Combining Genetic Algorithms, Neural Networks and Wavelet Transforms for Analysis of Raman Spectra

Kenneth Hennessy, Michael G. Madden and Alan G. Ryder

hennessy@vega.it.nuigalway.ie, michael.madden@nuigalway.ie
alan.ryder@nuigalway.ie
National University of Ireland, Galway, Ireland

Abstract. The purpose of this research is to develop Machine Learning techniques for quantitative analysis of illicit narcotics from mixtures based on their Raman spectra, with the aim of improving on traditional chemometric quantification techniques. In this paper, we investigate the use of wavelet transforms for data and noise reduction, in conjunction with neural networks designed using genetic algorithms, for the prediction of cocaine concentration in mixtures using their Raman spectral data.

1 Introduction

Raman spectroscopy is the measurement of the intensity and wavelength of inelastically scattered light from molecules when they are excited by a monochromatic light source. Its versatility, due to ease of sampling via coupling to fibre optics and microscopes [1], allied to the ability to sample through glass and plastics [2], has made it a very practical technique for use by law enforcement agencies in the detection of illicit materials [3,4].

The presence of diluents in a drug mixture pose a problem for Raman spectroscopy in real-world situations as they may interfere with the correct identification and quantification of target materials by masking important peaks [4]. Fluorescent compounds, poor spectral reproducibility of spectral intensities and poor signal/noise ratios also have a detrimental effect on Raman spectra elucidation [2]. In order to develop methods to overcome these problems, noise reduction and Machine Learning (ML) techniques are being investigated.

The objective of this study is to develop accurate techniques for automatically estimating the concentration of cocaine in a sample containing diluents. In order to achieve this objective two main factors need to be addressed. The first is the noise inherent in spectroscopic data. Noise is a consequence of environmental and instrumental conditions and its removal from the data is imperative for accurate prediction. The second factor is a significant challenge to the ML community and lies in the high dimensionality and low sample number commonly found in this domain. Poor generalisation and overfitting to the training data are a consequence of this second problem. Accordingly, the first aim of this work is to assess the effect of simultaneous dimensionality and noise reduction of spectral data using a wavelet transform on the predictive ability of various machine learning methods. Wavelet transforms are used for noise reduction and compression in signals and for image compression. The specific transforms applied in this paper are the Daubechies D4 transform [9] and the Haar transform [9].

In spectroscopy, the most popular techniques for regression are chemometric methods (see Section 3.4) and feed-forward neural networks. Feed forward neural networks are proven tools for quantification and classification of spectroscopy data; however, they are often difficult to optimise [6,7], and small changes in the configuration of a neural network can result in dramatic changes in its predictive power [12]. Therefore, the second aim of this work is to assess the use of a genetic algorithm to evolve an optimal neural network configuration for cocaine prediction. This approach, however, is conditional on spectral data dimensionality reduction. High spectral dimensionality would render a combined neural network/genetic algorithm approach impractical due to long neural network training times.

Raman spectra were collected from 36 solid samples comprising different mixtures of cocaine (concentration varied between 0 and 100%), caffeine and glucose. Clearly, this is a very small training dataset. However, as discussed above, in many real-world applications, it is required to identify materials based on a small number of reference spectra. Caffeine and glucose are typical examples of diluents that could be mixed with illegal narcotics and make the quantification of cocaine from Raman spectra more difficult. Details of the data collection, preprocessing and the samples are available in previous publications [3, 4].

2 Related Research

This paper extends previous work by Madden & Ryder [4], which analysed the same dataset, who used a genetic algorithm for data reduction, wrapped around neural network and kNN predictors. That study is compared to this work in Section 4.4. The wavelet transform has been used to reduce noise [9] and compress data [10] in areas such as image recognition by neural networks [11] and damage detection of composite structures using neural networks [12]. Wavelet transforms have been used in spectroscopic applications also for noise reduction and the reduction of dimensionality. Trygg & Wold [5] successfully used the Daubechies D4 wavelet transform to pre-process near-infrared spectra prior to using Partial Least Squares (PLS) for regression. Shao *et al.* [13] showed that a wavelet transform could greatly decrease the time needed for resolution of nuclear magnetic resonance spectra using the immune algorithm.

The wavelet transform has been used in pattern recognition to extract sharp features from an image, with Osowski & Nghia [11] showing that wavelet-transformed data in conjunction with neural networks perform well for shape recognition. A similar amalgamation of wavelet transforms and neural networks has been used for the interpretation of infrared spectra by Tchistiakov *et al.* [14]. They successfully used a wavelet transform to reduce the dimensionality of their infrared spectra and increase the efficiency of their neural networks. They determined the neural network configuration manually, by varying the size of the hidden layer from 3 to 20 nodes, and using combinations of many different neural networks for hierarchical modelling.

Very little work has been published on the use of wavelet transforms in conjunction with neural networks and Raman spectra. Estienne & Massart [15] compared a wavelet transform and a Fourier transform for noise reduction before using Principal Component Regression with variable selection and Partial Least Squares for prediction. Interestingly, while they preferred the Fourier transform to the wavelet transform for noise reduction, they found that the performance of models based on transformed data were equivalent or worse than those based on the raw spectral data.

The use of genetic algorithms to optimise the number of inputs, number of hidden nodes and the number of training epochs for neural networks has been used for the calibration of voltammetric data by Richards *et al.* [6]. The G-Prop algorithm of Castillo et al. [7] for the optimisation of weights and structure of neural networks is also related to the method used in this paper to configure our neural networks.

3 Machine Learning Analysis

3.1 Data Reduction

There were two different wavelet transforms compared in this study, the Haar transform and Daubechies D4 transform. The Daubechies D4 requires that the starting number of inputs is a power of 2, so the 510 spectral data points was padded with zeros to make 512 data points. The Haar transform is a simple transform which has two scaling function coefficients describing a low-pass filter, listed in Eq. (1), and a scaling function given by Eq. (2).

$$(c(0), c(1)) = (\frac{1}{2}, \frac{1}{2}), \quad Z_i = c_0 S_i + c_1 S_{i-1} \qquad (1), (2)$$

The Daubechies D4 transform has four scaling function coefficients, given in Eq. (3), and the scaling function described by Eq. (4).

$$(c(0), c(1), c(2), c(3)) = (\frac{1+\sqrt{3}}{4\sqrt{2}}, \frac{3+\sqrt{3}}{4\sqrt{2}}, \frac{3-\sqrt{3}}{4\sqrt{2}}, \frac{1-\sqrt{3}}{4\sqrt{2}}) \qquad (3)$$

$$Z_i = c_0 S_{2i} + c_1 S_{2i+1} + c_2 S_{2i+2} + c_3 S_{2i+3} \qquad (4)$$

Each step of the wavelet transform applies the low pass filter to the data, effectively compressing the data by a half. Figure 1 illustrates successive compressions using the Haar transform and Daubechies D4 transform on a Raman spectrum from pure cocaine. Clearly, by the time 6 successive transforms have been applied, reducing the data to 8 points, most detail has been lost. Note that it is necessary to use different intensity scales for graphing the two transforms in Figure 1, because through successive compressions the Haar transformed data decreases in intensity while the Daubechies D4 transformed spectrum increases in intensity.

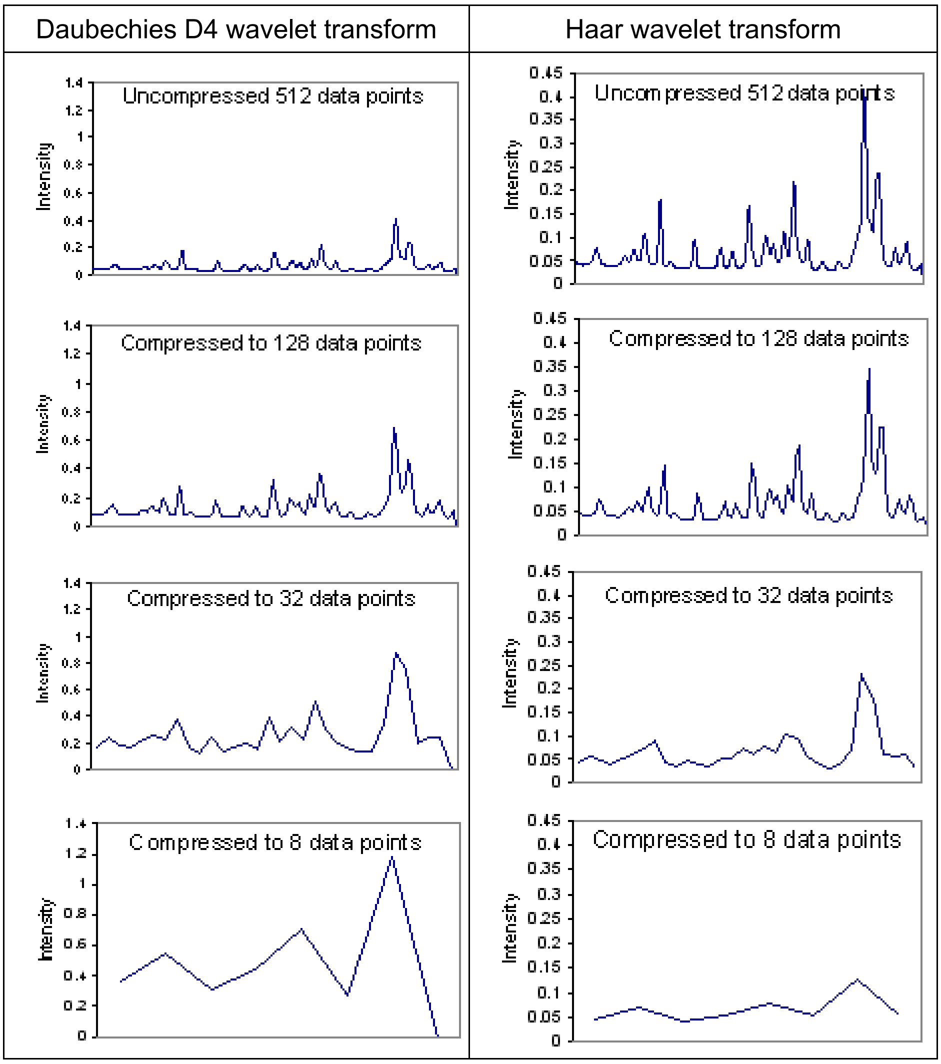

Figure 1 Successive compressions of 100% cocaine Raman spectrum

3.2 Neural Network Analyses

In this work, a conventional feed-forward neural network structure with one hidden layer was used. Our implementation is based on one by Brierley [17]. The inputs corresponded to Raman spectrum data, with transformations applied as discussed below, and the single output corresponded to a prediction of cocaine concentration.

The activation function used in this work is the *tanh* activation function and networks were trained using the standard backpropagation with momentum algorithm.

3.3 Genetic Algorithm for Evolution of Neural Network Configuration

In this study, each individual in the population represents a possible configuration for a neural network. Each individual has four components describing different aspects of the configuration: number of hidden nodes; learning rate from input to hidden layer; learning rate from hidden to output layer; and momentum term. A real valued representation [18] was used to describe these components. Each component was assigned maximum and minimum values, based on previous experiments and experience with the dataset; these are detailed in Table 1. To calculate the fitness of a member of the population, a neural network was built based on the information encoded within that individual. This was then trained for 300 epochs and tested on the wavelet-compressed data. The root mean squared error in prediction (RMSEP), calculated using leave-one-out cross-validation, was used as the fitness measure for each individual.

Table 1 Settings for components in genetic algorithm

Neural Network Component	Minimum	Maximum
Number of Hidden Nodes	1	30
Input–Hidden Learning Rate	0.001	0.181
Hidden–Output Learning Rate	0.001	0.181
Momentum	0.001	0.011

For this work, a parallel island model was used [18], where each island had a fixed population size of 50 and 19 islands were used. The parallel island model was used so that analysis could be distributed across multiple computers but is not strictly necessary for the working of the technique employed here. Our breeding strategy involves elitism, crossover, mutation and migration as follows.

- Elitism: The top two fittest individuals from each population are copied without mutation into next generation, to ensure a steady progression in fitness.
- Crossover: The preceding population was sorted by descending fitness and two randomly-selected individuals from the top 10 are crossed over. This was repeated until a new population was produced.
- Mutation: The mutation rate was set at 10%, i.e. one mutation in every 10 individuals. This high level of mutation is used to counter-balance the combined effects of using elitism and restricting crossovers to component boundaries.
- Migration: The migration rate between islands was set at 5%, i.e. every generation there was a 5% probability of incorporating the best individual from a randomly chosen island population (A) into a population (B) overwriting the worst individual in population (B).

The initial weights in the neural networks were set randomly. They were initially set using the Nguyen and Widrow algorithm [19]; however, this was not found to improve the predictive power of the network or decrease the training time.

3.4 Chemometric Methods

Partial Least Squares (PLS) regression [5,16] and Principal Component Regression (PCR) [15] were carried out on the data using Unscrambler, a software package from CAMO AS, Norway. Both PCR and PLS regression extract successive linear combinations of the predictors, called factors, with the intention that there is no correlation between the factor score variables used in the predictive regression model. The techniques differ, however, in the methods used in extracting factor scores. PCR regression produces factors reflecting the covariance structure between the predictor variables, while PLS regression produces factors reflecting the covariance structure between the predictor and response variables [20].

4 Experiments and Results

Two sets of experiments were performed:

1. An evaluation of the impact of the wavelet transforms, to compare the Haar and Daubechies D4 transforms and gauge their value for data reduction when used with various ML algorithms for quantification of Raman spectra.

2. An evaluation of the effectiveness of automatically optimising the neural network configurations, using a genetic algorithm approach. This set of experiments used the transform that had been found to be most promising in the first set of experiments.

These experiments are described in the following sub-sections.

4.1 Evaluation of Wavelet Transforms

In the first group of experiments, the effect of successive compressions of the spectral data using the wavelet transforms (see Section 3.1) on the predictive performance of various ML techniques is investigated. The algorithms used are MP5 trees (MP5), Support Vector Machines (SVM), k-Nearest Neighbour (kNN), Decision Table (DT), M5 Rules (M5R) and Neural Network (NN). Each of these algorithms was implemented using the Weka ML software [21] except for the neural network, see Section 3.2.

Figures 2 and 3 show results obtained using the Daubechies D4 transform and the Haar transform in conjunction with the six different ML techniques listed above. In the graphs, the y-axis measures the RMSEP of the ML techniques using leave-one-out cross-validation.

The figures demonstrate that compression of the data does not significantly increase and in most cases decrease the error of prediction in ML methods. The exception to this is kNN, whose error remains the same through one compression and then increases through successive compressions.

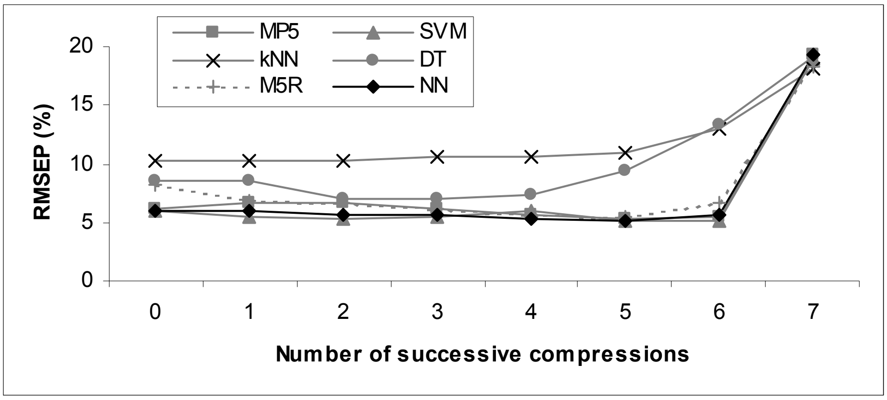

Figure 2 Effect of successive compressions using Daubechies D4 on RMSEP

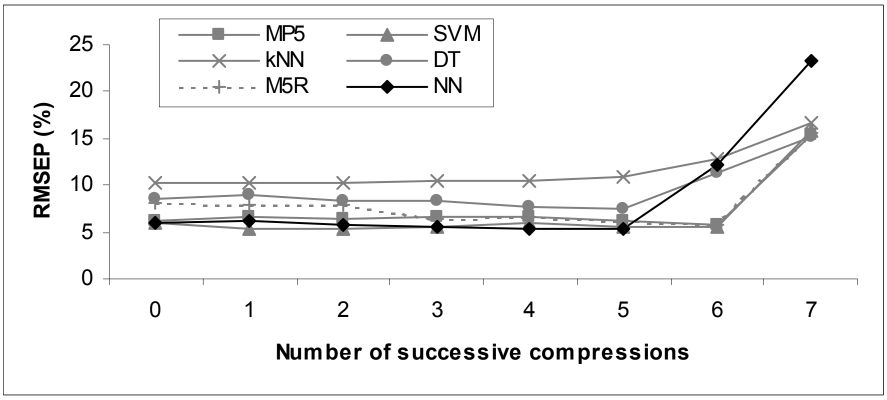

Figure 3 Effect of successive compressions using Haar on RMSEP

In order to determine which of the transforms to use in the second set of experiments the effect of both transforms on neural network prediction was compared. It can be seen from Figures 2 and 3 that both transforms in conjunction with neural networks perform quite well with similar accuracy; however the Daubechies D4 transform keeps its predictive power down to a compression level of 8 data points (6 successive compressions), and exhibits less variability relative to the network structure at 16 data points (5 successive compressions).

4.2 Results of Neural Network Evolution

The second set of experiments focussed on optimising neural network configurations automatically using the genetic algorithm approach described earlier in Section 3.3. The Daubechies D4 transform was chosen as the compression algorithm for these experiments, and spectra compressed to 16 and to 32 data points were both evaluated. (Because of the time-consuming nature of the genetic algorithm, spectra with less compression were not evaluated in this part of the study.) Figure 4 plots the RMSEP for the fittest neural network in a given generation, against the number of generations. It is interesting to note in Figure 4 that neural networks using the data compressed 5 successive times (16 points) are slightly less amenable to optimisation than those using the data compressed 4 times (32 points).

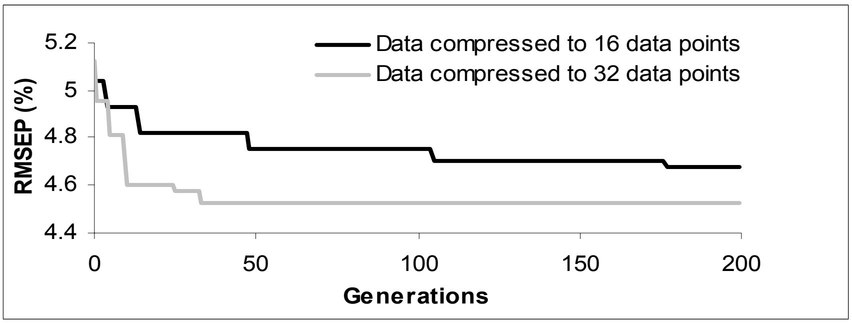

Figure 4 Fitness of best individual on island number 16

The configurations for the neural network with 16 and 32 inputs chosen by the genetic algorithm are detailed in Table 2.

Table 2 Optimal neural network configurations found using genetic algorithm

Structural Element	**32 Data Points**	**16 Data Points**
Number of Hidden Nodes	3	13
Learning Rate (Input-Hidden)	0.0015	0.0011
Learning Rate (Hidden-Output)	0.0897	0.1272
Momentum Term	0.0055	0.0079

4.3 Comparison with Chemometric Methods

The optimised neural network using wavelet-compressed data performed well relative to both of the traditional chemometric methods, PCR and PLS regression. The chemometric methods were both evaluated relative to the data using leave-one-out cross-validation, yielding the results listed in Table 3. The performance of the best evolved neural network (corresponding to the lower line in Figure 4) is also listed for comparison. It can be seen from Table 3 that a neural network using the compressed data with configuration evolved using a genetic algorithm can greatly improve upon traditional statistical methods for the prediction of cocaine concentration from Raman spectra.

Table 3 Results of prediction methods using leave-one-out cross validation

Method	**RMSEP (%)**
PLS Regression using Uncompressed Data	5.27
PCR using Uncompressed data	5.21
Evolved Neural Network using Wavelet-Reduced Data	4.55

4.4 Comparison with Previous Results

Two of the authors of this paper have previously studied the quantitative analysis of Raman spectra using the same data set as this study [10]. They compared traditional chemometric methods to the k-Nearest Neighbour algorithm and neural networks for the prediction of cocaine concentration. They used a genetic algorithm to choose points for input to the neural network in a wrapper approach to feature selection. They also used all their prediction methods in an ensemble, which produced better results than any of the individual predictors. Their main results are summarised in Table 4. The results below represent the best run of each algorithm.

Table 4 Results of Madden and Ryder [4] for the same data set

Method	**RMSEP (%)**
Neural Network	5.21
k-Nearest Neighbour	5.82
Ensemble	4.86

Comparing with Table 3 with Table 4, it would appear that the approach used in this study has produced better results.

5 Conclusions

Data reduction using successive applications of either the Haar transform or the Daubechies D4 transform does not weaken predictive performance until the number of remaining data points becomes very small, with the Daubechies D4 transform in particular showing good performance down to a compression level of 8 data points. In the case of many ML techniques the predictive performance on Raman spectral data is actually strengthened through the use of a wavelet transform. We believe this increase is due to noise and dimensionality reduction and would have applications in other spectroscopic and signal analysis areas.

In the second set of experiments, the resulting networks with optimised configuration, using wavelet transformed data, outperformed PLS and PCR algorithms for the prediction of cocaine concentration from Raman spectral data, and also outperformed previous ML approaches to analysing the same data.

Accordingly, it is concluded that the approach proposed here is an effective technique for use in Raman spectral prediction.

Acknowledgements

This research is supported by a grant from Enterprise Ireland's Commercialisation Fund Technology Development Programme, TD/03/212.

References

1. Harvey S.D., Vucelick M.E., Lee R.N., Wright B.W.: Blind field test evaluation of Raman spectroscopy as a forensic tool. Forensic Sci. Int. 125 (2002) 12–21.
2. Bulkin B.J.: The Raman effect: an introduction. In: Analytical Raman Spectroscopy. Chemical Analysis volume 114. New York: John Wiley and Sons, Inc, (1991) 1-19.
3. Ryder A.G., O'Connor G.M. and Glynn T.J.: Quantitative analysis of cocaine in solid mixtures using Raman spectroscopy and chemometric methods. J. Raman. Spectrosc. 31, (2000) 221-227.
4. Madden M. G., Ryder A. G.: Machine learning methods for quantitative analysis of Raman Spectroscopy. Data Proceedings of SPIE, the International Society for Optical Engineering, Vol. 4876, (2002) 1130-1139.
5. Trygg J., Wold S.: PLS regression on wavelet compressed NIR spectra. Chemometr. Intell. Lab. 42, (1998) 209-220.
6. Richards E., Bessant C., Saini S.: Optimisation of a neural network model for calibration of voltammetric data. Chemometr Intell. Lab. 61 (2002) 35– 49.
7. Castillo P.A., Merelo J.J., Prieto A., Rivas V. and Romero G.: G-Prop: Global optimization of multilayer perceptrons using Gas. Neurocomputing 35 (2000) 149-163.
8. Sexton R.S., Dorsey R.E., Johnson J.D.: Optimization of neural networks: A comparative analysis of genetic algorithm and simulated annealing. Eur. J. Oper. Res. 114 (1999) 589-60.
9. Bakhtazad A., Palazoglu A. and Romagnoli J.A.: Process data de-noising using wavelet transform. Intelligent Data Analysis. Vol. 3 Issue 4, (1999) 267-285.
10. Yoon J., Lee B., Han C.: Calibration transfer of near-infrared spectra based on compression of wavelet coefficients. Chemometr. Intell. Lab. 64, (2002) 1-14.
11. Osowski S., Do Nghia D.: Fourier and wavelet descriptors for shape recognition using neural networks—a comparative study. Pattern Recogn. 35 (2002) 1949–1957.
12. Yam L.H., Yan Y.J., Jiang J.S.: Vibration-based damage detection for composite structures using wavelet transform and neural network identification. Compos. Struct. 60 2003 403–412.
13. Shao X., Yu Z., Sun L.: Resolution of multicomponent NMR signals using wavelet compression and immune algorithm. Spectrochim. Acta. Part A 59, (2003) 1075-1082.
14. Tchistiakov V., Ruckebusch C., Duponchel L., Huvenne J-Pierre, Legrand P.: Neural network modelling for very small spectral data sets: reduction of the spectra and hierarchical approach. Chemometr. Intell. Lab. 54 (2000) 93–106.
15. Estienne F., Massart D.L.: Multivariate calibration with Raman data using fast Principal component regression and partial least squares methods. Anal. Chim. Acta 450 (2001) 123-129.
16. Luinge H.J., van der Maas J.H., Visser T.: Partial least squares regression as a multivariate tool for the interpretation of infrared spectra. Chemometr. Intell. Lab. 28 (1995) 129-138.
17. Brierley P.: Visual Neural Data Mining. http://www.philbrierley.com, (2004).
18. Whitley D.: An overview of evolutionary algorithms: Practical issues and common pitfalls. Inform. Software Tech. 43 (2001) 817-831.
19. Nguyen D., Widrow B.: Improving the learning speed of two layer neural networks by choosing initial values of the adaptive weights. 1990 IJCNN International Joint Conference on Neural Networks, 17 June 1990, San Diego, California, IEEE, New York, (1990) 21– 26
20. StatSoft, Inc.: Electronic Statistics Textbook. http://www.statsoft.com/textbook/stathome.html, (2004).
21. Witten I.H., Eibe F.: Data Mining: Practical machine learning tools with Java implementations. Morgan Kaufmann, San Francisco, 2000.