Research Article
Prediction of the Yield of Enzymatic Synthesis of Betulinic Acid Ester Using Artificial Neural Networks and Support Vector Machine

[1]Run Wang, [2]Qiaoli Mo, [1]Qian Zhang, [3]Fudi Chen and [3,4]Dazuo Yang
[1]College of Light Industry, Textile and Food Science Engineering,
[2]College of Chemistry, Sichuan University, Chengdu, Sichuan 610064, China
[3]Key Laboratory of Marine Bio-Resources Restoration and Habitat Reparation in Liaoning Province,
Dalian Ocean University, Dalian 116023, China
[4]College of Life Science and Technology, Dalian University of Technology, Dalian 116021, China

Abstract: 3β-*O*-phthalic ester of betulinic acid is of great importance in anticancer studies. However, the optimization of its reaction conditions requires a large number of experimental works. To simplify the number of times of optimization in experimental works, here, we use Artificial Neural Network (ANN) and Support Vector Machine (SVM) models for the prediction of yields of 3β-*O*-phthalic ester of betulinic acid synthesized by betulinic acid and phthalic anhydride using lipase as biocatalyst. General Regression Neural Network (GRNN), Multilayer Feed-forward Neural network (MLFN) and the SVM models were trained based on experimental data. Four indicators were set as independent variables, including time (h), temperature (°C), amount of enzyme (mg) and molar ratio, while the yield of the 3β-*O*-phthalic ester of betulinic acid was set as the dependent variable. Results show that the GRNN and SVM models have the best prediction results during the testing process, with comparatively low RMS errors (4.01 and 4.23 respectively) and short training times (both 1s). The prediction accuracy of the GRNN and SVM are both 100% in testing process, under the tolerance of 30%.

Keywords: Artificial neural network, betulinic acid ester, biocatalyst, support vector machine

INTRODUCTION

3β-*O*-phthalic ester of betulinic acid has clinical potential as an anticancer medicine, which can be synthesized from reaction of betulinic acid and phthalic anhydride using lipase as biocatalyst (Fig. 1). It has a variety of properties including inhibition of antibacterial, anti-inflammatory, anti-malarial, anthelmintic, antioxidant and Human Immunodeficiency Virus (HIV) (Yogeeswari and Sriram, 2005). According to previous studies, the introduction of polar groups at the C-3 and C-28 positions also highly increases the anticancer activity and hydro-solubility (Thibeault *et al*., 2007; Gauthier and Legault, 2008). However, the practical applications of betulinic acid in the pharmaceutical and medical industry is deeply constrained because it is insoluble in water (approximately 0.02 mg/mL) under ordinary circumstances, leading to great difficulties in preparation of injectable formulations for biological experiments and decreases the bioavailability.

The detailed approaches for the synthesis of 3β-*O*-phthalic ester of betulinic acid based on chemical catalytic esterification have been reported by previous research reports (Mukherjee *et al*., 2004; Kvasnica *et al*., 2005; Mukherjee *et al*., 2006; Rajendran *et al*., 2008), which have several disadvantages (e.g., high energy consumption and by-products) (Yasin *et al*., 2008). Compared with traditional chemical approaches, the application of enzymes in organic synthesis offers a series of advantages, including high catalytic efficiency, high selectivity, mild reaction condition and high product purity and quality (Loughlin, 2000; Zarevúcka and Wimmer, 2008). However, the best detailed conditions for the synthesis are difficult to obtain due to the large-scale and complex laboratory experiments. Moghaddam *et al*. (2010) used Artificial Neural Network (ANN) models to develop models for predicting the yield of enzymatic synthesis of betulinic acid ester. They successfully found that the quick propagation algorithm was the best model during their computational experiments. Nevertheless, previous ANN models are based on comparatively complex operations and the selection method of the best ANN model was based on a limited number of results, which are not robust and user-friendly enough, compared with

Corresponding Author: Dazuo Yang, Key Laboratory of Marine Bio-Resources Restoration and Habitat Reparation in Liaoning Province, Dalian Ocean University, Dalian 116023, China, Tel./Fax: +86-0411-84762290
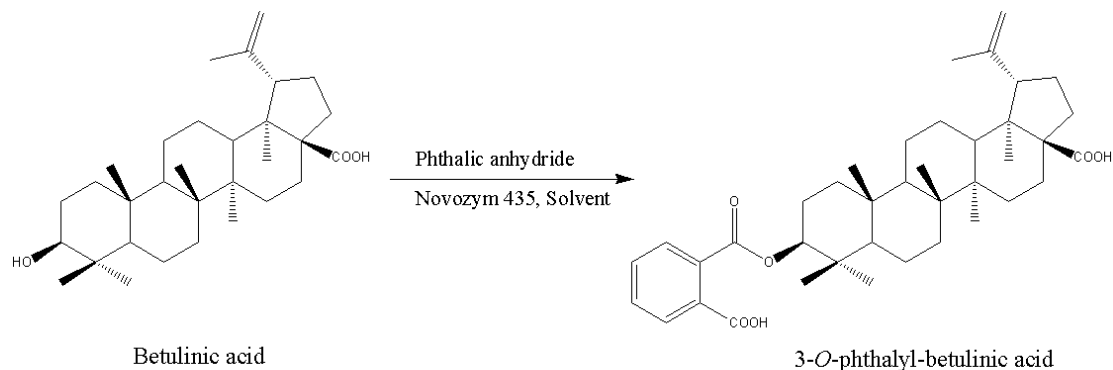
Fig. 1: Reaction between betulinic acid and phthalic anhydride utilizing Novozym 435 as biocatalyst

latest machine learning models. Here, we aim at using novel and user-friendly approaches of ANN models and Support Vector Machine (SVM) to train the data of the yield of enzymatic synthesis of betulinic acid ester and obtain a series of best machine learning models for the prediction of the yield. Comparisons are made in order to determine the most suitable machine learning model for the prediction.

MATERIALS AND METHODS

Data set: According to previous research, the synthetic conditions of enzymatic synthesis of betulinic acid ester includes time (h), temperature (°C), amount of enzyme (mg) and molar ratio (mmolbetulinic acid/mmolphthalic anhydride) (Moghaddam *et al*., 2010). Here, we aim at using novel ANN and SVM models to fit the four conditions and to predict the isolated yield (%) of the enzymatic synthesis.

A complete machine learning model consists of two parts, the independent variable (s) and the dependent variable (s). Here, we set the time (h), temperature (°C), amount of enzyme (mg) and molar ratio as independent variables, while the isolated yield (%) was set as the dependent variable. 65% data group was set as training set, which 35% data group was set as testing set.

ANN models: A series of statistical learning algorithms with the name of Artificial Neural Networks (ANN) could give judgments when inputted into a large amount of information (Hopfield, 1988; Dayhoff and DeLeo, 2001; Yegnanarayana, 2009). Like the brain, a biological neural network, ANNs are usually comprised of neurons that can make instant calculations in different conditions with the connection with each other. Different from ordinary networks with one or two layers, there are three layers in Artificial Neural Networks which can learn inputs efficiently and recognize patterns in a direct way. Furthermore, ANNs can also use complicated algorithms to make prediction and find the optimum solution. Therefore, when dealing with problems that are too complex to solve, ANNs can

take the place of human brains and the application of ANNs is more and more popular in the scientific research. In this passage, we introduce the use of two kinds of ANNs, Multilayer feed-Forward Neural Networks (MLFN) and General Regression Neural Networks (GRNN) to build the models to forecast the yield of enzymatic synthesis of betulinic acid ester.

Multilayer feed-forward neural Networks (MLFN): With the training of a back-propagation learning arithmetic, multilayer feed-forward neural networks, one of the most popular neural networks, can be used to predict a large range of chemical reactions (Johansson *et al*., 1991; Smits *et al*., 1994; Svozil *et al*., 1997).

Neurons in the MLFN models are put into different layers (Fig. 2). Input layer is the first layer and output layer is the last one. Between them, hidden layers play a role of calculating and modeling. To be specific, we could use the mapping function Γ that allocates a subset $\Gamma(i) \subseteq V$ to each neuron i to describe the neurons in a formal way and the subset $\Gamma(i)$ is made up of all ancestors of the neuron. Meanwhile, there is a subset $\Gamma(i)^{-1} \subseteq V$ containing all ancestors of the given neuron$i$. All neurons in a given layer is connected with any one of the neurons in the past layer. A weight coefficient ω_{ij} can be used to present the connection of the ith and jth neuron and we apply the threshold coefficient ϑ_i (Fig. 3) to present the$i$th neuron. The level of significance of a particular connection in the neural network can be indicated by the weight coefficient. Additionally, Eq. (1) and (2) can determine the output value (activity) of the ith neuron x_i. It holds that:

$$x_i = f(\xi_i) \tag{1}$$

$$\xi_i = \vartheta_i + \sum_{j \in r_i^{-1}} \omega_{ij} x_j \tag{2}$$

where, the potential of the *i* th neuron is represented by ζ_i and function $f(\zeta_i)$ indicates the transfer function
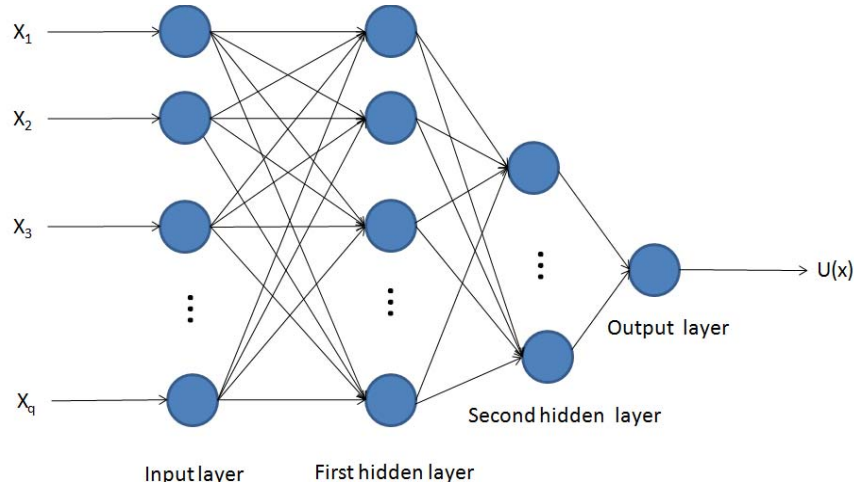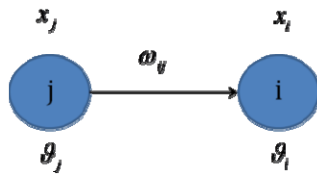
Fig. 2: Structure of the MLFN

Fig. 3: Connection between the two neurons

(the summation in Eq. (2) runs over all neurons j devolving the signal to the ith neuron). The threshold coefficient could be comprehend as a weight coefficient of the connection with regularly added neuron j, where $x_j = 1$(so-called bias).

For the transfer function it holds that:

$$f(\zeta) = \frac{1}{1+exp(-\zeta)} \qquad (3)$$

To minimize the total of the squared differences between the required and calculated output values, the weight coefficients ω_{ij} and threshold coefficients ϑ_i are devolved by the supervised adaptation process. Minimization of the objective function E can complete this by:

$$E = \sum_0 \frac{1}{2}\left(x_0 - \hat{x}_0\right)^2 \qquad (4)$$

where, x_0 and $\hat{x}_0$ are vectors comprised of the required and calculated processes of the output neurons and the summation carried out over all output neurons o.

General Regression Neural Network (GRNN): The Nadaraya-Watson kernel based General Regression Neural Network (GRNN) which is put forward by Specht (1991) is widely applied in medical diagnosis, pattern identification, forecasting, three-dimensional modeling, chemical engineering and function approximation (Hoskins and Himmelblau, 1988; Khan et al., 2001; Goulermas et al., 2007; Kandirmaz et al.,

2014; Li et al., 2014a, 2014b). Compared with other statistical neural networks like feed-forward networks, GRNN represents more accurate in regard to function approximation (Kandirmaz et al., 2014). Even if it is the first time to be used with the aim of function approximation, in some studies, GRNN is also applied to classification problems with small modifications (Kandirmaz et al., 2014). As can be seen from Fig. 4, there are 4 layers in GRNN, respectively, input layer, pattern layer, summation layer and output layer, with the characteristics of rapid learning, coherence and finding optimum with a great many specimens (Yang et al., 2014).

In the input layer, corresponding inputs are conserved and every input vector x can be devolved to pattern layer which contains different neurons for training data. As illustrated in Eq. (5), calculations are made about weighted squared Euclidean distance in the pattern layer. Before aggregated, inputs which are used for activation function, whether squares or absolute values, should be deducted from neuron values in pattern layer when applying to network. Then, neurons in summation layer, to which the outcomes are transferred, add dot product of pattern layer weights and outputs. As can be seen from Fig. 4, $f(x)K$ indicates weighted outputs of pattern layer where K is a Parzen window related constant. $Yf(x)K$ refers to multiplication of training data output Y values and pattern layer outputs. In output layer, $f(x)K$ separates $Yf(x)K$ to estimate desired values, held in Eq. (6) and (7) (Goulermas et al., 2007; Yang et al., 2014):

$$D_j = \left(x - x_j\right)^T \left(x - x_j\right) \qquad (5)$$

$$Y(x) = \frac{\int_{-\infty}^{\infty} Yf(x,Y)dY}{\int_{-\infty}^{\infty} Yf(x,Y)dY}, \qquad (6)$$

$$Y(x) = \frac{\sum_{j=1}^{p} y_j e[-D_j/2\sigma^2]}{\sum_{j}^{p} e[-D_j/2\sigma^2]} \qquad (7)$$
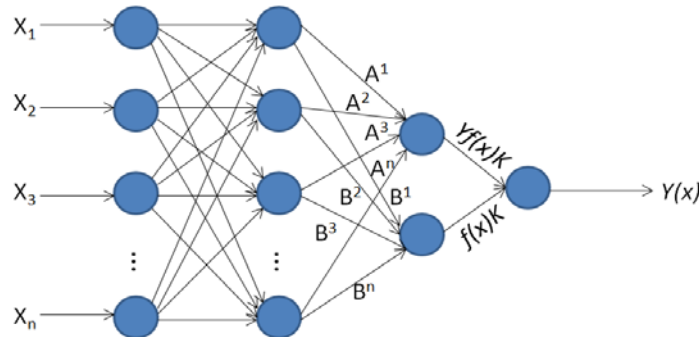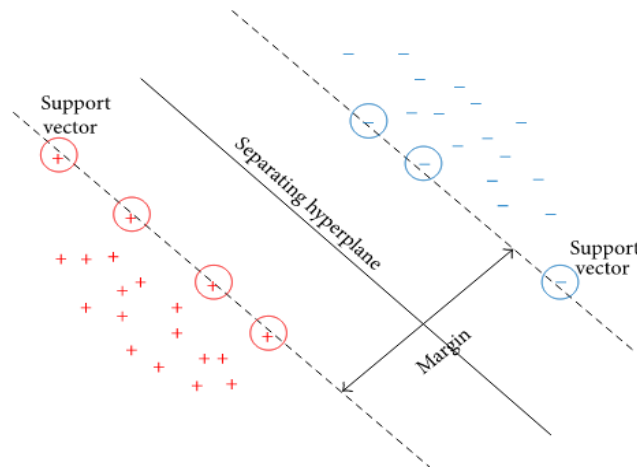
Fig. 4: Structure of the GRNN

Fig. 5: The support vectors determine the position of the optimal hyperplane

SVM model: There is a formidable machine learning technique with the name of Support Vector Machine (SVM) established from the statistical learning theory (Deng *et al*., 2012). In terms of increasing generalization, this theory can give an integral optimization in an efficient way, with restricted information of specimens between the learning capacity and the complicacy of models. Separating all specimens with the maximum margin, the main theory of SVM is a plane which has the ability of discovering the optimal hyperplane and linear separable dualistic classification (Zhong *et al*., 2013; Chen *et al*., 2015). Additionally, the plane can also increase the forecasting ability of the model and can decrease the mistake occurring accidentally when classifying. As can be seen from Fig. 5, specimens of type 1 are represented by "+" and specimens of type -1 are represented by "−" to shows the optimal hyperplane.

To explain the main structure of a representative support vector machine, Fig. 6 shows a small subset derived from the training data by related algorithm that contains the SVM. Kernels are characterized by the letter "*K*" (Kim *et al*., 2005). To have a forecasting accuracy, appropriate kernels and suitable parameters should be selected in terms of classification. However,
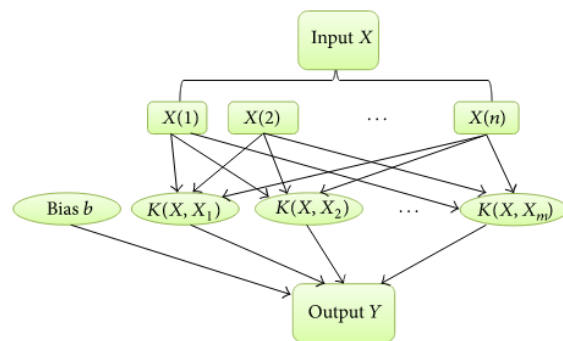

Fig. 6: The main structure of support vector machine

we could not find an available international standard to select these parameters. In most cases, to solve this task in a relatively reasonable way, we could take the advantage of the experiences from massive calculations, the contrast of experiment results and the application of cross validation which is realizable in program package (Fan *et al*., 2008; Guo and Liu, 2010; Chen *et al*., 2015).

Model development: The ANN prediction models were constructed by the Neural Tools® software (trial

Table 1: Best net search in different ANN models

| Model type	Mean RMS error	Training time	Prediction accuracy (%)
GRNN	4.01	0:00:01	100
SVM	4.23	0:00:01	100
MLFN (2 Nodes)	6.59	0:00:28	90.91
MLFN (3 Nodes)	8.68	0:00:29	90.91
MLFN (4 Nodes)	5.56	0:00:38	90.91
MLFN (5 Nodes)	7.99	0:00:46	90.91
MLFN (6 Nodes)	9.85	0:00:57	81.82
MLFN (7 Nodes)	8.98	0:01:04	81.82
MLFN (8 Nodes)	10.47	0:01:14	81.82
MLFN (9 Nodes)	11.93	0:01:26	72.73
MLFN (10 Nodes)	8.17	0:01:49	90.91
...	...	...	
MLFN (25 Nodes)	54.14	0:01:53	18.18
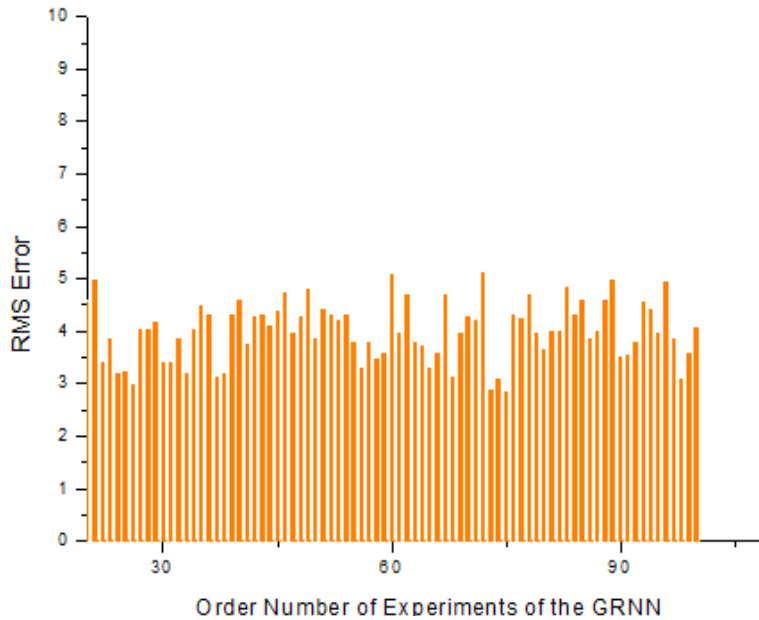
Fig. 7: Results of computational experiments of the GRNN

version, Palisade Corporation, NY, USA) (Pollar *et al*., 2007; Friesen *et al*., 2011; Vouk *et al*., 2011). The GRNN and MLFN was chosen as the learning machines of ANNs.

We used RMS error and training time as the indicators to measure the performances of ANN and SVM models (Table 1). The nodes of MLFN models were set from 2 to 25, from which we could find out the change regulation of the MLFN models when dealing with development processes.

Table 1 indicates that the GRNN, SVM and MLFN with 4 nodes have comparatively low mean RMS errors (4.01, 4.23 and 5.56, respectively). It is clear that the GRNN and SVM have the lowest RMS errors and training times, while the MLFN models have comparatively higher RMS errors and longer training times. In addition, the prediction accuracy (under the tolerance of 30%) of the GRNN and SVM are both 100%. Here, we discuss the availability of the GRNN, SVM and MLFN respectively in order to determine the most suitable model for the prediction of the yield.

RESULTS AND DISCUSSION

Comparison between the GRNN and MLFN: As for the GRNN, it has the lowest RMS error and training time during our research, compared with other 24 MLFN models. And according to the robustness of the principles of the GRNN, it has a high reproducibility, which has an overwhelming advantage compared with other ANN models during our research. In order to test the robustness of the GRNN, the computational experiments for the GRNN was repeated, which are shown in Fig. 7.

Figure 7 shows the RMS errors of the GRNN models in repeated experiments. It is significant that there is a stable fluctuation during repeated experiments, which shows that the GRNN model for the optimization process is robust. And what is more importantly, the mean RMS error is relatively low, which ensures the availability of the GRNN model.

To illustrate the change of the MLFN models with different numbers of nodes, Fig. 8 is used for showing the results of MLFN models with different nodes.
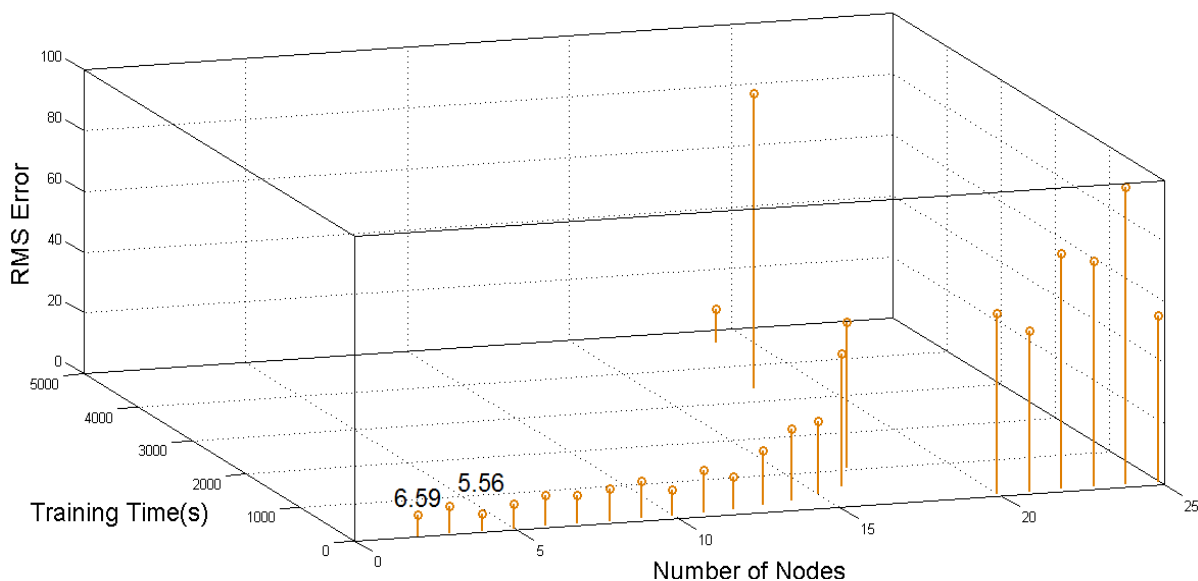
Fig. 8: RMS errors and training times of MLFN models with the change of nodes

It can be seen that with the increase of nodes, the RMS errors and training time of MLFN models become unsteadily fluctuant, which highly corresponds to the fluctuation character of typical MLFN models. It is worth mentioning that results in different MLFN models presented by Table 1 is not a fixed result, because of the effects of different random initial values chosen by the computer when training. However, it is still clear that the MLFN model may have a good result (relatively low RMS error and short training time) in a relatively low number of nodes. For practical applications, operators can use related software to find out the most suitable model for the optimization of reaction conditions in the range of low number of nodes. However, compared to the GRNN, MLFN models commonly cost longer training time and the fluctuations are not as stable as those of GRNN model.

Therefore, we still consider the GRNN model is a more suitable model for the prediction of the yield of enzymaticsynthesis of betulinic acid ester

Training and testing results of the GRNN and the SVM: Here, we use one of the typical examples of the training and testing results to present the availability of the GRNN and also illustrate the testing results of the SVM. Figure 9 and 10 are used for the illustration of the training and testing results of the GRNN, while Fig. 11 is used for the illustration of the testing results of the SVM. The training and testing sets of the GRNN and SVM are the same.
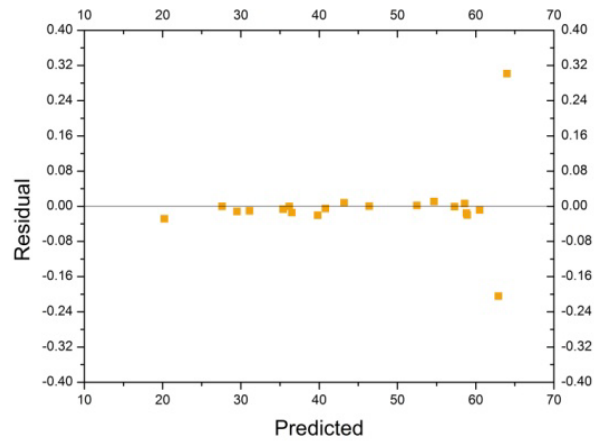
For showing the capacity for recall of the GRNN model for the optimization of the design, Fig. 9 is used for illustrating the training results of the GRNN.
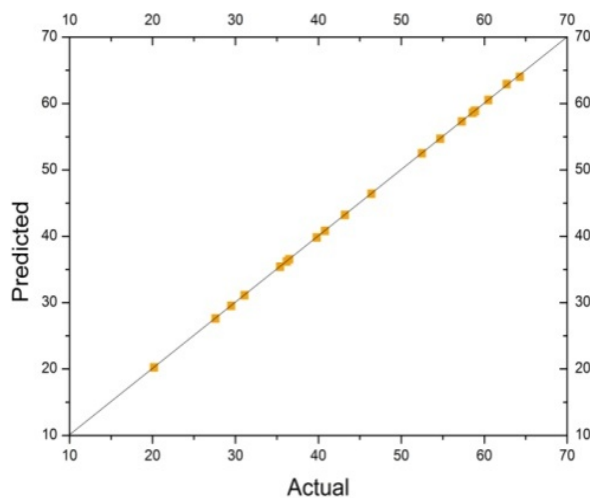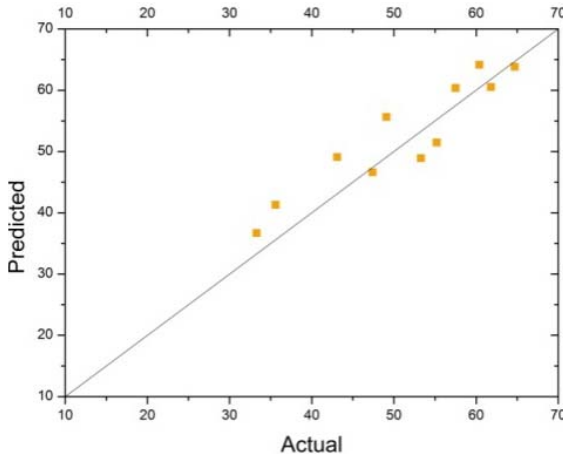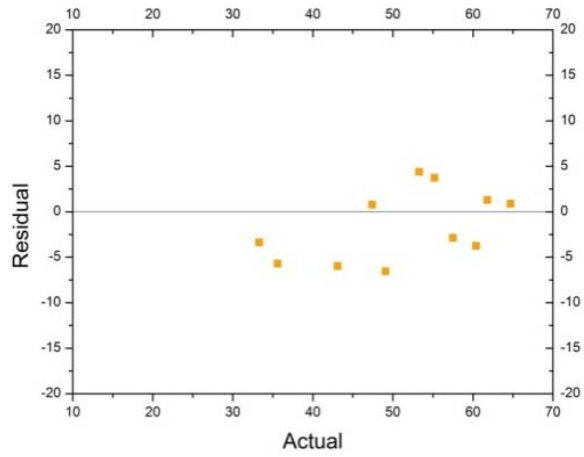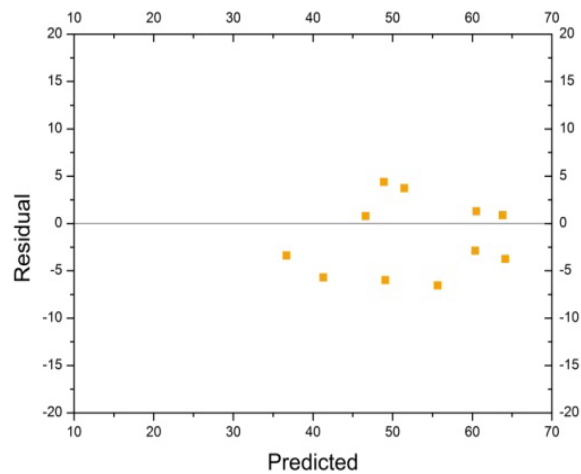

(a)
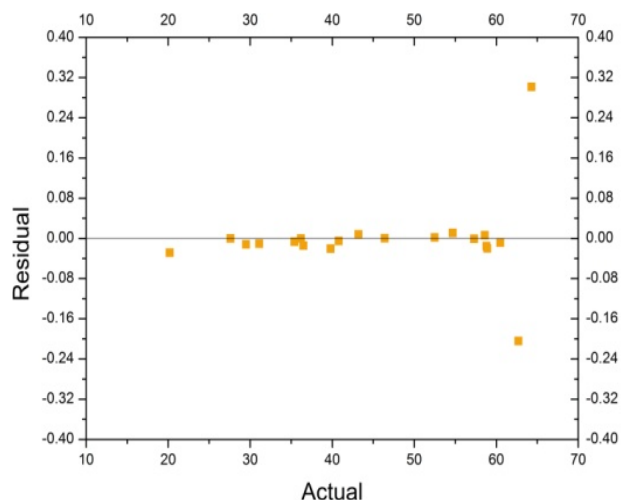
(b)

(c)

Fig. 9: Training results of the GRNN model; (a): Predicted values versus actual values; (b): residual values versus actual values; (c): residual values versus predicted values

(a)

(b)

(c)

Fig. 10: Testing results of the GRNN model; (a): Predicted values versus actual values; (b): residual values versus actual values; (c): residual values versus predicted values
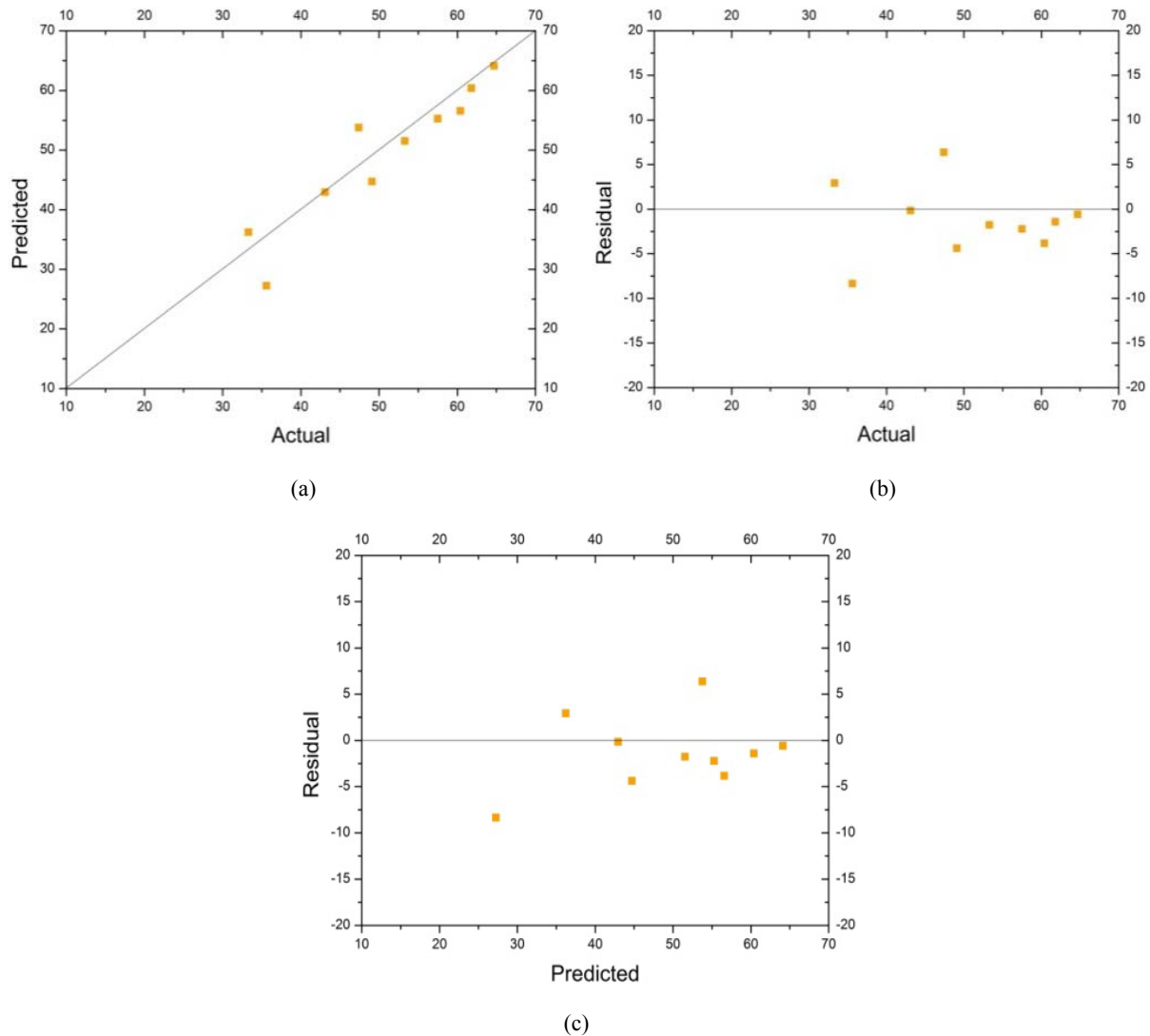
(a)

(b)

(c)

Fig. 11: Testing results of the SVM model; (a): Predicted values versus actual values; (b): residual values versus actual values; (c): residual values versus predicted values

Figure 9 shows that the GRNN model has a strong capacity for recall. The predicted values is highly close to the actual values (Fig. 9a), which indicates that the non-linear fitting effects of the model is highly decent. The comparisons between the residual values and actual/predicted values (Fig. 9b and 9c) also show that the residual values are relatively low, which suggests the robustness of the development of the GRNN model.

For showing the availability of the GRNN model after a training process, we use the data set which has not been used for the training process. Results are shown in Fig. 10.

Figure 10 shows the precise predicted results during the testing process. Predicted values are close to the actual values (Fig. 10a). Residual values presented by Fig. 10b and 10c show that the residual values are relatively low. Results present the robustness and availability of the GRNN model when testing.

In terms of the testing results of the SVM, Fig. 11 is illustrated for showing the correctness and robustness of the SVM in the prediction section.

Being similar to the results of the GRNN in the aspects of the RMS error and the training time, the testing results of the SVM are also highly similar to those of the GRNN. We can see that the SVM can generate a fairly analogical and precise result, compared with the testing results of the GRNN.

To make a comparison between the GRNN and SVM, we should firstly note that the initial values for the training process of the GRNN are random, leading to different results in repeated experiments. Compared with the GRNN, the SVM has very good repeatable results due to its principle. Therefore, it seems that the GRNN is not as robust as the SVM. However, results of repeated experiments (Fig. 7) show that regardless of

those fluctuations of RMS errors, the GRNN is also highly robust because the fluctuations are in a controllable range, which ensures the robustness of the GRNN. In terms of the training time, the GRNN and SVM are too short to find out the differences. However, we should note that the GRNN model can be mainly developed by packed software, while the SVM needs to use the Matlab and finish a series of processes, which requires a higher requirements of computer configuration and comparatively longer time. For a more convenient operation, the GRNN seems more practical than the SVM. Nevertheless, due to the high robustness and repeatability, the SVM should not be neglected in practical applications.

Here, enzymaticsynthesis of betulinic acid ester is a typical example for the application of machine learning techniques like ANNs and SVM to the prediction of yields in laboratory experiments. It shows that machine learning techniques have a huge potential applications for the prediction of yields in chemical reactions. And what is more importantly, we can optimize the reaction conditions via the "well-trained" models and predict their yields without trying repeatedly in laboratory experiments.

CONCLUSION

3β-*O*-phthalic ester of betulinic acid is of great importance in clinical research. Here, we successfully find that the application of the ANNs and SVM are useful for the prediction of yields of 3β-*O*-phthalic ester of betulinic acid synthesized by betulinic acid and phthalic anhydride using lipase as biocatalyst. Results show that the GRNN and the SVM model have the best prediction results during the testing process, with comparatively low RMS errors (4.01and 4.23, respectively) and short training times (both 1s). The prediction accuracy of the GRNN and SVM are both 100% in testing process, under the tolerance of 30%. Both the GRNN and SVM have very good repeatability and robustness. Our research successfully show that machine learning techniques like the ANNs and SVM can be used for the prediction of yields and optimization of conditions of traditional organic synthesis. What is more, it is also proved that support vector machine is a novel and strong machine learning tool for related research and applications.

REFERENCES

Chen, F., H. Li, Z. Xu, S. Hou and D. Yang, 2015. User-friendly optimization approach of fed-batch fermentation conditions for the production of iturin A using artificial neural networks and support vector machine. Electron. J. Biotechn., 18(4): 273-280.

Dayhoff, J.E. and J.M. DeLeo, 2001. Artificial neural networks: Opening the black box. Cancer, 91(8 Suppl.): 1615-1635.

Deng, N., Y. Tian and C. Zhang, 2012. Support Vector Machines: Optimization Based Theory, Algorithms and Extensions. 1st Edn., Chapman and Hall/CRC, New York.

Fan, R.E., K.W. Chang, C.J. Hsieh, X.R. Wang and C.J. Lin, 2008. LIBLINEAR: A library for large linear classification. J. Mach. Learn. Res., 9: 1871-1874.

Friesen, D., M. Patterson and B. Harmel, 2011. Comparison of multiple regression and neural networks for forecasting real estate values. Reg. Bus. Rev., 30: 114.

Gauthier, C. and J. Legault, 2008. Synthesis of two natural betulinic acid saponins containing α- 1 -rhamnopyranosyl-(1→2)-α- 1 -arabinopyranose and their analogues. Tetrahedron, 64: 7386-7399.

Goulermas, J.Y., P. Liatsis, X.J. Zeng and P. Cook, 2007. Density-driven generalized regression neural networks (dd-grnn) for function approximation. IEEE T. Neural Networ., 18(6): 1683-1696.

Guo, Q. and Y. Liu, 2010. Modeco: An integrated software package for ecological niche modeling. Ecography, 33(4): 637-642.

Hopfield, J.J., 1988. Artificial neural networks. IEEE Circuit. Devic., 4(5): 3-10.

Hoskins, J.C. and D.M. Himmelblau, 1988. Artificial neural network models of knowledge representation in chemical engineering. Comput. Chem. Eng., 12(9-10): 881-890.

Johansson, E.M., F.U. Dowla and D.M. Goodman, 1991. Backpropagation learning for multilayer feed-forward neural networks using the conjugate gradient method. Int. J. Neural Syst., 2(4): 291-301.

Kandirmaz, H.M., K. Kaba and M. Avci, 2014. Estimation of monthly sunshine duration in Turkey using artificial neural networks. Int. J. Photoenergy, 2014: 9.

Khan, J., J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann and P.S. Meltzer, 2001. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med., 7(6): 673-679.

Kim, D.W., K.Y. Lee, D. Lee and K.H. Lee, 2005. A kernel-based subtractive clustering method. Pattern Recogn. Lett., 26(7): 879-891.

Kvasnica, M., J. Sarek, E. Klinotova, P. Dzubak and M. Hajduch, 2005. Synthesis of phthalates of betulinic acid and betulin with cytotoxic activity. Bioorgan. Med. Chem., 13(10): 3447-3454.

Li, H., W.J. Leng, Y.B. Zhou, F. Chen, Z.L. Xiu and D.Z. Yang, 2014a. Evaluation models for soil nutrient based on support vector machine and artificial neural networks. Sci. World. J., 2014: 7.

Li, H., X. Wang, T. Yi, Z. Xu and X. Liu, 2014b. Prediction of Henry's law constants for organic compounds using multilayer feedforward neural networks based on linear solvation energy relationship. J. Chem. Pharm. Res., 6(6): 1557-1564.

Loughlin, W.A., 2000. Biotransformations in organic synthesis. Bioresource Technol., 74(1): 49-62.

Moghaddam, M.G., F.B.H. Ahmad, M. Basri and M.B.A. Rahman, 2010. Artificial neural network modeling studies to predict the yield of enzymatic synthesis of betulinic acid ester. Electron. J. Biotechn., 13(3): 2182-2185.

Mukherjee, R., M. Jaggi, P. Rajendran, M.J.A. Siddiqui, S.K. Srivastava *et al.*, 2004. Betulinic acid and its derivatives as anti-angiogenic agents. Bioorg. Med. Chem. Lett., 14(9): 2181-2184.

Mukherjee, R., V. Kumar, S.K. Srivastava, S.K. Agarwal and A.C. Burman, 2006. Betulinic acid derivatives as anticancer agents: Structure activity relationship. Anti-Cancer. Agent. Me., 6(3): 271-279.

Pollar, M., M. Jaroensutasinee and K. Jaroensutasinee, 2007. Morphometric analysis of Tor tambroides by stepwise discriminant and neural network analysis. World Acad. Sci. Eng. Technol., 33: 16-20.

Rajendran, P., M. Jaggi, M.K. Singh, R. Mukherjee and A.C. Burman, 2008. Pharmacological evaluation of c-3 modified Betulinic acid derivatives with potent anticancer activity. Invest. New Drug., 26(1): 25-34.

Smits, J.R.M., W.J. Melssen, L.M.C. Buydens and G. Kateman, 1994. Using artificial neural networks for solving chemical problems: Part I. Multi-layer feed-forward networks. Chemometr. Intell. Lab., 22(2): 165-189.

Specht, D.F., 1991. A general regression neural network. IEEE T. Neural Networ., 2(6): 568-576.

Svozil, D., V. Kvasnicka and J. Pospichal, 1997. Introduction to multi-layer feed-forward neural networks. Chemometr. Intell. Lab., 39(1): 43-62.

Thibeault, D., C. Gauthier, J. Legault, J. Bouchard, P. Dufour and A. Pichette, 2007. Synthesis and structure-activity relationship study of cytotoxic germanicane- and lupane-type 3beta-O-monodesmosidic saponins starting from betulin. Bioorgan. Med. Chem., 15(18): 6144-6157.

Vouk, D., D. Malus and I. Halkijevic, 2011. Neural networks in economic analyses of wastewater systems. Expert Syst. Appl., 38(8): 10031-10035.

Yang, D., H. Li, C. Cao, F. Chen, Y. Zhou and Z. Xiu 2014. Analysis of the oil content of rapeseed using artificial neural networks based on near infrared spectral data. J. Spectrosc., 2014: 5.

Yasin, Y., M. Basri, F. Ahmad and A.B. Salleh, 2008. Response surface methodology as a tool to study the lipase-catalyzed synthesis of betulinic acid ester. J. Chem. Technol. Biot., 83(5): 694-698.

Yegnanarayana, B., 2009. Artificial Neural Networks. PHI Learning Pvt. Ltd., New Delhi.

Yogeeswari, P. and D. Sriram, 2005. Betulinic acid and its derivatives: A review on their biological properties. Curr. Med. Chem., 12(6): 657-666.

Zarevúcka, M. and Z. Wimmer, 2008. Plant products for pharmacology: Application of enzymes in their transformations. Int. J. Mol. Sci., 9(12): 2447-2473.

Zhong, X., J. Li, H. Dou, S. Deng, G. Wang *et al.*, 2013. Fuzzy nonlinear proximal support vector machine for land extraction based on remote sensing image. PloS One, 8(7): e69434.