Geometric Comparison of Classifications and Rule Sets*

Trevor J. Monk, R. Scott Mitchell, Lloyd A. Smith and Geoffrey Holmes

Department of Computer Science
University of Waikato
Hamilton, New Zealand
tmonk@cs.waikato.ac.nz, rsm1@cs.waikato.ac.nz, las@waikato.ac.nz, geoff@waikato.ac.nz

Abstract

We present a technique for evaluating classifications by geometric comparison of rule sets. Rules are represented as objects in an n-dimensional hyperspace. The similarity of classes is computed from the overlap of the geometric class descriptions. The system produces a correlation matrix that indicates the degree of similarity between each pair of classes. The technique can be applied to classifications generated by different algorithms, with different numbers of classes and different attribute sets. Experimental results from a case study in a medical domain are included.

Machine Learning, Classification, Rules, Geometric Comparison

1. Introduction

Inductive learning algorithms fall into two broad categories based on the learning strategy that is used. In *supervised learning* (learning from examples) the system is provided with examples, each of which belongs to one of a finite number of classes. The task of the learning algorithm is to induce descriptions of each class that will correctly classify both the training examples and the unseen test cases. *Unsupervised learning*, on the other hand, does not require pre-classified examples. An unsupervised algorithm will attempt to *discover* its own classes in the examples by clustering the data. This learning mechanism is often referred to as 'learning by observation and discovery.'

One of the most important criteria for evaluating a learning scheme is the quality of the class descriptions it produces. In general, many descriptions can be found that cover the examples equally well, but most perform badly on unseen cases. Techniques are required for evaluating the quality of classifications, either with respect to a classification or simply relative to each other.

This paper presents a new method for comparing classifications, using a geometric representation of class descriptions. The similarity of classes is determined from their overlap in an n-dimensional hyperspace. The technique has a number of advantages over existing statistical or instance-based methods of comparison:

- Descriptions are produced that indicate *how* two classifications are different, rather than simply *how much* they differ.
- The algorithm bases its evaluation on descriptions of the classifications (expressed as production rules), not on the instances in the training set. This approach will tend to smooth out irregular or anomalous data that might otherwise give misleading results.
- Classifications using differing numbers or groups of attributes can be compared, provided there is some overlap between the attribute sets.
- The technique will work with any clustering scheme whose output can be expressed as a set of production rules.

In this study, the geometric comparison technique is used to evaluate the performance of a clustering algorithm (AUTOCLASS) in a medical domain. Two other techniques are also used to compare the automatically generated classifications against a clinical classification produced by experts.

* This project was funded by the New Zealand Foundation for Research in Science and Technology

1.1. Comparing Classifications and Rule Sets

Regarding evaluation of unsupervised 'clustering' type methods, Michalski & Stepp (1983) state that:

> " The problem of how to judge the quality of a clustering is difficult, and there seems to be no universal answer to it. One can, however, indicate two major criteria. The first is that the descriptions formulated for clusters (classes) should be 'simple', so that it is easy to assign objects to classes and to differentiate between the classes. This criterion alone, however, could lead to trivial and arbitrary classifications. The second criterion is that class descriptions should 'fit well' the actual data. To achieve a very precise 'fit', however, a description may have to be complex. Consequently, the demands for simplicity and good fit are conflicting, and the solution is to find a balance between the two."

The CLUSTER/2 algorithm used a combined measure of cluster quality based on a number of elementary criteria including the 'simplicity of description' and 'goodness of fit' mentioned above (Michalski & Stepp, 1983).

Hansen & Bauer (1987) use an information-theoretic measure of cluster quality in their WITT system. This *cohesion* metric evaluates clusters in terms of their within-class and between-class similarities, using the training examples that have been assigned to each class. Other measures are based on the class *descriptions*—in the form of decision trees, rules or a 'concept hierarchy' as used by UNIMEM (Lebowitz, 1987) or COBWEB (Fisher, 1987). Some clustering systems produce class descriptions as part of their normal, operation while others, such as AUTOCLASS (Cheeseman, *et. al.*, 1988) merely assign examples to classes. In this case, a supervised learning algorithm such as C4.5 (Quinlan, 1992) can be used to induce descriptions for the classification. This is the method used in this study for evaluating AUTOCLASS clusterings.

Mingers (1989) uses the two criteria *size* and *accuracy* for evaluating a decision tree (or an equivalent set of rules). Following the principle of Occam's Razor, it is generally accepted that the fewer terms in a model the better; therefore, in general, a small tree or rule set will perform better on test data. Accuracy is a measure of the predictive ability of the class description when classifying unseen test cases. It is usually measured by the error rate—the proportion of incorrect predictions on the test set (Mingers, 1989). Accuracy is often used to measure classification quality, but it is known to have several defects (Mingers, 1989; Kononenko & Bratko, 1991). An information-based measure of classifier performance developed by Kononenko & Bratko (1991) eliminates these problems and provides a more useful measure of quality in a variety of domains.

The remainder of this paper is organised as follows. The next section briefly describes WEKA, a machine learning workbench currently under development at the University of Waikato. This includes an overview of the AUTOCLASS and C4.5 algorithms used in our experiment. Section 3 describes our experimental methodology and the algorithm used by the geometric rule set comparison system. Results of the experiment, using a diabetes data set, are presented in Section 4. These results are discussed in Section 5, including some analysis of the performance of the geometric comparison algorithm. Section 6 contains some concluding remarks and ideas for further research in this area.

2. The WEKA Workbench

WEKA,[1] the Waikato Environment for Knowledge Analysis, is a machine learning workbench currently under development at the University of Waikato (McQueen, et. al, 1994). The purpose of the workbench is to give users access to many machine learning algorithms, and to apply these to real-world data.

WEKA provides a uniform interactive interface to a variety of tools, including machine learning schemes, data manipulation programs, and the LISP-STAT statistics and graphics package (Tierney, 1990). Data sets to be manipulated by the workbench use the ARFF (Attribute-Relation File Format) intermediate file format. An ARFF file records information about a relation such as its name, attribute names, types and values, and instances (examples). The WEKA interface is implemented using the TK X-Window widget set under the TCL scripting language (Ousterhout,

[1] The name is taken from the weka, a small, inquisitive native New Zealand bird related to the well-known Kiwi.

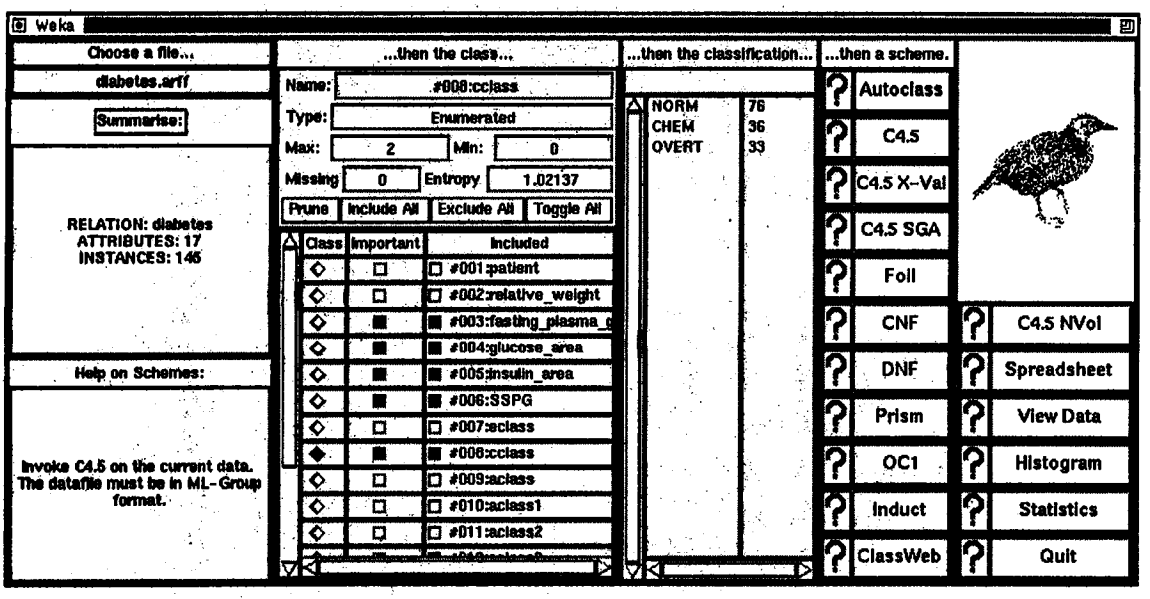

Figure 1. *The* WEKA *Workbench*

1993). ARFF filters and data manipulation programs are written in C. WEKA runs under UNIX on Sun workstations. Figure 1 shows an example display presented by the workbench.

2.1. AutoClass

AUTOCLASS is an unsupervised induction algorithm that automatically discovers classes in a database using a Bayesian statistical technique. The Bayesian approach has several advantages over other methods (Cheeseman, *et.al.*, 1988). The number of classes is determined automatically; examples are assigned with a probability to *each* class rather than absolutely to a single class; all attributes are potentially significant to the classification; and the example data can be real or discrete.

An AUTOCLASS run proceeds entirely without supervision from the user. The program continuously generates classifications until a user-specified time has elapsed. The best classification found is saved at this point. A variety of reports can be produced from saved classifications. A WEKA filter has been written that extracts the most probable class for each instance, and outputs this information in a form suitable for inclusion in an ARFF file. This allows the AUTOCLASS classification to be used as input to other programs, such as rule and decision tree inducers.

2.2. C4.5

C4.5 (Quinlan, 1992) is a powerful tool for inducing decision trees and production rules from a set of examples. Much of C4.5 is derived from Quinlan's earlier induction system, ID3 (Quinlan, 1986). The basic ID3 algorithm has been extensively described, tested and modified since its invention (Mingers, 1989; Utgoff, 1989) and will not be discussed in detail here. However, C4.5 adds a number of enhancements to ID3, which are worth examining.

C4.5 uses a new 'gain ratio' criterion to determine how to split the examples at each node of the decision tree. C4.5. This removes ID3's strong bias towards tests with many outcomes (Quinlan, 1992). Additionally, C4.5 allows splits to be made on the values of *continuous* (real and integer) attributes as well as enumerations.

Decision trees induced by ID3 are often very complex, with a tendency to 'over-fit' the data (Quinlan, 1992). C4.5 provides a solution to this problem in the form of *pruned* decision trees or production rules. These are derived from the original decision tree, and lead to structures that generally cover the training set less thoroughly but perform better on unseen cases. Pruned trees and rules are roughly equivalent in terms of their classification accuracy; the advantage of a rule representation is that it is more comprehensible to people than a decision tree (Cendrowska, 1987; Quinlan, 1992).

The first stage in rule generation is to turn the initial decision tree 'inside-out' and generate a rule corresponding to each leaf. The resulting rules are then generalised to remove conditions that do not contribute to the accuracy of the classification. A side-effect of this process is that the rules are no longer exhaus-

tive or mutually exclusive (Quinlan, 1992). C4.5 copes with this by using 'ripple-down rules' (Compton, et. al., 1992). The rules are ordered, and any example is then classified by the first rule that covers it. In fact, only the *classes* are ranked, with the twin advantages that the final rule set is more intelligible and the order of rules within each class becomes irrelevant (Quinlan, 1992). C4.5 also defines a *default class* that is used to classify examples not covered by any of the rules.

3. Methodology

3.1. The Data Set

Reaven and Miller (1979) examined the relationship between Chemical and Overt diabetes in 145 non-obese subjects. The data set used in this study involves six attributes: patient age, patient relative weight, fasting plasma Glucose, Glucose, Insulin and steady state plasma Glucose (SSPG). The data set also involves two classifications, labeled CClass and EClass—presumably representing 'clinical classification' and 'electronic classification'. Each classification describes three classes: Overt diabetics, those requiring Insulin injections; Chemical diabetics, whose condition may be controlled by diet; and a Normal group, those without any form of diabetes. The same data set is used in the present study with the omission of patient age. Reaven and Miller found the three attributes Glucose, Insulin, and SSPG to be more significant than any of the others.

Both the clinical and electronic classifications assume the presence of three classes. The scatter plot below (Figure 2) shows the data set and its clinical classification. Each of the six small plots shows a different pair of variables (the plots at the lower right are simply reflections of those at the upper left). For example, the plot in the upper left corner shows Glucose vs. SSPG. The clinical classification appears to be highly related to the Glucose measurement: in fact, the three classes appear to be divided by the Glucose measurement alone. This is particularly well illustrated in the plot of Glucose vs. SSPG in the upper left plot.

The Electronic classification (EClass) is generated using a clustering algorithm by Friedman and Rubin (1967). Each class is described by the mean of the three variables of the instances in that class (Glucose, Insulin

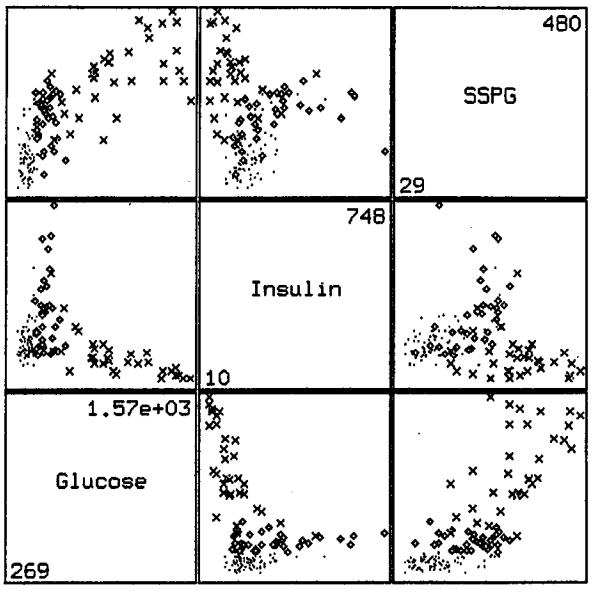

Figure 2. *Scatterplot matrix of diabetes data*

- • Normal class
- ◇ Chemical class
- × Overt class

and SSPG), giving a single point in 3-space. Each instance is assigned to the class whose mean lies closest to it (in terms of the minimum Euclidean distance). After each instance has been assigned to a class, the class means are recalculated. The process of assigning instances to classes and recalculating class means repeats until no patients are reassigned. The algorithm assumes prior knowledge of the class means and number of classes, to define the initial classes. Reaven and Miller used the results of a previous study (Reaven *et. al.*), 1976) to determine suitable starting means.

3.3. Classification using AutoClass

One objective of this study was to determine if the patients in the data set naturally fall into groups, without prior knowledge of the number of groups or the attributes of the groups. We also wished to determine the effectiveness of AUTOCLASS's classification of the diabetes data, and compare it to the clinical classification.

It is difficult to determine which attributes from the data set are reasonable ones to generate a classification. Some may be irrelevant, and a classification generated using them may produce insignificant classes. Reaven and Miller considered only Glucose, Insulin and SSPG to be relevant attributes. They found that fasting plasma Glucose exhibited a high degree of linear association with Glucose (r = 0.96), indicating that these two variables are

Classification	Attributes
AClass	Glucose, Insulin, SSPG
AClass1	Relative weight, Fasting Plasma Glucose, Glucose, Insulin, SSPG
AClass2	Glucose, SSPG
AClass3	Glucose, Insulin

Table 1. AUTOCLASS *classifications*

Classification	Error Rate
CClass	7.2%
EClass	5.6%
AClass	5.6%
AClass1	7.2%
AClass2	3.2%
AClass3	4.8%

Table 2. *Predicted error rate of rule sets*

essentially equivalent. Four classifications were made in the present study, using AUTOCLASS. Each used different selected attributes, as shown in Table 1. Since AUTOCLASS completes its classification after a specified time has elapsed, an arbitrary execution time of one hour was chosen. Classes did not appear to change significantly with longer execution times. However, a comprehensive study of the effects of differing execution time has not been performed.

Although different in detail, all the classifications divide the data set into classes that bear an obvious similarity to the clinical classification. Thus we were able to assign the names 'Normal', 'Chemical' and 'Overt' to the generated classes. This is unlikely to be easy to achieve in general. C4.5 was used to generate rule sets describing all six classifications. The attributes used to induce the rule sets were in all cases the same as those used to generate the initial classification. Cross-validation checks were performed on the rule sets to derive a reliable estimate of their accuracy. The predicted error rates of each of the rule sets on unseen cases is shown in Table 2. Three techniques were used to compare the generated classifications with the clinical classification: classification differences by instance, classification differences by comparison of means, and classification differences by comparison of rules.

3.3.1. Comparing individual instances

The automatic classification of each instance was compared to the clinical classification, and the differences were tallied. A difference in the classification of an instance from the clinical classification is assumed to be a misclassification. The percentage misclassification gives some indication of the 'goodness' of a classification. Some misclassifications are more important than others, and a single error statistic does not illustrate this. A large number of Normal patients misclassified as Chemical diabetics may not be important, since they are in no danger of dying from this classification error, however if Overt patients are misclassified as Normal then death could result. The Friedman and Rubin automatic classification (EClass) has 20 misclassifications, giving an error statistic of 13.8%. This was deemed acceptable by Reaven and Miller.

3.3.2. Class comparison using two sample comparison of means

Each class may be described by the means of the attributes of the instances it contains. This is similar to the generation of a classification using Friedman and Rubin's clustering algorithm, where the means characterize the classes. Each class is described by the means and standard deviations of the three main attributes: Glucose, Insulin, and SSPG. For example, the Glucose mean for the Normal class of the electronic classification (EClass) will be compared with the Glucose mean for the Normal class of the clinical classification (CClass).

We used a two way t-test (Nie, et. al., 1975) to compare the class means. The null hypothesis was that there is no difference between the two means. The level of similarity between two classifications is given by the number of rejected t-tests. All tests were performed at the 95% level of significance. For example, the SSPG mean of the Chemical class for EClass is significantly different from the SSPG mean of the Chemical class for CClass. Assuming that the means of the classes in one classification must be the same as the means of the classes in another classification for the two classes to be considered equivalent, then we cannot say that the Chemical class for CClass is equivalent to the Chemical class generated by Friedman and Rubin's classification algorithm (EClass).

3.3.3. Comparing rules for classification comparison.

Neither technique described above allows us to accurately compare different classifications. Comparing classifications by examining misclassified instances is dependent on

the two individual sets of data being compared. Examining misclassifications may not provide an accurate estimate of the classification error rate for unseen data.

Assuming that a rule set accurately describes the classification of a set of data, then by comparing two sets of rules we are effectively comparing the two classifications. The technique presented in this paper for comparing rules produces a new set of rules describing the differences between the two classifications. An analysis of this kind allows machine learning researchers to ask the question "Why are these two classifications different?" Previously it has only been possible to ask "How different are these two classifications?"

3.4. Multidimensional Geometric Rule Comparison

This technique represents each rule as a geometric object in n-space. A rule set produced using C4.5 is represented as a set of such objects. As a geometric object, a rule forms a boundary within which all instances are classified according to that rule. The *domain coverage* of a set of rules is the proportion of the entire domain which they cover. Domain coverage is calculated by determining the *hypervolume* (n-dimensional volume) of a set of rules as a proportion of the hypervolume of the entire domain. The 'ripple down' rules of C4.5 must be made mutually exclusive to ensure that no part of the domain is counted more than once. The size of the overlap between two sets of rules provides an indication of their similarity. The non-overlapping portions of the two rule sets are converted into a set of rules describing the differences between the two rule sets.

3.4.1. Geometric Representation of Rules

A production rule can be considered to delimit a region of an n-dimensional space, where n is the 'dimension' of the rule—the number of distinct attributes used in the terms on the left hand side of the rule. The 'volume' of a rule is then simply the volume of the region it encloses. Any instance lying inside this region will be classified according to the right hand side of the rule. There is a problem, however, with rules that specify only a single bound for some attributes. For example, the two-dimensional rule

$$\text{Glucose} < 418 \wedge \text{SSPG} < 145 \Rightarrow \text{NORM}$$

does not give lower bounds for either of the attributes. The volume of this rule is effectively infinite, making it very hard to compare against anything else. Unfortunately, most of the rules induced from our classifications take this form, as C4.5 works by splitting attributes rather than explicitly defining regions of space. In this study, our solution to this problem has been to define absolute upper and lower limits for each attribute. Rules that do not specify a particular bound are then assumed to use the appropriate absolute limit. We have used the maximum and minimum values in the data set for each attribute as our absolute limits. The limits for the Glucose attribute are zero and 1600; for SSPG they are zero and 500. The rule above can then be re-written as:

$$\text{Glucose} > 0 \wedge \text{Glucose} < 418 \wedge \text{SSPG} > 0 \wedge \text{SSPG} < 145 \Rightarrow \text{NORM}.$$

The geometric representation of the above rule is shown in Figure 3. The solid dots represent the Normal examples from the CClass classification.

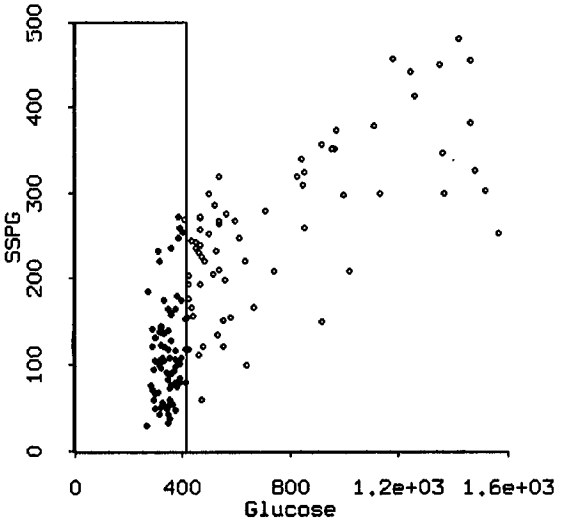

Figure 3. *Geometric representation of a rule*

An entire set of rules describing a data set can be represented as a collection of geometric objects of dimension m, where m is the maximum dimension of any rule in the set. Any rule with dimension less than the maximum in the set is promoted to the maximum dimension. For example, if another rule,

$$\text{Glucose} > 741 \Rightarrow \text{OVERT}$$

were added to the first, the dimension of this rule would have to be increased to two—the current maximum in the set. This is because two squares may be easily compared, but a line and a square, or a square and a cube,

may not. Promotion to a higher dimension is achieved by adding another attribute to a rule, but not restricting the range of that attribute. The rule

Glucose > 741 ⇒ OVERT

in one dimension is equivalent to

Glucose > 741 ∧ SSPG < 500 ∧ SSPG > 0 ⇒ OVERT

in two dimensions. The range of SSPG is not a factor in determining if a patient is Overt since no patient lies outside the range of SSPG specified in this rule.

Consider a comparison between the normal classes of EClass and AClass. The input rule sets are as follows:

EClass:

Glucose < 376 ⇒ NORM
Glucose < 557 ∧ SSPG < 204 ⇒ NORM

AClass:

Glucose < 503 ∧ SSPG < 145 ⇒ NORM
Glucose < 336 ⇒ NORM
Glucose < 418 ∧ SSPG < 165 ⇒ NORM

Figure 4 shows the geometric representation of the two rule sets. The solid boxes represent the EClass rules; the dotted ones represent the AClass rules.

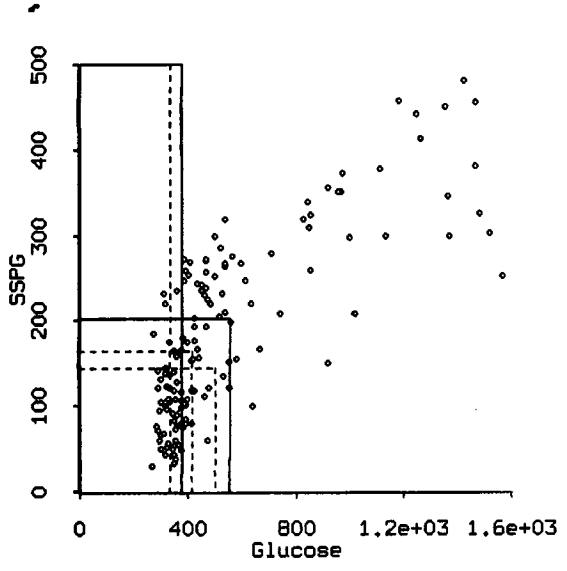

Figure 4. *Geometric input rules*

3.4.2. The Cutting Function

Making rules mutually exclusive, and determining their similarities and differences, all involve the cutting function. Consider the two dimensional example shown in Figure 5. Rule A overlaps Rule B in every dimension. Rule A is the cutting rule, and Rule B is the rule being cut.

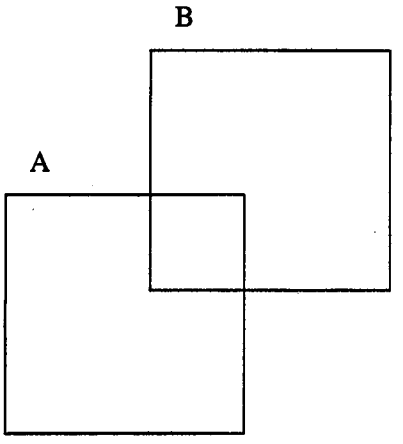

Figure 5. *Before cutting*

Each dimension in turn is examined; the first being the x dimension. The minimum bound (left hand edge) of rule A does not overlap rule B, so it need not be considered. The maximum bound (right hand edge) of rule A cuts rule B into two segments. Rule B becomes the section of rule B which was overlapped by A in the x dimension. A new rule, B1, is created which is the section of rule B not overlapped by Rule A in the x dimension (Figure 6).

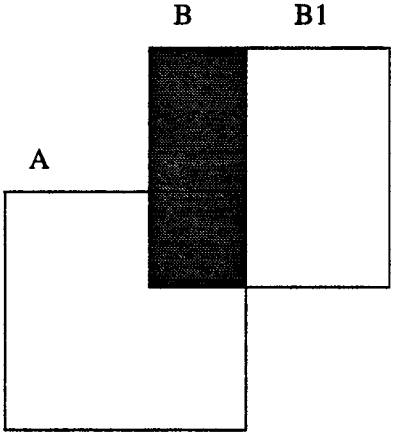

Figure 6. *After cutting x dimension*

Considering dimension 2, the y dimension: All references to rule B refer to the newly created rule B at the last cut. The minimum bound of rule A (the bottom edge) does not overlap rule B, so no cut is made. The maximum bound of rule A (the top edge) overlaps rule B, creating a new rule B2 which is the

section of rule B not overlapped by rule A. Rule B becomes the section of rule B which is overlapped by rule A (Figure 7). The remaining portion of the original rule B after all dimensions have been cut is the overlap between the two rules.

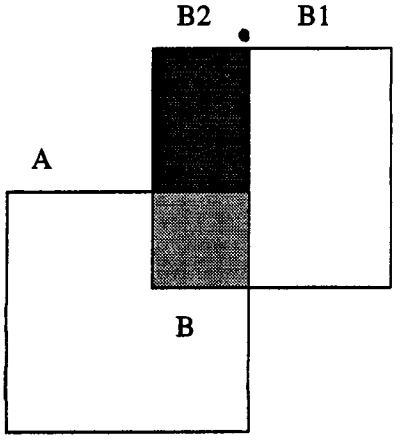

Figure 7. *After cutting y dimension*

The new rules A, B, B1, and B2 are all mutually exclusive. Rules B1 and B2 describe the part of the domain covered by the original Rule B, and not by Rule A. The cutting function generalises to N dimensions, since each dimension is cut independently of every other dimension.

Rule A is assumed to have a higher 'priority' than rule B. Any instances which lie in the overlap between rule A and rule B will remain within rule A after the cut.

3.4.3. Generating mutually exclusive rules

The 'ripple down' rules of C4.5 may be thought of as a priority ordering of rules—those that appear first are more important than those that follow. Any instances that fall into the overlap of two rules would be correctly classified by the higher priority rule, that is, the one which appears first in the list.

Obviously higher priority rules must not be cut by lower priority rules; the result would no longer correctly classify instances. In an ordered list of rules from highest to lowest priority, such as those output from C4.5, the rule at the head of the list will be used as a cutter for all those following it. Each cut rule is replaced by the segments not overlapped by the cutter. B is replaced by B1 and B2 in the example above. Once all the rules below the cutter have been cut, the rule following the cutter is chosen as the new cutter. When no cutters remain, the result is a list of mutually exclusive rules—which classify all instances exactly as before.

The mutually exclusive rules for the example comparison between EClass and AClass are shown below (Figure 8). As before, the solid boxes represent the EClass rules and the dotted boxes represent the AClass rules. The rules within each classification do not overlap.

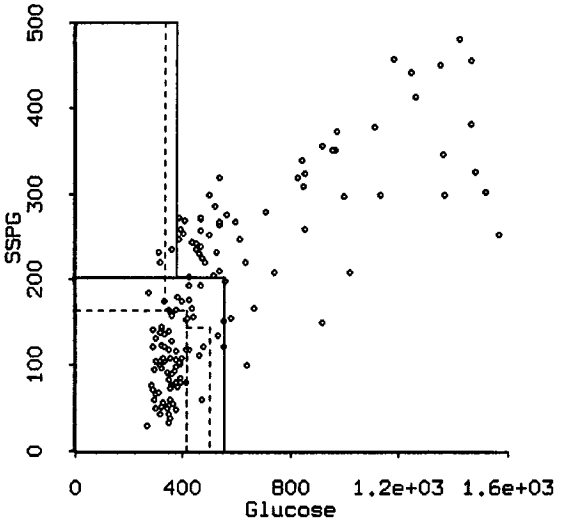

Figure 8. *Mutually exclusive rules*

3.4.4. Rule Set Differences

The objective of the geometric algorithm is to compare two sets of rules and determine their differences. Each class in the first set of rules is compared with each class in the second, generating a new rule set describing the differences between the two classes. Consider two rule subsets: A and B, describing two classes within different rule sets. Each rule from A is used to cut each rule from B and the resulting set of rules describes the part of the domain described by rule set B and not by rule set A. The same process is used in reverse (set B cutting set A) to describe the part of the domain covered by set A that is not covered by set B.

The difference matrix shows rule set coverage of the differences between rules describing two classes. This is not an accurate statistic since bigger rules cover more of the domain and their differences will therefore appear more significant than smaller rules, even though their relative differences may be similar.

For example, the rules below describe the part of the domain covered by EClass (class Norm) but not by AClass (class Norm). Figure 9 shows the geometric representation of these rules in relation to the data set.

Glucose > 336 ∧ Glucose < 376 ∧
SSPG > 165 ⇒ NORM

Glucose > 503 ∧ Glucose > 557 ∧
SSPG < 204 ⇒ NORM

Glucose > 418 ∧ Glucose < 503 ∧
SSPG > 145 ∧ SSPG < 204 ⇒ NORM

Glucose > 376 ∧ Glucose < 418 ∧
SSPG > 165 ∧ SSPG < 204 ⇒ NORM

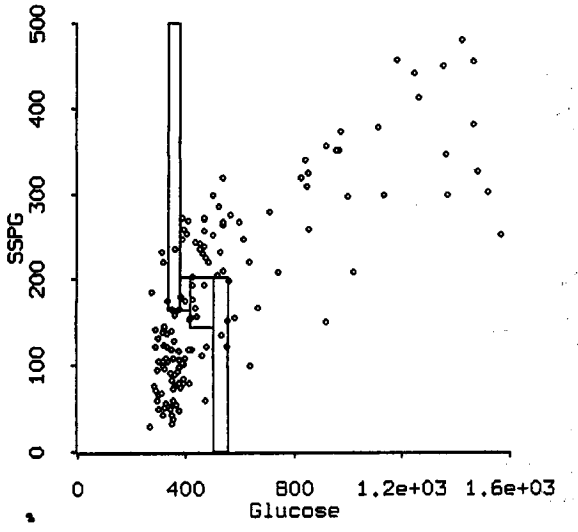

Figure 9. *Output rules.*

The EClass rules entirely encompass those of AClass, so there are no rules describing the part of the domain covered by AClass and not by EClass.

3.4.5. Correlation Matrix

The matrix provides a similarity measure, or correlation estimate, between two sets of rules. It describes the amount by which the two rule sets overlap. Each class from the first set of rules is compared to each class from the second set. Consider again two subsets of rules, A and B, each describing a class within two different sets of rules. A describes the ideal classification; Set B describes the classification to be compared against the ideal. As with the Difference Matrix, each rule from set A is compared to (cuts) each rule in set B. The hypervolume of the overlap between each pair of rules is taken as a percentage of the hypervolume of the rule from set A. This statistic indicates the proportion of the rule in set A overlapped by the one in set B. The sum of the proportions indicates the similarity between the two rule sets. Comparing a set of rules to itself will produce a correlation matrix with 100% along the main diagonal, and 0% everywhere else, indicating that each class completely overlaps itself and each class is mutually exclusive (since all the rules are mutually exclusive).

4. Results

This section presents the results produced by the three classification comparison methods described earlier—instance comparison, two sample comparison of means and geometric rule comparison. For each method, the five automatically generated classifications (EClass, AClass, AClass1, AClass2 and AClass3) are compared against the original clinical classification, CClass.

Table 3 shows the results of the instance comparison test. The numbers in each column indicate the number of examples classified differently to CClass. These values are also expressed as a percentage of the total training set of 145 instances. The rows of this table break the misclassifications down further to show what types of misclassification are occurring. For example, the row 'Norm⇒Chem' represents instances classified as Normal by CClass, but as Chemical by the automatic classifications.

Results of the two-sample t-test comparison are given in Table 4. Here the entries in the table indicate the attributes for which the t-test *failed*, for each class in the five classifications. A failed test is one where the null hypothesis was rejected, ie. the mean of the attribute is significantly different from the corresponding CClass mean, at the 95% significance level. The bottom row of this table shows the total number of rejections for each classification.

Tables 5–9 show the correlation matrix output produced by the geometric rule comparison system. The table entries indicate the amount of overlap between the two classes as a proportion of the class listed across the top of the table—in this case CClass.

5. Discussion

The correlation matrices produced by the geometric rule comparison system can be used

Difference Type	Classification EClass	AClass	AClass1	AClass2	AClass3
Norm⇒Chem	3 (2.1%)	9 (6.2%)	3 (2.1%)	5 (3.4%)	2 (1.4%)
Chem⇒Norm	10 (6.9%)	4 (2.8%)	13 (8.9%)	12 (8.3%)	19 (13.1%)
Overt⇒Norm	1 (0.7%)	0 (0.0%)	1 (0.7%)	3 (2.1%)	8 (5.5%)
Overt⇒Chem	6 (4.1%)	8 (5.5%)	4 (2.8%)	5 (3.4%)	2 (1.4%)
Total	20 (13.8%)	21 (14.5%)	21 (14.5%)	25 (17.2%)	31 (21.4%)

Table 3. *Results of classification instance comparison*

Class	Classification EClass	AClass	AClass1	AClass2	AClass3
Normal	—	—	Glucose	Glucose	Glucose, SSPG
Chemical	SSPG	—	SSPG	SSPG	Insulin, SSPG
Overt	—	—	—	—	Insulin
Total	1	0	2	2	5

Table 4. *Results of t-tests on classification means*

EClass	CClass Normal	Chemical	Overt
Normal	94.05%	31.33%	0.00%
Chemical	5.95%	68.67%	14.19%
Overt	0.00%	0.00%	85.81%

Table 5. *EClass vs. CClass correlation*

AClass	CClass Normal	Chemical	Overt
Normal	86.86%	13.62%	0.00%
Chemical	13.14%	86.38%	14.19%
Overt	0.00%	0.00%	85.81%

Table 6. *AClass vs. CClass correlation*

AClass1	CClass Normal	Chemical	Overt
Normal	94.89%	41.68%	1.64%
Chemical	5.11%	58.32%	2.76%
Overt	0.00%	0.00%	95.60%

Table 7. *AClass1 vs. CClass correlation*

AClass2	CClass Normal	Chemical	Overt
Normal	87.68%	37.20%	37.20%
Chemical	12.32%	62.80%	8.91%
Overt	0.00%	0.00%	53.89%

Table 8. *AClass2 vs. CClass correlation*

AClass3	CClass Normal	Chemical	Overt
Normal	35.25%	35.25%	8.29%
Chemical	64.75%	64.75%	20.76%
Overt	0.00%	0.00%	70.94%

Table 9. *AClass3 vs. CClass correlation*

to determine which classification rule set compares most closely to the clinical classification. Choosing the best rule set to describe the classification also depends on the particular misclassifications made by each rule set. We indicated previously that misclassifying Overt patients as Normal can result in death. Obviously it is more important to minimise these types of misclassifications rather than those from Normal to Chemical for example, where the effect of misclassification is not so important. The correlation matrices indicate AClass and EClass have 0% misclassification of Overt patients to Normal. AClass2 has a significant misclassification error of 37.2%. AClass3 has low correlation between similar classes, and significant overlap between the Normal and Chemical classes. 35.25% of Chemical diabetics are misclassified as Normal and 64.75% of Normal patients are misclassified as Chemical.

The Normal and Overt classes of AClass1 are very similar to the equivalent CClass classes, but the Chemical classes are not highly correlated between these two classifications. If misclassification of Chemical diabetics to Normal were considered unimportant, AClass1 would obviously be the best classification. However, AClass has good correlation between similar classes (all above 85%), and would be used if an acceptable

error rate over all classes were desired.

It is desirable to have a single metric that describes the similarity between two rule sets. This could perhaps be calculated as a weighted average of the correlation percentage for equivalent classes. In the tables above, equivalent classes lie on the main diagonal of the correlation matrix. If no misclassification is considered more important than any other, a weighting of 1.0 would be used for each class. Table 10 shows this metric, calculated using a weight of 1.0, for each rule set compared to the CClass rule set. This table indicates that AClass is the classification most similar to CClass, and that AClass3 is the least similar. This supports the 'intuitive' reading of the correlation matrices.

Classification	Similarity
EClass	82.84%
AClass	86.35%
AClass1	82.94%
AClass2	68.12%
AClass3	56.98%

Table 10. *Overall correlation with CClass*

The table of instance misclassifications can be used in the same way as the correlation matrix. The classification error for each type of misclassification corresponds roughly to those in the correlation matrix. For example, there are 19 misclassifications from Chemical to Normal for the AClass3 classification, giving an error rate of 13.1% (one of the highest rates in Table 2). The same misclassification in Table 7 gives an error rate of 37.2%—also one of the highest. We believe the similarity statistics in the correlation matrices are more indicative of potential misclassification error than the estimates of Table 2. The instance misclassification errors are only representative of the current data set. For large data sets this error may be close to the population error—however, small data sets are not representative of the population. The geometric rules are more indicative of classification errors because the rules are representative of the population domain—that is, it is assumed the rules will be used to classify unseen cases. Misclassifications indicated in Table 2 are not always accounted for by the rules. C4.5 reclassifies some instances when the rules are constructed. The single misclassification from the Overt class to the Normal class by EClass for example, is not represented in Table 4.

An advantage of the instance misclassifications is that the distribution of the data is inherent in the misclassifications themselves. The geometric representation of the classifications has no knowledge of the underlying distribution of the data; the similarity estimates assume a uniform distribution of each attribute across the domain.

The t-test comparison of classes is very imprecise. It imparts very little information about the quality of a classification compared with the clinical classification. AClass is indicated as a similar classification, since the means of all the variables for each class are equivalent. AClass3 obviously compares poorly to the clinical classification since the means are significantly different in all three classes—two out of three means are different in both the Normal and Chemical classes. The t-test results would indicate that AClass1 and AClass2 are comparable classifications. This is an obvious disagreement with the analysis of the geometric correlation matrices. The similarity metrics for AClass1 and AClass2 alone differ by 14.82%. 37.2% of Overt diabetics are classified as Normal by AClass2, compared with 1.64% by AClass1.

6. Conclusions

This paper has presented a new method for evaluating the quality of classifications, based on a geometric representation of class descriptions. Rule sets are produced that describe the difference between pairs of classes. Correlation matrices are used to determine the relative degree of similarity between classifications. The method has been applied to classifications generated by AUTOCLASS in a medical domain, and its evaluation compared to those of simple instance comparison and statistical methods.

The results obtained so far are encouraging. The evaluations produced by the geometric algorithm appear to correlate reasonably well with the simple instance comparison. We believe that the geometric evaluation is more useful because it reflects the performance of the classification in the real world, on unseen data. Instance based or statistical methods cannot reproduce this. However, several aspects of the technique require further experimentation and development.

The algorithm currently assumes a uniform distribution for all attributes, which in general is not valid. We plan to incorporate a mechanism for specifying the distribution of

attributes to the algorithm, or at least use 'standard' configurations such as a normal distribution.

The system is at present limited to comparing ripple-down rule sets. Ideally it would also be able to handle rule sets in which every rule has equal priority. There is a difficulty in this case with overlapping rules from different classes—unlike ripple-down rules there is no easy way to decide which rule, if any, should take priority.

Finally we plan to extend the system to handle enumerated as well as continuous attributes, and integrate it fully with the WEKA workbench.

Acknowledgements

We wish to thank Ray Littler and Ian Witten for their inspiration, comments and suggestions. Credit goes to Craig Nevill-Manning for his last minute editing. Special thanks to Andrew Donkin for his excellent work on the WEKA Workbench.

References

Cendrowska, J., 1987. "PRISM: An algorithm for inducing modular rules". *International Journal of Man-Machine Studies*. 27, 349–370.

Cheeseman, P., Kelly, J., Self, M., Stutz, J., Taylor, W. & Freeman, D., 1988. "AUTOCLASS: A Bayesian Classification System". In *Proceedings of the Fifth International Conference on Machine Learning*, 54–64. Morgan Kaufmann Publishers, San Mateo, California.

Compton, P., Edwards, G., Srinivasan, A., Malor, R., Preston, P., Kang, B. & Lazarus, L., 1992. "Ripple down rules: turning knowledge aquisition into knowledge maintenance". *Artificial Intelligence in Medicine* 4, 47–59.

Fisher, D., 1987. "Knowledge Acquisition Via Incremental Concept Clustering". *Machine Learning* 2, 139–172.

Friedman, H. & Rubin, J., 1967. "On some invariant criteria for grouping data". *Journal of the American Statistical Association* 62, 1159–1178.

Hanson, S. & Bauer, M., 1989. "Conceptual Clustering, Categorization, and Polymorphy". *Machine Learning* 3, 343–372.

Kononenko, I. & Bratko, I., 1991. "Information-Based Evaluation Criterion for Classifier's Performance". *Machine Learning* 6, 67–80.

Lebowitz, M., 1987, "Experiments with Incremental Concept Formation: UNIMEM". *Machine Learning* 2, 103–138.

McQueen, R., Neal, D., DeWar, R. & Garner, S., 1994. "Preparing and processing relational data through the WEKA machine learning workbench". Internal report, Department of Computer Science, University of Waikato, Hamilton, New Zealand.

Michalski, R. & Stepp, R., 1983. "Learning from Observation: Conceptual Clustering". In Michalski, R., Carbonell, J. & Mitchell, T. (eds.), *Machine Learning: An Artificial Intelligence Approach*, 331–363. Tioga Publishing Company, Palo Alto, California.

Mingers, J., 1989. "An Empirical Comparison of Pruning Methods for Decision Tree Induction". *Machine Learning* 4, 227–243.

Nie, N., Hull, H., Jenkins, J., Steinbrenner, K. & Bent, D., 1975. SPSS: *Statistical Package for the Social Sciences*, 2nd ed. McGraw-Hill Book Company, New York.

Ousterhout, J., 1993. *An Introduction to* TCL *and* TK. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts.

Quinlan, J., 1986. "Induction of Decision Trees". *Machine Learning* 1, 81–106.

Quinlan, J., 1992. C4.5: *Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, California.

Reaven, G., Berstein, R., Davis, B. & Olefsky, J., 1976. "Non-ketoic diabetes mellitus: insulin deficiency or insulin resistance?". *American Journal of Medicine* 60, 80–88.

Reaven, G. & Miller, R., 1979. "An Attempt to Define the Nature of Chemical Diabetes Using a Multidimensional Analysis". *Diabetologia* 16, 17–24.

Tierney, L., 1990. LISP-STAT: *An Object-Oriented Environment for Statistical Computing and Dynamic Graphics*. John Wiley & Sons, New York.

Utgoff, P., 1989. "Incremental Induction of Decision Trees". *Machine Learning* 4, 161–186.