Intelligent Classification of Orange Growing Areas by Using Near-Infrared Spectra

Xia Jiang, Yifan Cai, Simon X. Yang and Gauri S. Mittal

School of Engineering, University of Guelph
Guelph, Ontario, N1G 2W1, Canada
E-mails: jiangx7@hotmail.com; {ycai, syang, gmittal}@uoguelph.ca

ABSTRACT

Near-Infrared spectroscopy (NIR) is a fast and non-destructive method to identify orange growing areas. In this paper, a principal component analysis (PCA) approach was used to obtain the features of orange NIR spectra by reducing the divisions in the analysis. An artificial neural network (ANN) was developed to achieve enhanced classification accuracy, while a support vector machine (SVM) model was proposed for higher classification accuracy. A hybrid genetic algorithm (GA) SVM model was designed, with the most valuable data from the PCA selecting by GA. The simulation results showed that the hybrid GA-SVM classifier achieved the best accuracy of 89.717%.

Keywords: Near-Infrared spectroscopy, principal component analysis, artificial neural network, support vector machine, Canada.

1. INTRODUCTION

The ability to trace the growth area of oranges is one of the greatest concerns to juice manufacturers. The geographical areas of orange producing determine the quality and the flavor of the productions. Identification of the different growing areas with high accuracy and reliable classification results becomes an essential issue to the orange industry.

In this paper, the Near-Infrared Spectroscopy (NIR) generated with high efficiency to obtain the orange characteristic datasets. A total number of 1589 samples were collected. In this study, three classifiers were developed to distinguish 16 places of orange growing areas. The total dataset was randomly divided into two parts: training set and testing set. The classifier was adjusted according to the training data's error, and an ANN model [1] was proposed as it has fast learning rates, in addition to the ability of adjusting the connection weights between each layer. Furthermore, a SVM classifier [2] was proposed, as it could avoid the dram back of the ANN model. In addition, a hybrid GA-SVM

[3] was created which offers the best results in this study. After data was processed by PCA [4], it was then fed into a GA algorithm [5], in order to find optimal data. This resulted in producing a more valuable dataset for the SVM model to process.

2. The Proposed Approaches

In this study, PCA was used to utilize the extraction feature of raw orange NIR spectra. The purpose of PCA was to keep enough information for ANN, SVM and hybrid GA-SVM classifiers on the basis of reduction in the dimensionality of datasets.

2.1 The proposed artificial neural network (ANN) classifier

The ANN model was developed for orange original growing provenance classification. There were three layers. In this proposed ANN, a two-way-neural network was built, which involved the forward transmission of the inputs and the backward propagation of errors in Figure 1.

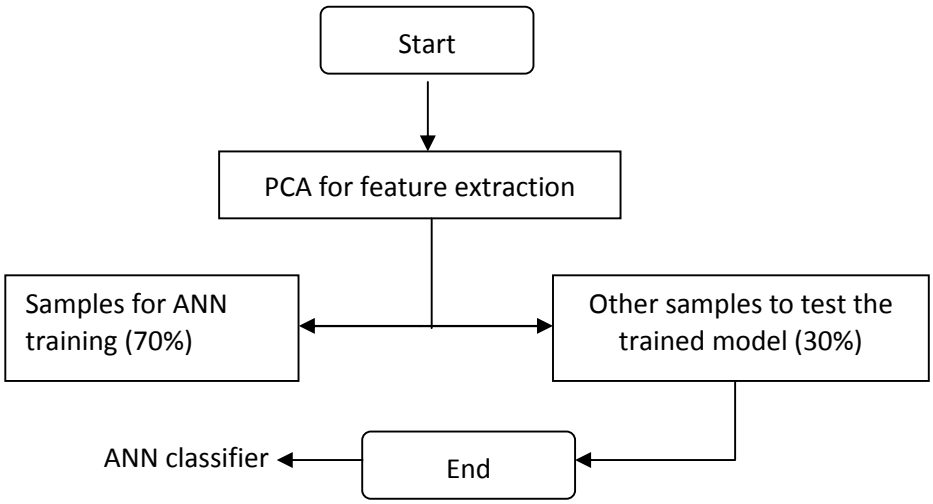

Figure 1. Chat of ANN classifier for orange original growing provenance.

Step 1: Get the input elements into the ANN system, the output of neuron *j* in the hidden layer could be obtained as

$$H_j = f\left(\sum_{t=0}^{n} \mu_{ij} y_i\right) , \quad (1)$$

where y_i represents 25 input nodes. In this study, these input nodes of the PCA results (PCs) have been normalized in the range of [-1, 1]; all the PCs obtained by PCA are normalized as y_i, which are the inputs of the following three methods; *n* represents the number of ANN input nodes, and the

range of n is [3,18]; μ_{ij} represents the weight among the hidden nodes between the input node i and the hidden node j, and f is the activation function of the hidden layer and it is defined as

$$f(a) = \tanh(\sim) = \frac{e^{\sim} - e^{-\sim}}{e^{\sim} + e^{-\sim}} \quad , \tag{2}$$

Step 2: Get the final output results. The final output of ANN is a computation on all the nodes on hidden layer and summing them with connection weight as

$$Z = \sum_{j=0}^{m} v_j h_j \, , \tag{3}$$

where h_j represents the output of neuron j, and has been computed with the inputs of first layer, and the value range of h_j is [0, 13]. The parameter M represents the number of nodes on the hidden layer, while V represents the weight between hidden nodes and the output, for example, connection weight between the input node i and the hidden node j is v_j.

Step 3: Obtain the error backward propagation part. After putting the training sample into the algorithm, the network is formed, and the error range could be obtained. An error function is needed to adjust connection weights and thresholds, and it is express as

$$E = \frac{1}{2} e^2 = \frac{1}{2} (t - z)^2 \, , \tag{4}$$

where e represents the target error value, t is the test sample target value, and z is the prediction value of the ANN.

2.2 The proposed SVM model

The SVM classifier was evaluated as a way to research about the relationship between the orange original growing provenance and the orange NIR spectra. The total process can be conclude as follows:

Step 1: Illustrate the basic binary classification problem with the essence of SVM. Assuming set of samples express as $(\vec{y}, z)$, $\vec{y}_i \in R_d \, (i = 1, 2, \cdots, N)$ represents the input vector, and the corresponding class label is $y \in [-1, +1]$. A hyperplane of linear decision is defined as

$$g(\vec{y}) = \vec{\mu} \cdot \vec{y} + b \, , \tag{5}$$

where $g(\vec{y})=0$ represents the separating hyperplane, $\mathrm{sgn}\left(g(\vec{y})\right)$ is the corresponding classier, and the variable $\vec{\mu}$ represents the weight vector and *b* is the bias. The signal "·" is the inner product operator.

The following formula is used to compute the Euclidean distance between any instances $\vec{y}_i$ to the separating hyper-plane as

$$¢ = \frac{1}{\left\|\vec{w}\right\|}\left|g(\vec{y})\right|, \tag{6}$$

where $\left\|\vec{w}\right\|$ is the second norm of vector $\vec{w}$. Thus, maximizing the margin between $g(\vec{y})=1$ and $g(\vec{y})=-1$ is equivalent to minimizing $\left\|\vec{w}\right\|^2/2$.

Step 2: Choose the kernel function. It is important to select the function, as it reflects the essential relation between linearly separable data in *H* and nonlinearly separable data in R^d.

Step 3: Extend SVM into a multi-class classifier. It is hard to limit the real-world applications to binary classification; and it can be easily increased to multi-category issues.

2.3 The proposed GA-SVM model

The PCA method for feature extraction used to preprocess the orange NIR spectra. The whole process of a standard GA is as follows:

Step 1: Select a set of initialized samples randomly from first 25 PCs. The feasible solution number is pre-configured and the rule of choosing the value number is to obtain 100% information of orange NIR spectra. *A* range of [1, 25] positive integers formed resulting from the individual input subset $y=(y_1, y_2, \cdots, y_n)$ in the population. The number of inputs is the length of chromosome l.

Step 2: Describe the fitness evaluation within the dashed rectangle. Measure the fitness value for each individual to predict the accuracy of SVM model. The fitness function could be expressed as

$$f_i = \sum_{i=1}^{n} y_i\,, \tag{7}$$

where f_i is the raw fitness for the i^{th} individual, y_i is selected valuable datasets though the input datasets. Assuming there are *n* fitness cases, the mean fitness can be given by

$$\overline{f} = \frac{1}{n}\sum_{i=1}^{n} f_i\,. \tag{8}$$

Step 3: Adapt fitness value biased roulette method to conduct parent selection. If the fitness value of input subset in population is larger, there will be higher probability that the input subsets are parents. Those parents will benefit more sensitive genetic information for the SVM classifier.

Step 4: Mate the parents and conduct the reproduction.

It could select the high value features or elements for classification. The GA part could provide the valuable training data for SVM model to simulate.

3. Simulation Results and Discussions

In this experiment, the raw data were processed by the PCA method which is used to find the most valuable parameters. The MATLAB GUI (Graphical User Interface) was used to create an interface. Before processing the program, the PCA analysis must be completed in advance.

3.1 Graphical user interface tool of MATLAB

A graphical user interface (GUI) is a pictorial program in MATLAB. There three elements are required in the principal: components, figures and callbacks. The components supply the input methods and display the figures. Components of GUI are arranged in a figure. Callbacks is a method to perform an action which is code executed in response to an event.

3.2 The results using the proposed ANN method

For the proposed ANN model, the number of hidden layers and learning rate were the significant elements. In this paper, two elements were selected by empirically testing. Both the amount of hidden layers and learning rate critically impact the function. The recognition rate can be reduced by too many or too few hidden layers. Also, the size of training epoch impacts the ability of the whole system. The larger size of the epoch cause the training set over fit. However, the overly small size of epoch could impair the recognition of system. This network was trained by Levenbery-Marquardt back-propagation algorithm (trainlm). The number of hidden neurons was 13 and the learning rate was 0.9. The highest accuracy in this experiment was achieved at 81.4528%.

According the experiment results, the definition of training, testing and validation are shown in Figure 2. Training is presented to the training network, and the network is adjusted according to its error. Validation is used to measure network generalization, and to halt training when generalization stops improving. Testing has no effect on training and provides an independent measure of network performance during and after training.

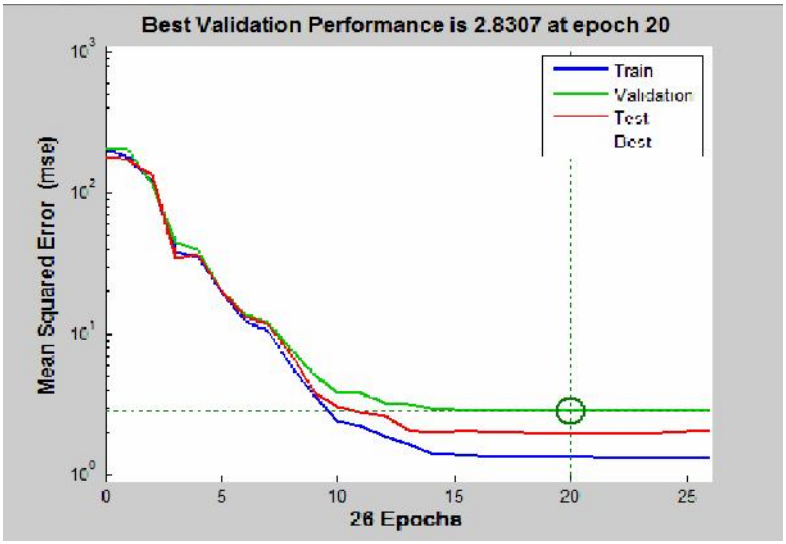

Figure 2. Best validation performances within the 13 hidden neurons.

A regression (*R*) plot is shown in Figure 3, which presents the relationship between the outputs of the network and the targets. The axes symbolize the training data, validation data and testing data. When the outputs equal to targets, the perfect results are presented by dashed line in each axis. The solid line indicated the best fit linear regression line which is between outputs and targets. The value of *R* is an indication of the outputs and targets relationship. If R=1, there is a linear relationship between outputs and targets. If *R* is close to zero, there is nonlinear relationship between outputs and targets. In these experiments, the training data indicates a good fit. With training data and testing datasets, the overall R, which is the correlation between the output and the target, was 0.96078, which is closed to 1, which means that there is strong correlation.

3.3 The results using the proposed SVM model

The SVM algorithm aims to finding OSH so that the samples from different classes can be separated as accurately as possible. In this study, the training and testing dataset were the same as the previous proposed method, which occupied 70% and 30% of the whole raw data. After several experiments, the results reflect that the SVM model could obtain high prediction accuracy from the training set.

In order to highlight the advantages of the SVM model, the ANN model was tested in the same condition. As shown in the Table 1, the SVM model provided better P_a than ANN model in all the cases. The highest improvement result was 14.5283% when the number of 20 PCs. The best performance of SVM model was achieved at 86.9811% in 20 PCs.

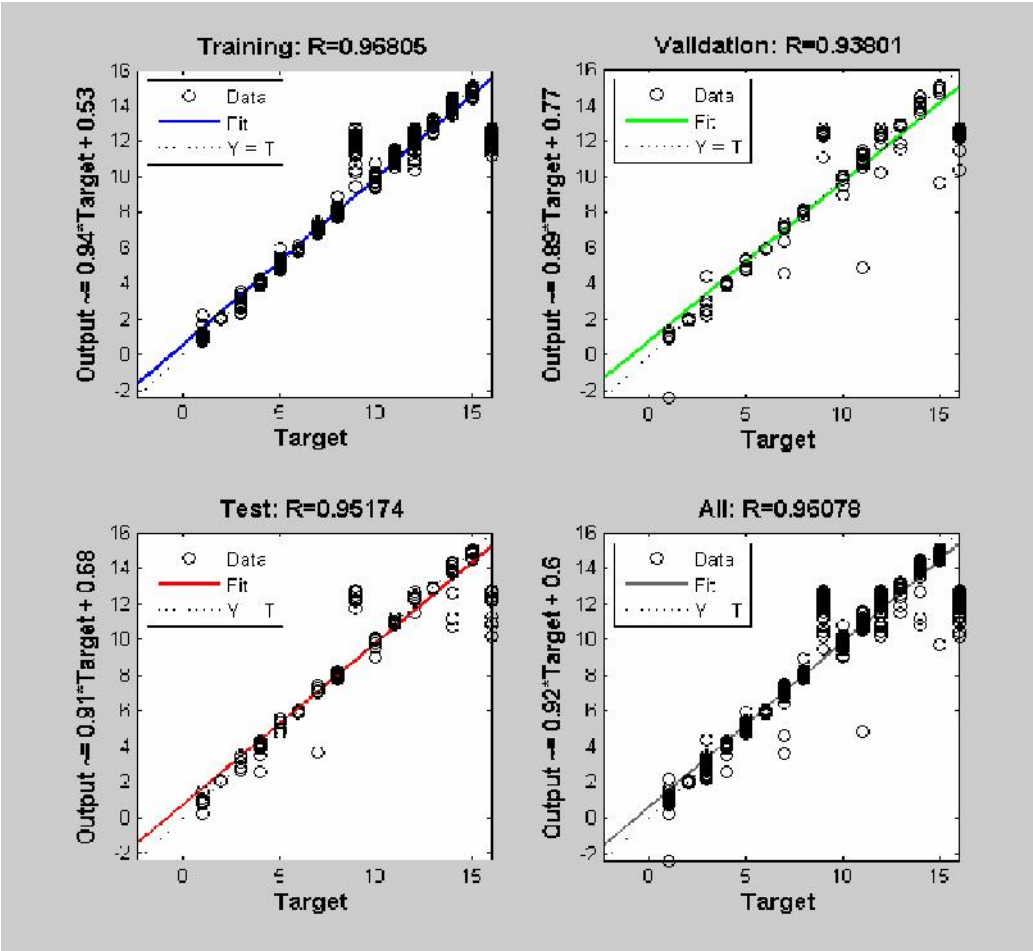

Figure 3. Regression results about the relationship between the outputs of the network and the targets.

3.4 The results using the proposed GA-SVM model

The whole procedure of GA is to obtain the best input set for SVM model. One point crossover and uniform mutation are applied. Eventually, after the generation the genes with higher fitness will be obtained. In the experiment, the raw data was processed by PCA method same as the previous models. The parameters of this hybrid GA-SVM model were settled as: population size (200);

Table 1. Accuracy of SVM model for orange juice growing area classification. P_a is the prediction accuracy; N is the number of principal components PCs.

N	5	10	15	20	25
SVM P_a(%)	85.6604%	85.8491%	84.8763%	86.9811%	85.6604%
ANN P_a(%)	74.7179%	81.4528%	76.2264%	72.4528%	72.1132%

size (100); probability of crossover: (0.7); probability of mutation: (0.001). The accuracy was achieved at 89.717%. The proper parameters was crucial for the quality of outputs. Also, the feasible input for the GA was from 1 to 25 in positive integers. The population size relate to the search speed and convergence rate. Too small or too large of the population size could affect the results. Too small could lose some good solutions. But too large would increase the search complexity and search time. This is no arithmetical method to select the size of population in the researches.

In this study, several population sizes were tested at the final stage in hybrid GA-SVM model. In general, the crossover probability is good to choose within the range of [0.5, 1], the rate of mutation is good smaller than 0.1. In this experiment, to find the proper solutions and keep the search space variety, different rates are tested.

4. Conclusion

From experiment result, the achievement of orange growing areas classification could be increase to 89.717% by using hybrid GA-SVM model. The comparison experiment results indicate that the ANN had the ability to self learn and organize, resulting in ease of use, and it can detect possible interactions by the predictor variables. The SVM model with the kernel function can be used to process the nonlinear input data, and it could perform well even if the datasets have many attributes. For the last method, when GA is combined with SVM model, classification is greatly enhanced. The benefits of GA assist the SVM model to decide the numbers of support vectors, and the hybrid GA-SVM obtained higher significantly accuracy than only SVM model.

5. References

[1] Dettmar, H. P., Barbour, G. S., Blackwell, K. T., Vogl, T. P., Alkon, D. L., Fry, F. S., and Chambers, T. L. (1996). Orange juice classification with a biologically based neural network. *The Journal of Computers and chemistry*, 20(2), pp.261-266.

[2] Alba, E., Garcia-Nieto, J., Jourdan, L., and Talbi, E. G. (2007). Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms. *In Evolutionary Computation, 2007. CEC 2007. IEEE Congress* on pp. 284-290.

[3] Ding, Y., X. Song and Zen, Y. (2008). Forecasting financial condition of Chinese listed companies based on support vector machine. *The Journal of Expert Systems with Applications*, 34(4), pp.3081-3089.

[4] Erenturk, S., and Erenturk, K. (2007). Comparison of genetic algorithm and neural network approaches for the drying process of carrot. *Journal of Food Engineering*, 78(3), pp.905-912.

[5] Karuppathal, R. and Palanisamy, V. (2013). Hybrid GA-SVM for feature selection to improve Automatic Bayesian classification of Brain MRI Slice. *Life Science Journal*, pp.10.