Fuzzy Support Vector Machines based on Adaptive Particle Swarm Optimization for Credit Risk Analysis

M D Murjadi*1 and Z Rustam1

1 Department of Mathematics, Universitas Indonesia, Indonesia.
*Corresponding author: melina.dewi@ui.ac.id

Abstract. Every bank has loaning activities. Banks have several criteria for determining whether credit will give or not because every credit loan has a risk that the credit cannot return. In other words, banks need to analyze the credit applicant before granting the loan. Credit loan is a case of binary classification. The classification from applicant’s data might be helpful for the bank in consideration whether the applicant will return the loan or not. Support Vector Machines (SVM) is a classification technique based on structural risk minimization, which is effective for binary classification. This method developed into Fuzzy Support Vector Machines (FSVM), which is able to minimize the influence of outlier in finding the best hyper plane. Adaptive Particle Swarm Optimization (APSO) is an extension of Particle Swarm Optimization (PSO). In APSO-based FSVM, APSO used to determine the fuzzy score by finding the class center of each attribute that may give the highest accuracy. The result of this study is APSO-based FSVM can give the highest accuracy for each process. The highest rate of accuracy is 75.67%, which used APSO-based FSVM with 70% of training data and linear kernel.

1. Introduction
In this era, credit risk analysis is essential. In credit loaning, creditor and debitor make an agreement for some things, like how much loan will be granted and how long the installment period is. After the loan is given, the debitor has a responsibility to pay the loan as in the agreement. However, the fact is that many debitors fail to pay the loan. The creditor is at risk that the debitor may not be able to fulfill the agreement. This is why banks have some criteria to determine which applicants will be granted the loan [1]. These criteria can be translated into data that can be used for credit risk analysis. This problem is about classification of which applicants will fulfill the agreement and which applicants will not fulfill the agreement.

Some research has been done to identify which model is efficient for classification and prediction in credit risk analysis with statistical method like multiple discriminant analysis [2], classification trees [3], linear programming[4], KNN classifiers [5], and the others. However, these models fail to fulfillmultivariate normality assumption for independent variables. Recent research for this problem is developed on artificial intelligence methods like genetic algorithm[6], Support Vector Machines (SVM) [7], Artificial Neural Network (ANN) [8], and other methods[9].

SVM is a machine learning method for classification that first introduced by Vapnik in 1992. SVM has been widely applied in classification cases such as for digit handwriting recognition [10], voice identification [11], face detection [12], text classification [13], cancer classification [14], protein classification [15], credit risk data classification [7], and so forth. The development of the SVM technique continues as its application in measuring credit score inference [16], cancer classification by
selection of Artificial Bee Colony features [17], prediction of stock movements [18], interference
detection systems [19], and predictors of bankruptcy [20].

In 2002, Lin and Wang developed the SVM formula into Fuzzy Support Vector Machines (FSVM)
by adding fuzzy values to each data point [21]. In this way, the FSVM method can classify
unrecognized data on the SVM method. In 2014, Punniyamoorthy and Sridevi used the FSVM method
to analyze credit data using Fuzzy C-Means (FCM) to assign fuzzy values [9]. In 2014, Almasi et al.
modified the FSVM method by using Adaptive Particle Swarm Optimization (APSO) method to
assign fuzzy values for each data point. This modification can handle data with noise attributes that
have not been handled by the previous FSVM [22].

2. Support Vector Machines
This section reviews the mathematical formulation of SVM. Assume a training data set \( S \) with the
labeled training points \((x_i, y_i)\) where every training data \( x_i \in \mathbb{R}^n \) belongs to a class labeled by \( Y_i \in \{1, -1\}\) for \( i = 1, ..., n \). The optimal
hyperplane can be obtained by solving the quadratic optimization problem as in equation (1).

\[
\min \frac{1}{2} \|w\|^2 + C \sum \xi_i \\
\text{subject to } y_i (w^T x_i + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \forall i = 1, ..., n
\]

The parameter \( \xi_i \) is a slack variable. The parameter \( C \) is defined by user as a
regularization parameter to control the model complexity. For non linear data, data can be mapped to a higher dimensional
feature space where data can be separated by a hyperplane. This can be done by using the kernel trick.
The primal problem in equation (1) can be transformed to its dual form as in equation (2).

\[
\max \left( -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) + \sum_{i=1}^{n} \alpha_i \right) \\
\text{subject to } \sum_{i=1}^{n} \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C, \forall i = 1, ..., n
\]

The kernel function is written as \( K(., .) \). Some of the kernel functions are shown in Table 1. In solving
equation (2), \( \alpha_i > 0 \) is the support vector.

| Name                | Kernel function                                      |
|---------------------|------------------------------------------------------|
| Linear Kernel       | \( K(x_i, x_j) = x_i^T x_j \)                        |
| Polynomial Kernel   | \( K(x_i, x_j) = (t + x_i^T x_j)^d \)                |
| RBF Kernel          | \( K(x_i, x_j) = \exp \left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right) \) |
| Sigmoid Kernel      | \( K(x_i, x_j) = \tanh(\beta_0 x_i^T x_j + \beta_1) \) |

3. Fuzzy Support Vector Machines
While SVM is effective for binary classification, SVM is sensitive to outlier. In 2002, Lin and Wang
developed the extension method of SVM called Fuzzy Support Vector Machines. In this method,
every point on training data will be given fuzzy membership values \( \iota \). Assume a training data set \( S \)
with the labeled training point \((x_i, y_i, \iota_i)\) where every training data \( x_i \in \mathbb{R}^n \) belongs to a class labeled
by $Y_i \in \{1,-1\}$ for $i = 1, \ldots, n$. The optimal hyperplane can be obtained by solving the quadratic optimization problem as in equation (3).

$$\min \frac{1}{2} ||\mathbf{w}||^2 + C \sum \xi_i$$
subject to $y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i \text{and} \xi_i \geq 0 \ \forall \ i = 1, \ldots, n$ (3)

The difference between SVM and FSVM is on the terms $\xi_i$. In FSVM, the measure of error has different weights. The FSVM can also be solved by its dual form.

3.1. Generation of membership.

The fuzzy membership can be obtained by setting it as a function of the distance between the data point and its fixed class center. Data is divided into two classes, that is positive class and negative class. The class center is defined by the mean vector of the attributes as in equation (4) and (5),

$$\mathbf{x}_+ = \frac{1}{n_+} \sum_{\mathbf{x}_i \in C^+} \mathbf{x}_i$$
$$\mathbf{x}_- = \frac{1}{n_-} \sum_{\mathbf{x}_i \in C^-} \mathbf{x}_i$$

where $\mathbf{x}_+$ is the mean from positive class and $\mathbf{x}_-$ is the mean from negative class. Meanwhile, $n_+$ and $n_-$ are the total observations from class $C^+$ and $C^-$ respectively. Then, the maximum radius from the positive class and the negative class are defined as in equation (6) and (7).

$$\mathbf{r}_+ = \max|\mathbf{x}_+ - \mathbf{x}_i| \text{ if } \mathbf{x}_i \in C^+$$
$$\mathbf{r}_- = \max|\mathbf{x}_- - \mathbf{x}_i| \text{ if } \mathbf{x}_i \in C^-$$

The fuzzy membership $\delta_i$ is defined as a function of mean and radius from each class as in equation (8) (Almasi, Gooqeri, Asl, dan Tang, 2015).

$$s_i = \begin{cases} 
1 - \frac{||\mathbf{x}_+ - \mathbf{x}_i||}{r_+ + \delta} & \text{if } y = 1 \\
1 - \frac{||\mathbf{x}_- - \mathbf{x}_i||}{r_- + \delta} & \text{if } y = -1 
\end{cases}$$

where $\delta$ is a positive constant with a small value to avoid the case where $s_i = 0$.

4. Adaptive Particle Swarm Optimization

In the first step of the PSO algorithm, a number of particles are generated. Each particle has speed and position. At each iteration, the speed and position of each particle are calculated and updated according to equation (9) and (10):

$$\mathbf{v}_{k+1} = w \mathbf{v}_k + c_1 r_1 (\mathbf{p}_k - \mathbf{x}_k) + c_2 r_2 (\mathbf{p}^g - \mathbf{x}_k)$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{v}_{k+1}$$

where $\mathbf{x}_k$ is the $i$-th particle position on the $k$-th iteration and $\mathbf{v}_k$ is the velocity of the $i$-th particle on the $k$-th iteration. The best position of a particle at each iteration is denoted by $\mathbf{p}_k$ and the global best particle position is denoted by $\mathbf{p}^g_k$. The cognitive parameters and social parameters of particles are denoted by $c_1$ and $c_2$ of nonnegative value, $r_1$ and $r_2$ are random numbers between 0 and 1. The optimal solution will be obtained through the iteration process. The coefficient of inertia is denoted by $w$ at interval $[0,1]$. To improve the searching ability of PSO, $w$ is defined as in equation (11) [23]:

$$w(t) = w_{\text{max}} - \frac{w_{\text{max}} - w_{\text{min}}}{\text{iter}_{\text{max}}} \times t$$

In the early stages, the nature of the exploration is more necessary than the nature of exploitation so that it takes the large value of cognitive parameters and small values of social parameters. When approaching the final stage, the nature of exploitation is more necessary than the nature of the
exploration so that the large value of social parameters and small values of cognitive parameters are required. Therefore, the values of cognitive parameters and values of social parameters need to be adapted so that the PSO algorithm is modified into Adaptive Particle Swarm Optimization (APSO). Equations (12) and (13) are the formulas used to convert the values of cognitive parameters and values of social parameters adaptively.

\[
\text{If } c_1^{\text{final}} < c_1, \text{then } c_1 = (c_1^{\text{final}} - c_1^{\text{initial}}) \left( \frac{\text{iter}}{\text{iter}_{\text{max}}} \right) + c_1^{\text{initial}} \\
\text{If } c_2^{\text{final}} > c_2, \text{then } c_2 = (c_2^{\text{final}} - c_2^{\text{initial}}) \left( \frac{\text{iter}}{\text{iter}_{\text{max}}} \right) + c_2^{\text{initial}}
\]

where \(c_1\) is the cognitive parameter and \(c_2\) is the social parameter.

5. FSVM based on APSO

In general, the value of fuzzy membership is obtained through the distance function between the data points and the class center. The development of fuzzy membership valuation method is based on this concept. Although with this method, FSVM can handle noise and outlier, this concept has not been able to solve noise problem in attribute. In the APSO-based FSVM method, APSO is used to change the attributes of a positive class center and a negative class center such that the error generalization can be minimized. After APSO convergence, the function of fuzzy membership assignment is made based on the center position of the refined class [22].

APSO is used to change the position of the class center attribute. By modifying the position of the class center of each class, each data point can achieve different weights that are relevant to its class point. Therefore, the upper and lower limits for each class attribute are set, and APSO is used to change the class center of each attribute within these limits. The structure of each particle in APSO can be seen in Table 2. Each particle consists of two parts, the first part is used to locate the optimal positive class center attribute and the second part is used to locate the optimal negative class center attribute.

| Parameter 1 | Parameter m | Parameter m+1 | Parameter 2m |
|-------------|-------------|---------------|--------------|
| Minimum value | \(x_{+11} - \alpha\) | ... | \(x_{-11} - \alpha\) | ... | \(x_{-im} - \alpha\) |
| Maximum value | \(x_{+11} + \alpha\) | ... | \(x_{-11} + \alpha\) | ... | \(x_{-im} + \alpha\) |

Here is the algorithm for FSVM based on APSO [22]:
1. Initialize FSVM model parameters
2. Initialize APSO population
3. Generate fuzzy memberships based on initial class centers
4. Train FSVM with initial model parameters
5. Evaluate the cost function
6. Update the initial population based on the procedure of APSO until the optimization approach terminate
7. Use the global best of the APSO as the class centers

6. Computational experiments

The data used in this paper is the German Credit version of the numerical data obtained through the UCI repository. This data is obtained from Prof. Dr. Hans Hofmann in 2000. The data has 24 attributes in numerical version. The attributes are the status of existing checking account, duration (in months), credit history, purpose, credit amount, savings account, length of work, installment rate of disposable income, gender, marital status, other debtors, length of residency, property owned, age (in years),
other installment plans, housing, number of existing credits at this banks, job, number of people being liable to provide, telephone and foreign worker [24]. German Credit data was processed several times by modifying the amount of training data. The amount of training data used is 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, and 90%. Each group of training data is processed with SVM, FSVM, and FSVM programs based on APSO. MATLAB R2015a is used for these experiments.

The model obtained from each method is used to process the testing data. The programs use linear kernel and polynomial kernels with \( d = 2 \) and \( d = 3 \). The number of particles used for APSO is 10 and the number of maximum iterations is 100. The cognitive and social parameters are given as follows: \( c_1^{initial} = 1.6, c_1^{final} = 0.7, c_2^{initial} = 0.9, c_2^{final} = 1.7 \). The value of \( \alpha \) is 0.1 [22]. Predicted results are evaluated by confusion matrix method.

The results of the SVM, FSVM, and FSVM based on APSO can be seen in Table 3. The highest accuracy rate is 75.67% which is obtained from FSVM based on APSO with training data 70% and linear kernel. The results show that FSVM based on APSO has better generalization performance than SVM and FSVM.

| Kernel Method | Training data |
|---------------|---------------|
|               | 10% | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% |
| Linear        |     |     |     |     |     |     |     |     |     |
| SVM           | 33.33% | 41.63% | 35.00% | 44.33% | 42.80% | 35.75% | 38.67% | 36.50% | 34.00% |
| FSVM          | 46.89% | 70.13% | 69.71% | 69.50% | 72.00% | 69.75% | 72.67% | 72.50% | 69.00% |
| FSVM - APSO   | 75.26% | 71.25% | 71.29% | 72.00% | 73.40% | 74.00% | 75.67% | 74.50% | 70.00% |
| Poly (d = 2)  |     |     |     |     |     |     |     |     |     |
| SVM           | 66.56% | 64.62% | 62.71% | 62.50% | 62.80% | 60.75% | 63.67% | 60.50% | 61.00% |
| FSVM          | 66.67% | 64.88% | 62.57% | 63.17% | 61.80% | 61.00% | 63.67% | 63.50% | 67.00% |
| FSVM - APSO   | 69.00% | 66.37% | 63.00% | 63.83% | 62.20% | 61.50% | 66.00% | 67.50% | 67.00% |
| Poly (d = 3)  |     |     |     |     |     |     |     |     |     |
| SVM           | 70.22% | 69.63% | 68.29% | 68.17% | 66.00% | 68.25% | 69.00% | 66.00% | 61.00% |
| FSVM          | 70.33% | 69.50% | 68.57% | 68.17% | 67.00% | 68.00% | 68.67% | 65.50% | 60.00% |
| FSVM - APSO   | 70.56% | 69.75% | 68.57% | 68.83% | 68.00% | 69.25% | 69.67% | 67.00% | 61.00% |

7. Conclusions
The main difference between SVM and FSVM method is the fuzzy membership. Therefore, choosing a proper function to generate fuzzy membership is the main problem of FSVM. APSO method is used to find the purified positive and negative class centers. The effects of outliers from each attributes are reduced. From this experiment, the FSVM based on APSO can achieve the highest accuracy among SVM and FSVM.

8. References
[1] Dexu Hand Miao We 2016 Financial Exclusion and Inclusive Finance China Economist.11(3)pp 64–76
[2] Pinches GE and Mingo K A 1975 The role of subordination and industrial bond ratings The Journal of Finance.30(1)pp 201–06
[3] Makowski P 1985 Credit scoring branches out Credit World. 5(1)pp 30–37
[4] Glover F1990 Improved linear programming models for discriminant analysis Decision Sciences.21(4)pp 771–85
[5] Henley W E and Hand DJ 1996 A k-nearest-neighbor classifier for assessing consumer credit risk The Statistician.45pp 77–95
[6] Chen MC and Huang SH2003Credit scoring and rejected instances reassigning through evolutionary computation techniques Expert Systems with Applications. 24(4) pp433–41
[7] Huang Z, et al. 2004 Credit rating analysis with support vector machines and neural networks: a market comparative study Decision Support Systems. 37(4) pp 543–558
[8] Lai K K, Yu L, Wang S and Zhou L 2006 Neural network meta learning for credit scoring Intelligent Computing, Springer Berlin Heidelberg pp 403–08
[9] Punniyamoorthy MandSridevi P 2016 Identification of a standard AI based technique for credit risk analysis Benchmarking: An International Journal 23(5) pp 1381–90
[10] Cortes C and Vapnik VN 1995Support vector networks Machine Learning 20(3) pp 273–97
[11] Schmidt M 1996 Identifying speaker with support vector networks Interface ‘96 Proceedings, Sydney
[12] Osuna E, Freund R andGirosi F 1997 Training support vector machines: an application to face detectionIEEE Conference on Computer Vision and Pattern Recognition pp 130–136
[13] Joachims T 1997 Text categorization with support vector machinesTechnical report LS VIII Number 23, University of Dortmund
[14] Guyon I, et al. 2002 Gene Selection for Cancer Classification using Support Vector Machines Machine Learning 46(1–3) pp 389–422
[15] Leslie, et al. 2004 Mismatch String Kernels for Discriminative Protein Classification Bioinformatics 20(4) pp 467–76
[16] Rustam Z and Yaurita F 2017 Application of Support Vector Machines for Reject Inference in Credit ScoringThe 3rd International Symposium on Current Progress in Mathematics and Sciences AIP Proceeding Conference, Bali, Indonesia
[17] Nadira T and Rustam Z 2017 Classification of Cancer Data Using Support Vector Machines with Features Selection Method Based On Global Artificial Bee ColonyThe 3rd International Symposium on Current Progress in Mathematics and SciencesAIP Conference, Bali, Indonesia
[18] Rustam Z, Vibranti D F, and Widya D 2017 Predicting the Direction of Stock Price Movement using Support Vector Machines and Fuzzy Kernel C-MeansThe 3rd International Symposium on Current Progress in Mathematics and SciencesAIP Conference, Bali, Indonesia
[19] Olivera N and Rustam Z 2017 Comparison of Fuzzy Robust Kernel C-Means and Support Vector Machines for Intrusion Detection Systems using Modified Kernel Nearest Neighbor Feature SelectionThe 3rd International Symposium on Current Progress in Mathematics and SciencesAIP Conference, Bali, Indonesia
[20] Rustam Z and Yaurita F 2018 Insolvency Prediction in Insurance Companies Using Support Vector Machines and Fuzzy Kernel C-MeansJournal of Physics: Conference Series 1028(1), 012118
[21] Lin C F and Wang S D 2002 Fuzzy support vector machines IEEE Transactions on Neural Network 13(2) pp 464–71
[22] Almasi O N, et al. 2015 A New Fuzzy Membership Assignment Approach for Fuzzy SVM Based on Adaptive PSO in Classification Problems Journal of Mathematics and Computer Science 14 pp 171–82
[23] Shi Y and Eberhart R 1998 A modified particle optimizer Proceedings of the 1998 IEEE World Congress on Computational Intelligence pp 69–73
[24] Lichman M 2013 UCI Machine Learning Repository [http://archive.ics.uci.edu/ml] Irvine, CA: University of California, School of Information and Computer Science