

A multi-class IC package type classifier based on kernel-based nonlinear LS-SVM method

Yung-Hsiang Hung*

National Chin-Yi University of Technology

Department of Industrial Engineering and Management

No.57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung 41170, Taiwan (R.O.C.)

E-mail:hys502@ncut.edu.tw

Mei-Ling Huang

National Chin-Yi University of Technology

Department of Industrial Engineering and Management

No.57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung 41170, Taiwan (R.O.C.)

E-mail:huangml@ncut.edu.tw

Received 21 May 2012

Accepted 30 December 2013

Abstract

Least Squares Support Vector Machine (LS-SVM) is powerful to solve problems such as multi-class nonlinear classification. In this study, a LS-SVM with kernel-based was applied to multi-class IC packaging type dataset classification problem. In the first stage, the greedy search algorithm in feature selection that reduced 15 features to 9 features was used to improve the LS-SVM. Then, the hyperparameters of LS-SVM were tuned by using 10-fold cross-validation procedure and a grid search mechanism. This study compared the classification performance of two classifiers, namely the LS-SVM with RBF kernel, LS-SVM with polynomial kernel and NN method, in 63 classes of IC packaging type dataset in the full model and feature redundant model. The results showed that, for the classification problem of multi-class IC packaging type dataset, in the full model and reduced model, the classification performance of LS-SVM with RBF kernel is better than that of LS-SVM with polynomial kernel and NN classifier. The accuracy rates of the two classifiers in the full model were 81.12%, 72.38% and 81.68%, respectively, and 85.29%, 78.43% and 83.65% in the reduced model. In sum, regarding the multi-class IC packaging type classification problem, the classifier using the LS-SVM with RBF of the greedy feature reduced model has the highest classification performance in reducing the dataset complexity, with an accuracy rate at 85.29%.

Keywords: IC Packaging Type, Feature selection, Least Squares Support Vector Machine, Kernel-Based, Classification Accuracy.

1. Introduction

In recent years, with the increasing demands of IC products, the IC packaging industry has developed prosperously. Under continuous development and revolution of electronics manufacturing technologies, the market demands IC chips to be lighter, thinner, shorter, smaller, and with higher functionality and shorter product life cycle. As a result, the packaging technologies are being continuously renovated and developed. The micro-components have more and more I/O pins, the packaging thickness become thinner, and the packaging area becomes smaller. Generally, the packaging technology will affect the operational frequency, power consumption, reliability, and cost of products, and it is also the major factor determining whether the 3C product is competitive. Therefore, IC packaging structure and technological capabilities have become a critical part in IC design and manufacturing process¹. As the technological leap in the global IC industry constantly promotes the reduction in size and enhancement in packaging performance, IC design not only has to take into consideration of the specification, production process, circuit design, and reliability, but also technical problems of IC packaging, such as packaging size, I/O pin pitch, number of pins, chip size, frequency, transmission speed, and trust conditions³. Therefore, in the IC product design and development process, designers must first choose the appropriate IC packaging type and specifications before proceeding to circuit design and production process planning.

To IC designers in the upstream of the industry, how to accomplish product design that meets the market demands at reasonable costs and shortest possible time is the main challenge². Therefore, if the complete and timely IC packaging product knowledge database can be established to help IC designers gaining the latest and feasible knowledge on IC packaging types, so that they can determine the IC design approach and shorten the design time. However, related knowledge on IC packaging types is often scattered in different functional departments or relevant departments of the IC packaging foundry. Therefore, this study integrated the complete information on IC packaging product and applied the machine learning algorithm, in order to establish the automatic classifier for IC packaging product, thus to be served as the product knowledge

bank for IC design engineers at the product design and development stages. It can provide real-time operational platform for clients, sales representatives, customer service representatives, and management on related product type, thus effectively shortening the product planning time of IC design. This study summed up 15 characteristic features in relation to the commonly used packaging technology types, and collected a total of 2496 pieces of product data in 63 categories of end products manufactured by 5 different technological types, including TFBGA, LGA, PBGA, FCBGA and QFP, in order to carry out study on automatic classification problem of IC packaging product.

In recent years, due to the development of data mining technologies, the application of classification problem has been widely discussed and studied. The classification algorithms based on machine learning can analyze a large amount of data automatically or semi-automatically by computer according to known variables, and find potential and useful information, thus to identify meaningful relationships or rules³. For example, trend, feature and relevant process can be used to establish effective rules and models. Some well-known approaches to classification problems include (binary) Logistic Regression, K-Nearest Neighbor, Decision Trees, Naïve Bayes classifier, Linear and Quadratic Discriminant Analysis, Neural Networks, Rough set theory (RST), and more recently, Support Vector Machines (SVMs)⁵. Machine learning is closely related to statistics because both concern data analysis. However, the difference between machine learning and statistics in that machine learning emphasizes on the complexity of computation and realization of algorithms. Machine learning has been regarded as a new learning algorithm for various applications, such as pattern recognition⁶, audio classification and categorization⁷. Recently, SVMs have been proposed as popular tools for learning from experimental data. SVMs for classification and regression are powerful methods of machine learning. The benefit of SVM, as compared to statistical classifiers, is its general applicability to nonlinear problems. The Least Squares Support Vector Machine (LS-SVM) can transform a quadratic programming problem into a linear programming problem, thus reducing the computational complexity⁸. LS-SVM finds the solution by solving a linear system, instead of quadratic programming. SVMs are essentially

binary classifiers, and designed to separate only two classes from each other. However, in most of the real applications, multi-class classification is required.

This study applied LS-SVM to establish a multi-class IC packaging product classifier, which serves as a useful tool for IC packaging manufacturers to establish the information platform for IC packaging products and technological data query. In this article, we consider the problem of classification and feature selection for IC packaging type dataset with multi-class types. This study has two important contributions: 1) it demonstrates that this radial basis function (RBF) kernel measure is significantly better for IC packaging type dataset than the polynomial kernel used in SVM-based methods; 2) it applies a greedy search algorithm to reduce the dimensional complexity, and improves the computational time and classification accuracy for kernel-based LS-SVM classifier. Overall, IC package type dataset is an important topic to IC designers and is very significant in contribution to practical study. This study not only presents another case of classification. Since the products involve fast changes, and the manufacturing specifications and technologies involve many expert areas (electrical, electronics, materials, etc.), the variables are highly correlated, and difficult to be distinguished. This paper is organized as follows. Section 1 outlines the most significant technological challenges to continued performance growth in multi-class IC package type classification. Section 2 describes the data. Section 3 describes the experimental framework used to evaluate the performance of the Multi-class LS-SVM algorithms. Section 4 presents the results. Finally, further steps are discussed in conclusions.

2. Materials and method

2.1. Related studies on IC packaging

The main purposes of IC packaging include: 1) electric transmission, 2) signal transmission, 3) heat removal, 4) circuit protection. Generally speaking, IC packaging types can be divided into several technology types according to shape and function. Each technology can be regarded as a product family, such as Flip Chip Ball Grid Array (FCBGA), Land Grid Array (LGA) and QFP (Quad Flat Pack). Hung¹ collected expert opinions and

related knowledge in aspects of product development, electricity, product design, manufacturing process development, product reliability, and business engineering of IC packaging foundries, and integrated five product families, which are TFBGA Package Family, QFP Package Family, PBGA Package Family, LGA Package Family and FCBGA Package Family (product category). The related 14 features include: Package Size Range, Package Height (mm), Ball Pitch (mm), Lead Count (max.), Wafer Size (inch), Stacked Die Quantity, Substrate Layer, Frequency (GHz) max., MCM, Speed (Gbps) max., Power (W) max., Reliability (Level) and Reliability, for a total of 2496 pieces of product data. This paper demonstrated a neural network classifier model based on rough set feature selection in the multiclass IC package products classify. It compared the performance of neural networks, with and without rough set preprocessing, and discovered that when the number of features is low, the RSAR_BPNN classifier offers better performance. In the other, Hung² applied Variable Precision Rough Set (VPRS) and Back-propagation Neural network (BPNN) to establish the prediction models and classification criteria for the 5 product families. The results showed that the testing accuracy rate of the prediction model was 99.8%. Except for TFBGA and LGA that have conditional features hard to recognize, the test accuracy rates of the other 3 product families (PBGA, QFP and FCBGA) were 100%. In addition, the study summed 88 classification criteria. Hsieh⁹ added a feature to discuss whether the packaging manufacturing process uses the Ball Grid to solve the problem of difficulty in recognizing the conditional features in case of TFBGA and LGA. The results showed that the classification results based on approximate set could reach 100% after adding the characteristic feature of Ball Grid to correctly distinguish 5 product families.

2.2. IC package product dataset

Although Hsieh⁹ was able to correctly distinguish 5 product families by 100%, there was still limitations in practical application because in each product family, different end products could be extended given different shapes, functions, reliability conditions, and manufacturing conditions. Take LGA packaging family as an example, in case of SVTLGA packaging, the

maximum number of chip stacking is 2; the operational chip sizes are 8 inch and 12 inch; the packaging height is 1.0mm; the packaging size ranges from 5*5mm² to 19*19mm²; the pin distances at I/O ends are 0.5mm, 0.65mm, 0.8mm and 1.0mm; the highest electric frequency is 5GHz; the highest transmission speed is 10Gbps; the highest power is 4W; the condition of trust is Level 3; the highest circuit welding temperature is 260°C; the number of supporting layers is 2. The aforesaid 5 types of product families have a total of 63 types of different end products, as shown in Table 1. Therefore, this study applied the features and decision-making variables (product classification) in a total of 15 variables organized by Hsieh ⁹ as the features for this study, and recoded the 63 types of end products as the decision-making variables of the data set, as shown in Table 2.

2.3. LS-SVM

SVMs for classification and regression are powerful

Table 2. Features and specification range in the IC packaging dataset.

| Attributes | Specification range |
|--------------------------------|--------------------------|
| Package Family | TFBGA/LGA/PBGA/FCBGA/QFP |
| Package Size(mm ²) | 3x3~50x50 |
| Package Height (mm) | 0.5~3.75 |
| Ball Pitch (mm) | 0.4~1.27 |
| Lead Count (max.) | 8~1521 |
| Wafer size (inch) | 8/12 |
| Stacked Die Quantity | 1/2/3/4/5/6 |
| MCM | Yes/No |
| Substrate Layer | 2/4/6/8 |
| Frequency (GHz) max. | 2.4~10 |
| Speed (Gbps) max. | 0.4~10 |
| Power (W) max. | 2~10 |
| Reliability (Pre-con : Level) | 2/3 |
| Reliability (IR : DegC) | 245/260 |
| Ball Grid | Yes/No |

methods of machine learning. SVMs have several benefits, as compared to traditional classifiers (such as Neural networks) ¹⁰. The most important characteristic

Table 1. Description of the multi-class in the IC Package Type Dataset.

| Package Family | # Classes | Class name (# samples) | | |
|----------------|-----------|------------------------|-----------------|------------|
| TFBGA (632) | 16 | MCM-VFBGA(16) | 6S-STFBGA(16) | VFBGA(32) |
| | | MCM-STFBGA(64) | EDHS-TFBGA(32) | WFBGA(16) |
| | | MCM-TFBGA(64) | 3S-SWFBGA(8) | TFBGA(64) |
| | | 3S-S2TFBGA(64) | TFTBGA(64) | S2BGA(64) |
| | | 4S-STFBGA(32) | SVFBGA(16) | -- |
| | | 5S-STFBGA(16) | STBFGA(64) | -- |
| QFP (272) | 15 | EDHS-SQFP(12) | EDHS-LQFP(8) | TQFP(24) |
| | | EDHS-SQFP(12) | E-PADTQFP(24) | S2QFP(12) |
| | | MCM-QFP(12) | DGHS-QFP(12) | LQFP(28) |
| | | MCM-LQFP(28) | DDLQFP(28) | VQFP(4) |
| | | E-PADLQFP(28) | SLQFP(28) | QFP(12) |
| PBGA (352) | 8 | EDHS-MCMBGA(56) | MCMBGA(56) | PBGA(56) |
| | | EDHS-PBGA(56) | HS-MPBGA(8) | SPBGA(56) |
| | | MCMS2BGA(56) | MPBGA(8) | -- |
| LGA (424) | 17 | MCM-S2VTLGA(16) | MCM-LGA(64) | LGA(64) |
| | | MCM-UTLGA(16) | S2VTLGA(8) | SLGA(64) |
| | | MCM-XTLGA(8) | S2-LGA(64) | WTLGA(8) |
| | | MCM-VTLGA(16) | SVTLGA(16) | VTLGA(16) |
| | | MCM-SUTLGA(8) | SUTLGA(8) | UTLGA(16) |
| | | 4S-S2VTLGA(8) | WTLGA(16) | -- |
| FCBGA (816) | 7 | EHS-MCM-FCBGA(96) | EHS2-FCBGA(128) | FCBGA(112) |
| | | EHS-MP-FCBGA(32) | EHS-FCBGA(144) | -- |
| | | MCM-FCBGA(144) | DHA-FCBGA(160) | -- |
| Total class | 63 | -- | -- | -- |

is that they can handle very large feature spaces, and their generalization ability and computational efficiency are both independent of the dimension of the input space. Another benefit, as compared to conventional algorithms, is that SVMs can result in a globally optimal solution for the problem. In other words, it has a fast optimization algorithm, which can be applied to very large datasets, and has a very efficient implementation of the cross-validation. The ability of SVM to separate a set of positive data from a set of negative data that may or may not be linearly separable with maximum margin hyperplane makes it a good classification model¹¹. As a result, it is considered especially efficient in high dimensional classification problems¹²⁻¹⁵. Recently, an empirical study by Van Gestel et al.,¹⁶ has shown that LS-SVM is comparable to SVM in terms of generalization performance. This study demonstrated the use of the principle proposed herein by formulating a two-class classification problem, and considered the multiclass classification by using the Error Correcting Output Codes (ECOCs) strategy.

A linear SVM was developed to classify the data set, which contains two separable classes. It is assumed that given training data set $D = \{ (x_i, y_i) : x_i \in R^P, y_i \in R ; i = 1, 2, \dots, n \}$, and a P dimensional input space and $y_i \in \{-1, +1\}$ for both the output classes. It is also supposed that $\Phi(\cdot)$ is a nonlinear function that maps the input space into a higher dimensional feature space. LS-SVM maps the input vectors $x \in R^P$ into vectors $\Phi(x)$ of a higher dimensional feature space S , in order to solve a linear classification problem in a Φ - space. The solution for an indicator function can be written as:

$$y = f(x, w) = \text{sign} [w^T \Phi(x) + b]. \tag{1}$$

which is a linear classifier in a feature space, and can create a nonlinear separating hypersurface in the primal space. The simplest method would be to form the following learning problem

$$\min \frac{1}{2} \|w\|^2 + C (\text{number of misclassified data}). \tag{2}$$

where C is the regularization parameter, and a constant denoting a trade-off between minimizing the sum of total training error and maximizing the classification margin. Large C leads to small number of

misclassifications, small C means that the large classification margin is favored. Recently, LS-SVM is well-known to solve optimization problems with high accuracy^{10, 17}. Different from Vapnik's SVM¹⁸, the error term of the LS-SVM is defined as

$$y_i(x)[w^T \Phi(x) + b] = 1 - \xi_i, \quad i = 1, 2, \dots, n. \tag{3}$$

The LS-SVM technique can solve the following optimization problem:

$$\min_{w, b, \xi} \zeta(w, b, \xi) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i^2. \tag{4}$$

$$\text{s.t. } y_i(x)[w^T \Phi(x) + b] = 1 - \xi_i, \quad i = 1, 2, \dots, n \text{ and } \xi_i \geq 0.$$

where $\Phi(\cdot)$ is the feature map that is defined by a Mercer kernel, the weight vector w and the bias b are determined by minimizing the empirical sum of a specific margin-based loss function. The optimization problem is transformed into an unconstrained problem by the saddle point of the Lagrange function:

$$\mathcal{L}(w, b, \xi, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i \{ f_i(w^T \Phi(x) + b) - 1 + \xi_i \} \tag{5}$$

where α_i is Lagrange multiplier. Equation (5) is minimized with respect to w , b and maximized with respect to $0 \leq \alpha_i \leq 1$. $\|w\|^2$ and α are obtained from the LS-SVM, which is updated at each iteration. The decision function $f(x, w)$ in the dual space is obtained as

$$y(x) = f(x, w) = \text{sign} [\sum_{i=1}^n \alpha_i y_i K(x, x_i) + b] \tag{6}$$

where

$$\text{sign}(v) = \begin{cases} +1 & \text{if } v > 0 \\ -1 & \text{otherwise} \end{cases}.$$

is a signum function. $K(x, x_i) = (\Phi(x), \Phi(x_i))$ is a kernel function satisfying Mercer's conditions. Therefore, these training data points are often not linearly separable, and difficult for linear classifiers to make predictions accurately. Nonlinear kernel may provide better performance than linear kernels on moderate or large size samples. Although linear kernels are fast, Gaussian kernels have better performance in terms of accuracy¹⁹. In LS-SVMs, the most popular

kernel functions are polynomials and RBF with Gaussian kernels. Hsu and Lin²⁰ proposed that the radiation kernel function could classify non-linear and high-dimensional data. The complexity in search of parameters could be immediately reduced by adjusting σ^2 and γ . For the nonlinear LS-SVM, which uses the Gaussian kernel:

$$K(x, x_i) = \exp^{-\|x-x_i\|^2 / 2\sigma^2} . \quad (7)$$

where parameter σ^2 is the variance of the Gaussian function. More details concerning LS-SVM can be found in¹⁰.

A typical SVM is a two-class classifier that organizes all training sets into two classes. They are designed to separate only two classes from each other. However, in most of the real applications, multi-class classification is required, and the IC Package classification datasets are usually high-dimensional and imbalance.

3. Multi-class LS-SVM Classifier and Class Coding

Fundamentally, the binary LS-SVM classifier can be extended for multi-class problems as well. There are different options to extend SVM for multi-class cases, these include one-against-all, one-against-one, and ECOCs method. The underlying principle of ECOCs is to assign a particular binary code to each class, and train individual classifiers to identify each bit in the code Hill and Doucet²⁰. For example, 63 IC packaging type class can be represented using a 6-bit code ([-1,-1,-1,-1,-1,-1], [-1,-1,-1,-1,-1,+1], etc.) instead of a single output value y , the multiple output values $y^{(j)}$ with $j = 1, \dots, m_y$ where m_y is the number of output values. Each LS-SVM separates one class of IC packaging types from the rest of the database. The multi-class classification system is based on the output of the binary classifiers, as shown below:

$$\begin{cases} y^{(1)}(x) = \text{sign}[w^{(1)T} \Phi^{(1)}(x) + b^{(1)}] \\ y^{(2)}(x) = \text{sign}[w^{(2)T} \Phi^{(2)}(x) + b^{(2)}] \\ \vdots \\ y^{(m_y)}(x) = \text{sign}[w^{(m_y)T} \Phi^{(m_y)}(x) + b^{(m_y)}] \end{cases} \quad l = 1, 2, \dots, m_y \quad (8)$$

Thus, the LS-SVM multi-class classifier can be presented as

$$y^l(x) = \text{sign} \left[\sum_{j=1}^n \alpha_j^{(l)} y_j^{(l)} K^{(l)}(x, x_j) + b^{(l)} \right] \quad l = 1, 2, \dots, m_y \quad (9)$$

The purpose of such aggregations is to expand SVM binary classification to multi-class classification.

3.1. Feature selection for multi-class

It is well-known that the training of SVM is very slow, as compared with other classifiers, because there are too many attributes requiring extensive CPU and memory resources for processing. A feature selection process can be used to remove terms in the training sets that are statistically uncorrelated with the class labels. This will reduce the data dimensionality before passing it to the LS-SVM classifier, thus improving both efficiency and accuracy²². Wrapper is one of the well-known feature selection methods, and it often resorts to heuristic methods, and particularly, greedy approaches. Wrapper method is based on backward feature elimination, such as greedy algorithm, and on the best first heuristic, guiding a backward search mechanism through the feature space²³. However, in the most complex case of multi-class classification problems, LS-SVMs are frequently used. A greedy search algorithm is also used to find a suitable classifier that is likely to generate the selected individuals. This approach is used with a specific classification algorithm, as the outcome of the evaluation is used during the search.

3.2. Performance evaluation

The accuracy of a classifier refers to the ability of a given classifier to correctly predict the class label of new or previously unseen data. Performance on training set is definitely not a good indicator of performance on an independent test set. The commonly adopted estimation methods for accuracy estimation include holdout, random subsampling, k-fold cross-validation, and bootstrap methods²⁴. The holdout method, also known as the test sample estimation, partitions the data into two mutually exclusive subsets, called a training set and a test set, or holdout set. In general, two-thirds of the data are allocated to the training set, and the remaining one-third is allocated to the test set. The training set is used to derive the model, whose accuracy is estimated with the test set. However, a simple variant

forms the basis of an important statistical technique, called cross-validation. Cross-validation²⁵ is commonly used to obtain a reliable estimate of the test error for performance estimation, which is known as k-fold cross-validation. K-fold cross-validation is a technique to reduce the variance of the results of an (classification) experiment²⁶. Stratification means that all classes are present all the samples in the same proportion as in the whole data set. K-fold cross-validation provides a relatively fast and still precise estimate of the performance of such models. In the method, the original sample is split into K disjoint partitions. Then it trains a network using data from K-1 segments, and tests its performance by evaluating the error function, using the remaining segment. There will be K runs for the experiment, and the average of these results will be taken. That is, the error (predictive accuracy) estimates are averaged to yield an overall error (predictive accuracy) estimate. Finally, the final classifier is generated by learning from all the data. In general, stratified 10-fold cross-validation is recommended for estimating accuracy due to its relatively low bias and variance.

4. Experimental Results and Discussion

In this study, a real-word IC packaging type dataset was used to test the performance of LS-SVM multi-class classifier and NN classifier. First, the ECOC strategy was utilized to implement the 63-class IC packaging type classification. The LS-SVM methods for feature selection were all implemented in KULeuven's LS-SVMlab1.5 MATLAB/C Toolbox²⁷ under MS Windows XP. All experiments were conducted using in the Matlab software. The initial LS-SVM and NN were started with all 15 features, and a removal operator (backward elimination) that allows the lowest generalization error was used. For each classifier implement the 10-fold cross-validation to obtain the training error rate, and then feed in the test samples to obtain the test error rate. The CPU time for training and testing individual classifier is recorded as a measure of cost through the whole procedure.

In NN models, the architecture used in these experiments consisted of eight tan-sigmoid hidden units and one output purelin output unit. Weights were updated after presentation of each training vector using

a variable learning rate and momentum to speed training. There are two features of the Neural Network Toolbox that are designed to improve network generalization - regularization and early stopping. Classification results for NN models, for both training and test data, are presented along with the statistics of the other network paradigms in Table 3 as an average for all 10 networks.

In LS-SVM models, after feature selection, which removes nonfeature terms, the resulting "cleansed" training documents can be used for effective classification. For kernel types of multi-class IC packaging type classifier, RBF kernel was used with Gaussian kernel and polynomial kernel with degree 2, 4, 6, 8, 10. In case of LS-SVM with RBF kernel, two parameters were tuned, which are kernel setting (σ^2) and γ . A grid search technique using 10-fold cross-validation was employed to find the optimal parameter values²⁸. The model was trained with the optimal parameter settings with the whole training set, and tested with a test set. In this simulation, [0.01 0.1 0.5 0.75 1.0 1.25 1.50 1.75 2.00] and [0.01 1 5 10 15 20 25 30 35 40 45 50] were selected. To evaluate the classification accuracy, 10-fold cross-validation, which reserves a small amount of the training set to test the classifier's ability to generalize to novel data, was used.

In practice, training an SVM on the data set is slow and the extension of SVM to multiple classes is not as natural as NN. However, the experimental results demonstrated that, in the full model, RBF kernel has the classification performance accuracy rate over 81.12%; in the reduced model, this study extracted 9 variables by the greedy search algorithm based on backward feature elimination. As shown in Table 3, the reduced model not only effectively reduced the complexity of the original data, but also showed classification performance over 78.43% in LS-SVM with polynomial and RBF kernel classification results. In classification results based on different kernel functions, this study found that LS-SVM with RBF kernel is better than LS-SVM with polynomial kernel, in terms of classification accuracy rate for multi-class IC packaging type dataset classification performance in both the full model and reduced model. By comparison, the accuracy rate of LS-SVM with RBF kernel and LS-SVM with polynomial kernel in the full model is 82.12% and 74.23%, respectively, and 85.29% and 78.43%, respectively, in reduced model. The results are shown in Table 3. Fig. 1

Table 3. Average classification accuracy of multi-class IC packaging type dataset with NN, RBF and Polynomial kernels.

| | LS-SVM (RBF) | | LS-SVM (Polynomial) | | NN | |
|-------------------|---------------------|---------------------|---------------------|--------------------|---------------------|---------------------|
| | Full Model | Reduced Model | Full Model | Reduced Model | Full Model | Reduced Model |
| Training Accuracy | 0.8212 ± 0.0174 | 0.8429 ± 0.0151 | 0.7423 ± 0.0180 | 0.7913 ± 0.017 | 0.8257 ± 0.1043 | 0.8352 ± 0.1353 |
| Testing Accuracy | 0.8112 ± 0.0114 | 0.8529 ± 0.0161 | 0.7238 ± 0.0210 | 0.7843 ± 0.019 | 0.8168 ± 0.1036 | 0.8365 ± 0.1276 |
| Ave. CPU time(s) | 64.585 | 42.135 | 49.232 | 40.134 | 21.617 | 10.3084 |
| features | 15 | 9 | 15 | 9 | 15 | 9 |

illustrates the experimental results of executing the multi-class IC packaging type classifier with RBF kernel for 10 times. However, LS-SVM with RBF kernel classifier performance is better than other classification methods. The worst test accuracy rate is 0.8311, and the best is 0.8691.

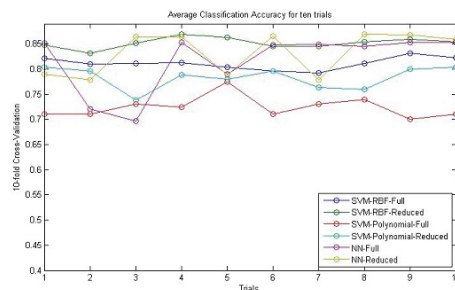


Fig. 1. Result of ten trials for multi-class IC packaging type classifier.

5. Conclusion

Greedy search algorithm has proved to be an effective method of feature selection for a multi-class LS-SVM with kernel learning methods. Construction of a LS-SVM module requires two parameters: γ , which is the regularization parameter, and σ , which is the Gaussian bandwidth. The γ determines the trade-off between the fitting error minimization and the smoothness. Therefore, 10-fold cross validation and a grid search mechanism that provide the number of hyperparameters to be tuned is relatively small. This study applied the LS-SVM with RBF kernel and LS-SVM with polynomial kernel classifiers to discuss the classification performance of 63 classes of IC packaging type dataset in the full model and reduced model. The results showed that the data complexity is simpler and the classification performance is better in the reduced

model than in the full model. In addition, the classification results of LS-SVM with RBF kernel showed better stability than the LS-SVM with polynomial kernel in both the full model and reduced model (std for RBF kernel = 0.0161, std. for polynomial kernel = 0.019) with classification performance as high as 85.29%. In sum, the results demonstrated that classifiers integrating LS-SVM with RBF kernel with greedy feature selection can effectively enhance the classification speed and performance of the multi-class IC packaging type datasets. Based on previous studies on IC package database, this study used a Multi-class classifier to assist IC design engineers to quickly evaluate the suitable packaging technology at the design stage, so as to increase the operating efficiency and shorten the time course. Moreover, by developing the IC packaging manufacturing prediction technology, IC package information can be integrated into the design process, in order to provide IC design process with packaging awareness, and assist IC designer to shorten the design time course and reduce the manufacturing cost.

Acknowledgement: The authors would like to thank for the anonymous referees for their careful reading of the paper and several suggestions which improved the paper. This work has been supported by the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC 98-2221-E-167-010-MY2. The author thanks these sources of support.

References

1. Y. H. Hung, A Neural Network Classifier with Rough Set-based Feature Selection to Classify Multiclass IC Package Products, *Advanced Engineering In formatics*, **23**(3) (2009) 348–357. doi: 10.1016/j.aei.2009.04.001
2. Y. H. Hung, Improving Classification Accuracy of IC Packaging Products Database Based on Variable

- Precision Rough Sets, *Information Technology Journal*, **7**(3) (2008) 440–449. DOI: 10.3923/itj.2008.440.449
3. P. Gregory, Advanced Interconnect Technologies. (Available from: <<http://www.ecnchinamag.com>>, 2007).
 4. M.J.A. Berry and G Linoff, *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management* (Wiley Computer, New York, 2004).
 5. I.H.m. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd Edition. (Elsevier, 2011).
 6. P. Dhanalakshmi, S. Palanivel and V. Ramalingam, Classification of audio signals using AANN and GMM, *Applied Soft Computing*, **11**(1) (2011) 716–723. doi:10.1016/j.asoc.2009.12.033
 7. A. Khan, M. Farhan and A. Ali, Speech Recognition: Increasing Efficiency of Support Vector Machines, *International Journal of Computer Applications*, **35**(7) (2011) 17–21. doi: 10.5120/4413-6131
 8. J.A.K. Suykens and J. Vandewalle, Least squares support vector machine classifiers, *Neural Processing Lett.* **9**(3) (1999) 293–300. DOI: 10.1023/A:1018628609742
 9. H.C. Hsieh, *Study of Application of Rough Set Theory in IC Package Classification*, (Master Thesis, Department of Institute of Industrial Engineering and Management, Master Thesis, National Chin-Yi University of Technology, Taiwan, Republic of China, 2008).
 10. J.A.K. Suykens, T.J. Gestel, D. Brabanter, B. Moor and D.J. Vandewalle, *Least Squares Support Vector Machines*, (World Scientific, working paper, Singapore, 2002).
 11. T. Joachims, A statistical learning model of text classification for support vector machines, *In SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, (ACM Press, 136, NY, USA, 2001), pp. 128.
 12. V. Vapnik, *Statistical Learning Theory*, 1st Edition, (Wiley, New York, 1998).
 13. An Wensen, A. Cecilio and S. Yanguang. Support Vector Regression with Interval-Input Interval-Output, *International Journal of Computational Intelligence Systems*, **1**(4) (2008) 299–303.
 14. R.A. Anjum, Y.Q. Zhang and Robert W. Harrison, Granular Decision Tree and Evolutionary Neural SVM for Protein Secondary Structure Prediction, *International Journal of Computational Intelligence Systems*, **2**(4) (2009) 343–352. doi:10.1080/18756891.2009.9727666
 15. X. Liu, H. Fu and W. Lin, A Modified Support Vector Machine model for Credit Scoring, *International Journal of Computational Intelligence Systems*, **3**(6) (2010,) 797–803. doi: 10.1080/18756891.2010.9727742
 16. T.V. Gestel, J. Suykens, B. Baesens, S. Viaene, J. Vanthienen, G. Dedene, B. Moor and J. Vandewalle, Benchmarking least squares support vector machine classifiers, (Internal Report, E SAT-SISTA, K.U.Leuven, Leuven, Belgium, 2000), pp. 00–37.
 17. K.S. Chua, Efficient computations for large least square support vector machine classifiers, *Pattern Recognition Letters*, **24**(1-3) (2003) 75–80. doi:10.1016/S0167-8655(02)00190-3
 18. V. Vapnik, *Statistical Learning Theory*, (New York: John Wiley 1998).
 19. D.M.J. Tax and R.P.W. Duin, Support vector domain description, *Pattern Recognition Letters*, **20**(11-13) (1999) 1191–1199. doi:10.1016/S0167-8655(99)00087-2
 20. C.W. Hsu and C.J. Lin, A comparison of methods for multiclass support vector machines, *IEEE T. Neural Networ.* **13**(2) (2002) 415–425. doi:10.1109/72.991427
 21. S.I. Hill and A. Doucet, A Framework for Kernel-Based Multi-Category Classification, *Journal of Artificial Intelligence Research*, **30** (2007) 525–564. doi:10.1613/jair.2251
 22. S. An, W. Liu, S. Venkatesh and R. Tjahyadi, A Fast Feature-based Dimension Reduction Algorithm for Kernel Classifiers, *Neural Processing Lett.* **24**(2) (2006) 137–151. DOI: 10.1007/s11063-006-9016-7
 23. X.W. Chen, X. Zeng and D.V. Alphen, Multi-class feature selection for texture classification, *Pattern Recognition Letters*, **27**(14) (2006) 1685–1691. doi:10.1016/j.patrec.2006.03.013
 24. W. Jiang and R.A. Simon, A comparison of bootstrap methods and an adjusted bootstrap approach for estimating the prediction error in microarray classification, *Statistics in Medicine*, **26**(29) (2007) 5320–5334. DOI: 10.1002/sim.2968
 25. M. Stone, Cross-validatory choice and assessment of statistical predictions, *Journal of the Royal Statistical Society Series B*, **36**(2) (1994) 111–147. doi:10.2307/2984809
 26. L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees*, (Wadsworth, 1984).
 27. K. Pelckmans, J. Suykens, T.J. Gestel, de. Brabanter, L. Lukas, B. Hamers, B. Moor and D.J. Vandewalle, *LS-SVMlab: A MATLAB/C toolbox for Least Squares Support Vector Machines*, (Available at: <http://www.esat.kuleuven.ac.be/sista/lssvmlab>, accessed 17 July 2008).
 28. V. Cherkassky and Y. Ma, Practical selection of SVM parameters and noise estimation for SVM regression, *Neural Networks*, **17**(1) (2004) 113–126. doi:10.1016/S0893-6080(03)00169-2