CROATICA CHEMICA ACTA
CCACAA **78** (4) 557–561 (2005)

ISSN-0011-1643
CCA-3048
Conference Paper

Extraction of Comprehensible Logical Rules from Neural Networks. Application of TREPAN in Bio and Chemoinformatics*

Brian D. Hudson,[a], David C. Whitley,[a] Antony Browne,[b] and Martyn G. Ford[a]**

[a]*Centre for Molecular Design, Institute of Biomedical and Biomolecular Sciences, University of Portsmouth, Portsmouth, Hants, PO1 2DY, UK*

[b]*School of Computing, University of Surrey, Guildford, Surrey, GU2 7XH, UK*

RECEIVED JULY 25, 2003; REVISED AUGUST 29, 2003; ACCEPTED JANUARY 20, 2004

Keywords
bioinformatics
chemoinformatics
neural networks
rule induction
decision trees

TREPAN is an algorithm for the extraction of comprehensible rules from trained neural networks. The method has been applied successfully to biological sequence (bioinformatics) problems. It has now been extended to handle chemoinformatics (QSAR) datasets. The method has been shown to have advantages over traditional symbolic rule induction methods such as C5. Results obtained for bioinformatics and chemoinformatics problems using the TREPAN algorithm are presented.

INTRODUCTION

Artificial Neural Network (ANN) solutions are traditionally viewed as classification systems whose internal representations are extremely difficult to interpret. Simpler techniques are often of more utility due to the comprehensibility of the resulting models.[1] It is now becoming apparent that algorithms can be designed which extract understandable representations from trained neural networks, enabling them to be used for data mining, *i.e.* the discovery and explanation of previously unknown relationships present in data.

TREPAN[2] is an algorithm for the extraction of comprehensible rules from trained artificial neural networks. The aim is to overcome the »black box« nature of ANN models. We have recently described a generalized implementation of this algorithm,[3,4] and applied it to some problems in bioinformatics.

The aim of this paper is to describe further applications of the TREPAN methodology, specifically to datasets comprising real-valued variables as are commonly found in chemoinformatics problems. The examples include a brief summary of the original bioinformatics dataset together with three new examples where TREPAN is applied to chemoinformatic sets. The results are compared with equivalent results obtained using the well-known C5 rule induction algorithm.[5]

METHODS

The ANNs used in this study were feed-forward, back-propagation networks with a single hidden layer and a single output unit. Negative and positive training cases were assigned target values at the output unit of 0 and 1, respectively. The networks employed tanh transfer functions for the hidden units and a logistic transfer function

* Presented at the 18th International Dubrovnik Course & Conference MATH/CHEM/COMP 2003

** Author to whom correspondence should be addressed. (E-mail: brian.hudson@port.ac.uk) for the output unit. Network training was performed using the scaled conjugate gradient method to minimize the cross-entropy error, and a weight decay regularizer was employed to guard against over-training.[6] Network training was carried out using Netlab.[7]

The training protocol followed a similar procedure to that adopted by Manallack *et al*.[8] The details for dataset 1 appear elsewhere.[4] For datasets 2 and 3, to determine an optimal network, each dataset was divided randomly into training and validation sets in the ratio of 3:1. The number of hidden units was varied between 2 and 5, and in each case 20 networks were minimized over the training set starting from random initial weights. In addition to the weight decay error term, an early stopping rule based on the validation set was used to provide further protection against over-training. The network with the lowest validation set error was selected for analysis by TREPAN.

The procedure for dataset 4 was similar, except that a ratio of 2:1 was used for the division into training and validation sets, to maintain consistency with the original analysis.

Decision trees were constructed using the generalized implementation of the TREPAN algorithm.[4] A sample size of 1000 was used together with a maximum tree size of 9 nodes. This was found to be sufficient to describe the datasets. The equivalent C5 decision trees were extracted using Clementine v7. For these results the default values for C5 were used.

The decision trees in the figures are structured so that a positive result for a test leads down the left hand branch of the tree. Negative results are to the right. For the TREPAN trees, unclassified, means that there are further nodes in this branch of the tree that have not been shown in the figure. The numbers in the leaf nodes for the TREPAN trees indicate the number of training set examples reaching that node.

Data were obtained from the original sources as described in the relevant part of the results and discussion section below.

RESULTS AND DISCUSSION

The four applications studied are:

1. Identifying splice junction sites in human DNA sequences.
2. Distinguishing drugs from leads.
3. Identifying conformational classes from molecular dynamics simulations.
4. QSAR analysis.

Table I shows the accuracy of the various methods on the original training sets expressed as a percentage of correct classifications. In this table the ANN column refers to the accuracy of the original ANN from which the TREPAN tree was trained.

TABLE I. Accuracy for chemoinformatic datasets

Dataset	C5 / %	Neural Network / %	TREPAN / %
Splice Junction Donor	91.9	93.9	90.7
Drug/Lead	69.3	68.6	65.7
Conformation	99.0	94.4	94.4
QSAR	91.8	98.0	91.8

Dataset 1: Splice Junction prediction

This dataset is the clean dataset of human splice junction sites from Thanaraj.[9] The set comprises a training set of 567 positive and 943 negative sequences and an external test set of 229 positive and 373 negative sequences. The set was chosen because it is a well-studied problem and the answer, in terms of a consensus sequence, is well known to be the sequence:[10]

C/G A G | G T A/G A G T

From Table I it can be seen that all three methods perform reasonably well on this dataset. Although C5 gives accurate results, the decision tree it uses to do this, is highly complex. In contrast the decision tree produced by TREPAN (Figure 1) is a simple M-of-N rule, where '-2 = A' describes an Adenine at position -2 and '-1 = G' describes a Guanine at position -1 *etc*. Furthermore, the rule is cast in a form that is very recognizable to the practicing biologist.

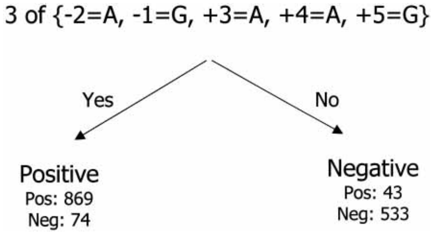

Figure 1. TREPAN tree for splice junction dataset.

Dataset 2: Distinguishing Drugs from Leads

This dataset comprises a set of 137 drug like molecules classified into drugs or leads.[11] For each of these a set of 7 descriptors, analogous to those in the original paper, was calculated using Cerius-2.[12] These parameters are *A*log*P*, MW, MR, N donors, N acceptors, N rot bonds and the number of Lipinski violations. This dataset is interesting as the intention is to derive useful »rules of thumb« in the same spirit as the Lipinski rules.[13] Indeed, as these rules are inherently cast as an M-of-N rule (*i.e.* If two of four conditions are met the compound is unlikely to be bioavailable) the formalism of the TREPAN rules may be very suitable for this type of problem.

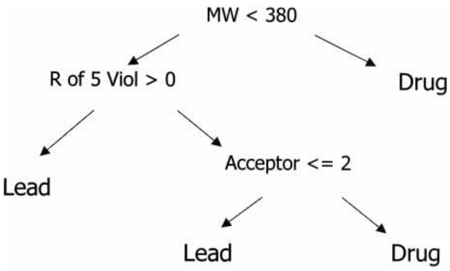

Figure 2a. C5 tree for drug/lead dataset.

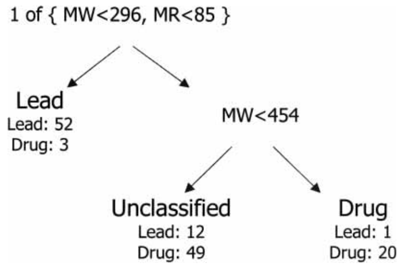

Figure 2b. TREPAN tree for drug/lead dataset.

The network training protocol resulted in a network with 3 hidden units. The C5 and TREPAN trees are shown in Figures 2a and 2b, respectively. Both methods give sensible trees consistent with the findings of the original paper. However the trees are different. The C5 tree splits first on MW then on the number of Lipinski violations and finally on the number of H bond acceptors. The tree seems very reasonable. The TREPAN tree also splits on MW but also on MR. Of course these two are highly correlated but the TREPAN tree shows the advantage of the M-of-N formalism. The 1-of-2 rule derived is essentially an OR condition, something which a binary split method such as C5 cannot achieve.

Dataset 3: Conformational Analysis

This dataset concerns the analysis of conformational data using statistical techniques. The data is derived from molecular dynamics simulations of the anti-diabetic agent rosiglitazone (Figure 3). These simulations were performed for a period of 5ns, using standard techniques. A sample structure was taken every 1ps leading to 5000 data points. Each of these conformations was classified as either a folded or an extended structure based on the measured distance between the two ends of the molecule. Figure 4 shows these distances across the time series and it can be seen that the conformations are divided roughly 50:50 between the folded (<10A) and extended (>10A) structures. The data themselves are the dihedral angles of each of the 8 flexible torsion angles T1-T8.

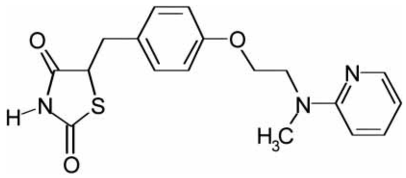

Figure 3. Structure of rosiglitazone.

In this case the network training procedure resulted in a network with 4 hidden units. The C5 algorithm produces highly accurate classifications of the conformations but does this using a very complicated decision tree (not described here). By contrast, the TREPAN tree (Figure 5) is straightforward (note that the TREPAN analysis was performed on a subset of the original data). Essentially, this tree is showing how the molecule adopts a folded conformation. Firstly, the molecule must fold about one of the central torsion angles, in this case T5. In addition, it needs to have required values for two outer torsion angles T2 and T7. This second rule again shows the value of the M-of-N formalism. In this case the 2-of-2 rule is analogous to a logical AND operation.

Dataset 4: QSAR

The final example is a standard QSAR dataset.[14] This comprises a set of 48 inhibitors of HIV-1 protease. These compounds were categorized into low (pIC50 < 8) and high (pIC50 > 8) inhibitory activity against the enzyme. The X block is a set of 14 parameters described in Ref. 14. The original paper utilizes a 3 component PLS model

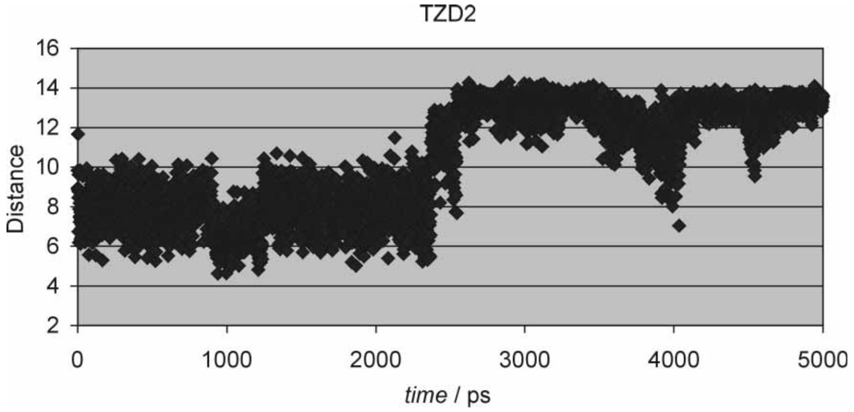

Figure 4. Distance *vs.* time for molecular dynamics simulation data.

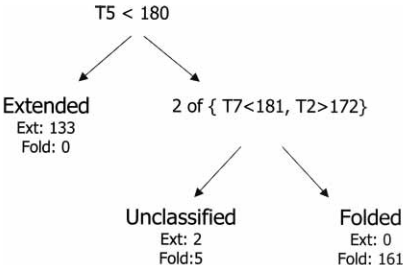

Figure 5. TREPAN tree for conformation dataset.

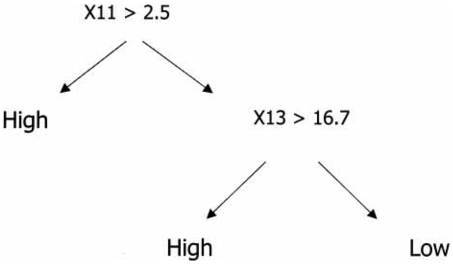

Figure 6a. C5 tree for QSAR dataset.

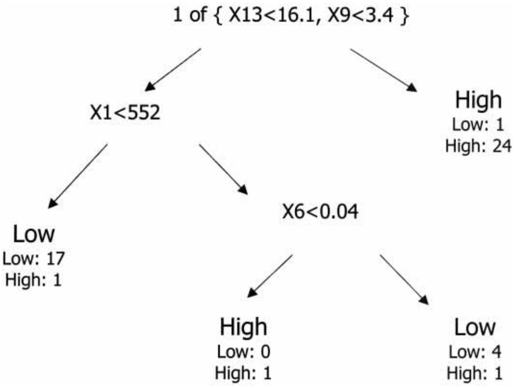

Figure 6b. TREPAN tree for QSAR dataset.

with an R^2 of 0.91 and a Q^2 of 0.85. The highest loadings of the parameters of the model were X9, X11, X10 and X13 all of which were positive.

The optimal network was found to have 2 hidden units. The C5 tree (Figure 6a) gives a simple splitting on the values of X11 and X13. Both of these have high loadings in the original PLS model. The TREPAN tree (Figure 6b) differs from this. Again the primary split is a 1-of-2 rule analogous to a logical OR. Once again the tree is readily comprehensible and preserves the features of the original model.

CONCLUSIONS

The examples illustrated show the utility of decision tree approaches to common problems in bioinformatics and chemoinformatics. In particular, the M-of-N formalism employed by TREPAN appears to be particularly useful both for the analysis of bioinformatic data, where the result is often a consensus sequence and also to chemoinformatic problems, where rule of thumb solutions can have advantages over more precise QSAR predictions in that the derived rules are much easier to interpret and are therefore, arguably, of more value to the practicing medicinal chemist who has to '*make the next molecule*'.

A further advantage of the TREPAN methodology is that it can be generalized to extract rules from other classifiers. It would be of interest to apply TREPAN using C5 as the classifier. This approach could combine the accuracy and computational efficiency of C5 with the comprehensibility of the TREPAN formalism.

Acknowledgements. – This work was supported by the BBSRC, grant ref. BIO/12005. We are also grateful to SPSS Inc. for supplying a copy of the Clementine software to perform the C5 calculations.

Note. – The TREPAN software is available for download from http://www.cmd.port.ac.uk/.

REFERENCES

1. B. Lucic, D. Nadramija, I. Basic, and N. Trinajstic, *J. Chem. Inf. Comput. Sci.* **43** (2003) 1094–1102.
2. M. W. Craven, *Extracting comprehensible models from trained neural networks*, Ph.D. Thesis, University of Wisconsin, Madison, 1996, available at http://www.cs.wisc.edu/~shavlik/abstracts/craven.thesis.abstract.html.
3. A. Browne, B. D. Hudson, D. C. Whitley, M. G. Ford, and P. Picton, *Rule Extraction from trained Neural Networks* in: D. J. Livingstone and M. G. Ford (Eds.), *Proceedings of 14th European QSAR Conference*, Blackwell Scientific, London, 2003, pp. 391–393.
4. A. Browne, B. D. Hudson, D. C. Whitley, M. G. Ford, and P. Picton, *Neurocomputing* **57** (2004) 275–293.
5. R. Kohavi and J. R. Quinlan, in: *Handbook of data mining and knowledge discovery*, Oxford University Press, New York, 2002.
6. C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, New York, 1995.
7. I. T. Nabney, *Netlab: Algorithms for Pattern Recognition*, Springer, London, 2002.
8. D. T. Manallack, B. G. Tehan, E. Gancia, B. D. Hudson, M. G. Ford, D. J. Livingstone, D. C. Whitley, and W. R. Pitt, *J. Chem. Inf. Comput. Sci.* **43** (2003) 674–679.
9. T. A. Thanaraj, *Nucleic Acids Res.* **27** (1999) 2627–2637.
10. J. D. Watson, N. H. Hopkins, J. W. Roberts, J. Argetsinger, and A. Weiner, *Molecular Biology of the Gene*, 4th Ed., Benjamin Cummings, Menlo Park, CA, 1987.
11. T. I. Oprea, A. M. Davis, S. J. Teague, and P. D. Leeson, *J. Chem. Inf. Comput. Sci.* **41** (2001) 1308–1315.
12. Cerius-2, MSI Inc., San Leandro, CA.
13. C. A. Lipinski, F. Lombardo, B. W. Dominy, and P. J. Feeney, *Adv. Drug Deliv. Res.* **46** (2001) 3–26.
14. R. Kiralj and M. M. C. Ferreira, *J. Mol. Graphics Modell.* **21** (2003) 435–448.

SAŽETAK

Izlučivanje razumljivih logičkih pravila iz neuronskih mreža. Primjena TREPAN algoritma u bioinformatici i kemoinformatici

Brian D. Hudson, David C. Whitley, Antony Browne i Martyn G. Ford

TREPAN je algoritam za izlučivanje razumljivih pravila iz neuronskih mreža nakon provedenoga postupka učenja. Metoda je uspješno primjenjivana na probleme u bioinformatici, za analizu bioloških sekvencija. Primjena TREPAN metode sada se proširuje i na analizu skupova podataka u kemoinformatici (QSAR). Pokazano je da metoda ima prednosti u odnosu na uobičajene postupke koji se rabe za indukciju simboličkih pravila poput metode C5. Prikazani su rezultati koji su dobiveni u analizi bioinformatičkih i kemoinformatičkih problema s pomoću algoritma TREPAN.