Haiku: interactive comprehensible data mining

Russell Beale
School of Computer Science
University of Birmingham
Edgbaston, Birmingham, B15 2TT, UK
R.Beale@cs.bham.ac.uk
+44 121 414 3729

Andy Pryke
School of Computer Science
University of Birmingham
Edgbaston, Birmingham, B15 2TT, UK
A.N.Pryke@cs.bham.ac.uk
+44 121 414 5142

ABSTRACT

This paper discusses a novel data mining system devised and developed at Birmingham, which attempts to provide a more effective data mining system. It ties in with the workshop goals in a number of different areas. The system tries to allow users to see the big picture and not get caught up in irrelevant detail too early; it uses perception-based approaches to data mining, and it uses interactive visualisation techniques to assist in this process.

BIG QUESTIONS

In data mining, or knowledge discovery, we are essentially faced with a mass of data that we are trying to make sense of. We are looking for something "interesting". Quite what "interesting" is is hard to define, however - one day it is the general trend that most of the data follows that we are intrigued by - the next it is why there are a few outliers to that trend. In order for a data mining to be generically useful to us, it must therefore have some way in which we can indicate what is interesting and what is not, and for that to be dynamic and changeable.

The second issue to address is that, once we can ask the question appropriately, we need to be able to understand the answers that the system gives us. It is therefore important that the responses of the system are represented in ways that we can understand.

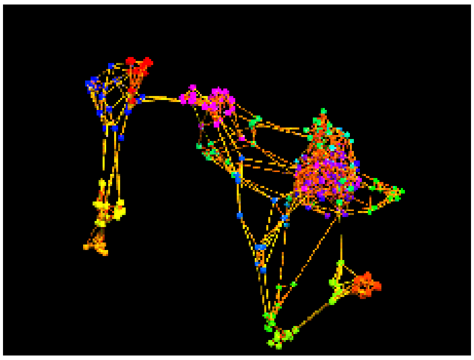
Thirdly, we should recognise the relative strengths of users and computers. The human visual system is exceptionally good at clustering, at recognising patterns and trends, even in the presence of noise and distortion. Computer systems are exceptionally good at crunching numbers, producing exact parameterisations and exploring large numbers of alternatives.

An ideal data mining system should, we would argue, offer the above characteristics and use the best features of both the user and the computer in producing its answers. This leads us towards a system that will be interactive, in order to be flexible and work towards a solution. It should use visualisation techniques to offer the user the opportunity to do both perceptual clustering and trend analysis, and to offer a mechanism for feeding back the results of machine-based data mining. It should have a data mining engine that is powerful, effective, and which can produce humanly-comprehensible results as well.

The Haiku system was developed with these principles in mind, and offers a symbiotic system that couples interactive 3-d dynamic visualisation technology with a novel genetic algorithm.

VISUALISATION

The visualisation engine used in the Haiku system provides an abstract 3-d perspective of multi-dimensional data. The visualisation consists of nodes and links, whose properties are given by the parameters of the data. Data elements affect parameters such as node size, mass, link strength and elasticity, and so on. Multiple elements can affect one parameter, or a subset of parameters can be chosen. Nodes are scattered randomly into the 3d space, with their associated links. This 3d space has obeys a set of physical-type laws, which affect this initial arrangement. Links tend to want to assume a particular length, and tend to pull inwards until they reach that length, or push outwards if they are compressed, just as a spring does in the real world. Nodes tend to repel each other, based on their mass. This can be seen as a force directed graph visualisation. The physics of the space are adjustable, but are chosen so that a steady state solution can be reached that is static - this is unlike the real world, in which a steady state exists that involves motion, with one body orbiting another. This initial state is then allowed to evolve, and the links and nodes shuffle themselves around until they reach a local minimum, low energy steady state. This is then static, and can be explored at will by rotating it, zooming in and flying through and around it. It is a completely abstract representation of the data, and so has no preconceptions built in. Different data to attribute mappings will clearly give different structures, but the system can at least produce a view of more than 3 dimensions of the raw data at once.



A typical structure is shown in Figure 1.

Figure 1. Nodes and links self-organised into stable structure.

PERCEPTION-BASED DISCOVERY

From this initial visualisation, the user can view the structure from whatever viewpoint they wish, and can indicate to the system the data that they are interested in - whether it is a particular cluster, or a number of outliers, or a suspected trend through the space, the relevant areas can be selected. This process takes full advantage of the abilities of our visual system.

From this, the processing moves towards the computer, as the genetic algorithm-based process takes over.

GENETIC ALGORITHMS FOR DATA MINING

We use a genetic algorithm (GA) approach for a number of reasons. The first is that a GA is able to effectively explore a large search space, and modern computing power means we can take advantage of this within a reasonable timeframe. We use a special type of GA that evolves rules; these produce terms to describe the underlying data of the form:

> IF `term` OP `value|range` (AND …) THEN `term` OP `value|range` (AND …)

where `term` is a class from the dataset, OP is one of the standard comparison operators (<, >, =, ≤, ≥), `value` is a numeric or symbolic value, and `range` is a numeric range. A typical rule would therefore be:

IF colour = red AND consistency = soft THEN fruit = strawberry

A set of these rules can, in principle, describe any arbitrary situation. There are two situations that are of interest to us; classification, when the left hand side of the equation tries to predict a single class (usually known) on the right hand side, and association, when the system tries to find rules that characterise portions of the dataset.

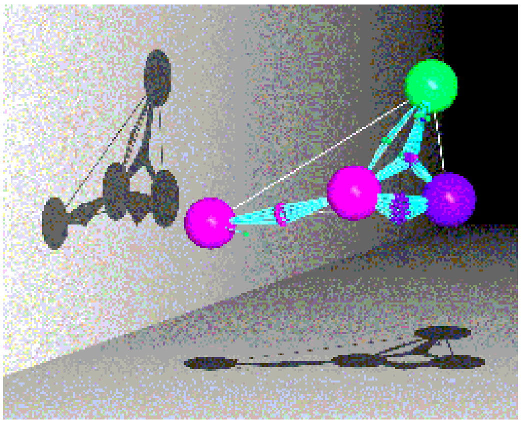
The algorithm follows fairly typical genetic algorithmic approaches in its implementation, but with specialised mutation and crossover operators, in order to explore the space effectively. Statistically principled comparisons showed that this technique is at least as good as conventional machine learning at classification (Pryke, Andy, thesis [1]), but has advantages over the more conventional approaches in that it can perform clustering operations too. The genetic algorithm aims to optimise an objective function, and manipulation of this function allows us to explore different areas of the search space.

One of the key design features is to produce a system that has human-comprehensible results. Rules of this form are inherently much more understandable than decision trees or probabilistic or statistical descriptions. However, since the GA is trying to minimise an objective function, we can manipulate this function to achieve different results. If we insist that the rules produced must be short (and hence easier to understand) then the system will trade off accuracy and/or coverage but will give us a general overview that is appropriate for much of the data. When we have this higher level of comprehension, we can allow the rules to become longer and hence more specific, and accuracy will then increase.

FEEDBACK

The results from the GA are fed back into the visualisation: identified clusters can be coloured, for example, or rules added and linked to the data that they classify, as in Figure 2.

Figure 2: Rules and classified data



In this figure, rules are the large purple, fuschia and green spheres, with the data being the smaller spheres. Links are formed between the rules and the data that is covered by the rule, and the visualisation has reorganised itself to show this clearly. We have additionally coloured the data according to its correct classification.

A number of things are immediately apparent from this visualisation, much more easily than would the case from a textual description. On the very left of the figure, one rule, the fuschia sphere, covers exactly the same data as the other fushia sphere, except it also misclassifies one green data point. But the rightmost fushia rule, whilst correctly classifying all the fushia data also misclassifies much of the other data as well. On the right hand side, the purple rule clearly does very well; it covers all its data and doesn't misclassify anything. The green rule at the top has mixed results.

The system is fully interactive, in that the user can now identify different characteristics and instruct the GA to describe them, and so the process continues.

This synergy of abilities between the rapid, parallel exploration of the structure space by the computer and the user's innate pattern recognition abilities and interest in different aspects of the data produces a very powerful and flexible system.

CLASSIC CASE STUDY: WELL KNOWN DATASETS

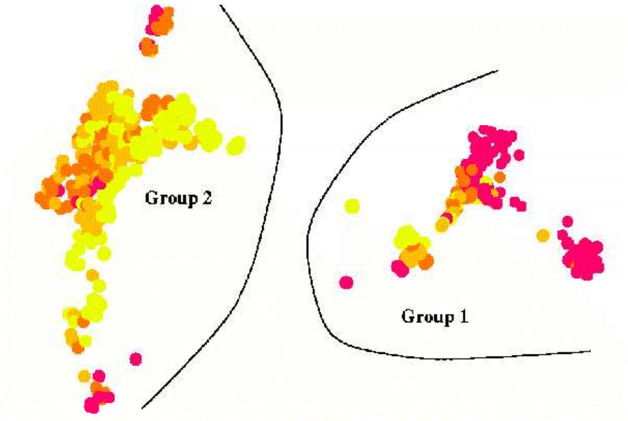
Several machine learning datasets from the UCI Machine Learning Repository[2] were used to benchmark the performance of data mining and classification. It should be noted that it focuses on quantitative performance, whereas the qualitative experience and use of perception-based mining techniques is not assessed. However, good results on these datasets in quantitative terms will give us confidence when analysing new datasets.

The GA-based approach gave perfectly acceptable results, with statistically analysis showing it performed better than C4.5[6] on the "Australian Credit Data" (p=0.0018). No significant difference in performance were found for the other two datasets. These results are summarised below

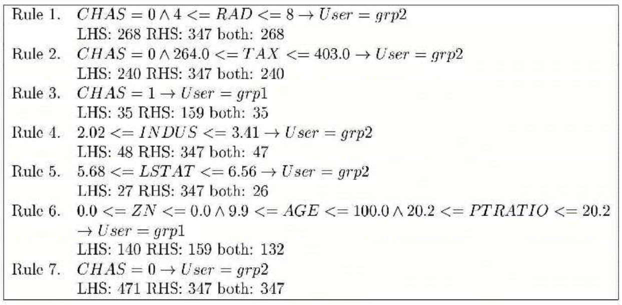
Dataset	Genetic algorithm % Correct	C4.5 [] % Correct
Australian Credit [3]	86%	82%
Boston Housing [4]	64%	65%
Pima Indians Diabetes [5]	73%	73%

CASE STUDY: INTERACTIVE DATA MINING OF HOUSING DATA

The figure below shows a 2D view of the systems visual clustering of the Boston Housing data. Two used selected groups have been indicated.



Ga based data mining was then applied to these user identified groups. The fitness function was chosen so as to bias the system towards the discovery of rules which are short and accurate. The rules are shown below:



Rule 1. $CHAS = 0 \wedge 4 <= RAD <= 8 \rightarrow User = grp2$
LHS: 268 RHS: 347 both: 268
Rule 2. $CHAS = 0 \wedge 264.0 <= TAX <= 403.0 \rightarrow User = grp2$
LHS: 240 RHS: 347 both: 240
Rule 3. $CHAS = 1 \rightarrow User = grp1$
LHS: 35 RHS: 159 both: 35
Rule 4. $2.02 <= INDUS <= 3.41 \rightarrow User = grp2$
LHS: 48 RHS: 347 both: 47
Rule 5. $5.68 <= LSTAT <= 6.56 \rightarrow User = grp2$
LHS: 27 RHS: 347 both: 26
Rule 6. $0.0 <= ZN <= 0.0 \wedge 9.9 <= AGE <= 100.0 \wedge 20.2 <= PTRATIO <= 20.2$
$\rightarrow User = grp1$
LHS: 140 RHS: 159 both: 132
Rule 7. $CHAS = 0 \rightarrow User = grp2$
LHS: 471 RHS: 347 both: 347

This case study illustrates the following. The interactive visual discovery approach has revealed new structure in the data by visual clustering. Application of a data mining algorithm has generated concrete information about these "soft" discoveries.

Together, interactive data mining has delivered increased knowledge about a well known dataset.

ACKNOWLEDGEMENTS

This work was partially supported by grants from British Telecom, Integral Solutions Ltd and British Maritime Technology. Thanks to Nick Drew and Bob Hendley for their work on the visualisation parts of the system, and to colleagues for their comments and help.

REFERENCES

[1] Pryke, Andy Neil. - Data mining using genetic algorithms and interactive visualisation. Ph.D thesis University of Birmingham, 1999.

[2] Blake, C.L. & Merz, C.J. (1998). UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html].

Irvine, CA: University of California, Department of Information and Computer Science

[3] J. Ross Quinlan ,Simplifying decision trees, Int. J Man-Machine Studies 27, Dec 1987, pp. 221-234.

[4] Quinlan,R. (1993). Combining Instance-Based and Model-Based Learning. In Proceedings on the Tenth International Conference of Machine Learning, 236-243, University of Massachusetts, Amherst. Morgan Kaufmann.

[5] Smith, J.W., Everhart, J.E., Dickson, W.C., Knowler, W.C., Johannes, R.S. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In Proceedings of the Symposium on Computer Applications and Medical Care (pp. 261--265). IEEE Computer Society Press.

[6] Quinlan, R. (1992) C4.5: Programs for Machine Learning", Morgan Kaufmann.