Using A Bayesian Averaging Model for Estimating the Reliability of Decisions in Multimodal Biometrics

Carsten Maple Vitaly Schetinin
Institute for Research in Applicable Computing, University of Luton, UK
carsten.maple@luton.ac.uk vitaly.schetinin@luton.ac.uk

Abstract

The issue of reliable authentication is of increasing importance in modern society. Corporations, businesses and individuals often wish to restrict access to logical or physical resources to those with relevant privileges. A popular method for authentication is the use of biometric data, but the uncertainty that arises due to the lack of uniqueness in biometrics has lead there to be a great deal of effort invested into multimodal biometrics. These multimodal biometric systems can give rise to large, distributed data sets that are used to decide the authenticity of a user. Bayesian Model Averaging (BMA) methodology has been used to allow experts to evaluate the reliability of decisions made in data mining applications. The use of Decision Tree (DT) models within the BMA methodology gives experts additional information on how decisions are made. In this paper we discuss how DT models within the BMA methodology can be used for authentication in multimodal biometric systems.

Keywords: Identification of person, Decision-making, Bayes procedure, Stochastic approximation, Trees

1. Introduction

There is an increasing requirement to place restrictions on access to both physical and logical resources. The days where businesses had open door policies, or even people in their own home leaving back doors unlocked and open are long gone. Businesses and individuals now have richer assets than ever before and the risk of these being stolen has increased. If we consider individuals, many now have expensive electronic equipment that is easy to remove from an office or home if unauthorised access has been gained. The problem is however far worse than simply losing a replaceable asset. Modern businesses and consumers have data-rich assets that if lost can have serious ramifications.

Industrial espionage no longer requires the theft of large equipment or files, rather the theft of a database or a portable computer carrying sensitive information can often pose a much more significant threat. In order to overcome such issues, both businesses and individuals often employ some level of security. This security barrier is designed to allow only those with the correct level of authorisation into a building or a computer system.

Traditionally much of the physical security was implemented using locks and keys. Similarly the majority of computer systems were originally accessed using a text-based password, if there was any restriction at all. However, over time there has been a shift away from these traditional methods of authentication for a number of reasons.

Physical access systems have moved away from physical devices such as the lock and key to authentication through keypads (by way of a Personal Identification Number (PIN) or a General Access Code) and swipe cards (using technologies such as Radio-Frequency Identification (RFID), magnetic strips or embedded chips (smart cards)).

Conversely, access to computer systems has started to move away from passwords and PINs to the use of physical devices such as proximity tokens (utilising technologies such as RFID) and dongles.

These well-established techniques for authentication are known as token-based and knowledge-based methods and can be categorised as shown in Table 1.

Table 1. Classification of authentication techniques.

Classification	Example
Biometrics-based	
- static	Physical attribute, e.g. fingerprint, iris scan
- dynamic	Behavioral attribute, e.g. keystroke dynamics, signature dynamics
Token-based	Something you possess, e.g. swipe card, key
Knowledge-based	Something you know, e.g. password, PIN

This table also features the two forms of biometric technique that can be used for authentication: static and dynamic. Basically these can be thought of as things you are and things you do. These biometric techniques are used to overcome some of the weaknesses of token-based

Proceedings of the First International Conference on Availability, Reliability and Security (ARES'06)
0-7695-2567-9/06 $20.00 © 2006 IEEE

and knowledge-based methods. Often a combination of biometrics is used for authentication. However the mixing of biometric data coming from different sources makes the estimation of uncertainty in decisions a difficult problem. For this reason there is a requirement for an effective decision-making process [1, 2]. In this paper we discuss the use of Decision Trees (DTs) within the Bayesian Model Averaging (BMA) methodology [3, 4] as a tool to assist in making reliable decisions on whether to authenticate a user.

Practically, the BMA can be implemented on the base of a Markov Chain Monte Carlo (MCMC) approximation technique based on the random sampling from the posterior distribution [5, 6]. For real-world applications when the dimensionality of a model cannot be predefined, the MCMC technique has been extended by Reversible Jumps (RJ) introduced in [7]. It is also important to note that when domain experts cannot give the priors on structure and parameters of DTs, the required priors can be defined implicitly within a sweeping strategy of the Bayesian DT averaging suggested in [8].

In our experiments we compare the performance of the existing and suggested BMA techniques on some data sets taken from the UCI Machine Learning Repository [9]. The classification reliability is compared within an Uncertainty Envelope technique dealing with the class posterior distribution and a given confidence probability described in [10]. This technique provides realistic estimates of the reliability, which can be interpreted in statistical terms [11]. Using such an evaluation technique in our comparative experiments, we find that the Bayesian DT technique with the sweeping strategy is superior to the existing RJ MCMC technique.

Section 2 of the paper introduces the problem of biometric authentication, and then sections 3 and 4 present the bases of BMA and RJ MCMC techniques. In section 5 we describe the idea of Bayesian DT technique with a sweeping strategy and then in section 6 briefly describe the Uncertainty Envelope technique used in our comparative experiments. The experimental results are presented in section 7, and section 8 concludes the paper.

2. Biometric Authentication

Biometric authentication methods are becoming increasingly widespread. Popular techniques involving static biometrics include:

- Facial Recognition Systems – both well-established 2D systems and the recently developed 3D systems such as that from A4Vision [12]
- Retina Scanning Systems
- Iris Scanning Systems
- Fingerprint Analysis Systems
- DNA Analysis
- Facial Thermogram
- Hand Geometry
- Vascular patterns – Fujitsu are leading research on a non-intrusive method for authentication that relies on a near infrared beam being directed at the palm. The haemoglobin in the blood absorbs the IR rays that give rise to an image of the vascular pattern within the palm
- Palmprint
- Ear Shape

In addition to these methods involving static biometrics, the use of dynamic biometrics, while currently not as mature, are starting to gain popularity. The major methods for dynamic biometric authentication are:

- Gait
- Voice Recognition
- Keystroke Dynamics
- Signature Recognition Systems – these have become increasingly sophisticated, and involve matching the pressure, speed, characteristics and overall image of a signature

Multimodal biometric authentication systems use multiple applications to capture different types of biometrics and then aggregate this data in order to make a decision whether to authenticate or not. For example, a multimodal system might involve information from sensors detecting gait. As a user approaches a physical resource, such as a computer laboratory, cameras can capture the motion and store the data digitally. The user may then have to say a particular phrase into a microphone outside the door to the laboratory and another camera can be used to capture an image of the users face. The data from all three sensors is then used in the decision-making process.

Having gained access into the facility, the user may be required to present a fingerprint, say, in order to gain entry to the logical resources. The earlier data from the three sensors can now be used to inform the level of tolerance required in the fingerprint match. If a user is then permitted entry to the system, keystroke dynamics might be used to verify that only the authorised user is accessing the system at any point in time. Further to being able to use the data from a number of sensors to decide upon authenticity, multimodal biometric authentication systems also have the advantage that they can compensate for missing or poor data from a particular sensor. In the case where a sensor is a camera, it may be that some dust or foreign body has corrupted the view through that camera. This degradation in the data can be partially overcome by considering the data from the other sensors. Had this been the sole capture device, problems would have arisen that required immediate attention.

Problems regarding the quality or consistency of the capture of biometric data may not necessarily be due to a fault or error in the sensor. It is estimated that 5% of the population does not have fingerprints that are legible. The data gathered through the microphone may not match closely enough to the stored template if the user has a cold or is short of breath due to running along the

corridor. Again if either of these sensors was the sole data capture device, obvious problems would ensue. In a multimodal biometric authentication system such issues are less of a problem.

The use of such varied sources of data obviously makes the estimate of uncertainty in decisions a difficult problem. To tackle this problem we suggest the use of the BMA methodology described next.

3. The Bayesian Model Averaging Methodology

Nowadays the methodology of BMA is widely used for estimating the reliability of decisions. Domain experts responsible for making reliable decisions are also interested in interpretability of decision models. For this reason DT providing a graphical presentation of decisions is an attractive model for the experts [1-6]. The main idea of using DT models is to recursively partition data points in an axis-parallel manner. Such models provide natural feature selection and uncover the features which make the important contribution to the outcome. The resultant DT models can be easily understood by experts.

By definition, DTs consist of splitting and terminal nodes, which are also known as tree leaves. DTs are said to be binary if the splitting nodes ask a specific question and then divide the data points into two disjoint subsets, say the left or the right branch. Note that the number of the data points in each split should not be less than that predefined by a user. The terminal node assigns all data points falling in that node to a class of majority of the training data points residing in this terminal node. Within a Bayesian framework, the class posterior distribution is estimated for each terminal node [5, 6].

The required estimates can be achieved on the base of Bayesian MCMC methodology of sampling from the posterior distribution [4-6]. This technique has revealed promising results when applied to some real-world problems.

The MCMC methodology has been extended by Reversible Jumps (RJ) in order to deal with models of a variable dimensionality [7]. The RJ MCMC technique making such moves as *birth* and *death* allows the DTs to be induced under the priors given on the shape or size of the DTs. However, making such moves, the RJ MCMC should keep the balance between the birth and death moves in order to obtain the unbiased estimates of the posterior [5-8].

Within the RJ MCMC technique the proposed moves for which the number of data points falling in one of splitting nodes becomes less than the given number are assigned unavailable. Obviously the priors given on the DTs are dependent on the shape of class boundaries as well as on the level of noise in training data. Therefore the lack of *a priori* information can cause the overfitting of DTs and, as a consequence, the bias in the desired class posterior estimates [13].

Moreover the standard RJ MCMC technique of averaging over DTs cannot keep the balance between the death and birth moves. This happens because within the RJ MCMC some proposed DTs which cannot provide the given number of data points allowed being in the splitting nodes are assigned unavailable [8].

When *a priori* information of the favourite shape of DTs is unavailable, the Bayesian DT technique with a sweeping strategy has revealed a better performance [8]. Within this strategy the prior given on the number of DT nodes is defined implicitly and dependent on the given number of data points allowed being at the DT splits. So the sweeping strategy gives more chances to induce the DTs containing a near optimal number of splitting nodes required to provide the best generalisation. At the same time the number of data points allowed to be in the splitting nodes can be reasonably reduced without increasing the risk of overfitting the DTs.

4. The Bayesian Decision Tree Technique

For a classification model given with vector of parameters $\boldsymbol{\theta}$, the predictive distribution we are interested is written as an integral over the parameters $\boldsymbol{\theta}$

$$p(y \mid \mathbf{x}, \mathbf{D}) = \int_{\theta} p(y \mid \mathbf{x}, \boldsymbol{\theta}, \mathbf{D}) p(\boldsymbol{\theta} \mid \mathbf{D}) d\boldsymbol{\theta}$$

where y is the predicted class (1, …, C), $\mathbf{x} = (x_1, \ldots, x_m)$ is the m-dimensional input vector, and $\mathbf{D}$ are the given training data.

This integral can be analytically calculated only in simple cases, and in practice part of the integrand, which is the posterior density of $\boldsymbol{\theta}$ conditioned on the data $\mathbf{D}$, $p(\boldsymbol{\theta} \mid \mathbf{D})$, cannot usually be evaluated. However if values $\boldsymbol{\theta}^{(1)}$, …, $\boldsymbol{\theta}^{(N)}$ are the samples drawn from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{D})$, we can write

$$p(y \mid \mathbf{x}, \mathbf{D}) \approx \sum_{i=1}^{N} p(y \mid \mathbf{x}, \boldsymbol{\theta}^{(i)}, \mathbf{D}) p(\boldsymbol{\theta}^{(i)} \mid \mathbf{D})$$

$$= \frac{1}{N} \sum_{i=1}^{N} p(y \mid \mathbf{x}, \boldsymbol{\theta}^{(i)}, \mathbf{D})$$

This is the basis of the MCMC technique for approximating integrals [6]. To perform such an approximation, we need to run a Markov Chain until it has converged to a stationary distribution. Then we can collect N random samples from the posterior $p(\boldsymbol{\theta}|\mathbf{D})$ to calculate the desired predictive posterior density.

Let us define a classification problem presented by data $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$, where n is the number of data points, and $y_i \in \{1, \ldots, C\}$ is a categorical response. Using DTs for the classification, we need to find the probability φ_{tj} with which an input $\mathbf{x}$ is assigned by terminal node $t = 1, \ldots, k$ to the jth class, where k is the number of terminal nodes in the DT. Initially we can assign a Dirichlet prior for each terminal node.

The DT parameters are defined as $\boldsymbol{\theta} = (s_i^{\text{pos}}, s_i^{\text{var}}, s_i^{\text{rule}})$, $i = 1, \ldots, k - 1$, where s_i^{pos}, s_i^{var} and s_i^{rule} define the

position, *predictor* and *rule* of each splitting node, respectively. For these parameters the priors can be specified as follows. First we can define a maximal number of splitting nodes, say, $s_{\max} = n - 1$. Second we draw any of the m predictors from a uniform discrete distribution $U(1, \ldots, m)$ and assign $s_i^{\text{var}} \in \{1,..., m\}$. Finally the candidate value for the splitting variable $x_j = s_i^{\text{var}}$ can be drawn from a discrete distribution $U(x_j^{(1)}, \ldots, x_j^{(L)})$, where L is the number of possible splitting rules for variable x_j, either categorical or continuous.

Such priors allow us to explore DTs which split data in as many ways as possible. However the DTs with different numbers of splitting nodes should be explored in the same proportions [6].

For a case when there is knowledge of the favoured structure of the DT, we can define a prior by assuming the *a priori* probability of further splits to be dependent on how many splits have already been made above them [6]. For example, for the ith terminal node the probability of its splitting is written as

$$p_{split}(i) = \gamma(1 + d_i)^{-\delta},$$

where d_i is the number of splits made above i and γ, $\delta \geq 0$ are given constants. The larger δ, the more the prior favours "bushy" trees. For $\delta = 0$ each DT with the same number of terminal nodes appears with the same *a priori* probability.

To sample DTs of a variable dimensionality, the MCMC technique exploits the Reversible Jump extension [7]. To implement the RJ MCMC technique, Chipman *et al.* [5] and Denison *et al.* [6] have suggested exploring the posterior probability by using the following types of moves.

Birth. Randomly split the data points falling in one of the terminal nodes by a new splitting node with the variable and rule drawn from the corresponding priors.

Death. Randomly pick a splitting node with two terminal nodes and assign it to be one terminal with the united data points.

Change-split. Randomly pick a splitting node and assign it a new splitting variable and rule drawn from the corresponding priors.

Change-rule. Randomly pick a splitting node and assign it a new rule drawn from a given prior.

The first two moves, *birth* and *death*, are reversible and change the dimensionality of θ. The remaining moves provide jumps within the current dimensionality of θ. Note that the *change-split* move is included to make "large" jumps which potentially increase the chance of sampling from a maximal posterior whilst the *change-rule* move does "local" jumps.

The RJ MCMC technique starts drawing samples from a DT consisting of one splitting node whose parameters were randomly assigned within the predefined priors. So we need to run the Markov Chain while a DT grows and its likelihood is unstable. This phase is called *burn-in* and it should be preset enough long in order to stabilize the Markov Chain. When the Markov Chain becomes stable enough, we can start sampling. This phase is called *post burn-in*.

It is important to note that the DTs grow very quickly during the first burn-in samples. This happens because an increase in log likelihood value for the birth moves is much larger than that for the others. For this reason almost every new partition of data is accepted. Once a DT has grown the *change* moves are accepted with a very small probability and, as a result, the MCMC algorithm tends to get stuck at a particular DT structure instead of exploring all possible structures.

The size of DTs can rationally decrease by defining a minimal number of data points, p_{min}, allowed to be in the splitting nodes [2-6]. If the number of data points in new partitions made after the birth or change moves becomes less than a given number p_{min}, such moves are assigned unavailable, and the RJ MCMC algorithm resamples such moves.

However, when the moves are assigned unavailable, this distorts the proposal probabilities p_b, p_d, and p_c given for the birth, death, and change moves, respectively. The larger the DT, the smaller the number of data points falling in the splitting nodes, and correspondingly the larger is the probability with which moves become unavailable. Resampling the unavailable moves makes the balance between the proposal probabilities biased as described in [8].

Because DTs are hierarchical structures, the changes at the nodes located at the upper levels can significantly change the location of data points at the lower levels. For this reason there is a very small probability of changing and then accepting a DT split located near a root node. Therefore the RJ MCMC algorithm collects the DTs in which the splitting nodes located far from a root node were changed. These nodes typically contain small numbers of data points. Subsequently, the value of log likelihood is not changed much, and such moves are frequently accepted. As a result, the RJ MCMC algorithm cannot explore a full posterior distribution properly.

One way to extend the search space is to restrict DT sizes during a given number of the first burn-in samples as described in [6]. Indeed, under such a restriction, this strategy gives more chances of finding DTs of a smaller size which could be competitive in term of the log likelihood values with the larger DTs. The restricting strategy, however, requires setting up in an *ad hoc* manner the additional parameters such as the size of DTs and the number of the first burn-in samples. Unfortunately, in practice, it often happens that after the limitation period the DTs grow quickly again and this strategy does not improve the performance.

Alternatively to the above approach based on the explicit limitation of DT size, the search space can be extended by using a restarting strategy as Chipman *et al.* have suggested in [5]. Clearly, both these strategies cannot guarantee that most of DTs will be sampled from a model space region with a maximal posterior. In the next section we describe our approach based on sweeping the DTs.

5. The Bayesian Averaging with a Sweeping Strategy

In this section we describe our approach to decreasing the uncertainty of classification outcomes within the Bayesian averaging over DT models. The main idea of this approach is to assign the *a priori* probability of further splitting DT nodes to be dependent on the range of values within which the number of data points will be not less than a given number of points, p_{min}. Such a prior is explicit because at the current partition the range of such values is unknown.

Formally, the probability $P_s(i, j)$ of further splitting at the ith partition level and variable j can be written as

$$P_s(i, j) = \frac{x_{\max}^{(i,j)} - x_{\min}^{(i,j)}}{x_{\max}^{(1,j)} - x_{\min}^{(1,j)}}, \qquad (1)$$

where $x_{\min}^{(i,j)}$ and $x_{\max}^{(i,j)}$ are the minimal and maximal values of variable j at the ith partition level.

Observing this prior, we can see that $x_{\max}^{(i,j)} \le x_{\max}^{(1,j)}$ and $x_{\min}^{(i,j)} \ge x_{\max}^{(1,j)}$ for all the partition levels $i > 1$. On the other hand there is partition level k at which the number of data points becomes less than a given number p_{min}. Therefore, we can conclude that the *a priori* probability of splitting P_s ranges between 0 and 1 for any variable j and the partition levels i: $1 \le i < k$.

From (1) it follows that for the first level of partition, probability P_s is equal to 1.0 for any variable j. Let us now assume that the first partition split the original data set into two non-empty parts. Each of these parts contains less data points than the original data set, and consequently for the (i = 2)th partition either $x_{\max}^{(i,j)} < x_{\max}^{(1,j)}$ or $x_{\min}^{(i,j)} > x_{\max}^{(1,j)}$ for new splitting variable j. In any case, the numerator in (1) decreases, and probability P_s becomes less than 1.0. We can see that each new partition makes values of the numerator and consequently the probability (1) smaller. So the probability of further splitting nodes is dependent on the level i of partitioning of the data set.

The above prior favors splitting the terminal nodes which contain a large number of data points. This is clearly a desired property of the RJ MCMC technique because it allows accelerating the convergence of the Markov Chain. As a result of using prior (1), the RJ MCMC technique of sampling DTs can explore an area of a maximal posterior in more detail.

However, prior (1) is dependent not only on the level of partition but also on the distribution of data points in the partitions. Analyzing the data set at the ith partition, we can see that value of probability P_s is dependent on the distribution of these data. For this reason the prior (1) cannot be implemented explicitly without the estimates of the distribution of data points in each partition.

To make the birth and change moves within prior (1), the new splitting values $s_i^{\text{rule,new}}$ for the ith node and variable j are assigned as follows. For the birth and change-split moves the new value $s_i^{\text{rule,new}}$ is drawn from a uniform distribution: $s_i^{rule,new} \sim U(x_{\min}^{1,j}, x_{\max}^{1,j})$.

The above prior is "uninformative" and used when no information on preferable values of s_i^{rule} is available. As we can see, the use of a uniform distribution for drawing new rule $s_i^{\text{rule,new}}$, proposed at the level $i > 1$, can cause the partitions containing fewer data points than p_{min}. However, within our technique such proposals can be avoided.

For the change-split moves, drawing $s_i^{\text{rule,new}}$ follows after taking new variable $s_i^{\text{var,new}}$: $s_i^{var,new} \sim U\{S_k\}$, where $S_k = \{1, \ldots, m\} \backslash s_i^{\text{var}}$ is the set of features excluding variable s_i^{var} currently used at the ith node.

For the change-rule moves, the value $s_i^{\text{rule,new}}$ is drawn from a Gaussian with a given variance σ_j: $s_i^{rule,new} \sim N(s_i^{rule}, \sigma_j)$, where $j = s_i^{\text{var}}$ is the variable used at the ith node.

Because DTs have hierarchical structure, the change moves (especially change-split moves) applied to the first partition levels can heavily modify the shape of the DT, and as a result, its bottom partitions can contain fewer data points than p_{min}. As mentioned in section 4, within the Bayesian DT techniques [5, 6] such moves are assigned unavailable.

Within our approach after birth or change moves there arise three possible cases. In the first case, the number of data points in each new partition is larger than p_{min}. The second case is where the number of data points in one new partition is larger than p_{min}. The third case is where the number of data points in two or more new partitions is larger than p_{min}. These three cases are processed as follows.

For the first case, no further actions are taken, and the RJ MCMC algorithm runs as usual.

For the second case, the node containing an unacceptable number of data points is removed from the resultant DT. If the move was of birth type, then the RJ MCMC resamples the DT. Otherwise, the algorithm performs the death move.

For the last case, the RG MCMC algorithm resamples the DT.

As we can see, within our approach the terminal node, which after making the birth or change moves contains fewer than p_{min} data points, is removed from the DT. Clearly, removing such unacceptable nodes turns the random search in a direction in which the RJ MCMC algorithm has more chances to find a maximum of the posterior amongst shorter DTs. As in this process the unacceptable nodes are removed, we named such a strategy *sweeping*.

After a change move the resultant DT can contain more than one node splitting fewer than p_{min} data points. However this can happen at the beginning of the burn-in phase, when the DTs grow, and this is unlikely to happen, when the DTs have grown.

Next we describe the Uncertainty Envelope technique suggested to estimate the classification uncertainty of multiple classifier systems.

6. The Uncertainty Envelope Technique

In general, the Bayesian DT strategies allow sampling the DTs induced from data independently. In such a case, we can naturally assume that the inconsistency of the classifiers on a given datum **x** is proportional to the uncertainty of the DT ensemble. Let the value of class posterior probability $P(c_j|\mathbf{x})$ calculated for class c_j be an average over the class posterior probability $P(c_j|K_i, \mathbf{x})$ given on classifier K_i:

$$P(c_j \mid \mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} P(c_j \mid K_i, \mathbf{x}),$$

where N is the number of classifiers in the ensemble.

As classifiers $K_1, \ldots, K_N$ are independent of each other and their values $P(c_j|K_i, \mathbf{x})$ range between 0 and 1, the probability $P(c_j|\mathbf{x})$ can be approximated as follows

$$P(c_j \mid \mathrm{x}) \approx \frac{1}{N}\sum_{i=1}^{N} I(y_i, t_i \mid \mathbf{x}),$$

where $I(y_i, t_i)$ is the indicator function assigned to be 1 if the output y_i of the ith classifier corresponds to target t_i, and 0 if it does not.

It is important to note that the right side of the above equation can be considered as a *consistency* of the outcomes of the DT ensemble. Clearly, values of the consistency,

$$\gamma = \frac{1}{N}\sum_{i=1}^{N} I(y_i, t_i \mid \mathbf{x}),$$

lie between $1/C$ and 1.

So we can conclude that the classification confidence of an outcome is characterized by the consistency of the DT ensemble calculated on a given input **x**. Clearly, the values of γ are dependent on how representative the training data are, what classification scheme is used, how well the classifiers were trained within a classification scheme, how close the datum **x** is to the class boundaries, how the data are corrupted by noise, and so on.

From the above consideration, we can assume that there is some value of consistency γ_0 for which the classification outcome is confident, that is the probability with which a given datum **x** could be misclassified is small enough to be acceptable. Given such a value, we can now specify the uncertainty of classification outcomes in statistical terms. The classification outcome is said to be *confident and correct*, when the probability of misclassification is acceptably small and $\gamma \geq \gamma_0$.

Additionally to the confident and correct output, we can specify a *confident but incorrect* output referring to a case when almost all the classifiers assign a datum **x** to a wrong class whilst $\gamma \geq \gamma_0$. Such outcomes tell us that the majority of the classifiers fail to classify a datum **x** correctly. The confident but incorrect outcomes can happen for different reasons, for example, the datum **x** could be mislabelled or corrupted, or the classifiers within a selected scheme cannot distinguish the data **x** properly.

The remaining cases for which $\gamma < \gamma_0$ are regarded as *uncertain classifications*. In such cases the classification outcomes cannot be accepted with a given confidence probability γ_0 and the DT ensemble labels them as uncertain.

The above three characteristics, the confident and correct, confident but incorrect, and uncertain outcomes, seem to provide a practical way of evaluating different types of DT ensembles on the same data sets. Comparing the ratios of the data points assigned to be one of these three types of classification outcomes, we can quantitatively evaluate the classification uncertainty of the DT ensembles. Depending on the costs of types of misclassifications in real-world applications, the value of the confidence consistency γ_0 should be given, say, equal to 0.99.

Next we use the Uncertainty Enveloped technique to compare the performance of the existing and proposed Bayesian RJ MCMC techniques on some real world data sets.

7. Experiments

Table 2 lists the characteristics of the Image and Satimage data sets, taken from the UCI Repository [9], which are used in our experiments; here *m*, *train*, and *test* are the numbers of classes, input variables, training and test examples, respectively. For both domain problems, the number of classes $C = 7$.

Table 2. UCI domain problems.

#	Data	Data Characteristics			
		C	*m*	*train*	*Test*
1	*Image*	7	19	210	2100
2	*Satimage*	7	36	4435	2000

First we applied the Bayesian DT technique with the restarting strategy, described in [5], running 2000 samples for burn-in and 2000 post burn-in samples 50 times. The value of p_{min} was set equal to 5 for the Image data and 20 for the Satimage data.

Table 3 shows the performance of this technique. The Table 3 also provides the quantitative evaluations of uncertainty in classification outcomes calculated within the Uncertainty Envelope technique.

Table 3. Performances of Bayesian DTs with the restarting strategy.

#	Data	Perform,	Uncertainty Envelope, %		
		%	Correct	Uncertain	Incorrect
1	*Image*	**94.3**	22.4	77.6	0.0
2	*Satimage*	**87.9**	39.7	59.7	0.5

The performance of the Bayesian DT technique with the sweeping strategy is shown in Table 4 which also provides the quantitative evaluations of the uncertainty.

Comparing the above Bayesian DT techniques, we can see that on the Image and Satimage data, the restarting strategy slightly outperforms the sweeping strategy.

Table 4. Performances of Bayesian DTs with the sweeping strategy.

# Data	Perform %	Uncertainty Envelope, %		
		Correct	Uncertain	Incorrect
1 *Image*	93.0	**63.1**	36.4	0.5
2 *Satimage*	86.5	**65.1**	32.8	2.1

However, the comparison in the terms of the classification uncertainty shows us that the proposed sweeping strategy significantly outperforms the restarting strategy. On both domain problems sure correct classification rates of the sweeping strategy are higher that those of the restarting strategy. An explanation of this is that the Bayesian DT technique with the sweeping strategy is able to find more, shorter DTs than those of the technique with the restarting strategy. Clearly, for a shorter DT, the classification uncertainty is smaller. Thus we can conclude that within the sweeping strategy the Bayesian DT technique can provide more stable results in terms of the classification uncertainty.

It is important to note also that the Bayesian DTs sampled within the sweeping strategy always have a smaller proportion of uncertain classifications. Certainly it is an important property for authentication systems.

8. Conclusion

In this paper we have discussed issues surrounding the effective authentication of users based upon biometrics. Using multimodal biometrics gives rise to large heterogeneous data sets that require fast effective searching for decision-making.

The use of the RJ MCMC methodology of stochastic sampling from the posterior distribution makes Bayesian DT techniques effective for applications in which risk evaluation is of crucial importance. Existing techniques, exploring the space of DTs parameters, may prefer sampling DTs from the local maxima of the posterior instead of properly representing the posterior. This affects the evaluation of the posterior distribution and, as a result, causes an increase in the decision uncertainty. This negative effect can be reduced by averaging the DTs obtained in different starts or by restricting the size of DTs during the burn-in phase.

As an alternative way of reducing the classification uncertainty, we have suggested the Bayesian DT technique exploiting the sweeping strategy. Within this strategy, DTs are modified after birth or change moves by removing the splitting nodes containing fewer data points than acceptable.

References

[1] R. Duda, P. Hart, and D. Stork, *Pattern classification*, Wiley, New York, 2001.

[2] L. Kuncheva, *Combining pattern classifiers: Methods and algorithms*, Wiley, 2004.

[3] L. Breiman, J. Friedman, R. Olshen and C. Stone, *Classification and regression trees*. Belmont, CA: Wadsworth, 1984.

[4] W. Buntine, "Learning classification trees", *Artificial Intelligence Frontiers in Statistics*, D. Hand (ed.), Chapman & Hall, London, 1993, pp. 182-201.

[5] H. Chipman, E. George, and R. McCullock, "Bayesian CART model search", *J. American Statistics*, 93, ASA, CA: Alexandria, 1998, pp. 935-960.

[6] D. Denison, C. Holmes, B. Malick, and A. Smith, *Bayesian methods for nonlinear classification and regression*. Wiley, 2002.

[7] P. Green, "Reversible jump Markov Chain Monte Carlo computation and Bayesian model determination", *Biometrika*, 82, Oxford University Press, Oxford, 1995, pp. 711-732.

[8] V. Schetinin, J.E. Fieldsend, D. Partridge, W.J. Krzanowski, R.M. Everson, T.C. Bailey, and A. Hernandez, "The Bayesian decision tree technique with a Sweeping Strategy", *Proceedings of the Int. Conference on Advances in Intelligent Systems - Theory and Applications* (AISTA 2004) *in cooperation with IEEE Computer Society,* IEEE Computer Society Press, Luxembourg, 2004, ISBN 2-9599776-8-8.

[9] C.L. Blake and C.J. Merz, *UCI Repository of machine learning* data *set*, Irvine, University of California, 1998. Available at http://www.ics.uci.edu/~mlearn/MLRepo-sitory

[10] J.E. Fieldsend, T.C. Bailey, R.M. Everson, W.J. Krzanowski, D. Partridge, and V. Schetinin, "Bayesian inductively learned modules for safety critical systems", *Proceedings of the 35th Symposium on the Interface*, *Computing Science and Statistics*, Interface Foundation, Salt Lake City, 2003, pp. 110-125.

[11] V. Schetinin, D. Partridge, W.J. Krzanowski, R.M. Everson, J.E, Fieldsend, T.C. Bailey, and A. Hernandez, "Experimental comparison of classification uncertainty for randomized and Bayesian decision tree ensembles", *Proceedings of the Int Conference on Intelligent Data Engineering and Automated Learning (IDEAL'04)*, Z.R. Yang, H. Yin, and R. Everson (eds), Lecture Notes in Computer Science, Springer, 2004, pp. 726-732.

[12] "The Applications for Vision", A4Vision, 2006. Available from http://www.A4vision.com

[13] P. Domingos, "Bayesian averaging of classifiers and the overfitting problem", *Proceedings of the 17 Int. Conference on Machine Learning*, P. Langley (ed), Stanford, CA, Morgan Kaufmann, 2000, pp. 223-230.

0-7695-2567-9/06 $20.00 © 2006 IEEE