Pattern Recognition Letters xxx (2010) xxx–xxx

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Predicting incomplete gene microarray data with the use of supervised learning algorithms

Bhekisipho Twala *, Motee Phorah

Modelling and Digital Sciences, Council of Scientific and Industrial Research (CSIR), Digital Intelligence Research Group, P.O. Box 395, Pretoria 0001, South Africa

ARTICLE INFO

Article history:
Received 14 October 2009
Available online xxxx
Communicated by T. Vasilakos

Keywords:
Supervised learning
Microarray data
Incomplete data
Prediction

ABSTRACT

Motivation: With the wealth of sequence data and the huge amount of data generated from molecular technologies, the issue of gene classification/prediction has become a central challenge in the field of microarray data analysis. This has led to the application of many well-established supervised learning (SL) algorithms in an attempt to provide more accurate and automatic diagnosis class (cancer/non cancer) prediction. Virtually all research on SL addresses the task of learning to classify complete domain instances. However, in some research situations we often have to classify instances given incomplete vectors, which can affect the predictive accuracy of learned classifiers. The task of learning an accurate incomplete data classifier from instances raises a number of new issues some of which have not been properly addressed by bioinformatics research. Thus, an effective missing value estimation method is required for improving predictive accuracy.

Results: The essence of the approach is the proposal that prediction using supervised learning can be improved in probabilistic terms given incomplete microarray data. This imputation approach is based on the a priori probability of each value determined from the instances at that node of a decision tree (PDT) that have specified values. The proposed approach exploits the total probability and Bayes' theorems and it has three versions. We evaluate our approach with other supervised learning techniques including C5.0, classification and regression trees (CART), *k*-nearest neighbour (*k*-NN), linear discrimination (LD) naïve Bayes classifier (NBC), Repeated Incremental Pruning to Produce Error Reduction (RIPPER) and support vector machines (SVMs), from the point of view of their effect or tolerance of incomplete test data. Eight cancer related gene expression datasets are utilized for this task. Experimental results are provided to illustrate the efficiency and the robustness of the proposed algorithm.

1. Introduction

It is generally accepted that the highest accuracy results that a SL system can achieve depend on the quality of data and the appropriate selection of a learning algorithm for the data. One of the central tasks of SL algorithms is classifying instances from some domain of application, i.e., determining whether a particular instance belongs to a specified class, given a description of that instance. The wealth and complexity of microarray data lends itself well to the application of classifier or SL methods for prediction or classification of prognosis of diseases according to their gene expression signatures as measured by microarrays.

Virtually all research on supervised learning addresses the task of learning to classify complete domain instances (Osareh and Shadgar, 2009). However, in some research situations we often have to classify instances given incomplete data vectors. The frequency of poor data quality is one of the most vexing problems for functional genomics and bio-medical researchers, especially those dealing with microarray gene expression data. Incomplete microarray data could be caused by administrative error, defective technique, or technology failure. For example, an intended replication may be omitted, or a feature of the robotic apparatus may fail. A scanner may have insufficient resolution, or an image may be corrupted. Another complication could be project managers who flatly refuse to participate in the study. Some researchers follow the practice of flagging readings that are suspect, and these may be converted to missing values or otherwise excluded from the analysis before proceeding. For instance, spots with dust particles, irregularities or other bad features may be flagged manually. Spots may be flagged as 'absent' or 'feature not found' when nothing is printed in the location of a spot. Expression readings are barely above the background correction (using a criterion such as less than two background standard deviations above may also be flagged.

One primary concern of classifier learning is prediction accuracy. Recent research has shown that missing values in either the training data or test (unseen) data affect prediction accuracy of

* Corresponding author. Tel.: +27 (0) 12 841 4711; fax: +27 (0) 12 841 3550.
E-mail address: btwala1@csir.co.za (B. Twala).

0167-8655/$ - see front matter © 2010 Published by Elsevier B.V.
doi:10.1016/j.patrec.2010.05.006

learned classifiers (Quinlan, 1993). The task of learning an accurate incomplete data classifier from instances raises a number of new issues some of which have not been properly addressed by bioinformatics research. First, the types of processes that can cause an instance to have missing attribute values have to be considered. For example, whether this omission is randomly missing, uninformative, partially informative, or even misleading. Second, classification or prediction on incomplete data versus training on the artificially completed instances can also be considered. Intuitively, complete data give the learner more information about each instance, and hence, should make classification easier.

Robustness has a two fold meaning in terms of dealing with missing values in supervised learning. The toleration of missing values in the training set is one, and the toleration of missing data in the test (validation) set is the other. For the training set, both attribute values and/or class labels could be missing, while for the test set, only attribute values could be missing. When missing features are encountered in the test set, some *ad hoc* approaches of listwise deletion or imputation have been utilized by biomedical researchers to form a complete-data format. Some researchers have used supervised learning imputation for handling incomplete data. For purposes of this paper we are assuming that the class labels are not missing; only attribute values in the test set are considered as missing.

Although the problem of incomplete data has been treated adequately in various real world datasets, there are rather few published works or empirical studies in biomedical research concerning the task of assessing learning and classification accuracy with incomplete data using supervised ML algorithms such as DTs. In fact most of the biomedical studies have focussed on developing missing value estimation methods for incomplete microarray data (Troyanskaya et al., 2001; Walszak and Massart, 2001; Zhou et al., 2003; Oba et al., 2003; Bø et al., 2004; Kim et al., 2004; Nguyen et al., 2004; Kim et al., 2005; Sehgal et al., 2005; Gan et al., 2006; Williams et al., 2007; Brás and Menezes, 2007; Tuikkala et al., 2008; Zhang et al., 2008; García-Laencina et al., 2009) than developing techniques for prediction or classification using incomplete microarray data. Other researchers have focussed on classification of incomplete data in other fields like data mining or knowledge discovery (Hawarah et al., 2006; 47; 36; Farhangfar et al., 2008; Saar-Tsechansky and Provost, 2007; Twala et al., 2008; Twala, 2009; Branden and Verboven, 2009). To this end this paper provides:

- The largest number of popular and modern classifiers, namely, C4.5 (Quinlan, 1993), *k*-NN (Hand, 1997), LD (Fisher, 1936; Hand, 1997), NBC (Michie et al., 1994), CART (Breiman et al., 1984), RIPPER (Cohen, 1996), SVMs (Vapkin, 1995);
- The range of two missing data patterns and three missing data mechanisms for consistent amounts of missing data (5%, 10%, 20% 35% and 50%) for all datasets;
- The largest number of datasets – eight datasets ranging between 22 and 308 instances, 200 and 15,154 attributes, and 2 and 26 classes.

The purpose of this paper is to develop probabilistic methods for classifying incomplete test data using DTs, i.e. methods that could be used to handle incomplete software project test data. This approach is based on the *a priori* probability of each value determined from the instances at that node that have specified values. The missing attribute values can be either continuous or nominal. For purposes of this study, we assume that training data has no missing values. The proposed method follows the total probability and Bayes' theorems (Bernado and Smith, 1994) and it has three versions. We note that although some of these classifiers including C4.5 have their own internal approaches of handling unknown attribute values; it is not clear how they would react to external imputation methods.

The following section presents details of eight supervised learning techniques that are used in this paper. The framework of the proposed probabilistic method is introduced and described in Section 3. Section 4 presents related work. Section 5 empirically evaluates the robustness and accuracy of the new technique in comparison with on eight microarray domains. We close with a discussion and conclusions, and then directions for future research.

2. Missing data patters and mechanisms

The two most common tasks when dealing with incomplete data is to investigate the pattern (which values are missing) and the law generating the missing values (whether missingness is related to the study variables). When missing values are confined to a single variable we have a univariate pattern; monotonic pattern occurs if a subject, say Y_j, is missing then the other variables, say $Y_{j+1}, \ldots, Y_p$, are missing as well; arbitrary patterns occur when any set of variables may be missing for any unit.

The law generating the missing values seems to be the most important task since it facilitates how the missing values could be estimated more efficiently. If data are missing completely at random (MCAR) or missing at random (MAR), we say that missingness is *ignorable* (Little and Rubin, 1987; Schafer, 1997). For example, suppose that you are modelling oral cancer as a function of a white or a red patch on the gums. There may be no particular reason why some gums had white or red patches and others did not. Such data is considered to be MCAR. Furthermore, oral cancer may not be identified or diagnosed due to a given specific type of patch on the gums. Such data are considered to be MAR. MAR essentially says that the cause of missing data (oral cancer) may be dependent on the observed data (red or white patch on the gums) but must be independent of the missing value that would have been observed. MAR is a less restrictive model than MCAR, which says that the missing data cannot be dependent on either the observed or the missing data. For data that is informative missing (IM), we have *non ignorable* missingness (Rubin, 1987; Little and Rubin, 1987), that is, the probability that oral cancer results are missing depends on the unobserved values of oral cancer themselves. For example, medical doctors may be less likely to reveal oral cancer diagnosis test results of very young or very old patients with severe symptoms of oral cancer.

3. Existing supervised learning imputation methods

Some SL methods are inherently tolerant to incomplete data and thus require mechanisms for handling missing attribute values. An overview of the current supervised learning imputation methods (their strengths and limitations) used for comparative purposes to assess the performance of PDTI is now presented.

3.1. C4.5

Fractioning of cases is a missing value strategy used for the C4.5 decision tree (DT) learning system (Quinlan, 1993). Quinlan (1993) borrows the probabilistic complex approach by Cestnik et al. (1987) by "fractioning" instances or cases (FC) based on a priori probability of each value determined from the instances at that node that have specified values. Quinlan starts by penalising the information gain measure by the proportion of unknown instances and then splits these instances to both subnodes. For classification, Quinlan's technique is to explore all branches below the node in question and then take into account that some branches are more probable than others. The weights of the instance fragments clas-

Please cite this article in press as: Twala, B., Phorah, M. Predicting incomplete gene microarray data with the use of supervised learning algorithms. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.05.006

sified in different ways at the leaf nodes of the tree are summed and then the class with the highest probability or the most probable classification is chosen. C4.5 does not consider that association or dependencies among the attributes, thus, we shall assume a MCAR mechanism.

3.2. Classification and regression trees

One other sophisticated but refined tree-based method worthy of note and study is the surrogate variable splitting (SVS), which has been used for the classification and regression trees (CART). CART handles missing values in the database by substituting "surrogate splitters". Surrogate splitters are predictor variables that are not as good at splitting a group as the primary splitter but which yield similar splitting results; they mimic the splits produced by the primary splitter; the second does second best, and so on. The surrogates are used for tree nodes when there are missing values. The CART system relies on the dependencies of the attributes when dealing with missing values. Hence, we shall assume that the mechanism generating the missingness is MAR.

3.3. k-Nearest neighbour

One of the most venerable algorithms in statistical pattern recognition is the nearest neighbour. *k*-nearest neighbour (*k*-NN) can also be considered a supervised learning algorithm where the result of a new instance query is classified on majority of *k*-nearest neighbour category. Of late, such an algorithm has become popular in imputing missing microarray data (Troyanskaya et al., 2001; Kim et al., 2007). *k*-NN methods are sometimes referred to as memory-based reasoning or instance-based learning or case-based learning techniques and have been used for classification tasks. They essentially work by assigning to an unclassified sample point the classification of the nearest of a set of previously classified points. The entire training set is stored in the memory. *k*-NN requires that data are MCAR.

3.4. Linear discriminant

Originally developed in 1936 by Fisher (1936), linear discriminant analysis (LDA) finds a linear transformation ('discriminant function") of two predictors, say, *X* and *Y*, which yields a new set of transformed values that provides a more accurate discrimination than either predictor alone. Linear discriminants use a mean imputation strategy, i.e. replacing missing values of an attribute with the mean of the attribute. This strategy is applicable for continuous data. For discrete data of the corresponding attribute, the most frequent value (mode) was utilised. LDA is based on the assumption that data is MCAR.

3.5. Naïve Bayes classifier

The naïve Bayes classifier (NBC) is perhaps the simplest and most widely studied probabilistic learning method. It learns from the training data, the conditional probability of each attribute A_i, given the class label C (Kononenko, 1991; Michie et al., 1994). The strong major assumption is that all attributes A_i are independent given the value of the class C. Classification is therefore done applying Bayes rule to compute the probability of C given $A_1 \ldots, A_n$ and then predicting the class with the highest posterior probability. The probability of a class value C_i given an instance $X = \{A_1 \ldots, A_n\}$ for n observations is given by:

$$P(C_i|X) = \frac{p(X|C_i).p(C_i)}{p(X)} = p(A_1, \ldots, A_n|C_i).p(C_i) = \prod_{j=1}^{n} p(A_j|C_i).p(C_i).$$

The assumption of conditional independence of a collection of random variables is very important for the above result. Otherwise, it would be impossible to estimate all the parameters without such an assumption. This is a fairly strong assumption that is often not applicable. However, bias in estimating probabilities may not make a difference in practice – it is the order of the probabilities, not the exact values that determine the probabilities.

To perform imputation, we treat each attribute that contains missing values as the class attribute, then fill each missing values for the selected class attribute with the class predicted from the conditional probabilities established during training.

3.6. Ripper

RIPPER (Cohen, 1996) is a rule-based learning that builds a set of rules that identify classes while minimizing the amount of error. The error is defined by the number of instances misclassified by the rules. RIPPER incorporates a bias against missing values into a rule building process; any test of an attribute whose value is unknown (missing) returns a failure, so that the learner focuses on completely known (non-missing) features in selecting rule pre-conditions. The assumption made about the law generating the missing values when using RIPPER is that the data is MCAR.

3.7. Support vector machines

Support vector machines (SVMs) are pattern classifiers that can be expressed in the form of hyper-planes to discriminate positive instances from negative instances pioneered by Vapkin (1995). The principal goal of the SVM approach is to fix the computational problem of predicting with kernels (Breiman et al., 1984). The basic idea of SVMs is to determine a classifier or regression machine which minimizes the empirical risk (i.e., the training set error) and the confidence interval (which corresponds to the generalisation or test set error). In other words, the idea is to fix the empirical risk associated with architecture and then use a method to minimize the generalisation error. Motivated by statistical learning theory, SVMs have successfully been applied to numerical tasks, including regression and classification. They can perform both binary classification (pattern recognition) and real valued function approximation (regression estimation) tasks. Like artificial neural networks, the standard formulation of SVMs does not allow for missing values for any of the attributes in an instance being learned or classified. However, for the handling of missing values in SVM classifiers, the maximal variation approach by Bhattacherjee et al. (2001) is followed in this paper.

4. A new supervised imputation method

Although many supervised learning imputation methods have already been developed, we still propose a new technique. The motivation for introducing this imputation method is three fold: current imputation methods consider either a very local optimisation criterion resulting in less accurate results, or a more global imputation approach at high computational cost, other global methods are very fast but very inaccurate. The new strategy we propose approaches the imputation of missing attribute values in a global way but keeps the computation time under control. To construct the new method, DT learning and estimation of probabilities using logit models are utilized as described in the following sections.

4.1. Decision trees

A DT (Breiman et al., 1984; Quinlan, 1993) is a model of the data that encodes the distribution of the class label in terms of the

Please cite this article in press as: Twala, B., Phorah, M. Predicting incomplete gene microarray data with the use of supervised learning algorithms. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.05.006

predictor attributes. The root of the decision tree (DT) does not have any incoming edges. Every other node has exactly one incoming edge and zero or more outgoing edges. If a node n has no outgoing edges we call n a leaf node, otherwise we call n an internal node. Each leaf node is labelled with one class label; each internal node is labelled with one predictor attribute called the splitting attribute. Each edge e originating from an internal node n has a predicate q associated with it where q involves only the splitting attribute of n.

A DT can be used to predict the values of the target or class attribute based on the predictor attributes. To determine the predicted value of an unknown instance, you begin at the root node of the tree. Then decide whether to go into the left or right child node based on the value of the splitting attribute. You continue this process using the splitting attribute for successive child nodes until you reach a terminal or leaf node. The value of the target attribute shown in the leaf node is the predicted value of the target attribute.

4.1.1. The probabilistic approach

The missing value problem addressed in this paper can be defined as follows:

Given: A decision tree, a complete set of training data, and a set of instances for testing, described with attributes and their values. Some of the attribute values in the test instances are unknown.

Find: A classification rule for a new instance using the tree structure given that it has an unknown attribute value and by using the known attribute values.

Let A be the attribute associated with a particular node of the tree that could either be discrete or numerical. A discrete attribute has a certain number of possible values J and a continuous attribute may attain any value from a continuous interval. Each node is split into two sons (left and right sons). Hence, a new instance could either go to the left (L) or to the right (R) of each internal node. Further, let V be the binarised value for attribute A.

Let C denote a class and let there be k classes, $J = 1 \ldots, k$. The total probability theorem is used to predict class membership of an unknown attribute value by computing the conditional probability of a class C given the evidence of known attribute values.

For individual j, divide the attributes in the tree into classes for both $\underline{K}$ (the known attribute values) and $\underline{M}$ (the missing attribute values). Assuming that $\underline{K}$ and $\underline{M}$ are statistically independent, the conditional probability that a known attribute value belongs to a certain class is given by the following equation:

$$P(C_j|\underline{K}) = \sum P(C_j|\underline{K},\underline{M})P(\underline{M}|\underline{K}) = \sum P(C_j|\underline{K},\underline{M})P(\underline{M}),$$

where

$$P(\underline{M}|\underline{K}) = \frac{P(\underline{M},\underline{K}))}{P(\underline{K})} = \frac{P(\underline{M})P(\underline{K})}{P(\underline{K})} = P(\underline{M}).$$

The sum is over all possible combinations of values that branch to the left (L) or right (R) at each respective internal node, taken by the vector of the missing attribute values $\underline{M}$. For the unknown attribute values, the unit probability may be distributed across the various leaves to which the new instance could belong. These probabilities are going to be estimated in by using logit models.

For illustration purposes, suppose that the DT shown in Fig. 1 is constructed using a superficial dataset of, say, 40 instances. Further, consider the values for the categorical attribute 1 (A_1) and the numerical attribute 3 (A_3) are missing; attribute 2 (A_2), a continuous attribute, is the only attribute with non-missing values.

From the example, it appears that all the attributes with no missing values would be used when estimating the probabilities. However, this does not have to be the case. The attributes that are used are determined by where the instance branches at a particular internal node. For example, say, A_1 was not missing. For any instance branching to the left of that would mean non-utilisation of attribute A_2 which is connected to the right of branch A_1.

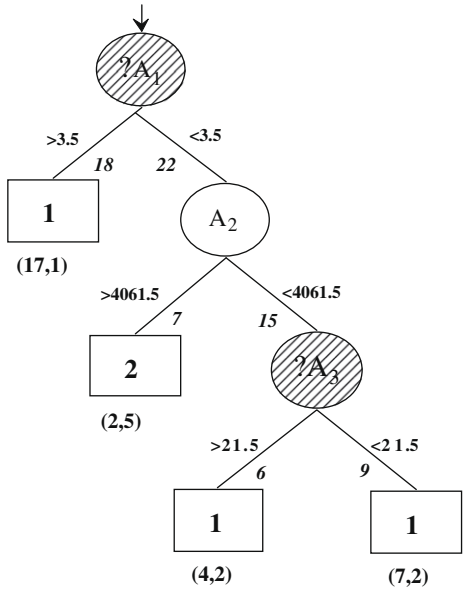

Fig. 1. Example of a binary decision tree from a set of 40 training instances that are represented by three attributes and accompanied by two classes. Note: Figures in brackets are the number of instances in each terminal node for class 1 and 2, respectively.

First case: Class membership for a new instance is predicted given that it will branch to the right of the internal node $A_2(A_2^R)$ given that both A_1 and A_3 have unknown attribute values. We can define the probability that the predicted class membership will be class 1 given that it branches to the right of internal attribute 2, that is, $P(C_1|A_2^R)$ can be defined by:

$$\begin{aligned} P\left(C_1|A_2^R\right) &= P\left(C_1|A_2^R,A_1^L,A_3^L,\right)P\left(A_1^L,A_3^L,|A_2^R\right) \\ &+ P\left(C_1|A_2^R,A_1^L,A_3^R,\right)P\left(A_1^L,A_3^R,|A_2^R\right) \\ &+ P\left(C_1|A_2^R,A_1^R,A_3^L,\right)P\left(A_1^R,A_3^L,|A_2^R\right) \\ &+ P\left(C_1|A_2^R,A_1^R,A_3^R,\right)P\left(A_1^R,A_3^R,|A_2^R\right) \end{aligned} \quad (389)$$

and

$$\begin{aligned} P\left(C_2|A_2^R\right) &= P\left(C_2|A_2^R,A_1^L,A_3^L,\right)P\left(A_1^L,A_3^L,|A_2^R\right) \\ &+ P\left(C_2|A_2^R,A_1^L,A_3^R,\right)P\left(A_1^L,A_3^R,|A_2^R\right) \\ &+ P\left(C_2|A_2^R,A_1^R,A_3^R,\right)P\left(A_1^R,A_3^R,|A_2^R\right) \\ &+ P\left(C_2|A_2^R,A_1^R,A_3^L,\right)P\left(A_1^R,A_3^L,|A_2^R\right) \end{aligned} \quad (392)$$

or $P\left(C_2|A_2^R\right) = 1 - P\left(C1|A_2^R\right)$. (393)

4.1.2. Full estimation of probabilities from training data using logit models

The binary logit model (BLM) is used to estimate probabilities for those datasets that have two classes while a multinomial logit model (MLM) is used to estimate probabilities for datasets with three or more classes (Hosmer and Lameshow, 1989). Both models are described below.

Let $C \in \{0, 1\}$ be the dependent or response variable and let $a = a_{i1} \ldots, a_{ip}$ be the predictor attributes vector. A linear predictor η_i is given by $\beta_0 + \beta' a$ where β_0 is the constant and β' is the vector of regression coefficients $(\beta_1 \ldots, \beta_p)$ to be estimated from the data.

They are directly interpretable as log-odds ratios or in terms of $exp(\beta')$, as odds ratios.

The *a posteriori* class probabilities are computed by the logistic distribution:

$$P(C = 1|a = a_{i1}, \ldots, a_{ip}) = \pi_i = \frac{exp\{\pi_i\}}{1 + exp\{\pi_i\}}$$

β' are estimated by maximising the likelihood function

$$L(\beta_1, \ldots, \beta_p) = \prod_{i=1}^{n} \pi_i^{C_i}(1 - \pi_i)1 - C_i.$$

Computational details can be found in (Menard, 1995)

The estimated predicted value $\hat{\eta}_j$ and the estimated probability $\hat{\pi}_j$ for a new observation $a_{j1}, \ldots, a_{jp}$ are given by $\hat{\eta}_j = \hat{\beta}_0 + \hat{\beta}'a$ and

$$\hat{\pi}_j = \pi(a, \hat{\beta}) = \frac{exp\{\hat{\eta}_j\}}{1 + exp\{\hat{\eta}_j\}}.$$

These terms are often referred to as "predictions" for given characteristic vector a. One advantage of using a binary logit model (rather than LDA) is that it is relatively robust, i.e., many types of underlying assumptions lead to the same logistic formulation. By contrast the LDA approach is strictly applicable only when the underlying variables are jointly normal with equal covariance matrices.

For example, suppose that we have a dataset with p attributes $(A_1 \ldots, A_p)$ and two classes (C_1, C_2). Then the probability that an object with values $a_1 \ldots, a_v$ belongs to C_1 as a logistic function of the attribute variables could be modelled as:

$$P(C_1|A) = \frac{exp\{\beta_0 + \beta_1\beta_1 + \cdots + \beta_k\beta_k\}}{1 + exp\{\beta_0 + \beta_1\beta_1 + \cdots + \beta_k\beta_k\}}.$$

The unknown parameters β_i can be estimated from the training data on instances with known classifications.

Using the example in our illustration, $P(C_1|A_2^L, A_1^L, A_3^L)$ would be estimated by:

$$log\left[\frac{P\left(C_1|A_2^L, A_1^L, A_3^L\right)}{P\left(C_2|A_2^L, A_1^L, A_3^L\right)}\right] = \beta_0 + \beta_1 A_2^L + \beta_2 A_2^L + \beta_2 A_1^L + \beta_3 A_3^L.$$

The generalisation of the binary logit approach to the case of three or more classes is knows as the MLM and the derivation is similar to that of the BLM To give a flavour of how this model can be used for probability estimation purposes, the procedure for a three-class case is sketched out. In this case, the probabilities of an observation belonging to each of the three classes, given a particular characteristic vector, are given by the following expressions:

$$P(\pi_1|a) = \frac{exp\{\hat{\eta}_1\}}{1 + exp\{\hat{\eta}_1\} + exp\{\hat{\eta}_2\}},$$

$$P(\pi_2|A) = \frac{exp\{\hat{\eta}_2\}}{1 + exp\{\hat{\eta}_1\} + exp\{\hat{\eta}_2\}},$$

$$P(\pi_3|A) = \frac{1}{1 + exp\{\hat{\eta}_1\} + exp\{\hat{\eta}_2\}}.$$

Given estimates of the values for the population parameters for the model, the first expression can be used to calculate the probability of a new observation with characteristic vector x belonging to class 1, the second expression can be used to calculate the probability of a new observation with characteristic vector x belonging to class 2, and the third expression can be used to calculate the probability of a new observation belonging to class 3.

Given the fact that there are only three classes, these probabilities must sum to unity. Then the classification rule is stated as follows: If faced with the problem of classifying a new observation with characteristic vector x, then classify it as belonging to the class with the highest calculated probability. Extensions to the four-class case and beyond are straightforward.

In order to identify the parameters of the model, β_{k+1} is set to 0 (a zero vector) as a normalisation procedure and thus:

$$P(C_{k+1}) = \frac{1}{\sum_{j=1}^{k+1} exp\{\hat{\eta}_j\}}.$$

In the MLM model the assumption is that the log-odds of each response follow a linear model. Thus, the jth logit has the following form:

$$log\left[\frac{P(C_j)}{P(C_{k+1})}\right] = \hat{\eta}_j.$$

This model is analogous to the BL model, except that the probability distribution of the response is multinomial instead of binomial and there are k equations instead of one. The k multinomial logit equations contrast each of categories $j = 1, \ldots, k$ with category $k + 1$, whereas a single binary logit equation is a contrast between successes and failures.

If $k = 1$ the ML model reduces to the usual binary logit model. The ML model is in fact equivalent to running a series of BL models. For purposes of this paper, the ML model was not used to estimate probabilities based on all the attributes given in the dataset, but to estimate only the unknown probabilities of the given attributes specifically related to the problem. For this method the unknown instance will be classified as belonging to class with the highest probability.

An important property of MLM is the assumption of independence from irrelevant alternatives (IIA), which could be a major drawback for some practical applications. The property of IIA could be stated as follows: the ratio of the choice of probabilities of any two alternatives is unaffected by the systematic utilities of any other alternatives. In other words, the odds of outcome 1 (say, Path 1) versus outcome 2 (say, Path 2) do not depend on what other outcomes (say, a and b) are available. For more details about the logit model and how the logits and probabilities are modelled, the reader is referred to Hosmer and Lameshow (1989).

5. Experiments

5.1. Experimental set-up

In order to empirically evaluate the performance of the proposed probabilistic technique, we conducted a series of simulation and experimental studies on eight microarray datasets in terms of misclassification error rate. The primary goal of the evaluation was to analyze the impact of erroneous data on predictive cancer diagnosis accuracy. Each dataset defines a different learning problem as shown in Table 1.

The selected datasets cover a comprehensive range for each of the following characteristics:

- the size of datasets expressed in terms of the number of instances ranges between 22 and 308;
- the number of attributes ranges between 2000 and 15,154; the number of classes ranges between 2 and 26.

In general the datasets were selected in order to assure reasonable comprehensiveness of the results. The first five involve datasets with only two classes and the last three involve datasets with more than two classes.

Sources of the datasets in terms of diagnostic tasks are given as: BC (Lee et al., 2003); CC (Alon et al., 1999); LUK (Golub et al., 1999); PC (Singh et al., 2002); OC (Shital and Kusiak, 2007); BT (Pomeroy

Please cite this article in press as: Twala, B., Phorah, M. Predicting incomplete gene microarray data with the use of supervised learning algorithms. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.05.006

Table 1
Datasets used for the experiments.

Datasets	Instances (no. of samples)	Attributes (no. of genes)	Classes	
Two classes:				
	BC	22	3226	2
	CC	62	2000	2
	LUK	72	7129	2
	PC	102	10,529	2
	OC	253	15,154	2
More than two classes:				
	BT	90	5920	5
	LC	203	12,600	5
	TUM	308	15,009	26

BC = Breast cancer (breast cancer and normal tissues).
CC = Colon cancer (colon cancer and normal tissues).
LUK = Leukaemia (acute lymphoblastic and acute Myelogenous).
PC = Prostrate cancer (prostrate cancer and normal tissues).
OC = Ovarian cancer (ovarian cancer and normal tissues).
BT = Brain tumour (5 brain tumour types).
LC = Lung cancer (4 lung cancer types and normal tissues).
TUM = Tumours (14 human tumour and 12 normal tissues).

et al., 2002); LC (Bhattacherjee et al., 2001); and TUM (Ramasway et al., 2001).

Since the distribution of missing values among attributes and the missing data mechanism were two of the most important dimensions of this study, three suites of data were created corresponding to MCAR, MAR and IM.

In order to simulate missing values on attributes, the original datasets are run using a random generator (for MCAR), and for MAR and IM, a quintile attribute-pair approach is utilized.

For MAR, the idea is to condition the generation of missing values based upon the distribution of the observed values. Attributes of a dataset are separated into pairs, say, (A_X, A_Y), where A_Y is the attribute into which missing values are introduced and A_X is the attribute on the distribution of which missingness of A_Y is conditioned. For example, to generate missingness in half of the attributes for a dataset with, say, 12 attributes $(A_1, \ldots, A_{12})$, the pairs (A_1, A_2), (A_3, A_4) and (A_5, A_6) could be utilised. We assume that A_1 is highly correlated with A_2; A_3 highly correlated with A_4, and so on. For the (A_1, A_2) pairing, A_1 is used to generate a missing value template of zeros and ones utilizing the quintile approach. The template is then used to "knock off" values (i.e., generating missingness) in A_2, and vice versa.

IM data arise due to the data missingness mechanism being explainable, and only explainable by the very variable(s) on which the data are missing. For conditions with IM data, a procedure identical to MAR was implemented. However, for IM, the missing values template was created using the same attribute variable for which values are deleted in different proportions. Both of these procedures have the same percentage of missing values as their parameters. These two approaches were also run to get datasets with four levels of proportion of missingness p. The experiment consists of having p% of data missing from only the testing (classification) set.

For each dataset, two suites were created. First, missing values were simulated on half of the attributes (MCAR*half*, MAR*half*, IM*half*). Second, missing values were introduced on all the attribute variables (MCAR*all*, MAR*all*, IM*all*). For both suites, the missingness was evenly distributed across all the attributes. To measure the performance of methods, the training set/test set methodology is employed as shown in Fig. 2 (supervised classification with incomplete data).

For each run, each dataset is split randomly into 80% training and 20% testing, with different percentages of missing data (i.e., 5%, 10%, 20%, 35% and 50%) in the covariates for testing set. five fold

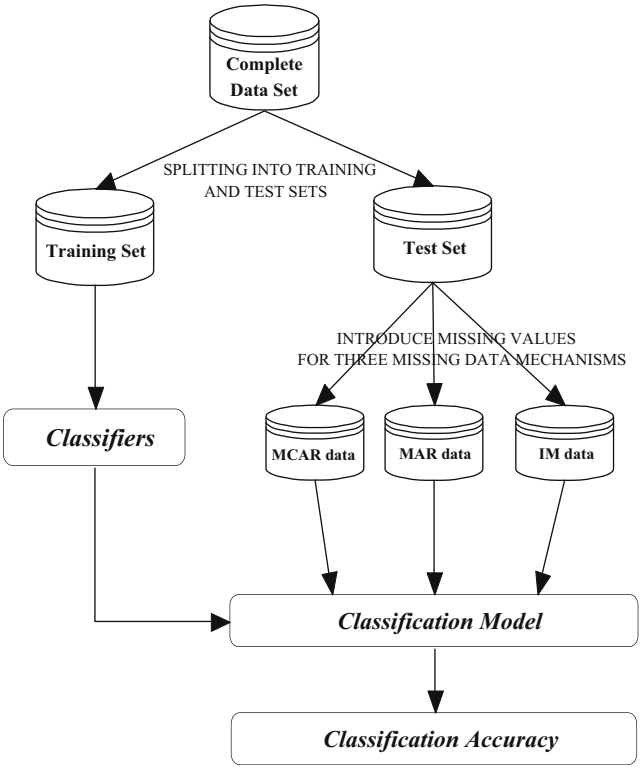

Fig. 2. On supervised classification with incomplete data.

cross validation was used for the experiment. It was also reasoned that the condition with no missing data should be used as a baseline and what should be analysed is not the error rate itself but the increase or excess error induced by the combination of conditions under consideration. Therefore, the excess error is the error achieved given that the dataset is incomplete less the error exhibited given that dataset is complete.

All statistical tests were conducted using the MINITAB statistical software program (MINITAB, 2002). Analyses of variance, using the general linear model procedure were used to examine the main effects and their respective interactions. The comparison of means was conducted by using the Tukey post hoc test (Kirk, 1982). This was done using a 5-way factorial design experiment, with four fixed effect factors (the testing methods; number of attributes with missing values; missing data proportions; and missing data mechanisms) and random effect factor (eight datasets). Results were averaged across five folds of the cross-validation process before carrying out the statistical analysis. The averaging was done as a reduction in error variance benefit.

5.2. Experimental results

Experimental results on the effects of new and existing methods for handling incomplete and test data (testing methods) on cancer diagnostic predictive accuracy are described.

5.2.1. Main effects

All the main effects were found to be significant at the 5% level of significance ($F = 11.2$, df = 7 for testing methods; $F = 9.8$, df = 1 for number of attributes with missing values (pattern); $F = 129.4$, df = 2 for missing data proportions; $F = 52.8$, df = 2 for missing data mechanisms; $p < 0.05$ for each main effect).

Fig. 3 plots the overall excess error rates for eight testing methods, which shows PDT achieving the highest accuracy rates, followed by SVM, NBC, and RIPPER. The worst overall performance

Please cite this article in press as: Twala, B., Phorah, M. Predicting incomplete gene microarray data with the use of supervised learning algorithms. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.05.006

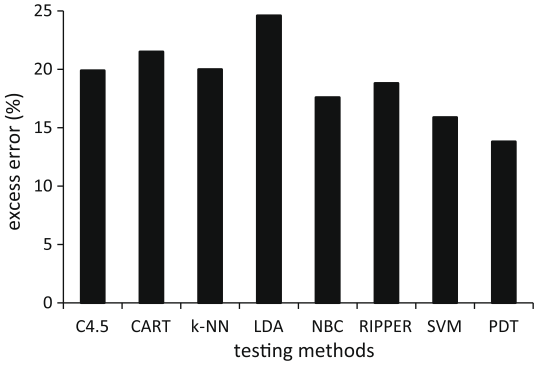

Fig. 3. Overall means for current and new testing methods.

is by LDA. The difference in error rate between PDT and the existing methods was found to the statistically significant at the 5% level ($p < 0.05$). Also, no significant differences in performance were found between CART and *k*-NN.

From Fig. 4, it appears that for all the testing methods (both current and new) missing values have a greater effect when they are distributed among half of the attributes compared with when missing values are in all the attribute variables.

The performance of all the testing methods degrades with increases in missing values, and vice versa. Also, the relationship between performance and missing data proportions appears to be linear (Fig. 5).

The severe impact of IM data on predictive accuracy is quite noticeable on Fig. 6. The excess error exhibited by the eight testing methods is 19.2% compared with an error rate of 9.7% for MCAR data. The significant difference between the MCAR and MAR is also significant with the impact of the latter mechanism more severe.

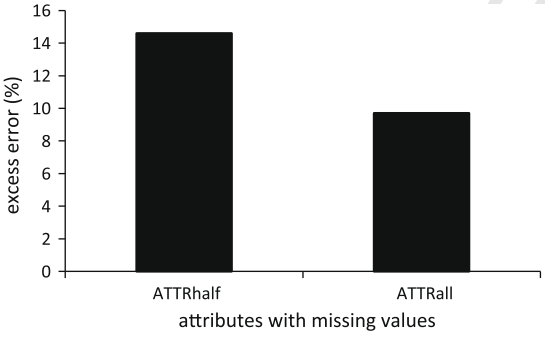

Fig. 4. Overall means for number of attributes with missing values (current and new testing methods).

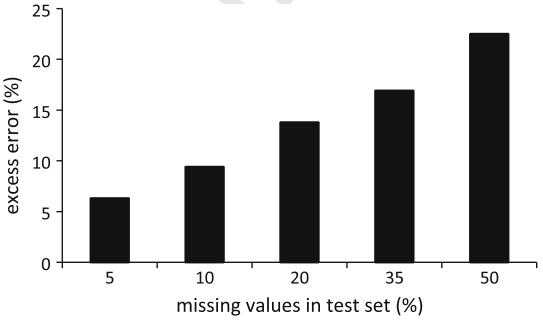

Fig. 5. Overall means for missing data proportions (current and new testing methods).

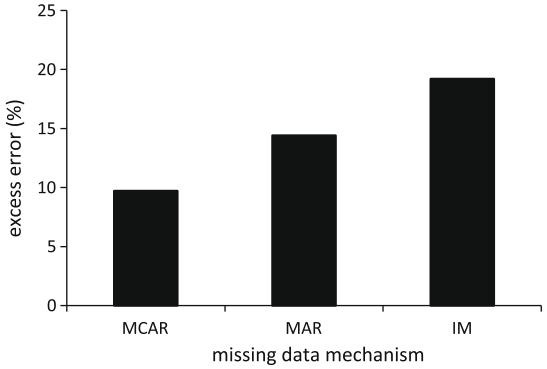

Fig. 6. Overall means for missing data mechanisms (current and new testing methods).

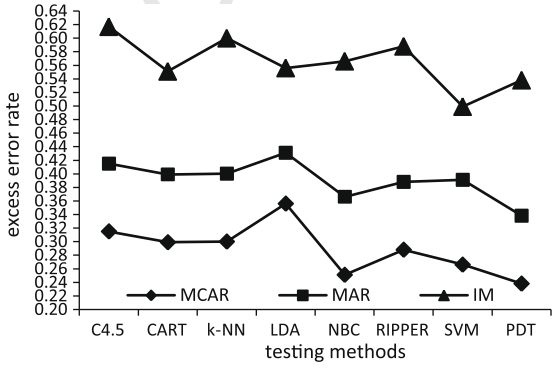

Fig. 7. interaction between current and new testing methods and missing data mechanisms.

5.2.2. Interaction effects

The only interaction effect that was found to be statistically significant at 5% is between testing methods and missing data mechanisms as displayed in Fig. 7. Fig. 7 further shows all the testing methods performing worse under the IM condition compared with when data are either MCAR or MAR. Compared with SVM, PDT appears to handle both MCAR and MAR data better with SVM coping better with IM data. Surprisingly LDA (which is based on the MCAR assumption) struggles with both MCAR and MAR data but does better than well-established supervised methods such as C4.5 and *k*-NN for IM data.

6. Conclusion and discussion

Our main contribution is the development of a novel probabilistic algorithm for the classification of incomplete (erroneous) microarray data. By making a couple of mild probabilistic assumptions, the proposed approach solves the incomplete microarray data problem in a principled manner, avoiding imputation heuristics.

As expected, the performance of the classifiers differs among the datasets even though our results were averaged. For example, the overall performance by classifier methods on datasets with more than two classes is poor compared with performances on two-class-problem domains. The former is intrinsically more difficult because the classification algorithm has to learn to construct a great number of separation boundaries or relations; each class has to be defined explicitly.

All the current and proposed SL methods exhibit bigger error rates when missing values are distributed among half of the attributes compared with when the missing values are in all attributes.

Please cite this article in press as: Twala, B., Phorah, M. Predicting incomplete gene microarray data with the use of supervised learning algorithms. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.05.006

The experiments further showed missing data mechanism as having more impact on the performances of the SL methods. The impact of MCAR data seems less severe on classification accuracy compared to MAR or IM. These results are in support with statistical theory and to our prior results reported in (Twala, 2009). Also, it was not surprising that all the techniques struggled with IM data (which is always a difficult assumption to deal with).

Overall, PDT is the most effective with serious competition from SVM. The strength of PDT lies on its MAR data assumption and its variance reduction ability due to its 'naïve' but still very reliable variable probability estimation strategy. Even if not an especially good model, the logistic model is less prone to over-fitting the data and hence better predictions (since one does not slavishly follow the idiosyncrasies of the training data. This is crucial in the microarray domain, which is characterized by thousands of genes/attributes and a very few samples. Unlike most other algorithms (for example *k*-NN), PDT performs very well because of its feature selection (DT induction) strategy.

The theoretical advantage of SVM made it competitive with PDT. For SVM, the idea of margin and stability mitigates the problem of over-fitting the training data as already discussed. SVM is also helpful when there are few training samples. Other algorithms are made stable by removing the noisy genes and reducing the number of features.

LDA is evidently the worst overall method due to its single imputation strategy that does not permit assessment of the uncertainty due to imputation. In other words, LDA does not adequately represent the uncertainty about the missing value.

The poor performance of SVS could be attributed to low correlations among attributes for some of the datasets. Both methods are suitable for domains in which strong relation exist between the attributes.

The difficulty with *k*-NNS could have been the choice of the distance metric *k* that is unknown for finite *n*. The good performance of NBC for handling MCAR data is attributed to its strategy of computing distributions of attributes; those distributions do not change for any missing data pattern. However, NBC encounters problems similar to *k*-NN, in particular because they rely on Euclidean of Mahalanobis distance for density estimation that generally require exponential sample to the data dimensionality.

The bias of RIPPER against attributes with missing values is quite appropriate. However, it seems that the way the rules are extracted makes the algorithm more sensitive to missing values.

Although numerous algorithms about classification with incomplete data have been developed, what sets apart our proposed novel strategy is its maximum achievable variance reduction ability or equivalently a maximum achievable smoothness of the probabilities. The proposed algorithm was also successful even when a high percentage of features are missing. Furthermore, the proposed approach does not make representational assumptions or pre-supposes other model constraints. Therefore, it is suitable for a wide variety of datasets. Despite its strengths, the proposed approach can be quite a slow, computationally intensive process especially for big DTs. This is because several branches must do the calculation simultaneously. So, if, say, *K* branches do the calculation, then the central processing unit time spent is *K* times the individual branch calculation.

Several exciting directions exist for future research. One topic deserving future study would be to assess the impact of missing values when they are in both the training and test sets and using model-based statistical imputation methods such as multiple imputation. PDT was also applied on only eight small datasets. This work could be extended by considering a more detailed simulation study using much more balanced types of datasets required to understand the merits of the technique, especially larger datasets.

Conflict of interest

None declared.

Acknowledgements

This work was supported by the Council of Scientific and Industrial Research, Modelling and Digital Sciences Grant (MDSARR1). We would like to thank David Hand for his constructive comments and suggestions.

References

Alon, U., Barkai, A., Gish, K., 1999. Broad patterns of gene expression revealed by clustering analysis of tumour and normal colon tissues probed by oligonucleotide arrays. Proc. Natl. Acad. Sci. 96, 6745–6750.
Bernado, J., Smith, A., 1994. Bayesian Theory. John Wiley, Chichester.
Bhattacherjee, A., Richards, W., Staunton, J., Li, C., 2001. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. Proc. Natl. Acad. Sci. 98, 13790–13795.
Bø, T.H., Dysvik, B., Jonaseen, I., 2004. LSimpute: Accurate estimation of missing values in microarray data with least squares methods. Nucl. Acids Res. 32 (3), e34.
Branden, J.V.K., Verboven, S., 2009. Robust data imputation. Comput. Biol. Chem. 33, 7–13.
Brás, L., Menezes, J.C., 2007. Improving cluster-based missing value estimation of DNA microarray data. Biomol. Eng. 24, 273–282.
Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., 1984. Classification and Regression Trees. Chapman and Hall Inc..
Cestnik, B., Kononenko, I., Bratko, I., 1987. Assistant 86: A knowledge-elicitation tool for sophisticated users. In: Bratko, I., Lavrac, N. (Eds.), European Working Session on Learning – EWSL87. Sigma Press, Winslow, England.
Cohen, W., 1996. Learning trees and rules with set-valued features. Amer. Assoc. Artifi. Intell. (AAAI).
Farhangfar, A., Kurgan, L., Dy, J., 2008. Impact of imputation of missing values on classification error for discrete data. Pattern Recognition 41, 3692–3705.
Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugenics 7, 179–188.
Gan, S., Liew, A.W.-E., Yan, H., 2006. Microarray missing data imputation based on a set theoretic framework and biological knowledge. Nucl. Acids Res. 34 (5), 1608–1619.
García-Laencina, P.J., Sancho-Gómez, J.-L., Figueiras-Vidal, A.R., Verleysen, M., 2009. K nearest neighbours with mutual information for simultaneous classification and missing data imputation. Neurocomputing 72, 1483–1493.
Golub, T., Slonim, D., Huard, C., 1999. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. J. Sci. 258, 531–537.
Hand, D.J., 1997. Construction and Assessment of Classification Rules. John Wiley & Sons.
Hawarah, L., Simonet, A., Simonet, M., 2006. Dealing with missing values in a probabilistic decision tree during classification. In: IEEE Internat. Conf. on Data Mining, December, pp. 325–329.
Hosmer, D.W., Lameshow, S., 1989. Applied Logistic Regression. Wiley, New York.
Kim, K.-Y., Kim, B.-J., Yi, G.-S., 2004. Reuse of imputed data in microarray increases imputation efficiency. BMC Bioinf. 5, 160.
Kim, H., Golub, G.H., Park, H., 2005. Missing value estimation for DNA microarray gene expression data: Local least squares imputation. Bioinformatics 21 (2), 187–198.
Kim, D.-W., Lee, K.-Y., Lee, K.H., Lee, D., 2007. Towards clustering of incomplete microarray data without the use of imputation. Bioinformatics 23 (1), 107–113.
Kirk, R.E., 1982. Experimental Design, second ed. Brooks, Cole Publishing Company, Monterey, CA.
Kononenko, I., 1991. Semi-naïve Bayesian classifier. In: Proc. Eur. Conf. on Artificial Intelligence. pp. 206–219.
Lee, K., Sha, N., Dougherty, E., Vannucci, E., 2003. Gene classification: A Bayesian variable selection approach. Bioinformatics 19, 90–97.
Little, R.J.A., Rubin, D.B., 1987. Statistical Analysis with Missing Data. Wiley, New York.
Menard, S., 1995. Applied Logistic Regression Analysis. Sage, London.
Michie, D., Spiegelhalter, D.J., Taylor, C.C., 1994. Machine learning of rules and trees. In Machine Learning, Neural and Statistical Classification. Ellis Harwood, 1994.
MINITAB, 2002. Statistical Software for Windows 9.0. MINITAB, Inc., PA, USA.
Nguyen, D.V., Wang, N., Carroll, R.J., 2004. Evaluation of missing value estimation for microarray data. J. Data Sci. 2, 347–370.
Oba, S., Sato, M.-A., Takemasa, I., Monden, M., Matsubara, K.-I., Ishii, S., 2003. A Bayesian missing value estimation method for gene expression profile data. Bioinformatics 19, 2088–2096.
Osareh, A., Shadgar, B., 2009. Classification and diagnostic prediction of cancers using gene microarray data analysis. J. Appl. Sci. 9 (3), 459–468.
Pomeroy, S., Tamayo, P., Gaasenbeek, M., Sturla, L., 2002. Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature 415, 436–442.
Quinlan, J.R., 1993. C.4.5: Programs for Machine Learning. Morgan Kauffman Publishers, Inc., Los Altos, California.

Ramasway, S., Tamayo, P., Rifkin, R., Mukherjee, S., 2001. Multiclass cancer diagnosis using tumour gene expression signature. Proc. Natl. Acad. Sci. 98, 15149–15154.

Rubin, D.B., 1987. Multiple Imputation for Nonresponse in Surveys. Wiley, New York.

Saar-Tsechansky, M., Provost, F., 2007. Handling missing values when applying classification models. J. Machine Learn. Res. 8, 1625–1657.

Schafer, J., 1997. Analysis of Incomplete Multivariate Data. Chapman and Hall, New York.

Sehgal, M.S.B., Gondal, I., Dolley, L.S., 2005. Collateral missing value imputation: A new robust missing value estimation algorithm for microarray data. Bioinformatics 21, 2417–2423.

Shital, S., Kusiak, A., 2007. Cancer gene search with data mining and genetic algorithms. Comput. Biol. Med. 37, 251–261.

Singh, D., Febbo, P., Jackson, J., Manola, J., Ladd, C., 2002. Gene expression correlates of clinical prostate cancer behaviour. Cancer Cell 12, 203–209.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibsharini, R., Botstein, D., Altman, R.B., 2001. Missing value estimation methods for DNA microarray. Bioinformatics 17, 520–525.

Tuikkala, J., Elo, L.L., Navalainen, O.S., Aitookallio, T., 2008. Missing value imputation improves clustering and interpretation of gene expression microarray data. BMC Bioinf. 9, 202.

Twala, B., 2009. An empirical evaluation of techniques for handling incomplete data using decision trees. Appl. Artif. Intell. 23 (5), 373–405.

Twala, B., Jones, M.C., Hand, D.J., 2008. Good methods for coping with missing data in decision trees. Pattern Recognition Lett. 29 (7), 950–956.

Vapkin, V., 1995. The Nature of Statistical Learning Theory. Springer-Verlag, Berlin.

Walszak, B., Massart, D.L., 2001. Tutorial, dealing with missing data, part 1. Chemometr. Intell. Lab. Systems 58, 15–27.

Williams, D., Liao, X., Xue, Y., Carin, L., Krishnapuram, B., 2007. On classification with incomplete data. IEEE Trans. Pattern Anal. Machine Intell. 29 (3), 427–436.

Zhang, X., Song, X., Wang, H., Zhang, H., 2008. Sequential local least squares imputation estimating missing value of microarray data. Comput. Biol. Med. 38, 1112–1120.

Zhou, X., Wang, X., Dougherty, E.R., 2003. Missing-value estimation using linear and non-linear regression with Bayesian gene selection. Bioinformatics 19, 2302–2307.

Please cite this article in press as: Twala, B., Phorah, M. Predicting incomplete gene microarray data with the use of supervised learning algorithms. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.05.006