ASIC: Supervised Multi-class Classification using Adaptive Selection of Information Components

Zongxing Xie, Thiago Quirino, Mei-Ling Shyu
Department of Electrical and Computer Engineering
University of Miami, Coral Gables, FL 33124, USA
z.xie1@umiami.edu, t.quirino@umiami.edu, shyu@miami.edu

Shu-Ching Chen
Distributed Multimedia Information System Laboratory
School of Computing and Information Sciences
Florida International University, Miami, FL 33199, USA
chens@cs.fiu.edu

Abstract

In this paper, a supervised multi-class classification approach called Adaptive Selection of Information Components (ASIC) is presented. ASIC has the facilities to (i) handle both numerical and nominal features in a data set, (ii) pre-process the training data set to accentuate the spatial differences among the classes in the training data set to reduce further computational load requirements, and (iii) conduct supervised classification with the C-RSPM (Collateral Representative Subspace Projection Modeling) approach. Experimental results on a variety of data sets have shown that the proposed ASIC approach outperforms other well-known supervised classification methods such as C4.5, KNN, SVM, MLP, BN, RF, Logistic, and C-RSPM, with higher classification accuracy, lower training and classification times, and reduced memory storage and processing power requirements.

1. Introduction

Supervised classification is a fundamental component in many data mining applications. For any classification framework, in addition to high classification accuracy performance, operational merits in terms of faster algorithmic execution speed, lower requirements of memory and storage, among others, are also crucial performance evaluation measures. Unfortunately, most of the existing supervised classification methods have difficulty in providing both satisfactory classification accuracy and operational merits. In particular, real-world applications usually involve a large number of classes and/or features, and their computational complexity typically increases correspondingly. Therefore, advanced techniques to address this high dimensionality issue to achieve operational merits is needed.

In the literature, techniques such as PCA (Principal Component Analysis) [10] and MDA (Multiple Discriminant Analysis) have been widely employed to handle high dimensionality issues. PCA is a linear transformation that computes an orthogonal coordinate system for a data set such that the greatest variance by any projection of the data set comes to lie on the first axis (known as the first principal component), the second greatest variance on the second axis, and so on. One of PCA's most desirable features is its provision of data dimensionality reduction capabilities, which is especially useful in applications where the employment of high-dimensionality feature vectors is required, such as in face recognition [14], SIMCA [23], RSIMCA [1] and other applications [24]. After the execution of PCA, different distance measures can be defined in the transformed projection space to facilitate the classification task. On the other hand, MDA adopts a perspective similar to that of PCA, but can be used to define spatial patterns and to assist in the meaningful interpretation of these patterns. In [5], a probabilistic MDA approach, integrated with the Expectation Maximization (EM) framework, was proposed to determine class discriminating features to improve classification performance in content-based image retrieval. The experimental results showed both an improved image retrieval precision and a reduced database search time. In another study, a recursive partition tree was proposed where MDA is applied to each local tree node to facilitate a faster

tree structure search process [3]. In [13], an effective fingerprint classification method was proposed based on MDA, where features were calculated from Gabor filtered images and the derived feature vectors were classified into one of five classes.

In this paper, a novel supervised multi-class classification approach called Adaptive Selection of Information Components (ASIC) is proposed which incorporates a WMCA/MDA-based data pre-processing method and the effective C-RSPM (Collateral Representative Subspace Projection Modeling) approach with the attempt to achieve both favorable operational merits and high classification accuracy. ASIC consists of two levels. The core of the $Global$ level is our proposed WMCA/MDA-based data preprocessing method which includes the WMCA (Weighted Multiple Correspondence Analysis) and MDA techniques. The main functionality of the WMCA component is to broaden ASIC's generality by facilitating it with the capability of handling both numerical and nominal features which occur commonly in real-world applications. WMCA replaces the traditional MCA (Multiple Correspondence Analysis) technique [8] that utilizes only a sequential, limited, and static number of principal components in scaling value computations with the introduction of a principal component similarity information extraction method, while also providing a higher efficiency of utilization of available derived statistical information. In [17], MCA was used to derive, using the first two principal axes, numerical scaling values for all nominal features possessing two or more categories. The MDA component aims at maximizing the ratio of the inter-class scatter to the intra-class scatter while also reducing a data set's dimensionality to $C-1$, where C is the number of classes in the data set, resulting in an improved execution of C-RSPM in the $Local$ level. The core of the $Local$ level is the C-RSPM approach which adaptively selects representative and possibly non-consecutive principal components and was shown to achieve good performance in classification accuracy and operational merits [16].

Experiments on a variety of data sets are conducted to assess the performance of the proposed ASIC approach in comparison to several well-known supervised classification methods such as C4.5 [15], KNN (K-Nearest Neighbors) [19], Bayes' Nets (BN) [6], Multi-Layer Perceptrons (MLP) [21], Random Forest (RF) [2], Logistic [9], and SVM (Support Vector Machines) [18] (all of them are available in WEKA [21]), in addition to our previously proposed C-RSPM approach [16]. The promising experimental results have demonstrated that ASIC outperforms all the methods in the comparison experiments in both classification accuracy and operational benefits.

The remainder of this paper is organized as follows. Section 2 illustrates the motivations for the proposed approach. Section 3 presents the proposed ASIC approach. The details of the experiments and comparative analysis of performance evaluation are described in Section 4. Finally, Section 5 concludes our study.

2. Motivation

In our earlier study [16], the C-RSPM classification approach was proposed to deal with high data dimensionality classification tasks. C-RSPM is capable of adaptively selecting nonconsecutive principal components from a training data set with the purpose of accurately modeling a representative subspace for each class in the data set via a series of collaterally executed RSPM (Representative Subspace Projection Modeling) classifiers. Experimental results in [16] showed that C-RSPM outperforms several supervised classification methods. However, in spite of its good performance, our further experiments with a larger variety of data sets have revealed the following limitations:

- *Ineffective classification with data sets possessing a very high number of classes.* C-RSPM sometimes did not perform as well as expected for a data set with a relatively large number of classes, especially when the classes' distributions are very similar. Hence, a more effective method that can identify and accentuate the differences among classes is more desirable.

- *Ineffective classification with data sets possessing a low number of training data instances.* C-RSPM requires the number of training data instances in each class to be at least as large as the number of features in the data set to successfully model each class. However, in many real-world applications, it may be very difficult to provide sufficient training data instances for each class. Several examples can be found in the data sets in [11][20]. Currently, this issue is resolved by simply duplicating randomly selected existing data instances, which inevitably leads to the "over-fitting" problem to some extent.

- *Relatively high initial computational load in each RSPM classifier.* PCA is employed in each RSPM classifier in a manner that an eigenspace transformation step takes place first, without any prior dimensionality reduction, followed by the adaptive principal component selection technique that captures the representative and possibly nonconsecutive principal components to model the training data classes. Consequently, each RSPM classifier needs to load all the dimensional information of the training data set in training, although most of such information is discarded after the eigenspace transformation and representative component selection steps. This results in a relatively high initial computational load.

It has been inferred that PCA maximizes the variance in all dimensions and is thus capable of modeling the degree of similarity among the data classes, just as is performed in C-RSPM. On the other hand, MDA is mathematically different from PCA in terms of what dimensional features it attempts to maximize. That is, MDA attempts to maximize the distance between class data clusters; while PCA does not take into account any class information in its variance maximization process. To capitalize the advantages of both PCA and MDA, some research studies have been developed. For example, in [4], a novel two-layer PCA/MDA scheme for hand posture recognition was proposed, where the PCA layer acts as a crude classifier, followed by applying local MDA for more precise classification. As a result, the dimensionality of the fingerprint feature patterns analyzed by each local MDA is substantially reduced, leading to a better overall classification performance.

These observations inspired us to develop a new and enhanced supervised classification approach. In contrast to [4], our proposed approach employs WMCA and MDA in the *Global* level for the purpose of crude data preprocessing, while harnessing the benefits of our previously proposed C-RSPM for a more precise classification at the *Local* level. In this manner, each RSPM classifier is able to both model the characteristics of their corresponding classes more accurately and discriminate among different class data instances more precisely. Furthermore, since MDA reduces the dimensionality of a training data set prior to dispatching the data to the local RSPM classifiers, it also relaxes PCA's requirements for the number of data instances in each class, which consequently reduces the possibility of "over-fitting". Moreover, the employment of MDA also lightens the initial storage load requirements in each local RSPM classifier due to the reduced data dimensional information, which further improves ASIC's overall operational benefits.

3. The ASIC approach

The proposed ASIC approach is composed of *Global* and *Local* levels. Figure 1 illustrates the components of each level and provides a top-down depiction of the execution flow. The core of the *Global* level is the proposed preprocessing method including the WMCA and MDA components. When a training data set is supplied to the *Global* level, it is divided into numerical and nominal features. The nominal features are then numerically scaled by WMCA prior to being re-grouped with the numerical features. Next, the generated new numeric feature set is then dispatched to the MDA component which accentuates the spatial differences among the training classes, in addition to transforming the number of features in the training data set to $C-1$, where C is the number of classes in the training set.

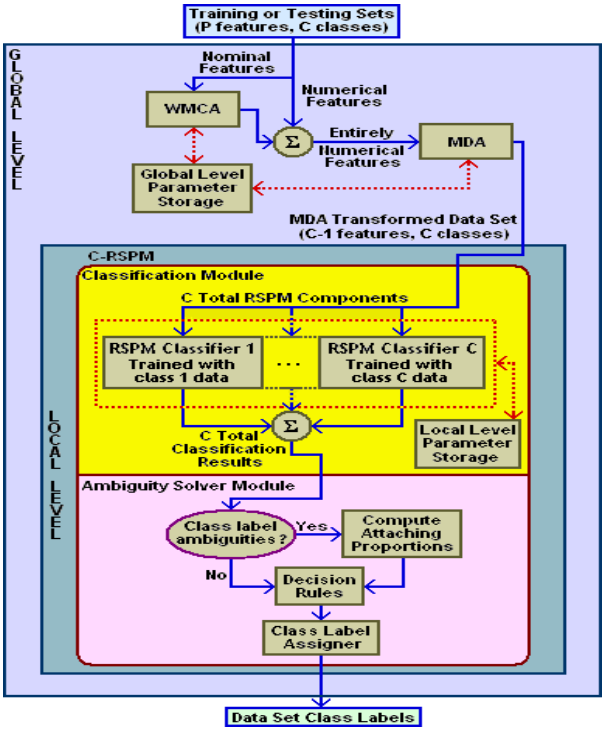

Figure 1. The proposed ASIC supervised classification approach.

From Figure 1, it can be noted that both the WMCA and MDA components store their derived scaling and transformation parameters in the *Global Level Parameter Storage* component, which symbolizes either a computer file or physical memory, for future use with the testing data sets (i.e., the red dotted arrow between WMCA and the storage component and the red dotted arrow between MDA and the storage component). This indicates that the parameters can be stored into or retrieved from the storage component. Finally, the transformed training data set is dispatched to the *Local* level, where the C-RSPM component adaptively trains C respective RSPM classifiers to recognize the C classes in the training data set. Similar to the components in the *Global* level, C-RSPM stores the derived classification parameters of the C classifiers into the *Local Level Parameter Storage* component for future use. The classification task follows naturally from the same descriptive order.

When a testing data set is supplied to the *Global* level, the WMCA and MDA components retrieve their classification parameters from the storage component and, respectively, scale and transform the testing set prior to dispatching it to the *Local* level. Finally, C-RSPM's *Classification Module* retrieves the required classification parameters for its C RSPM classifiers from the storage component, classifies one testing instance at a time by individually dispatch-

ing them to its collaterally executing RSPM classifiers, and forwards their classification results to the *Ambiguity Solver* Module. The *Ambiguity Solver* Module then intercepts classification conflicts that arise, employs the *Compute Attaching Proportion* and *Decision Rules* components to solve class ambiguity issues, and finally executes the *Class Label Assigner* component to assign the corresponding class label of the testing instance.

3.1. Global level

3.1.1 WMCA nominal feature handling

Both numerical and nominal features can carry valuable information about a data set that can crucially affect the performance of a class discrimination algorithm, consequently also affecting the overall performance of the classification process. This leads us to search for an effective means of incorporating nominal features into our inherently numerical-based methods. MCA [8] is a well-known method that generates scaling values to capture the degree of similarity among nominal feature values and to effectively express the relative significance of the features in a data set. Our proposed Weighted Multiple Correspondence Analysis (WMCA) method attempts to improve the information utilization efficiency of the traditional MCA approach by utilizing all the information acquired statistically from a data set to generate numerical scaling values. WMCA is utilized in conjunction with MDA at the $Global$ level for pre-processing purposes.

The main idea of WMCA is to first represent the nominal values in a data set numerically through the derivation of a zero-one valued indicator matrix [7]. Next, an important measure of correlation among the nominal feature values in the data set is computed from a feature cross-tabulation derived from the inner-product of the indicator matrix. The resulting matrix carries the various degrees of similarities among the nominal values in the data set. This naturally leads to the application of Singular Value Decomposition (SVD) onto this matrix in order to extract its eigenvalue-eigenvector pairs which correspond to the values that maximize the derived correlative measure, namely, (λ_1, $\boldsymbol{E}_1$), (λ_2, $\boldsymbol{E}_2$), ..., (λ_p, $\boldsymbol{E}_p$), where p is the number of nominal feature values in the data set. Unlike traditional MCA which utilizes only the first major principal component in numerical scaling value computations and discards the remaining eigenvector-eigenvalue pairs (thus under-utilizing their information), WMCA combines all information into a derived weighted measure. To accomplish that, a refined eigenspace is derived by selecting only those eigenvector-eigenvalue pairs satisfying Equation (1) which employs the standard deviation to capture the degree of similarity among the principal components, while discarding all others.

$$STD(\boldsymbol{E}_\psi) \leq \mathrm{Mean}_{\mathrm{STD}}(\mathbf{E}). \quad (1)$$

In Equation (1), matrix $\mathbf{E}$ is composed of all column eigenvectors acquired through SVD, $Mean_{STD}(\mathbf{E})$ is the mean standard deviation value of all the eigenvectors, $STD(\boldsymbol{E}_\psi)$ corresponds to the standard deviation of the ψ^{th} eigenvector satisfying Equation (1), and $\psi \in \mathbf{W}$ denotes the refined eigenvector space comprised of all eigenvector-eigenvalue pairs satisfying Equation (1). These eigenvectors are then utilized to derive a single column vector $\boldsymbol{H}$ (with $p \times 1$ dimension) through a weighted average given by Equation (2), where each eigenvector's contribution to the summation (i.e., their assigned weight) is based on the magnitude of their respective eigenvalues.

$$\boldsymbol{H} = \sum_{\psi \epsilon W} \frac{\lambda_\psi}{\lambda_t} \boldsymbol{E}_\psi \quad (2)$$

In Equation (2), λ_ψ is the eigenvalue corresponding to the ψ^{th} eigenvector $\boldsymbol{E}_\psi$ and $\lambda_t = \sum_{\psi \in \mathbf{W}} \lambda_\psi$ is the total sum of the eigenvalues, all in the refined eigenspace $\psi \in \mathbf{W}$. Finally, the p elements in $\boldsymbol{H}$ are used to derive p numerical scaling values for the p nominal feature values in the data set, which correspond respectively to the p columns of the derived indicator matrix.

3.1.2 MDA pre-processing

The MDA method is employed to maximize the ratio of the inter-class scatter to the intra-class scatter with the purpose of accentuating the differences among the classes in a given data set. In the $Global$ level, upon completion of MDA's global reshaping action onto the training data instances, the data dimensionality has been reduced to $C - 1$, where C is the total number of classes in the data set. This transformation allows each RSPM classifier in the $Local$ level to model more accurately the similarity information of a corresponding training class and to achieve better classification accuracy performance in comparison with our previous work in [16].

Geometrically speaking, the instances of a training data set can be considered as point coordinates in a multidimensional space. MDA determines discriminating axes in this space which yield an optimal separation of the predefined class data groups. The first discriminant function maximizes the differences between the values of the dependent variable. The second function, based on the first factor, is orthogonal and uncorrelated to it, maximizing the differences between the values of the dependent variable, and so on. Though mathematically different, each discriminant function is a dimension that differentiates a case into categories of the dependent variable based on its values on the

independent variables. The first function is the most significant differentiating dimension. However, succeeding functions may also represent additional significant dimensions of differentiation.

For the multi-class case having, for instance, C total classes and d-dimensional data instances, MDA yields a linear transformation that maps the original feature space to a new l-dimensional feature space, where $l < d$, by maximizing the objective function given by Equation (3) [6].

$$\mathtt{J}(W) = \frac{|W^t \mathtt{S}_B W|}{|W^t \mathtt{S}_W W|}, \text{ where} \tag{3}$$

- W is the transformation matrix;
- $\mathtt{S}_B = \Sigma_{i=1}^{c} n_i (m_i - m)(m_i - m)^t$ is the between-class scatter matrix;
- $\mathtt{S}_W = \Sigma_{i=1}^{c} \Sigma_{x \in x_i} (x - m_i)(x - m_i)^t$ is the within-class scatter matrix;
- x is a training data instance;
- x_i is a training data instance in the i^{th} class;
- m_i is the mean value of all training data instances in the i^{th} class;
- m is the mean value of all training data instances; and
- n_i is the number of training data instances in the i^{th} class.

Assume that a training data set is given by matrix $\mathtt{X}$. Its projection $\mathtt{Y}$ is obtained from the MDA transformation through $\mathtt{Y} = W^t \mathtt{X}$, where the transformation matrix W is not necessarily unique and the columns of an optimal solution of W are the generalized eigenvectors that correspond to the largest eigenvalues in $\mathtt{S}_B w_i = \lambda_i \mathtt{S}_W w_i$ [6].

Generally, MDA is employed with high-dimensional data sets to demonstrate its dimensionality reduction features. For instance, assume that a data set has a total number of classes $C < d$. Then, l can be set to $C - 1$ as indicated in [6]. This value of l is adopted for our implementation of MDA for the proposed $Global$ level.

3.2. Local level

The *Local* level primarily consists of C-RSPM. C-RSPM has two modules, namely, the *Classification* and *Ambiguity Solver* modules, which naturally correspond to the two phases of a probabilistic supervised classification process. The *Classification* module is based on a powerful predictive model learning procedure known as RSPM. The number of classifier components in this module is adaptive to the number of classes required by any application. Each classifier is embedded with the RSPM algorithm and trained with a set of data instances belonging to a particular class of a given training set. As a result, each classifier either identifies a testing instance as normal (i.e., belonging to the class of its training instances) or abnormal. During the classification stage, all classifier components are executed concurrently, receiving and classifying the same incoming testing instance.

In brief, given a $N \times p$-dimensional normalized training data set matrix $\mathbf{Z}$ with N rows of instances and p columns of features, the main idea of the *Classification* module consists of (i) computing the robust estimate of the correlation matrix, (ii) computing the p eigenvector-eigenvalue pairs of the correlation matrix, that is, $(\lambda_1, \boldsymbol{E}_1)$, $(\lambda_2, \boldsymbol{E}_2)$, ..., $(\lambda_p, \boldsymbol{E}_p)$, and (iii) projecting the normalized training data instances into matrix $\mathbf{Y}$, also known as the score matrix. Then, all those principal components whose corresponding score matrix column vectors do not satisfy Equation (4) are discarded.

$$\phi < STD(\boldsymbol{R}_m) < \mathrm{a} + \mathrm{b} \times (1 - \exp(-\alpha)), \tag{4}$$

where ϕ is an adjustable coefficient set by default to the empirical value of 0.0001, α is the desired pre-set false alarm rate of the classifier, $STD(\boldsymbol{R}_m)$ is the standard deviation of the $(m)^{th}$ score column vectors of $\mathbf{Y}$, denoted by $\boldsymbol{R}_m$, satisfying the selection function and corresponding to the m^{th} eigenvector, a and b are both set to the mean of the standard deviation values of those score column vectors whose standard deviation values are greater than ϕ, and finally, $\boldsymbol{M}$ is defined as the row vector holding the indices of those eigenvectors satisfying Equation (4) and forming a new refined eigenspace. Utilizing the selected principal components in the refined eigenspace, a class deviation measure is computed for all training data instances, from which a class threshold measure is derived for the process of distinguishing normal and anomalous instances.

The *Ambiguity Solver* module captures and coordinates classification conflicts. It is possible that an instance is classified as normal by multiple classifiers, or not recognized as normal by any classifier. To handle the first issue of class label ambiguity, the $Ambiguity\ Solver$ module attempts to estimate the true class membership of an ambiguous instance by computing its *Attaching Proportion* measure with respect to each of the classes of the k classifier components claiming it as normal to their own training data sets. The *Attaching Proportion* reflects the degree of normality of a given ambiguous instance with respect to a training data set, where a smaller value indicates a stronger resemblance between an instance and the spatial distribution of a training data set. To handle the second issue, the module simply assigns an "Unknown" class label to the instance. For a more detailed description of C-RSPM, please refer to [16].

4 Experimental results and discussion

4.1. Experimental setup

Various experiments are conducted to evaluate and validate the performance and generality aspects of the proposed ASIC approach. The data sets used in these experiments exhibit different distributions, possess different quantities of numeric and nominal features and data instances, and are obtained from two well-known public data repositories, namely, the UCI [11] and UCR [12] archives.

Five groups of data sets are used to conduct the comparative analysis between the ASIC approach and other well-known algorithms including C4.5, KNN (K=5), Bayes' Nets (BN), Multi-layer Perceptrons (MLP), Random Forest (RF), Logistic, Support Vector Machines (SVM), and C-RSPM. In every experiment, a 10-fold cross-validation process is utilized to better evaluate the performance of all methods. The following describes the data groups employed in the experiments:

- Group 1: Xi (Face all) data set from the UCR Time Series Data Mining Archive [12]. It is composed of 2,250 instances within 14 classes and 131 numeric features. From them, 40 instances are selected from each class for training, corresponding to $\frac{2}{3}$ of the data set being randomly selected for training with cross-validation.

- Group 2: Leaf (Swedish Leaf) data set from the UCR Time Series Data Mining Archive [12]. It is composed of 1,125 instances within 15 classes and 128 numeric features. From them, 500 instances, or $\frac{2}{3}$ of the training data set, are randomly selected from each class for training with cross-validation.

- Group 3: 19 types of network attacks with 34 numeric and 7 nominal features, including Back, Teardrop, Smurf, and Neptune, among others, from the KDD CUP 1999 Data [11]. Different attack classes possess different numbers of instances, and thus the training data set is randomly selected for cross-validation purposes.

- Group 4: Credit (Credit-card) data set from the UCI KDD Archive [20]. It is composed of 690 instances within 14 classes and having 6 numeric and 10 nominal features. From them, $\frac{2}{3}$ of the instances from each class are selected for training with cross-validation.

- Group 5: Soybean data set from WEKA. It is composed of 683 instances within 19 classes and having 36 nominal features. From them, $\frac{2}{3}$ of the instances from each class are randomly selected for training with cross-validation.

The α value, or false alarm rate, of each RSPM classifier in the $Local$ level is set to 0.1%, which is a low false alarm rate value employed in many research areas [1][16]. In order to ensure fairness, all methods in the WEKA package had their parameters configured to promote their best performance [22]. In fact, WEKA's default parameter values are mostly appropriately configured for optimal performance, and thus we hereby mention only those parameters that were modified. For C4.5, the reduced error pruning option was set to true. For SVM, the polynomial kernel was set to quadratic and the $lowerOrderTerms$ option was set to true. The KNN classifier, implemented by IBk in WEKA, was configured to perform data set normalization. For RF, the number of randomly chosen attributes was set to $log_2(\#attributes + 1)$ [22], and the number of trees was set to 20. For the Logistic method, the ridge parameter was set to 1. For MLP, the $decay$ parameter was set to true, the number of hidden layers was set to $(\#attributes + \#classes) \div 2$, the $training\ time$ parameter was set to $1,000$, and the $validationSetSize$ was set to 20%. Finally, for BN, the $useADTree$ parameter was set to true, the $estimator$ algorithm was set to $BMAEstimator$ whose parameters $useK2Prior$ was set to true, and the $searchAlgorithm$ was set to $HillClimber$ whose parameter $markovBlanketClassifier$ was set to true.

4.2. Performance evaluation

Table 1 displays the classification accuracy of the ASIC approach in comparison to all the other methods with respect to each group of data sets. The standard deviation of the classification accuracy for each approach, resulting from the 10-fold cross-validation process, is also included in parentheses. A smaller standard deviation value indicates that the specific approach performs with a consistent stability, while a larger value indicates inconsistent or unstable performance.

From the results in Table 1, it is clearly observable that the ASIC approach outperforms all the other methods with accuracies above 95% for all data groups. This is indicative that for classification tasks with a high number of classes, the MDA component of our proposed preprocessing method allows the C-RSPM approach to distinguish among distinct classes with high accuracy, independent of the number of features or instances in the data set. In particular, the experiments with data Group 2 clearly demonstrates the ASIC accuracy performance improvement in comparison to that of C-RSPM. ASIC's classification accuracy is not only the highest (95.99%) but also significantly larger than the classification accuracy of all the other methods, which indicates obvious improvements of both robustness and stability in classification tasks with a high number of classes. The experiments with data Group 3

Table 1. Classification accuracy comparison among ASIC, C-RSPM, C4.5, KNN, BN, MLP, RF, Logistic, and SVM. Standard deviations are shown in parentheses.

Accuracy	Group 1	Group 2	Group 3	Group 4	Group 5
ASIC	99.16% (±0.07)	95.99% (±1.96)	98.41% (±1.12)	96.37% (±1.76)	97.21% (±1.04)
C-RSPM	99.05% (±0.08)	79.00% (±4.45)	98.28% (±1.44)	95.14% (±1.85)	95.63% (±1.68)
C4.5	64.50% (±7.19)	78.63% (±6.21)	95.63% (±1.86)	67.38% (±8.84)	91.10% (±1.08)
KNN	64.38% (±5.22)	78.61% (±6.54)	86.57% (±7.72)	41.65% (±13.59)	88.07% (±1.19)
BN	68.47% (±4.24)	83.76% (±5.87)	91.25% (±3.49)	67.38% (±9.35)	92.34% (±0.92)
MLP	81.38% (±3.62)	85.98% (±3.74)	93.77% (±3.54)	62.78% (±9.75)	89.50% (±1.67)
RF	63.85% (±4.81)	78.46% (±4.32)	89.98% (±3.03)	69.06% (±11.55)	92.17% (±1.53)
Logistic	73.69% (±7.35)	80.39% (±7.35)	90.87% (±2.74)	64.77% (±10.48)	88.25% (±2.44)
SVM	71.93% (±6.55)	84.16% (±3.86)	95.92% (±2.03)	65.54% (±17.68)	90.03% (±4.69)

demonstrate the potential advantages of the proposed ASIC approach in the intrusion detection domain, which has received a significant research attention in recent years.

Moreover, it can also be concluded from the experiments with data Group 4 where the number of nominal features is much larger than that of numeric features, and especially data Group 5 where the data set is composed entirely of nominal features, that the proposed WMCA component in the pre-processing method effectively derives numerical representations of nominal features. This allows ASIC to yield satisfactory classification results. Please note that ASIC's accuracy for these data groups is just as high as for those data groups with primarily numerical features (namely, data groups 1, 2 and 3). Thus, equipped with our proposed WMCA approach, ASIC can perform well with data sets where symbolic, rather than numeric, or mixed features are employed.

Furthermore, during the experimental process, ASIC has been observed to require significantly lower training and classification times than those of all the other methods. In comparison to C-RSPM, which has been observed as the fastest algorithm in [16], ASIC's overall required processing time is about $1/7$ for Group 1 and Group 2 data sets, about $2/5$ for Group 3 data set, about $3/5$ for Group 4 data set, and about $1/3$ for Group 5 data set. Table 2 shows the average combined time in seconds for the training and classification tasks, and for all the approaches under the same execution environment. The combined time measure was selected since it has been observed that the classification time for all methods is relatively negligible in comparison to the training time, except for KNN which is an instance-based method and does not generate predictive models but rather spends most of their effort in the classification task. From Table 2, it is clearly observable that the proposed ASIC approach presents a significantly lower combined time than those of all other methods and for all groups of data, which supports the high applicability of the proposed method in real-time demanding applications.

Table 2. Average combined training and classification times, in seconds, for ASIC, C-RSPM, C4.5, KNN, BN, MLP, RF, Logistic, and SVM.

Times	Group 1	Group 2	Group 3	Group 4	Group 5
ASIC	2.3	2.7	6.2	1.6	1.9
C-RSPM	15.6	18.5	15.3	2.7	5.9
C4.5	88.3	96.4	28.9	17	14
KNN	16000	18000	14000.0	4.3	5.7
BN	29.6	33.4	16.8	2.5	2.1
MLP	22000	23000	21000	327.4	1290.5
RF	62.8	73.2	18.5	8.3	7.1
Logistic	389.4	432.7	354.7	41.6	174.2
SVM	407.2	459.6	320.5	91.4	161.4

It was also observed from the experiments that the ASIC approach requires less memory storage and processing power to acquire the components attained during the training phase and required by the classification task. This is achieved by the employment of MDA in the $Global$ level, which results in a low initial computational load by each RSPM classifier. Finally, the diverse experiments with different data sets demonstrate that ASIC is a highly accurate, multi-class supervised classification approach presenting favorable operational merits such as low training and classification times and a lightweight characteristic. It performs better in both aspects of classification accuracy and operation benefits in comparison with our previously proposed C-RSPM [16] approach. All these characteristics and advantages make ASIC a desirable and powerful tool for various processing power & memory constrained real-world demanding applications.

5 Conclusion

In this paper, a novel ASIC supervised multi-class classification approach is proposed. The ASIC approach adopts the adaptive selection of information components and is facilitated with the capabilities of dealing with both nominal and numerical features and reducing the dimensionality of the data set in the $Global$ level, and the capability of classifying data instances in the $Local$ level. The functionality of the $Global$ level is achieved via the proposed WMCA/MDA-based data pre-processing method; while the functionality of the $Local$ level is achieved via our previously proposed C-RSPM approach. The classification accuracy and operational performance of the proposed ASIC approach were evaluated through comparison experiments

with various well-known supervised classification methods. The promising experimental results have demonstrated that ASIC maintains an average accuracy of over 95% on all cross-validation based experiments, revealing its superior stability in the task of generating predictive models capable of capturing even the smallest differences among different training classes and achieving several operational merits including faster algorithmic execution and low requirements on storage, memory, and processing power.

Acknowledgment

For Mei-Ling Shyu, this research was supported in part by NSF ITR (Medium) IIS-0325260 and National Oceanic and Atmospheric Administration (NOAA). This research was carried out in part under the auspices of the Cooperative Institute for Marine and Atmospheric Studies (CIMAS), a Joint Institute of the University of Miami and NOAA, cooperative agreement #NA17RJ1226. The statements, findings, conclusions, and recommendations are those of the author(s) and do not necessarily reflect the views of the funding agency. For Shu-Ching Chen, this work was supported in part by NSF EIA-0220562, NSF HRD-0317692, and Florida Hurricane Alliance Research Program sponsored by the National Oceanic and Atmospheric Administration (NOAA).

References

[1] K. Branden and M. Hubert. Robust classification in high dimensions based on the simca method. *Chemometrics and Intelligent Laboratory Systems*, (79):10–21, 2005.

[2] L. Breiman. Random forests. In *Machine Learning*, volume 45, pages 5–32, Boston, 2001. Kluwer Academic Publishers.

[3] Y. Cui and J. Weng. Appearance-based hand sign language recognition from intensity image sequences. *Computer Vision and Image Understanding*, 78(2):157 – 176, 2000.

[4] J.-W. Deng and H. Tsui. A novel two-layer PCA/MDA scheme for hand posture recognition. *Proc. 16th International Conference on Pattern Recognition*, 1(7):283–286, August 11-15 2002.

[5] A. Dong and B. Bhanu. Discriminant features for model-based image databases. *Proc. of the 17th International Conference on Pattern Recognition (ICPR 2004)*, 2:997–1000, 2004.

[6] R. Duda, P.E.Hart, and D.G.Stork. *Pattern Classification*. John Wiley & Sons, 2nd edition, 2001.

[7] M. Greenacre. *Theory and Applications of Correspondence Analysis*. Academic Press, London, 1984.

[8] M. Greenacre and J. Blasius. *Multiple Correspondence Analysis and Related Methods*. Chapman and Hall, Boca Raton, FL, USA, 2006.

[9] P. Hooper. Reference point logistic classification. *Journal of Classification*, 16:91–116, November 1999.

[10] I. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York, New York, 2nd edition, 2002.

[11] KDD. KDD Cup 1999 Data. *Available at http://kdd.ics.uci.edu/databases/kddcup99/*, Feb. 2007.

[12] E. Keogh, X. Xi, L. Wei, and C. Ratanamahatana. The UCR time series classification/clustering. *Available at http://www.cs.ucr.edu/∼eamonn/time_series_data/*, Feb. 2007.

[13] J. Malinen, V. Onnia, and M. Tico. Fingerprint classification based on multiple discriminant analysis. *Proc. of the 9th International Conference on Neural Information Processing (ICONIP02)*, 5:2469–2473, 2002.

[14] B. Moghaddam, W. Wahid, and A. Pentland. Beyond eigenfaces: Probabilistic matching for face recognition. In *Proc. of Int'l Conf. on Automatic Face and Gesture Recognition (FG'98)*, pages 30–35, Nara, Japan, April 1998.

[15] J. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Francisco, CA, 1993.

[16] T. Quirino, Z. Xie, M.-L. Shyu, S.-C. Chen, and L. Chang. Collateral representative subspace projection modeling for supervised classification. In *Proc. of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI06)*, pages 98–105, Washington D.C., USA, November 13-15 2006.

[17] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, and L. Chang. Principal component-based anomaly detection scheme. *Foundations and Novel Approaches in Data Mining*, 9:311–329, T.Y. Lin, S. Ohsuga, C.J. Liau, and X. Hu (editors), Springer-Verlag, 2006.

[18] Statsoft. Support vector machines. *Available at http://www.statsoft.com/textbook/stsvm.html*, Feb. 2007.

[19] J. Tou and R. Gonzalez. *Pattern Recognition Principles*. Addison-Wesley Publishing, Massachusetts, 1974.

[20] UCI. UCI KDD Archive. *http://kdd.ics.uci.edu/*, 2005.

[21] Weka. *Available at http://www.cs.waikato.ac.nz/ml/weka/*, Feb. 2007.

[22] I. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementation*, volume 45. Morgan Kaufmann Publishers, San Mateo, CA, 1 edition, 2000.

[23] S. Wold. *Pattern Recognition*, pages 127–139, 1976.

[24] Z. Xie, T. Quirino, M.-L. Shyu, S.-C. Chen, and L. Chang. A distributed agent-based approach to intrusion detection using the lightweight PCC anomaly detection classier. In *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC2006)*, pages 446–453, Taichung, Taiwan, R.O.C., June 5-7 2006.