reduction from 512 data points with Hadamard transformation), 25 hidden nodes, and 26 output nodes for the presence or absence of the functional groups listed in Table 1. An output value <0.5 was interpreted as an absent functionality, and a present one otherwise. There was no need to include an "indecision" region, as a graph of the training progress illustrates in Figure 1. The initial cluster of output errors around 0.5 was soon pushed toward the desirable state of 0.0, leaving few output errors in the middle region. A standard training epoch consisted of $N = 511$ spectra presented in random order, followed by flashcard algorithm evaluations outlined elsewhere.[18−20] Output errors between 20 and 95% were considered "large errors", and "extreme errors" >99.5% were reduced after each standard epoch up to a maximum of 20*N* attempts without the option of aborting training. After every presentation cycle of 511 training spectra (including training according to the flashcard algorithm), a test set of 500 spectra not used in the training was used to gauge the suitability of the network for functional

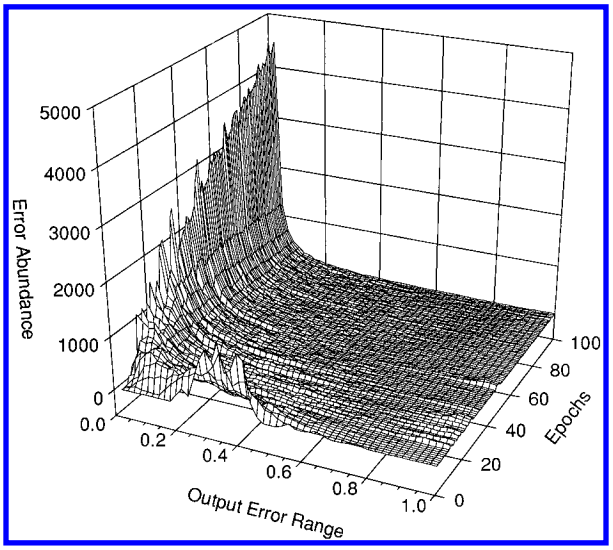

Figure 1. Error distribution progress during neural network training. Each datum represents the abundance of output errors in an error range bracket of 0.02.

group predictions, and the accuracy of present and absent predictions was noted each time. Prior to training, the network weights were assigned random values between −0.1 and +0.1, and 1.0% random peak-to-peak noise was added to each spectrum before it was processed by the neural network. This was accomplished by adding a random value between −0.05 and +0.05 to every data point (every spectrum had been previously normalized to maximum absorbance of 1.0). Neural network training proceeded with a learning rate $\eta = 0.02$, a momentum term $\alpha = 0.9$, a sigmoidal discrimination parameter $\beta = 1.0$, and a sigmoidal x-axis shift $\theta = 0.0$, and most presented data are averages of five training/testing results (Figures 2, 3, 5, 6, 9, 11−15, and 17).

NEURAL NETWORK TRAINING PARAMETERS

All attempts to improve neural network prediction accuracy in this paper are aimed at improving the prediction quality of *present* functional groups. Because training and test sets of almost any size lean strongly toward absent groups, they do not pose prediction problems, as success rates around 95−100% in the literature show. In addition to the quest for better recognition of present substructures, minimal training times are desirable when no prediction improvement can be achieved.

Learning Rate. Training of neural networks is largely controlled by the speed at which changes are incorporated into the weight structure. Too high a learning rate causes the network to jump across narrow local minima or bounce wildly across the error surface, while small learning rates mainly lead to slow training progress. But is it possible to improve the prediction quality by tweaking this neural network parameter? Figure 2 shows that this is not the case, although the learning rate η was varied across three orders of magnitude. In Figure 2a, line P traces the average of the best prediction accuracy for present functional groups found at any time during the training runs, while also showing the prediction quality for absent groups (A) at these points. The rates remain at almost the same level until learning rates >0.1 cause the network to become inefficient in its search for the global minimum. At the same time, the epoch curve

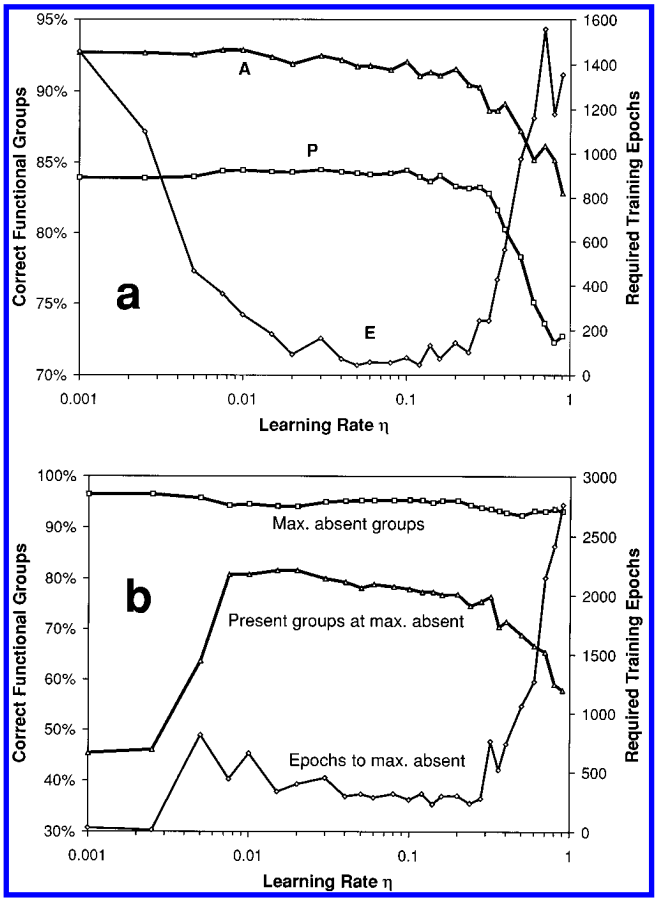

Figure 2. Influence of the learning rate η on functional group prediction and training effort. (a) Maximizing the prediction of present groups (P = average maximum of present groups found, A = average absent group prediction at that point, E = average epochs required to reach A). (b) Maximizing the prediction of absent groups.

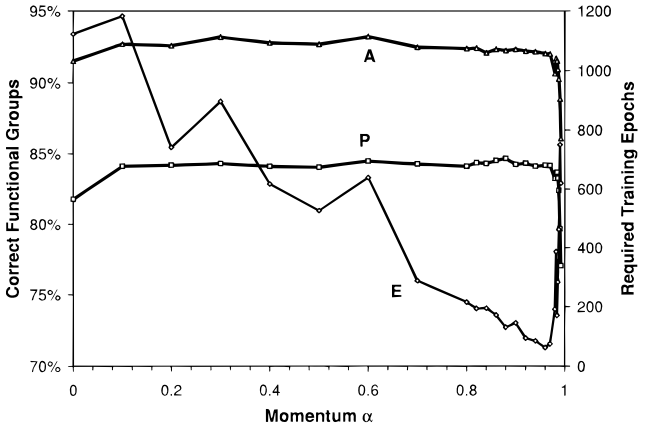

Figure 3. As in Figure 2a, but for the momentum term α.

(E) in Figure 2a confirms that for small η, training progress is slow, and for large η, the network has difficulties locating desirable minima efficiently, yielding lower quality networks in more time than necessary. Locating the best possible constellation for *absent* functionalities works in a very similar way. The curves in Figure 2b reveal that for small η, the best prediction rates for absent groups come and go while the network is still working on the present groups, and for higher η, the best recognition rates for present groups are reached before the rates peak for absent ones. For example, with $\eta = 0.04$, present groups are predicted best after 72.4 epochs with 84.3% (absent: 92.2%), and absent groups after

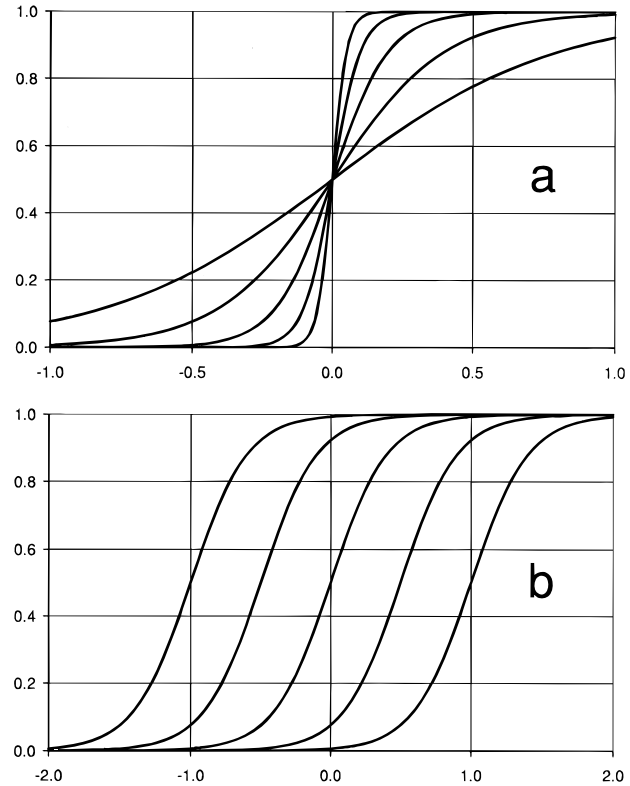

Figure 4. Sigmoidal function (eq 1): (a) varying discrimination parameter β and (b) shifting the curve along the x-axis by manipulating the bias term θ.

295.4 epochs with 95.2% (present: 79.3%). These numbers also are exemplary for another trend: With progressing training, the prediction quality of present groups loses more (−5.0%) than what is gained for absent groups (+3.0%), and in order to maintain good recognition rates, neural network training must be stopped well before the zenith of accurately predicted absent functionalities, and certainly before the RMS output error reaches the point of no more improvement. Slowly increasing η by 1% each epoch from 0.005 to a maximum of 0.1 during training[30] showed no significant effect on the prediction quality of either present or absent functional groups.

Momentum. Originally conceived to aid the back-propagation training algorithm to escape from local minima by maintaining the learning direction from the previous step, the momentum parameter α does just that—and not more. The curves in Figure 3 exhibit very flat prediction rates throughout the entire momentum range with less and less epochs required, until they go "berserk" for $\alpha > 0.95$. Beyond this point, the benefit of the momentum turns into a trap, where it becomes increasingly harder for the neural network to *reverse* training directions when needed.

Discrimination. Most researchers do not manipulate the steepness of the sigmoidal function (eq 1) around $x = 0$ (Figure 4a), which can be viewed as the nonlinear transition between predicting present and absent functional groups. With β close to 0, the sigmoidal curve turns into a flat horizontal line, providing almost no help to distinguish between large positive or negative values of x, so that the network state can flow almost freely between the two extremes during training, and only the most pronounced examples can be learned correctly. On the other hand, a very steep steplike transition makes the reversal of an

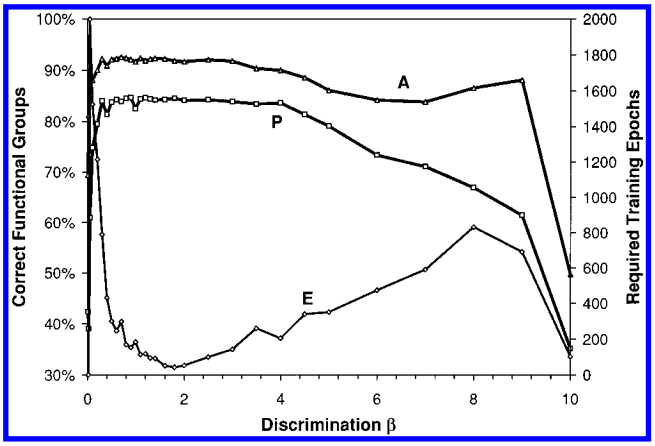

Figure 5. As in Figure 2a, but for the sigmoidal discrimination β.

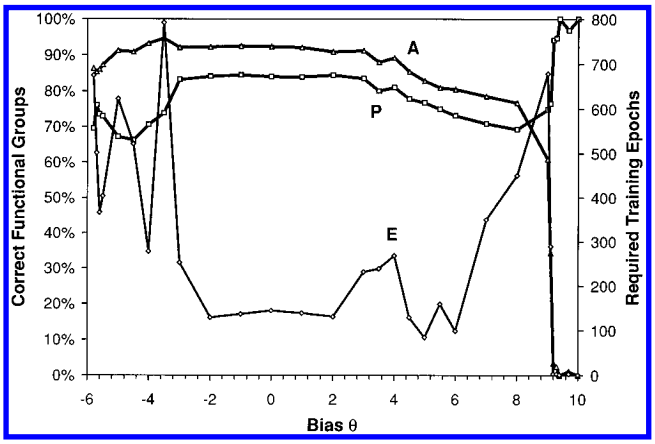

Figure 6. As in Figure 2a, but for the sigmoidal bias θ.

erroneous network state almost impossible, and once the network finds itself on either side of $x = 0$, it takes an overwhelming effort to switch sides. In addition, a neural network interpretation based on black-and-white decisions negates the purpose of its existence to turn continuous input data into decisions based on the weighted importance of each datum. Unfortunately, none of these considerations lead to improved predictions in Figure 5. Within the acceptable region of $\beta = 0.5-4.0$, the achievable recognition accuracy hovers at the same level, although the minimum in the epoch curve at $\beta = 1.9$ suggests that there is an optimum degree of discrimination. The poor prediction values outside the middle region support the initial assumptions made about β. At small values, the network has difficulties coming even to the most obvious conclusions and keeps dithering for a long time, and for large values of β, it makes "rash decisions" and is unable to correct them.

Bias. Moving the decision center away from $x = 0$ has benefits when one absolutely needs correct positive identifications but is not so concerned about false positives and vice versa. Also, when the x-values do not amount to anything significantly positive or negative, the introduction of a bias term θ moves the decision plane away from the "slippery" region around zero (Figure 4b). Neither of these scenarios applies to the interpretation of infrared spectra, and Figure 6 supports this notion. Between $\theta = -2.0$ to $+2.0$, everything is well, and the combination of present and absent group predictions are at a maximum level with a minimum in training effort. On the far right, the network becomes completely unusable, confirming the presence and denying the absence of every existing functional group, because θ

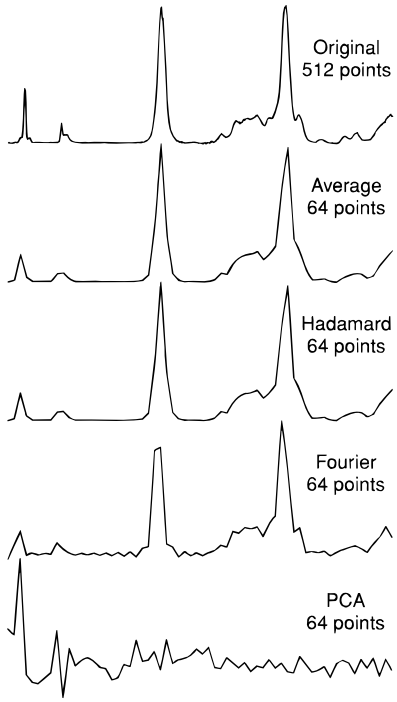

Figure 7. Four data reduction methods applied to the spectrum of 2-hydroxypropanoic acid.

has shifted the sigmoidal function so far into one direction that the steep deciding region cannot be reached. On the far left in Figure 6, the picture is just about to turn the other way, always predicting absent and never any present groups. Thus, in order to gain anything in the prediction of present functionalities by manipulating θ, the recognition quality for absent groups would have to be sacrificed entirely.

FEED-FORWARD NETWORK ARCHITECTURE

Input Data Reduction. What is the minimum number of data points we must use to obtain good prediction rates?[50] The correct answer to this question should be sought prior to doing anything else with neural networks, because a reduction from, *e.g.*, 825 to 64 data points speeds up training, prediction, and research progress by a factor of 13, not to mention the diminished impact on the research budget when less powerful computers will do the job just as well. Four data reduction methods were implemented to generate reduced data sets from a set of 512 data points per spectrum: boxcar averaging, Hadamard transformation, Fourier transformation, and principal component analysis (PCA). Boxcar averaging was chosen for its simplicity, where two or more data points are replaced by their average. Hadamard transformation works well when straight spectral baselines must be kept, while Fourier transformation has its advantage in preserving even minor peak shapes fairly well. However, the latter method also introduces "squiggle" artifacts into spectral baselines, as the comparison of reduction methods in Figure 7 reveals. Interestingly, the differences between Hadamard transformation and boxcar averaging are almost imperceptible. They give rise to nearly the same data sets, so that Figure 8 only shows the progression of Hadamard and Fourier reductions to fewer data points. At the lowest resolution, using the data set is almost pointless because spectral scaling always produces a

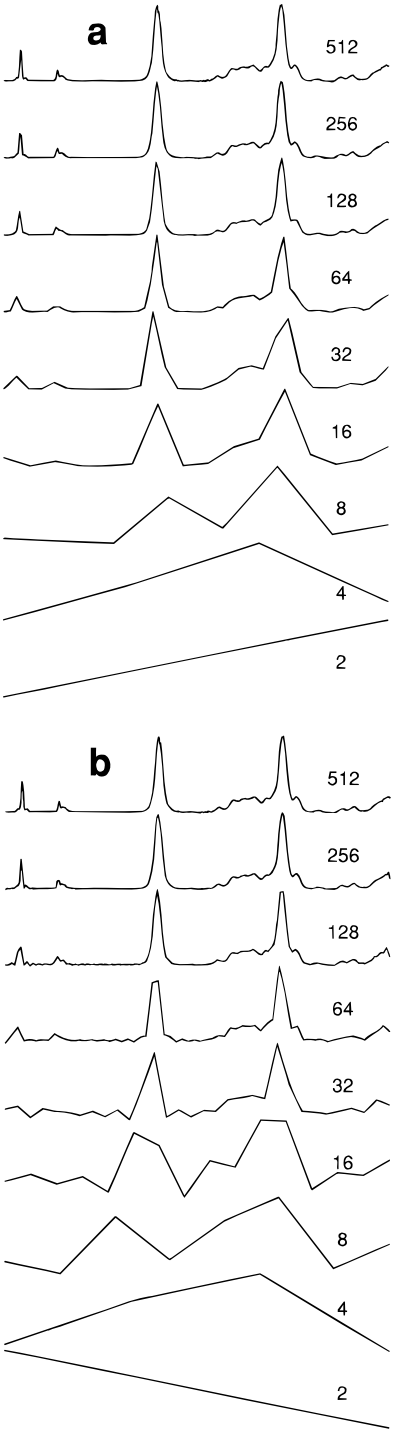

Figure 8. Change in spectral appearance reducing the number of data points down to a minimum: (a) Hadamard transformation or boxcar averaging (same results) and (b) Fourier transformation.

"spectrum" with the data points 0.0 and 1.0. PCA data do not bear any resemblance to the actual spectral due to the axis orthogonalization process, although it appears in Figure 9 that PCA data reduction produces slightly better results than any other method. However, a glance at Figure 10 shows that this result is merely a façade. The thick Hadamard lines follow a steady incline in prediction quality, accompanied by an even decrease in RMS prediction error. In contrast, the thin PCA curves jump wildly up and down, depicting a quite unstable system, but occasionally producing a prediction spike higher than the Hadamard curve. Because of this unsteady behavior, PCA was excluded from any

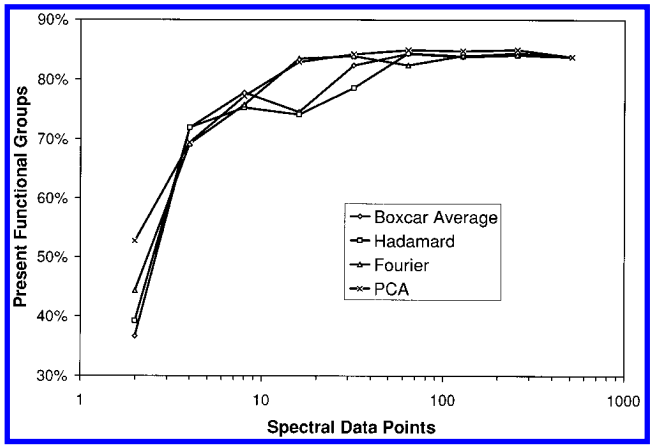

Figure 9. Correct prediction of present functional groups using a varying number of input data points and four different data reduction methods.

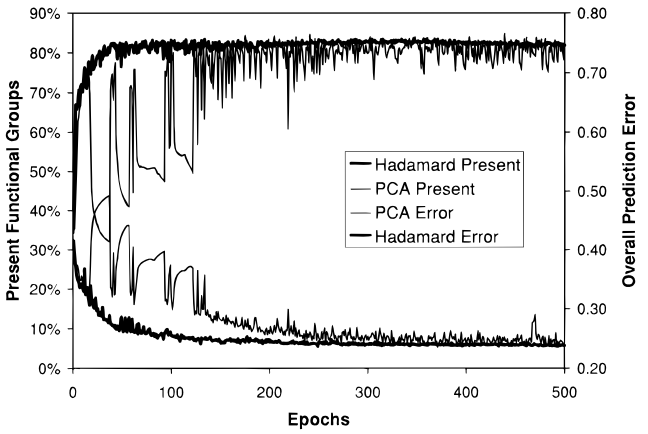

Figure 10. Comparison of prediction quality progress during neural network training for data reduction to 128 points with Hadamard transformation and principal component analysis (PCA).

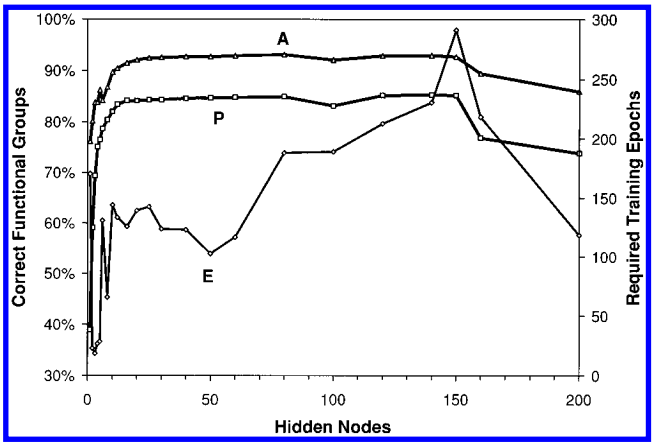

Figure 11. As in Figure 2a, but for hidden nodes.

further consideration for improving the prediction rates of present functional groups. A review of the traces in Figure 9 confirms that it makes almost no difference using 512, 256, 128, or 64 data points for neural network training, with Hadamard and boxcar average reductions being at a slight advantage. At 32 and 16 data points, the correct recognition of present functional groups still remains above 80%, but here Fourier reductions with peak shape preservation jump ahead with a wider margin, merging again at eight data points with ~77% correct recognition. This rate is quite remarkable, considering the fact that in order to collect an infrared spectrum with eight data points between 500 and 4000 cm^{-1}, the spectral resolution has to be in the neighborhood of 500 cm^{-1}! Even four data points with 70% correctly predicted functional groups produce much better answers than one would get by random guessing (21%) based on the test set composition (Table 1).

Hidden Nodes. No simple formula exists for the *a priori* determination of a comfortable number of hidden nodes to yield a satisfactory network without slow progress or overtraining. However, there is an absolute ceiling for any given data set, in that the number of adjustable network weights must not be larger than the number of input data points times the number of examples. In our case, the upper limit is 128 data points × 511 spectra = 65 408 weights, or 65 408 weights/(128 input + 26 output nodes) = 425 hidden nodes. Depending on the data set size and correlation among the examples, satisfactory training can be achieved utilizing less than 10% of the hidden nodes suggested by this ceiling. This assumption is supported by Figure 11, where 15−60 hidden nodes lead to approximately the same results in a minimum of training effort. Beyond this range, the network has to pour increasingly more effort into achieving the desired training state, and a drop-off occurs after 150 hidden nodes, when the network switches from generalization to a "grandmother" network, where the network weights merely function as storage units for the training set or even remain unused. The fact that this decrease happens after 150 and not after 425 hidden nodes suggests that 128 data points still contain a lot of "dead wood", with the optimum data set size possibly being 128 × (150/425) = 45 data points. And indeed, most of the curves in Figure 9 are beginning to slip between 64 and 32 data points, supporting this assumption. On the low end in Figure 11, it is clear that a minimum of 15 hidden nodes must be supplied to accommodate correlating the input nodes with the desired output nodes for this data set.

Specialized Networks. One of the most promising optimization steps follows the "divide and conquer" rule: Neural networks devoted to the recognition of a single functional group perform significantly better than networks which must work on 26 substructures at the same time. For this reason, 26 specialized networks were trained to predict the presence or absence of each individual functionality. All other network parameters including the number of hidden nodes remained the same. From the standpoint of efficiency, this may not be satisfactory because the reduction from 26 to one output node demands much less capacity from the neural network; the minimum number of hidden nodes was found to lie around 5. However, for the sake of changing only one parameter at a time, 25 hidden nodes were used in all specialized networks. Also, the effort to train 26 networks and keep them in the computer memory for prediction purposes is much greater than for a single network, but with constantly dropping hard disk prices and increased RAM capacity, this is a small price to pay for much better prediction accuracy, as Figure 12 illustrates. Every single one of the 26 present functional group prediction rates improves as a result of specialization, some by only a few percent (C═O, primary alcohol), but other figures leap by as much as 30% (tertiary amine, ketone). The overall prediction rate also improves from 84.0% to 91.7% for

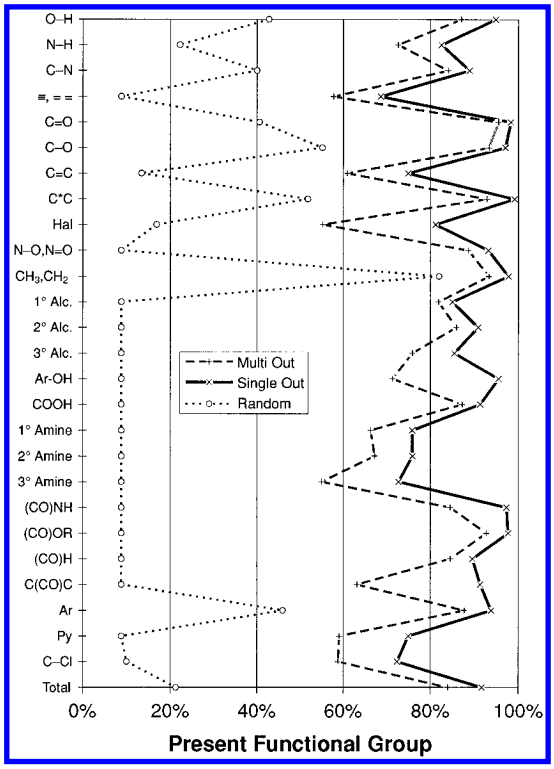

Figure 12. Correctly predicted present functional groups with random guessing, multioutput, and specialized networks for all 26 substructures used in this paper.

present functional groups, much higher than random guessing at 21%, which is depicted by the leftmost curve in Figure 12.

TRAINING SETS

Number of Spectral Examples. Many investigators use as many infrared spectra as they can get ahold of to conduct neural network training, *e.g.*, Robb and Munk's training set[10] consisted of 6695 spectra. In many cases, this is the correct approach, because neural networks operate best when based on a largely overdetermined data set—but only up to a certain point. In this part of the study, not only the number of training set spectra was varied but also the number of hidden nodes to accommodate the increased amount of data with higher network capacity. Figure 13 underscores the necessity of more hidden nodes using larger training sets, because the prediction quality curves for the same number of hidden nodes diverge for more training spectra. As Figure 14 shows, 511 training spectra are far from producing the best neural network, and the prediction quality drops considerably when using only half as many spectra. Increasing the training set size also produces the desired effect, a better overall prediction rate. However, at 2588 spectra (~50% of the entire NIST gas phase infrared database) we encounter the point of diminished returns, which leads to two possible conclusions: (a) Beyond a certain point, additional information will not add anything significant to the whole picture. While training mass spectra with the same set of functional groups,[20] the prediction quality actually *decreased* going from 1525 to 2588 training spectra. (b) The training set imbalance between less and highly abundant substructures becomes more pronounced with increasing training set size,[20] and they rapidly approach their actual distribution in the database.

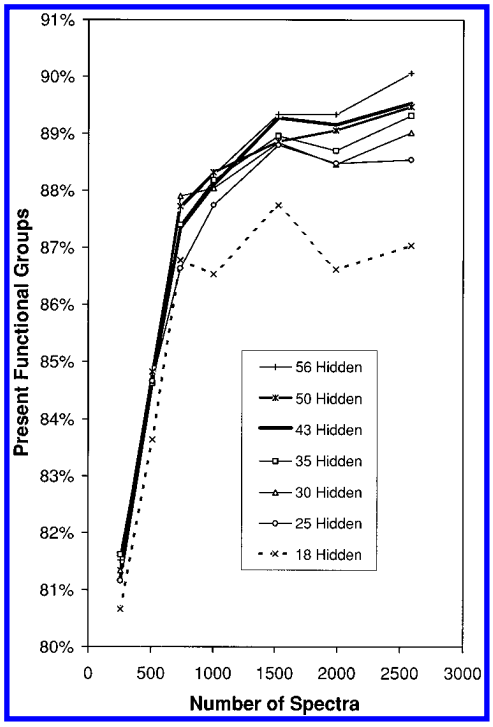

Figure 13. Present functional group prediction accuracy employing different sizes of training sets. The number of hidden nodes as an additional variable reflects the need for increased network capacity.

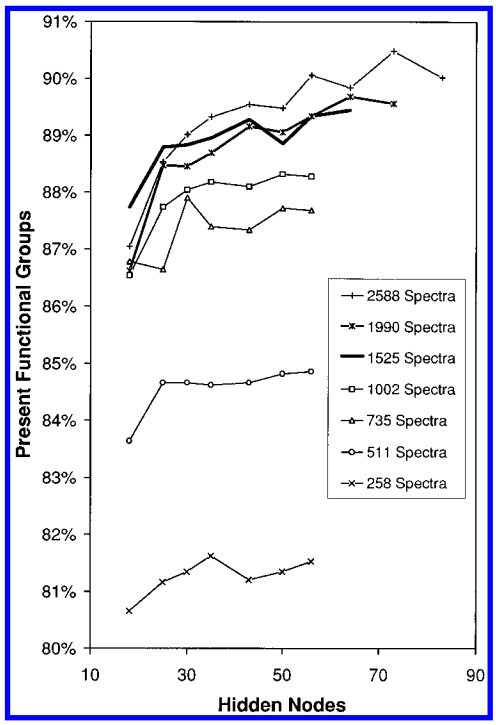

Figure 14. Prediction results for present functional groups with different numbers of training set spectra and hidden nodes.

Therefore it may be possible that when the same number of spectra are drawn from a much larger database stock, the point of diminished return will be reached closer to 100% correct recognition using more than 2588 spectra.

Noise. Low random noise levels added to the infrared spectra during training can avoid correlation of noise spikes in the training spectra with the presence of functional groups.[27] However, for the present data set, this danger is negligible, because the recognition curves in Figure 15 remain at the same level using random noise with peak-to-

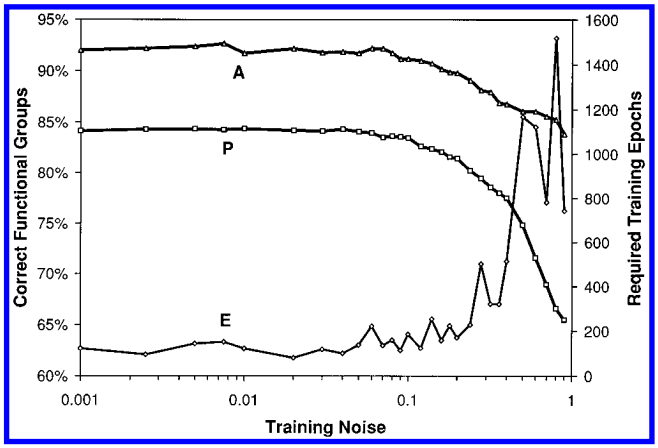

Figure 15. As in Figure 2a, but for added peak-to-peak random noise levels.

Table 2. Infrared Spectrum Data Ranges (2-Hydroxypropanoic Acid) After Preprocessing

preprocessing function	data range	
	minimum	maximum
range scaling	0.0	1.0
mean centered variance	−0.466	6.474
mean centering	−0.452	0.839
variance scaling	0.0	3.541
first derivative	−0.596	0.487
second derivative	−0.877	0.371
autoscaling	−1.092	2.947
normalization to unit length	0.0	0.558
Savitzky−Golay smoothing	−0.001	0.845
Fourier transform, real coefficients	−3.505	8.602
Fourier transform, imaginary coeff	−3.843	3.843
Hadamard transform	−4.235	8.602

peak levels of up to ∼10% of the maximum absorbance value. Only when the noise level approaches the signal size, prediction rates drop while training effort goes up. But even at a signal-to-noise ratio of 1.1 (90% noise), the network still predicts 65% of all present functional groups correctly.

Preprocessing. Statistical preprocessing of the entire training set is sometimes used to remove inadequacies in infrared spectra to express the presence of certain bonds with strong signals or to compensate for weakly populated spectral areas.[45,57] Several of these methods only work when the mean, variance, maximum, and minimum for every input data point of the training set is also made available for prediction purposes, even though it is statistically incorrect to apply the training set values to test spectra. But then again, does it matter when they are used in conjunction with statistically intractable neural networks? In the following equations, it is understood that an original data point x_{mn} (x'_{mn} after preprocessing) belongs to example m out of M spectra in the training set and to data point n out of N data points per spectrum. Other notational examples are as the minimum value of all spectra at data point n, the mean of all data points from spectrum m, and σ_n the variance at data point n for all spectra. Figure 16 depicts the effect of each of the preprocessing functions on the infrared spectrum of 2-hydroxypropanoic acid. In Table 2, the data range of this spectrum after preprocessing illustrates the results even further, because *no* scaling between 0.0 and 1.0 was performed prior to using the manipulated data sets for neural network training.

Range scaling (eq 2) confines the absorbance values of each spectral data point to a range between 0.0 and 1.0,

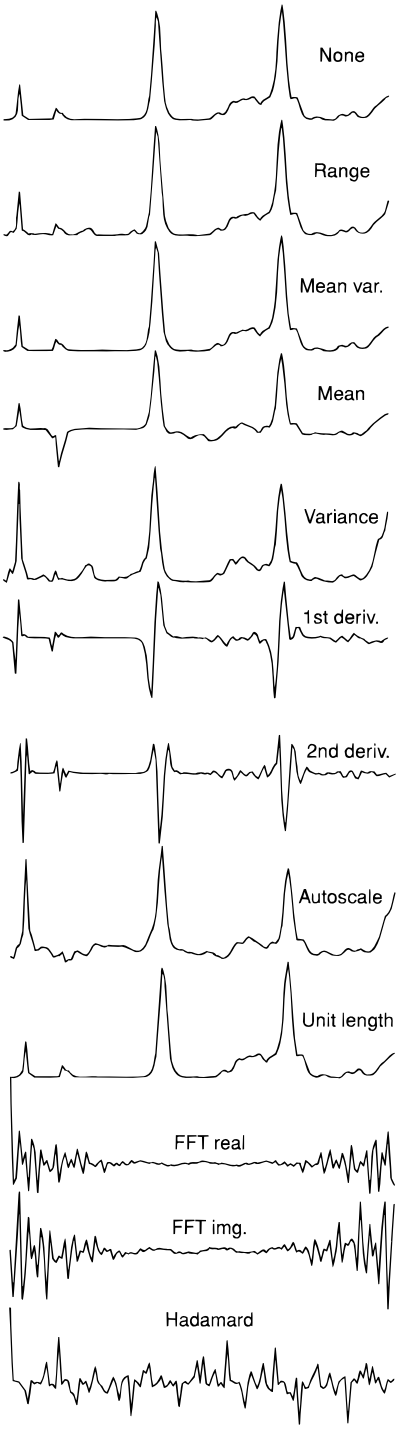

Figure 16. Effect of 11 different spectral preprocessing methods (Savitzky−Golay smoothing not shown) on the appearance of a 128-point spectrum of 2-hydroxypropanoic acid.

removing systematic baseline shifts and enlarging notoriously small bands. But it also has a negative impact on regions where there is usually no activity at all (2250−2600 cm^{-1}, >3700 cm^{-1}), promoting minor baseline humps to full-blown spectral bands.

$$x'_{mn} = \frac{x_{mn} - x_n^{\min}}{x_n^{\max} - x_n^{\min}} \qquad (2)$$

With normalization to mean centered variance (eq 3), each spectrum is centered around its own mean absorbance value,

divided by the variance of all absorbance values. This diminishes the influence of noisy spectra on the training progress.

$$x'_{mn} = \frac{x_{mn} - \bar{x}_m}{\sigma_m} \tag{3a}$$

$$\sigma_m = \sqrt{\frac{\sum_{i=1}^{M}(x_{mi} - \bar{x}_m)^2}{N-1}} \tag{3b}$$

Systematic baseline shifts or large absorbances for all spectra can be removed by mean centering each data point (eq 4), but it also reduces the significance of rare, but unambiguous bands such as the C≡N stretch around 2200 cm^{-1}.

$$x_{mn} = x_{mn} - \bar{x}_n \tag{4}$$

Variance scaling (eq 5) goes in the opposite direction, increasing the importance of these rare bands, which is especially important for neural network training of grossly underrepresented functional groups. In addition, it reduces the influence of areas with wildly fluctuating absorbance values, *e.g.*, the lower wavenumber region near the detector cut-off.

$$x'_{mn} = \frac{x_{mn}}{\sigma_n} \tag{5a}$$

$$\sigma_n = \sqrt{\frac{\sum_{i=1}^{M}(x_{in} - \bar{x}_n)^2}{M-1}} \tag{5b}$$

First and second derivatives remove the impact of sloping baselines and heighten the importance of very sharp peaks, which can have detrimental effects for sharp noise spikes. Autoscaling (eq 6) works similar to normalization to mean centered variance, but here the statistics considers a data point *n* throughout the spectral examples instead of a spectrum *m* by itself.

$$x'_{mn} = \frac{x_{mn} - \bar{x}_n}{\sigma_n} \tag{6}$$

In order to lessen the impact of peak-rich spectra in favor of sparse spectra in the neural network training process, normalization to unit length (eq 7) may be employed.

$$x'_{mn} = \frac{x_{mn}}{\sqrt{\sum_{i=1}^{N} x_{mi}^2}} \tag{7}$$

A five-point Savitzky-Golay smoothing function[68,69] was employed to iron out sharp spectral noise spikes and reduce initially present random noise prior to adding artificially produced noise. Fourier and Hadamard transformations have been added out of curiosity to study the influence of such drastic data manipulations on the prediction quality. Surprisingly, the achieved prediction rates for present functional groups (Figure 17) remain the same when using Hadamard coefficients and only drop slightly for real and imaginary Fourier coefficients with a significant increase in training speed for the imaginary coefficients. The fastest training coupled with the best prediction quality is achieved when data points are divided by σ_n, the variance of an input data point across all training spectra, as the results for autoscaling and variance scaling illustrate. Evidently, any help to signify rare bands and underrepresented functional groups for neural network training improves the overall recognition rate of present functional groups.

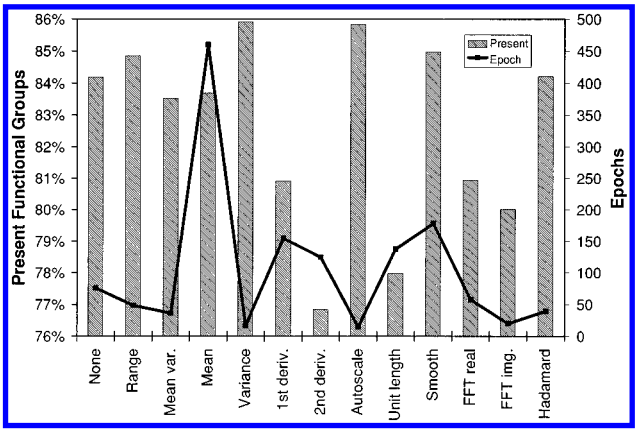

Figure 17. Impact of all 12 preprocessing methods on correct recognition of present substructures, including the required training effort for each method.

Table 3. Combination of Prediction Enhancement Methods: A = Increase Training Set to 2588 Spectra; B = Variance Scaling; C = Single Output Networks

method(s)	correct present[c] groups (%)	correct absent[c] groups (%)	quality figure present × absent[a]
regular	83.99 ± 0.09	92.47 ± 0.20	0.7767
A	89.77 ± 0.17	93.39 ± 0.36	0.8384
B	86.19 ± 0.32	90.95 ± 0.40	0.7839
C	91.65 ± 0.64	82.79 ± 2.50	0.7588
A + B	88.73 ± 0.24	93.92 ± 0.38	0.8334
A + C	95.43 ± 0.93	75.88 ± 4.59	0.7242
B + C	91.89 ± 0.98	88.49 ± 1.96	0.8131
A + B + C	92.88 ± 1.01	91.25 ± 3.18	0.8475
best results[b]	93.77%	95.69%	0.8973

[a] Multiplication of present × absent prediction accuracies as a measure of overall prediction quality. [b] Best individual functional group results from C, AC, BC, and ABC (see Table 4). [c] Data are averaged for five training runs each except for "best results".

OPTIMAL COMBINATIONS

A number of possibilities exist at this point to orchestrate a combined effort of the previously noted significant prediction improvements: (a) increase the number of neural network training examples to 2588 spectra; (b) employ variance scaling; and (c) use individual networks. All possible combinations of all three options (Table 3) reveal that utilizing increased training sets with specialized networks, but without variance scaling (methods A + C), yields the highest average correct prediction of functional groups: 95.4%. Unfortunately, the recognition quality for absent groups goes downhill to 75.9% at the same time, bringing the focus back to an *overall* improved prediction rate. The figure which best describes the neural network capability to

Table 4. Best Individual Functional Group Predictions Using Specialized Networks (See Also Caption for Table 3)[a]

functional group	methods A + C	methods A + B + C	methods B + C	method C only	Best present	best absent
O−H	0.9448	**0.9571**	0.9132	0.8499	96.7%	99.0%
N−H	**0.8527**	0.8492	0.8083	0.7652	91.9%	92.8%
C−N	0.8084	0.7810	**0.8310**	0.7875	90.0%	92.3%
X≡Y, X═Y═Z	0.8613	**0.8932**	0.7658	0.6681	90.9%	98.3%
C═O	0.9735	**0.9801**	0.9671	0.9623	99.0%	99.0%
C−O	0.9121	**0.9565**	0.8984	0.8666	97.8%	97.8%
C═C	**0.7528**	0.7508	0.6270	0.6194	83.6%	90.1%
C⋯C	0.9172	**0.9315**	0.8884	0.8873	98.5%	94.6%
halogen	**0.6923**	0.6869	0.6516	0.5876	75.0%	92.3%
N−O, N═O	**0.9730**	0.9071	0.9195	0.9180	97.7%	99.6%
CH_3, CH_2	**0.9671**	0.9043	0.8109	0.8751	97.8%	98.9%
primary alcohol	0.8888	**0.9085**	0.8553	0.7975	95.5%	95.2%
secondary alcohol	**0.8932**	0.8912	0.8493	0.7975	90.9%	98.3%
tertiary alcohol	0.7911	**0.8280**	0.7841	0.8057	84.1%	98.5%
Ar−OH	0.9781	0.9687	**0.9803**	0.7969	100.0%	98.0%
COOH	0.9257	**0.9693**	0.9671	0.8975	100.0%	96.9%
primary amine	0.8295	**0.8689**	0.8183	0.7322	88.6%	98.0%
secondary amine	**0.7428**	0.6937	0.7395	0.6593	81.8%	90.8%
tertiary amine	**0.7456**	0.6743	0.6699	0.6726	77.3%	96.5%
(CO)NH	**0.8752**	0.8106	0.8415	0.8295	90.9%	96.3%
(CO)OR	0.9665	**0.9912**	0.9644	0.9487	100.0%	99.1%
(CO)H	0.8709	**0.8896**	0.8875	0.8182	95.5%	93.2%
C(CO)C	0.7985	**0.8077**	0.7651	0.7370	84.1%	96.0%
aromatic C_6-ring	**0.8877**	0.8837	0.8260	0.8172	97.8%	90.7%
pyridine ring	0.6804	0.6874	**0.7270**	0.6615	77.3%	94.1%
C−Cl	0.6014	**0.6688**	0.6341	0.5698	72.0%	92.9%

[a] Figures in the first four columns are reported as correctly predicted present × absent groups, and an overall quality figure using the number in the last two columns appears in Table 3.

Table 5. Best Prediction Rates (%) Compared with Values From the Recent Literature, Where Applicable

	this work[a]		ref 15[b]		ref 50		ref 10[c]		ref 46	
functional group	(+)	(−)	(+)	(−)	(+)	(−)	(+)	(−)	(+)	(−)
O−H	96.7	99.0	89.2	94.2					91.1	96.5
N−H	91.9	92.8							84.8	96.4
C−N	90.0	92.3							73.1	90.3
X≡Y, X═Y═Z	90.9	98.3					88.6	99.1		
C═O	99.0	99.0					95.1	100.0	95.9	97.7
C−O	97.8	97.8					65.9	100.0		
C═C	83.6	90.1	58.5	88.5			50.0	92.9		
C⋯C	98.5	94.6					75.7	99.0		
halogen	75.0	92.3			57.0	91.0			65.8	87.1
N−O, N═O	97.7	99.6								
CH_3, CH_2	97.8	98.9								
primary alcohol	95.5	95.2	55.6	98.2					60.3	97.3
secondary alcohol	90.9	98.3							75.8	95.1
tertiary alcohol	84.1	98.5							62.5	99.9
Ar−OH	100.0	98.0							77.4	98.8
COOH	100.0	96.9	86.4	97.7	94.0	97.0	54.5	99.5	93.6	99.6
primary amine	88.6	98.0	58.3	99.1					83.2	98.8
secondary amine	81.8	90.8	65.0	99.1					71.4	96.5
tertiary amine	77.3	96.5							60.2	97.8
(CO)NH	90.9	96.3			64.0	99.0	65.7	99.6	51.0	99.4
(CO)OR	100.0	99.1	77.3	95.3	74.0	99.0	70.4	100.0	88.9	97.7
(CO)H	95.5	93.2	68.8	98.6	21.0	99.0	33.3	99.6	45.7	98.4
C(CO)C	84.1	96.0	64.0	96.2	59.0	97.0	42.9	100.0	75.4	96.2
aromatic C_6-ring	97.8	90.7	81.0	92.0					78.4	91.2
pyridine ring	77.3	94.1							49.1	98.3
C−Cl	72.0	92.9							51.7	88.9

[a] Correct prediction of present (+) and absent (−) groups (%). [b] Results for one hidden layer, single output nets. [c] Absent group prediction probably too optimistic, because no prediction evaluation was carried out for predictions between 0.01 and 0.5.

correctly separate a training and test set into present and absent groups was found[20] to be a multiplication of present and absent prediction rates, the last column in Table 3. Essentially it means relaxing the striving for the best possible present prediction a little bit in order to gain much more accuracy in the absent predictions. Table 4 lists the best individual predictions using specialized networks alone or in conjunction with any of the other two methods. The last two columns show the numbers that were originally used to calculate any of the boldface figures in Table 4, *e.g.*, for the O−H group, 0.9571 in column two results by multiplying 96.7% × 99.0%. The 26 best networks together gave an overall prediction accuracy of 93.8% present and 95.7% absent functionalities. To the authors' knowledge, these figures and most of the individual prediction rates are better than any previously reported functional group recognition

rates using infrared spectra and neural networks. The numbers in Table 5 compare the bottom line from this work with the results achieved by other researchers. Unfortunately, in many cases it was not possible to copy or compute numbers from the literature to allow fair comparison, because often a combined success rate for present and absent groups obscured the individual numbers, or results were presented in a completely different way. The figures from this work also compare favorably with reports from other authors,[70−72] using different methods than neural networks, but their results were not listed individually because they incorporated too few functional groups.

CONCLUSION

In a large-scale effort, many of the possible parameters influencing neural network training and prediction of substructures from gas phase infrared spectra were scrutinized for the maximum achievable prediction rate of present functional groups. Some of the results presented in this paper are not new, but the format of this study required reinvestigation with a unified target in mind. This was especially important because functional group predictions were carried out *during* training with a separate cross-validation set of 500 spectra not used in the training procedure. Monitoring the best prediction rates during training allowed the identification of the network state with the highest generalization capabilities and improved the prediction accuracy for present functional groups from 79.0% (after minimization of the RMS error) to 84.0% (saving the best generalization state). Furthermore, increasing the number of training spectra 5-fold resulted in another jump of this figure to 89.8%, and individual networks instead of a multioutput network gained an even higher increase from 84.0% to 91.7%. From the results of spectral preprocessing functions, only autoscaling and variance scaling produced significantly higher prediction rates by about 2%. Manipulation of all other neural network parameters including learning rate, momentum, sigmoidal discrimination and bias, and the number of input and hidden nodes or training noise resulted in no improvements whatsoever, and all of these parameters can be varied within a broad comfortable region without impact on the network generalization capabilities. Combining the most promising approaches, a 95.4% correct overall prediction for presence of functional groups probably comes near the limit of what can be achieved with neural networks, but improving only present predictions left the absent ones behind at 75.9%. A compromise between the two using the best individual network figures from all method combinations yielded an overall recognition accuracy of 93.8% for present and 95.7% for absent functional groups, the best general-purpose neural network results for infrared spectra reported so far in the literature.

Only a few items have been omitted from this study for future research: Using individual networks, it should be possible to produce a balanced training set with equal abundance of present and absent groups for each of the 26 functionalities, leaving room for investigating the needed quantity of present group occurrences for each substructure. Another study in the multidimensional optimization space would lead to comparison of different types of infrared spectra, answering the question whether matrix isolation, gas phase, or condensed phase infrared spectra are better suited for neural network predictions.

ACKNOWLEDGMENT

This work has been carried out with support from the National Science Foundation, Grant CHE-92-01277. Preliminary findings were presented at the PittCon'95 Conference in New Orleans, LA, March 1995, paper 712. The authors are indebted to Stephen Stein at the National Institute of Standards and Technology for providing the IR gas phase spectral library and to the Computing & Communications department at the University of California, Riverside for a grant of 1000 CPU h on three Digital AXP 21064A CPUs (275 MHz). We would also like to thank the SISAL computing group at the Lawrence Livermore National Laboratory for ∼30 h of Cray C90 CPU time.

REFERENCES AND NOTES

(1) Minsky, M. L.; Papert, S. *Perceptrons: an Introduction to Computational Geometry*; MIT Press: Cambridge, MA, 1969.
(2) Kowalski, B. R.; Jurs, P. C.; Isenhour, T. L.; Reilly, C. N. Computerized learning machines applied to chemical problems. Interpretation of infrared spectrometry data. *Anal. Chem.* **1969**, *41*, 1945−1953.
(3) *Artificial Intelligence Applications in Chemistry*; Pierce, T. H., Hohne, B. A., Eds.; ACS Symposium Series 306; American Chemical Society: Washington, DC, 1986.
(4) Zupan, J.; Gasteiger, J. Neural networks: A new method for solving chemical problems or just a passing phase? *Anal. Chim. Acta* **1991**, *248*, 1−30.
(5) Gasteiger, J.; Zupan, J. Neural networks in chemistry. *Angew. Chem., Int. Ed. Engl.* **1993**, *32*, 503−527.
(6) Tušar, M.; Zupan, J.; Gasteiger, J. Neural networks and modelling in chemistry. *J. Chim. Phys.* **1992**, *89*, 1517−1529.
(7) Tušar, M.; Zupan, J. Neural networks. *Proceedings of the Workshop "Computers in Chemistry";* Software Development in Chemistry 4; Springer-Verlag: Berlin, 1990; p 363−376.
(8) Kateman, G.; Smits, J. R. M. Colored information from a black box? Validation and evaluation of neural networks. *Anal. Chim. Acta* **1993**, *277*, 179−188.
(9) Jansson, P. A. Neural networks: An overview. *Anal. Chem.* **1991**, *63*, 357A−362A.
(10) Robb, E. W.; Munk, M. E. A neural network approach to infrared spectrum interpretation. *Mikrochim. Acta [Wien]* **1990**, *I*, 131−155.
(11) Munk, M. E.; Madison, M. S.; Robb, E. W. Neural network models for infrared spectrum interpretation. *Mikrochim. Acta [Wien]* **1991**, *II*, 505−514.
(12) Fessenden, R. J.; Györgyi, L. Identifying functional groups in IR spectra using an artificial neural network. *J. Chem. Soc., Perkin Trans. 2* **1991**, 1755−1762.
(13) Weigel, U.-M.; Herges, R. Automatic interpretation of infrared spectra: Recognition of aromatic substitution patterns using neural networks. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 723−731.
(14) Meyer, M.; Weigelt, T. Interpretation of infrared spectra by artificial neural networks. *Anal. Chim. Acta* **1992**, *265*, 183−190.
(15) Ricard, D.; Cachet, C.; Cabrol-Bass, D. Neural network approach to structure feature recognition from infrared spectra. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 202−210.
(16) Rumelhart, D. E.; McClelland, J. L. *Parallel Distributed Processing−Explorations in the Microstructure of Cognition.* The MIT Press: Cambridge, MA, 1986; Vol. 1, pp 318−362.
(17) Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533−536.
(18) Klawun, C.; Wilkins, C. L. A novel algorithm for local minimum escape in back-propagation neural networks: Application to the interpretation of matrix isolation infrared spectra. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 984−993.
(19) Klawun, C.; Wilkins, C. L. Neural network assisted rapid screening of large infrared spectral databases. *Anal. Chem.* **1995**, *67*, 374−378.
(20) Klawun, C.; Wilkins, C. L. Joint neural network interpretation of infrared and mass spectra. *J. Chem. Inf. Comput. Sci.* In press.
(21) Ajay: On better generalization by combining two or more models: a quantitative structure-activity relationship example using neural networks. *Chemom. Intell. Lab. Syst.* **1994**, *24*, 19−30.
(22) Elrod, D. W.; Maggiora, G.; Trenary, R. G. Applications of neural networks in chemistry. 1. Prediction of electrophilic aromatic substitution reactions. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 477−484.

(23) Aoyama, T.; Ichikawa, H. Reconstruction of weight matrices in neural networks—A method of correlating outputs with inputs. *Chem. Pharm. Bull.* **1991**, *39*, 1222−1228.
(24) Wythoff, B. J.; Levine, S. P.; Tomellini, S. A. Spectral peak verification and recognition using a multilayered neural network. *Anal. Chem.* **1990**, *62*, 2702−2709.
(25) Blank, T. B.; Brown, S. D. Data processing using neural networks. *Anal. Chim. Acta* **1993**, *277*, 273−287.
(26) Luinge, H. J.; Van der Maas, J. H.; Visser, T. Application of a neural network to the identification of pesticides from their infrared spectra. *Proc. SPIE Int. Soc. Opt. Eng.* **1991**, *1575*, 499−500.
(27) Wong, K. Y. M.; Sherrington, D. Neural networks optimally trained with noisy data. *Phys. Rev. E* **1993**, *47*, 4465−4482.
(28) Harrington, P. B. Temperature-constrained backpropagation neural networks. *Anal. Chem.* **1994**, *66*, 802−807.
(29) Hornik, K. Some new results on neural network approximation. *Neural Netw.* **1993**, *6*, 1069−1072.
(30) Jacobsson, S. P. Feature extraction of polysaccharides by low-dimensional internal representation neural networks and infrared spectroscopy. *Anal. Chim. Acta* **1994**, *291*, 19−27.
(31) Harrington, P. B. Sigmoid transfer functions in backpropagation neural networks. *Anal. Chem.* **1993**, *65*, 2167−2168.
(32) Kung, S. Y.; Hwang, J. N. An algebraic projection analysis for optimal hidden units size and learning rates in back-propagation learning. *IEEE International Conference on Neural Networks;* IEEE San Diego Section, IEEE Technical Activities Board Neural Network Committee, 1988; pp 363−370.
(33) Hush, D. R.; Salas, J. M. Improving the learning rate of back-propagation with the gradient reuse algorithm. *IEEE International Conference on Neural Networks;* IEEE San Diego Section, IEEE Technical Activities Board Neural Network Committee, 1988; pp 441−447.
(34) Huang, S.-C.; Huang, Y.-F. Bounds on the number of hidden neurons in multilayer perceptrons. *IEEE Trans. Neural Networks* **1991**, *2*, 47−55.
(35) Sartori, M.; Antsaklis, P. J. A simple method to derive bounds on the size and to train multilayer neural networks. *IEEE Trans. Neural Networks* **1991**, *2*, 467−471.
(36) Liu, Y.; Upadhyaya, B. R.; Naghedolfeizi, M. Chemometric data analysis using artificial neural networks. *Appl. Spectrosc.* **1993**, *47*, 12−23.
(37) Mittermayr, C. R.; Drouen, A. C. J. H.; Otto, M.; Grasserbauer, M. Neural networks for library search of ultraviolet spectra. *Anal. Chim. Acta* **1994**, *294*, 227−242.
(38) Curry, B.; Rumelhart, D. E. MSnet: A neural network which classifies mass spectra. *Tetrahedron Comput. Methodol.* **1990**, *3*, 213−237.
(39) Setiono, R.; Hui, L. C. K. Use of a quasi-Newton method in a feedforward neural network construction algorithm. *IEEE Trans. Neural Networks* **1995**, *6*, 273−277.
(40) Sharpe, R. N.; Chow, M.; Briggs, S.; Windingland, L. A methodology using fuzzy logic to optimize feedforward artificial neural network configurations. *IEEE Trans. Syst. Man Cybern.* **1994**, *24*, 760−768.
(41) Goodacre, R.; Neal, M. J.; Kell, D. B. Rapid and quantitative analysis of the pyrolysis mass spectra of complex binary and tertiary mixtures using multivariate calibration and artificial neural networks. *Anal. Chem.* **1994**, *66*, 1070−1085.
(42) Van Est, Q. C.; Schoenmakers, P. J.; Smits, J. R. M.; Nijssen, W. P. M. Practical implementation of neural networks for the interpretation of infrared spectra. *Vib. Spectrosc.* **1993**, *4*, 263−272.
(43) Smits, J. R. M.; Schoenmakers, P.; Stehmann, A.; Sijstermans, F.; Kateman, G. Interpretation of infrared spectra with modular neural-network systems. *Chemom. Intell. Lab. Syst.* **1993**, *18*, 27−39.
(44) Baxt, W. G. Improving the accuracy of an artificial neural network using multiple differentially trained networks. *Neural Comput.* **1992**, *4*, 772−780.
(45) Alam, M. K.; Stanton, S. L.; Hebner, G. A. Near-infrared spectroscopy and neural networks for resin identification. *Spectroscopy (Springf., Or.)* **1994**, *9(2)*, 30−40.
(46) Novič, M.; Zupan, J. Investigation of infrared spectra−structure correlation using Kohonen and counterpropagation neural networks. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 454−466.
(47) Affolter, C.; Clerc, J. T. Estimation of the performance of spectroscopic library search systems. *Fresenius J. Anal. Chem.* **1992**, *344*, 136−139.
(48) Bos, M.; Vrielink, J. A. M. The wavelet transform for pre-processing IR spectra in the identification of mono- and di-substituted benzenes. *Chemom. Intell. Lab. Syst.* **1994**, *23*, 115−122.
(49) Visser, T.; Luinge, H. J.; Van der Maas, J. H. Recognition of visual characteristics of infrared spectra by artificial neural networks and partial least squares regression. *Anal. Chim. Acta* **1994**, *296*, 141−154.
(50) Meyer, M.; Meyer, K.; Hobert, H. Neural networks for interpretation of infrared spectra using extremely reduced spectral data. *Anal. Chim. Acta* **1993**, *282*, 407−415.
(51) Gemperline, P. J.; Long, J. R.; Gregoriou, V. G. Nonlinear multivariate calibration using principal components regression and artificial neural networks. *Anal. Chem.* **1991**, *63*, 2313−2323.
(52) Kaiser, H. F. The application of electronic computers to factor analysis. *Educ. Psychol. Meas.* **1960**, *20*, 141−151.
(53) Malinowski, E. R. Determination of the number of factors and the experimental error in a data matrix. *Anal. Chem.* **1977**, *49*, 612−617.
(54) Foody, G. M. Using prior knowledge in artificial neural network classification with a minimal training set. *Int. J. Remote Sens.* **1995**, *16*, 301−312.
(55) Blank, T. B.; Brown, S. D. Nonlinear multivariate mapping of chemical data using feed-forward neural networks. *Anal. Chem.* **1993**, *65*, 3081−3089.
(56) Blaffert, T. Computer-assisted multicomponent spectral analysis with fuzzy data sets. *Anal. Chim. Acta* **1984**, *161*, 135−148.
(57) Zupan, J. *Algorithms for Chemists*; John Wiley & Sons: Chichester, 1989.
(58) Boas, M. L. *Mathematical Methods in the Physical Sciences*, 2nd ed.; John Wiley & Sons: New York, 1983.
(59) Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P. *Numerical Recipes in C*, 2nd ed.; Cambridge University Press: Cambridge, 1992 (corrected reprint 1994).
(60) Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P. *Numerical Recipes in C* (3.5" Diskette for IBM PC, version 2.02); Cambridge University Press: Cambridge, 1992.
(61) Smith, B. T.; Boyle, J. M.; Dongarra, J. J.; Garbow, B. S.; Ikebe, Y.; Klema, V. C.; Moler, C. B. *Matrix Eigensystem Routines—EISPACK Guide*, 2nd ed.; Springer-Verlag: Berlin, 1976.
(62) Garbow, B. S.; Boyle, J. M.; Dongarra, J. J.; Moler, C. B. *Matrix Eigensystem Routines—EISPACK Guide Extension*; Springer-Verlag: Berlin, 1977.
(63) Wilkinson, J. H.; Reinsch, C. *Handbook for Automatic Computation*; Springer-Verlag: New York, 1971; Vol. II.
(64) McGraw, J.; Skedzielewski, S.; Allan, S.; Oldehoeft, R.; Glauert, J.; Kirkham, C.; Noyce, B.; Thomas, R. *SISAL: Streams and Iterations in a Single Assignment Language. Language Reference Manual Version 1.2*; Lawrence Livermore National Laboratory: Livermore, CA, 1985.
(65) Cann, D. C. *SISAL 1.2: A Brief Introduction and Tutorial*; Lawrence Livermore National Laboratory: Livermore, CA, 1992.
(66) Cann, D. C. *The Optimizing SISAL Compiler: Version 12.0*; Lawrence Livermore National Laboratory: Livermore, CA, 1992.
(67) *Proceedings of the Second Sisal Users' Conference*; Feo, J. T.; Frerking, C.; Miller, P. J., Eds.; Lawrence Livermore National Laboratory: Livermore, CA, 1992.
(68) Savitzky, A.; Golay, M. J. E. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.* **1964**, *36*, 1627−1639.
(69) Enke, C. G.; Nieman, T. A. Signal-to-noise ratio enhancement by least-squares polynomial smoothing. *Anal. Chem.* **1976**, *48*, 705A−712A.
(70) Perkins, J. H.; Hasenoehrl, E. J.; Griffiths, P. R. The use of principal component analysis for the structural interpretation of mid-infrared spectra. *Chemom. Intell. Lab. Syst.* **1992**, *15*, 75−86.
(71) Hasenoehrl, E. J.; Perkins, J. H.; Griffiths, P. R. Rapid functional group characterization of gas chromatography/Fourier transform infrared spectra by a principal component analysis based expert system. *Anal. Chem.* **1992**, *64*, 705−710.
(72) Emmence, R. S.; Parker, M. E.; Smith, M. J. C.; Steele, D. Vibrational absorption intensities in chemical analysis. Part 7. On the extraction of group spectra. *J. Mol. Struct.* **1993**, *292*, 295−312.

CI950102M