

Article

A Novel Fuzzy Unsupervised Quadratic Surface Support Vector Machine Based on DC Programming: An Application to Credit Risk Management

Tao Yu ^{1,2}, Wei Huang ^{3,4} and Xin Tang ^{3,4,*} ¹ School of Mathematics, Harbin Institute of Technology, Harbin 150001, China; 11849454@mail.sustech.edu.cn² Department of Mathematics, Southern University of Science and Technology, Shenzhen 518055, China³ College of Business, Southern University of Science and Technology, Shenzhen 518055, China; huangw7@sustech.edu.cn⁴ National Center for Applied Mathematics Shenzhen, Southern University of Science and Technology, Shenzhen 518055, China

* Correspondence: tangx6@sustech.edu.cn

Abstract: Unsupervised classification is used in credit risk assessment to reduce human resource costs and make informed decisions in the shortest possible time. Although several studies show that support vector machine-based methods have better performance in unlabeled datasets, several factors still negatively affect these models, such as unstable results due to random initialization, reduced effectiveness due to kernel dependencies, and noise points and outliers. This paper introduces an unsupervised classification method based on a fuzzy unsupervised quadratic surface support vector machine without a kernel to avoid selecting related kernel parameters for credit risk assessment. In addition, we propose an innovative fuzzy membership function for reducing noise points and outliers in line with the direction of sample density variation. Fuzzy Unsupervised QSSVM (FUS-QSSVM) outperforms well-known SVM-based methods based on numerical tests on public benchmark credit data. In some real-world applications, the proposed method has significant potential as well as being effective, efficient, and robust. The algorithm can therefore increase the number of potential customers of financial institutions as well as increase profitability.



Citation: Yu, T.; Huang, W.; Tang, X. A Novel Fuzzy Unsupervised Quadratic Surface Support Vector Machine Based on DC Programming: An Application to Credit Risk Management. *Mathematics* **2023**, *11*, 4661. <https://doi.org/10.3390/math11224661>

Academic Editors: Antonella Basso and Anatoliy Swishchuk

Received: 14 October 2023

Revised: 10 November 2023

Accepted: 13 November 2023

Published: 16 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: credit risk assessment; unsupervised classification; kernel-free quadratic surface SVM; fuzzy membership function

MSC: 91B05

1. Introduction

With the rapid growth of data mining, classification, also called clustering, has become an important task for extracting information from the data in machine learning methods. The support vector machine (SVM) is a well-known supervised classification technique that introduces an optimal hyperplane to maximize the margin between two labeled classes [1]. In many fields, including credit risk assessment, credit card fraud detection, stock prediction and disease diagnosis, the SVM has demonstrated exceptional performance [2–5]. The SVM allows us to classify an arbitrary number of end states without making assumptions about the distribution of input factors or the target category. Due to this advantage, the SVM algorithm is widely used in the field of credit risk management [6]. The purpose of credit risk management is to predict the relative risk of default among borrowers [7]. Financial institutions can reduce human resource costs and make appropriate decisions within a short period of time using machine learning.

Although these SVM-based methods are effective on labeled credit datasets, they cannot solve unsupervised learning problems without labels. Typically, labeling data points in real-world applications is time-consuming and labor-intensive, while obtaining

large numbers of unlabeled data points is relatively easy [8]. In addition, a significant number of small and micro enterprises lack credit histories and have insufficient funding. In order to increase business opportunities for financial institutions, credit risk management should be extended to unlabeled datasets. The one-class SVM is a clustering method based on a single type of label [9]. The algorithm reduces reliance on dataset labels. The weight one-class SVM and fuzzy one-class SVM further improve the efficiency and accuracy of the OC-SVM [10,11]. However, OC-SVM-based methods rely heavily on the initial labeled data points. To overcome this drawback, an unsupervised absolute value inequality classifier (UAVIC) was proposed, which considers constructing hyperplanes containing data points [12]. With this unsupervised linear algorithm, dependence on dataset labels can be completely eliminated. In terms of linearly indivisible datasets, the unsupervised quadratic surface support vector machine (US-QSSVM) enhances the performance of the SVM-based unsupervised learning algorithms [8]. However, the classifier directly generated by UAVIC and the US-QSSVM model may suffer from noise points and outliers. This is due to the separation plane being positioned between the two planes, which must contain all samples.

In most supervised and unsupervised classification approaches based on SVM, noise points negatively affect decision plane learning. Fuzzy membership can be applied to each input point of the SVM, fuzzy SVM (FSVM), so that different input points can contribute differently to decision surface learning [13]. Fuzzy membership function based on Fisher Discriminant Analysis (FDA) further improves FSVM performance. In addition, the number of support vectors in fuzzy support vector machines can be reduced using the DC program [14]. It should be noted, however, that all these methods solve the problem of noise points through binary classification problems. The fuzzy one-class quadratic surface support vector machine (FOC-QSSVM) considers minimizing the within-class scatter, which solves the noise problem of OC-SVM [11]. However, fuzzy membership equations do not address the negative impact of noise points and outliers on unsupervised learning decision planes.

Financial institutions have suffered significant losses over the past few years as consumers and corporations default on loans. SVM models have often achieved superior performance among them [15]. However, the effectiveness and efficiency of general SVM-based credit scoring methods are also significantly influenced by the kernel used. Credit risk assessment therefore requires reducing the impact of kernel function selection on the robustness of classification results.

The motivation of this research is to propose an unsupervised learning method based on SVM for classification based on the connection between data points. We present a fuzzy unsupervised quadratic surface support vector machine (FUS-QSSVM) that optimizes the hyperplane by weakening outlier points with weak sample connections. This paper has several main contributions as follows:

- In this study, we propose a fuzzy unsupervised quadratic surface support vector machine for credit risk assessment. This kernel-free unsupervised learning method can solve classification problems on unlabeled credit datasets. The algorithm was tested on several public credit datasets, and its performance was compared with the previously mentioned unsupervised SVM-based methods. The results showed that the proposed method outperformed unsupervised classification methods in terms of accuracy and robustness;
- For weakening outlier points, we propose a fuzzy membership method based on Tomek link method. In unlabeled datasets, this method effectively reduces the impact of outliers and noise points on decision-making. With this fuzzy membership function, we can distinguish noise points that belong to different classes in unlabeled samples based on their connections and relationships;
- After proving its boundness and convergence, a new DC algorithm (DCA) was developed to implement the proposed nonconvex model on numerous artificial and real-world benchmark datasets.

The rest of the paper is organized as follows. In Section 2, we briefly review DC programming and the US-QSSVM algorithm. Our method is presented in Section 3, and then we show its simple reformulation together with the DCA algorithm to solve the FUS-QSSVM model.

In Section 4, the results of numerical experiments on several public benchmark datasets are shown to evaluate the performance of our method. Section 5 summarizes the main findings of this paper and provides suggestions for future work.

2. Review of DC Programming and the Unsupervised Quadratic Surface Support Vector Machine

In this section, we briefly review DC programming and the US-QSSVM algorithm, which is solved by DCA [8].

2.1. DC Programming and DCA

Over the past three decades, non-convex programming and global optimization have seen dramatic developments. There has been considerable research on differential convex function (DC) programming as one of the major nonconvex optimization problems [16]. According to the literature on mathematical programming and real-life applications, DC programming optimization problems can be divided into two types [17].

- $\inf\{g(x) - h(x) : x \in \mathbb{R}^n\}$, where g and h are convex functions;
- $\inf\{g(x) - h(x) : x \in C, f_1(x) - f_2(x) \leq 0\}$, where g, h, f_1, f_2 and C are convex functions.

Such a function is called a DC function, and $g - h$ is a DC decomposition of a DC function, while convex functions g and h are DC components [18]. DC algorithms are based on local optimality conditions and duality in DC programming. This method approximates the second DC component $h(x)$ using its affine minorization $h^k(x) := h(x^k) + \langle x - x^k, y^k \rangle$, where $y^k \in \partial h(x^k)$ for each iteration k and minimizes its convexity as a result [18]. DCA converges from any starting point as it is a global convergence descent method without a line search. For a more detailed description, please refer to literature.

2.2. Unsupervised Quadratic Surface Support Vector Machine

Given a dataset of n unlabeled points $\{x^i\}$, where $x^i = (x_1^i, x_2^i, \dots, x_m^i)^T \in \mathbb{R}^m$. The quadratic surface is determined by the parameter set (Q, f, c) in the US-QSSVM algorithm as follows:

$$g(x) \triangleq \frac{1}{2}x^T Qx + f^T x + c, \tag{1}$$

where $Q = (q_{ij})_{m \times m} \in \mathbb{S}^m$, $f = (f_i)_m \in \mathbb{R}^m$ and $c \in \mathbb{R}$. This model tries to include all unlabeled points between two hyperplanes as $\frac{1}{2}x^T Qx + f^T x + c = h$ and $\frac{1}{2}x^T Qx + f^T x + c = -h$ as tight as possible. Furthermore, this equates minimizing the distance between the hyperplanes $\frac{1}{2}x^T Qx + f^T x + c = h$ and $\frac{1}{2}x^T Qx + f^T x + c = 0$. However, the gradient directions of hyperplane $g(x) = h$ are different at different points. Then, the US-QSSVM estimates the distance by calculating the mean geometric margins of all points in the dataset, where the gradient directions are the directions at each point. Overall, the US-QSSVM formulations are as follows:

$$\begin{aligned} \min \quad & \frac{1}{n} \sum_{i=1}^n \frac{h}{\|Qx^i + f\|_2^2} + \eta_1 \sum_{i=1}^n \xi_i \\ \text{s.t.} \quad & \frac{1}{2}(x^i)^T Qx^i + f^T x^i + c \leq h + \xi_i, \quad i = 1, \dots, n, \\ & \frac{1}{2}(x^i)^T Qx^i + f^T x^i + c \geq -h - \xi_i, \quad i = 1, \dots, n, \\ & Q \in \mathbb{S}^m, (f, c) \in \mathbb{R}^{m+1}, \\ & h \geq 0, \xi_i \geq 0, \quad i = 1, \dots, n. \end{aligned} \tag{2}$$

where the penalty parameter $\eta_1 > 0$ needs to be determined beforehand, and the slack variable ξ_i measures the clustering error for x^i . The optimal solution $(Q^*, f^*, c^*, h^*, \xi_1^*, \dots, \xi_n^*)$ determines the classifier $g(x) = 0$. There are two main disadvantages to the US-QSSVM model. First, the hyperplane $g(x) = 0$ may not be an appropriate classifier because there may be differences in class distributions. Second, the distance between the two hyperplanes $\frac{1}{2}x^T Qx + f^T x + c = h$ and $\frac{1}{2}x^T Qx + f^T x + c = -h$ is influenced by noise points. The results of the algorithm may not be as reliable as expected.

3. Proposed Method

In this section, we first propose a new fuzzy membership function for unsupervised learning and then propose the FUS-QSSVM model.

3.1. Fuzzy Membership Function

It is important to choose a suitable fuzzy membership function, as different fuzzy membership functions may affect the classifier differently. The Euclidean distance between training points and their class centers is the most common membership function in FSVM models [11,19,20]. However, this method is not suitable for unsupervised classification.

In this paper, we propose a fuzzy membership function based on the distance between points, which is derived from the Tomek link method [21]. We define the function $d(x^i, x^j)$ as the Euclidean distance between the points x^i and x^j . For any $i \in n$, we have the distance between x^i and x^j , where $j = 1, \dots, n$ and $j \neq i$. After this, the distances are sorted in ascending order $d_1(x^i) \leq d_2(x^i) \leq \dots \leq d_{n-1}(x^i)$. The k -nearest neighbor of x^i can be written as $Kn(x^i) = \{x^j | d(x^i, x^j) \leq d_k(x^i), \forall j = 1, \dots, n, \& j \neq i\}$. Therefore, we can define the k -nearest neighbor link as an indicator function, as follows:

$$I_{ij} = \begin{cases} 1 & \text{if } x_i \in Kn(x^j) \ \& \ x_j \in Kn(x^i), \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

If the indicator function with x^i and x^j is equal to 1, then we call (x^i, x^j) a k -nearest neighbor link pair. In other words, point x^i and point x^j are k -nearest neighbors to each other. The point closer to the center of the class has a greater number of star link pairs k -nearest neighbor link pairs.

In this way, we can weaken points that are far from the center of the class by a new fuzzy membership function, as follows:

$$\hat{r}_i = \frac{\sum_{j=t}^k d(x_j, x_i) * I_{ij}}{\sum_{j=t}^k d(x_j, x_i)} \tag{4}$$

where t is the lower limit. Given a lower limit t , the fuzzy membership \hat{r}_i of the noise and outlier points is as close to 0 as possible, where $0 \leq \hat{r}_i \leq 1$. Figure 1 shows that our fuzzy membership function can weaken the outlier and noise points, where $k = \frac{n}{2}$ and $t = \frac{n}{4}$. In essence, fuzzy membership decreases in the direction of sample density reduction. It is an effective way to reduce the effect of outliers and noise points on the resulting analysis. In addition, this method is not dependent on the sample center.

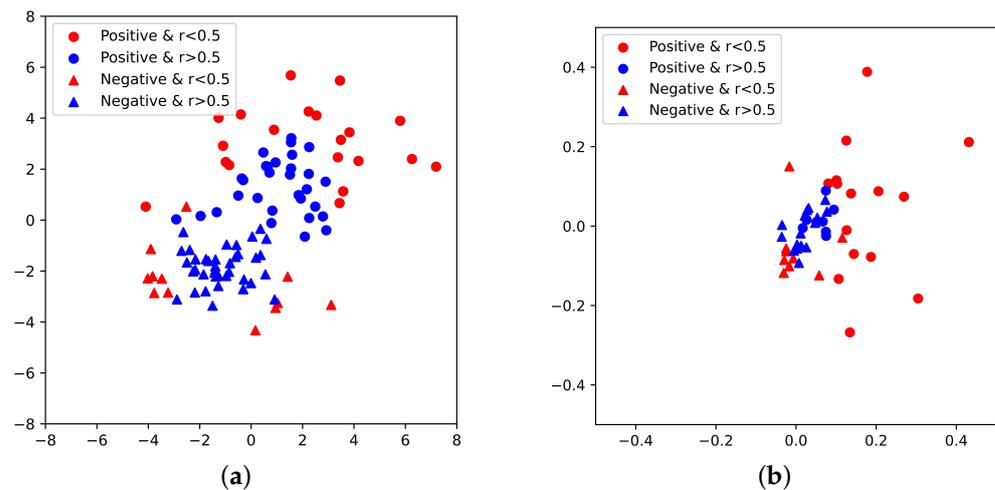


Figure 1. (a) The results of the fuzzy membership function in a Gaussian simulation dataset. (b) The results of the fuzzy membership function in a real-world dataset, Bankruptcy.

3.2. Fuzzy Unsupervised Quadratic Surface Support Vector Machine

The fuzzy US-QSSVM model also uses the fuzzy membership $\hat{r}^i, i = 1, \dots, n$, where $0 \leq \hat{r}_i \leq 1$, to deal with datasets containing outliers and noise.

In addition, sample points should be distributed more evenly on both sides of the separation plane to determine a more suitable hyperplane $g(x) = 0$. The fuzzy US-QSSVM model can then be used to create a more accurate prediction of the outcome. Therefore, the FUS-QSSVM is as follows:

$$\begin{aligned}
 \min \quad & \frac{1}{n} \sum_{i=1}^n \frac{h}{\|Qx^i + f\|_2^2} + \eta_3 \left| \sum_{i=1}^n \left(\frac{1}{2} (x)^T Qx^i + f^T x^i + c \right) \hat{r}_i \right| + \eta_1 \sum_{i=1}^n \hat{r}_i \xi_i \\
 \text{s.t.} \quad & \frac{1}{2} (x)^T Qx^i + f^T x^i + c \leq h + \xi_i, \quad i = 1, \dots, n, \\
 & \frac{1}{2} (x)^T Qx^i + f^T x^i + c \geq -h - \xi_i, \quad i = 1, \dots, n, \\
 & Q \in \mathbb{S}^m, \quad (f, c) \in \mathbb{R}^{m+1}, \\
 & h \geq 0, \quad \xi_i \geq 0, \quad i = 1, \dots, n.
 \end{aligned} \tag{5}$$

where ξ_i is the slack variable, and the penalty parameters $\eta_1 \geq 0$ and $\hat{\eta}_3 \geq 0$ need to be chosen beforehand. The term $\left| \sum_{i=1}^n \left(\frac{1}{2} (x)^T Qx^i + f^T x^i + c \right) \hat{r}_i \right|$ is a penalty function for balancing points on both sides of the separation plane. However, the objective function is one term of a nonlinear fraction that is difficult to solve. In this paper, we follow the literature to solve this issue [8,22]. The term $\frac{1}{n} \sum_{i=1}^n \frac{h}{\|Qx^i + f\|_2^2}$ can be Taylor expanded as $\frac{h_0}{u_0} + \frac{1}{u_0} (h - h_0) - \frac{h_0}{u_0^2} (\|Qx^i + f\|_2^2 - u_0) + R_1$, where $R_1 = o(h - h_0)^2 + (\|Qx^i + f\|_2^2 - u_0)^2 + (h - h_0)(\|Qx^i + f\|_2^2 - u_0)$ is a first-order Taylor remainder. The Taylor remainder R_1 can be omitted since $R_0 \rightarrow 0$ with $h \rightarrow h_0$ and $\|Qx^i + f\|_2^2 \rightarrow u_0$. Therefore, the term $\frac{1}{n} \sum_{i=1}^n \frac{h}{\|Qx^i + f\|_2^2}$ can be approximated to $\frac{h_0}{u_0} + \frac{1}{u_0} (h - h_0) - \frac{h_0}{u_0^2} (\|Qx^i + f\|_2^2 - u_0)$. Overall, the objective function can be rewritten as $\frac{h_0}{u_0} + \frac{1}{u_0} h - \frac{h_0}{u_0^2} \|Qx^i + f\|_2^2 + \eta_3 \left| \sum_{i=1}^n \left(\frac{1}{2} (x)^T Qx^i + f^T x^i + c \right) \hat{r}_i \right| + \eta_1 \sum_{i=1}^n \hat{r}_i \xi_i$, which can be simplified as $\frac{h_0}{u_0} + \frac{1}{u_0} (h - \hat{\eta}_2 \|Qx^i + f\|_2^2 + \hat{\eta}_3 \left| \sum_{i=1}^n \left(\frac{1}{2} (x)^T Qx^i + f^T x^i + c \right) \hat{r}_i \right| + \hat{\eta}_1 \sum_{i=1}^n \hat{r}_i \xi_i)$, where $\hat{\eta}_1 \triangleq \eta_1 u_0, \hat{\eta}_2 \triangleq \frac{h_0}{u_0}$ and $\hat{\eta}_3 \triangleq \eta_3 u_0$. Therefore, the FUS-QSSVM model can be approximated as follows:

$$\begin{aligned}
 \min \quad & h - \hat{\eta}_2 \|Qx^i + f\|_2^2 + \hat{\eta}_3 \left| \sum_{i=1}^n \left(\frac{1}{2} (x^i)^T Qx^i + f^T x^i + c \right) \hat{r}_i \right| + \hat{\eta}_1 \sum_{i=1}^n \hat{r}_i \zeta_i \\
 \text{s.t.} \quad & \frac{1}{2} (x^i)^T Qx^i + f^T x^i + c \leq h + \zeta_i, \quad i = 1, \dots, n. \\
 & \frac{1}{2} (x^i)^T Qx^i + f^T x^i + c \geq -h - \zeta_i, \quad i = 1, \dots, n. \\
 & Q \in \mathbb{S}^m, \quad (f, c) \in \mathbb{R}^{m+1}, \\
 & h \geq 0, \quad \zeta_i \geq 0, \quad i = 1, \dots, n.
 \end{aligned} \tag{6}$$

The optimal solution $(Q^*, f^*, h^*, c^*, \zeta_1^*, \dots, \zeta_n^*)$ determines the separation plane $g(x) \triangleq \frac{1}{2} x^T Q^* x + f^{*T} x + c^* = 0$. However, the problem in (6) is still difficult to calculate. Therefore, we can further simplify the problem in (6) by following the literature [8,11]. We create a vector formulation Θ based on the $\frac{m^2+m}{2}$ elements of the upper triangle of the matrix Q as follows:

$$\Theta \triangleq (q_{11}, q_{12}, \dots, q_{1m}, q_{22}, q_{23}, \dots, q_{2m}, \dots, q_{mm}) \in \mathbb{R}^{\frac{m^2+m}{2}} \tag{7}$$

Furthermore, we can construct an $m \times \frac{m^2+m}{2}$ matrix M_i for the point $x^i \in \mathbb{R}^m$ for $i = 1, 2, \dots, n$ as follows. For all $j = 1, 2, \dots, n$, the p th element in the j th row of M_i is assigned equal to x_k^i if the p th element of Θ is q_{jk} or q_{kj} for some $k = 1, 2, \dots, m$. Otherwise, the element is equal to 0. After this, we let $H_i \triangleq (M_i, I_{m \times m}) \in \mathbb{R}^{m \times (\frac{m^2+m}{2} + m)}$, where $i = 1, 2, \dots, n$. Then we have

$$G \triangleq \sum_{i=1}^n H_i^T H_i \in \mathbb{S}^{\frac{m^2+3m}{2}}, \tag{8}$$

$$v \triangleq (\Theta^T, f^T)^T \in \mathbb{R}^{\frac{m^2+3m}{2}}, \tag{9}$$

$$s_i \triangleq \left(\frac{1}{2} x_1^i x_1^i, \dots, x_1^i x_m^i, \frac{1}{2} x_2^i x_2^i, \dots, x_2^i x_m^i, \dots, \frac{1}{2} x_m^i x_m^i, x_1^i, x_2^i, \dots, x_m^i \right) \in \mathbb{R}^{\frac{m^2+3m}{2}}. \tag{10}$$

Summarizing this, the FUS-QSSVM model in (6) can be reformulated as

$$\begin{aligned}
 \min \quad & h - \hat{\eta}_2 v^T G v + \hat{\eta}_3 \left| \sum_{i=1}^n (s_i^T v + c) \hat{r}_i \right| + \hat{\eta}_1 \sum_{i=1}^n \hat{r}_i \zeta_i \\
 \text{s.t.} \quad & s_i^T v + c \leq h + \zeta_i, \quad i = 1, \dots, n, \\
 & s_i^T v + c \geq -h - \zeta_i, \quad i = 1, \dots, n, \\
 & G \in \mathbb{S}^{\frac{m^2+3m}{2}}, v \in \mathbb{R}^{\frac{m^2+3m}{2}}, c \in \mathbb{R}, \\
 & \|v\|_\infty \leq \hat{\delta}, \quad h \geq 0, \quad \zeta_i \geq 0, \quad i = 1, \dots, n.
 \end{aligned} \tag{11}$$

The matrix G is symmetric and positive semi-definite, which is easy to verify. It is worth noting that the constraint $\|v\|_\infty \leq \hat{\delta}$ ($\hat{\delta} \geq 0$ is a large enough constant) avoids the problem in (11) by becoming unbounded when the term $v^T G v \rightarrow -\infty$ with $\|v\|_\infty \rightarrow \infty$. However, the FUS-QSSVM model in (11) is not a convex QP problem, making it difficult to solve.

3.3. Decomposition Algorithm

As mentioned earlier, the problem in (11) is not a convex QP problem, but the difference between two convex functions. The DCA algorithm was developed to solve such problems [18,23]. In order to solve the proposed model, a DCA must be designed. At first,

we turned the problem into (11) from a constrained problem to an unconstrained problem. We define the function as follows:

$$S(v, h, c) = \begin{cases} |s_i^T v + c| - h & \text{if } |s_i^T v + c| > h, \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

$$\mathcal{X}_{\{h \geq 0 \ \& \ c \in \mathbb{R} \ \& \ \|v\|_\infty \leq \hat{\delta}\}} = \begin{cases} 0 & \text{if } h \geq 0 \ \& \ c \in \mathbb{R} \ \& \ \|v\|_\infty \leq \hat{\delta}, \\ +\infty & \text{otherwise,} \end{cases} \tag{13}$$

$$g(v, h, c) = \hat{\eta}_1 \sum_{i=1}^n \hat{r}_i S_i(v, h, c) + h + \mathcal{X}_{\{h \geq 0 \ \& \ c \in \mathbb{R} \ \& \ \|v\|_\infty \leq \hat{\delta}\}} + \hat{\eta}_3 \left| \sum_{i=1}^n (s_i^T v + c) \hat{r}_i \right|, \tag{14}$$

$$h(v, h, c) = \hat{\eta}_2 v^T G v. \tag{15}$$

Hence, the FUS-QSSVM model in (11) can be reformulated as an unconstrained difference convex (DC) program, as follows:

$$\min_{(v, h, c) \in \mathbb{R}^{\frac{m^2+3m}{2}+2}} \{g(v, h, c) - h(v, h, c)\} \tag{16}$$

We define the conjugate function of $g(v, h, c)$ as $g^*(v, h, c) \triangleq \sup_{(v, h, c)} \{(v, h, c)^T y - g(x)\}$ for $y \in \mathbb{R}^{\frac{m^2+3m}{2}+2}$. Then, we have two constructed sequences, $\{x^k\} = \{(v^k, h^k, c^k)\}$ and $\{y^k\}$, for the DC algorithm (DCA) as follows:

- **Step 1:** Set an initial estimation $x^0 = (v^0, h^0, c^0) \in \text{dom}(g(x) - h(x))$ and $k = 0$.
- **Step 2:** Calculate $y^k \in \partial h(x^k)$, (i.e., $y^k = (2\hat{\eta}_2 (v^k)^T G, 0, 0)^T$).
- **Step 3:** Calculate $x^{k+1} = \arg \min \{g(x) - \langle x, y^k \rangle\}$.
- **Step 4:** If $\|x^{k+1} - x^k\| > \epsilon$, then set $k = k + 1$ and go to **Step 2**; otherwise stop and output $x^{k+1} = (v^{k+1}, h^{k+1}, c^{k+1})$ as an optimal solution.

DCA can achieve better accuracy while reducing the complexity of calculations needed to solve the model. This makes it an ideal optimization technique for calculating FUS-QSSVM models. The proposed algorithm is shown to converge in the following part.

Theorem 1. For an optimal problem with the objective $\min\{g(x) - h(x)\}$, it holds that

1. The optimal value of the problem is finite;
2. The sequences x^k and y^k in the algorithm are bounded;
3. Each limit optimal point x^* of sequence x^k is a local optimal solution of the problem in (11).

The Appendix A contains details of the proof.

4. Results

4.1. Experimental Setup

In order to evaluate the performance of the proposed fuzzy unsupervised QSSVM (FUS-QSSVM) method, a number of public benchmark datasets were used. Several unsupervised SVM-based methods, such as Unsupervised Absolute Value Inequality Classifier (UAVIC), Unsupervised Quadratic Surface Support Vector Machine (US-QSSVM), and Unsupervised Quadratic Surface Support Vector Machine with Gold Rule (US-QSSVM-GOLD), were also tested on the same benchmark datasets for fair comparison. To improve the comparability of the results, we also used the well-known unsupervised K-means algorithm. Note that US-QSSVM-GOLD is a technique to optimize plane separation $g(x) = h^{p*}$ for the US-QSSVM model.

In order to evaluate the performance of the proposed fuzzy unsupervised QSSVM (FUS-QSSVM) method, a number of public benchmark datasets were used. Several unsupervised SVM-based methods, such as Unsupervised Absolute Value Inequality Classifier

(UAVIC), Unsupervised Quadratic Surface Support Vector Machine (US-QSSVM), and Unsupervised Quadratic Surface Support Vector Machine with Gold Rule (US-QSSVM-GOLD), were also tested on the same benchmark datasets for fair comparison. In order to improve the comparability of the results, we also used the well-known unsupervised K-means algorithm. Note that US-QSSVM-GOLD is a technique to optimize plane separation $g(x) = h^{p^*}$ for the US-QSSVM model. The literature provides details for simplicity of description [8].

To select the best penalty parameters for all the tested methods, the same grid method was used as follows: First, we selected half of the initial dataset for parameter tuning. Then, we evaluated the accuracy of all methods based on the actual labels of the points. To ensure statistical significance, we repeated this process ten times. Ultimately, we choose the most appropriate parameters based on the average of the 10 predicted results. Considering that the research in this paper concerns an unsupervised binary classification problem, the fuzzy membership of noise points and outliers should approach 0. Hence, we used the parameters $k = \frac{n}{2}$ and $t = \frac{n}{4}$ for the fuzzy function (4) in all experiments. According to our experience, the ratio of the two hyper-parameters $\hat{\eta}_1$ and $\hat{\eta}_3$ affects the experimental results; therefore, we provide a hyper-parameter in (11) as $\log_2 \hat{\eta}_3 = 2^6$. Following the literature, the suitable hyper-parameter $\hat{\eta}_1$ for the US-QSSVM and US-QSSVM-GOLD is $\log_2 \hat{\eta}_1 \in \{16, 17, \dots, 30\}$ [8]. However, the appropriate hyperparameter for the proposed method is $\log_2 \hat{\eta}_1 \in \{1, 2, \dots, 10\}$. FUS-QSSVM, US-QSSVM, and US-QSSVM-GOLD models were solved by DC algorithm. The UAVIC model was solved by linear programming. In addition, to avoid different initial values, all variables had unit vectors as their initial values. We computed all datasets without labels as training sets and tested them with actual labels as results. All reported results are the average of 20 experiments.

In this paper, five widely used public credit benchmark datasets from the UCI public database, StatLib and Kaggle, were selected for method validation. As far as possible, the datasets selected for this paper were diverse and representative of the algorithms being compared. These five datasets are widely used in many studies [24,25]. The Japanese (which removes missing data), Australian, and Credit approval datasets come from the UCI database. The Bankruptcy dataset comes from StatLib. The Prosper lending platform was founded in 2005 and is the second largest online lending platform in the United States. Kaggle provides the Prosper dataset, and all low-quality samples are removed. Table 1 shows the information for all tested datasets.

Table 1. Descriptions of the tested datasets.

Dataset	#Instances	#Negative	#Positive	#Attributes
Credit approval	690	383	307	15
Bankruptcy	100	50	50	5
Japanese ¹	651	357	294	15
Australian	690	383	307	14
Prosper ¹	20,222	13,062	7160	49

¹ Japanese and Prosper datasets delete missing data.

Due to the FUS-QSSVM, the US-QSSVM and UAVIC assume that the separation plane must be in the middle of the two separation planes, potentially affecting the accuracy between the two different classes. However, positive and negative samples are not distinguished by accuracy. The losses associated with defaulters’ forecasted errors are far greater than those associated with other types of forecast errors, particularly in credit classification. To assess the classifier’s classification ability for both positive and negative classes, we used four more comprehensive metrics—Accuracy, Recall, Precision, and F1-measure. To explain their definitions, let us first introduce a confusion matrix, as shown in Table 2, which is the accuracy evaluation matrix associated with them.

Table 2. Confusion matrix.

		Predicted Label	
		Positive	Negative
Actual label	Positive	TP	FN
	Negative	FP	TN

From Table 2, the following can be observed:

- **TP** (True Positive) represents the number of positive class samples that were accurately predicted by the classifier;
- **FN** (False Negative) means the number of positive class samples that the classifier mistakenly predicted to belong to the negative class;
- **FP** (False Positive) represents instances of the negative class that were incorrectly identified as members of the positive class by the classifier;
- **TN** (True Negative) corresponds to the number of negative class samples that were correctly identified by the classifier.

The definitions provided above clarify the interpretation of these three evaluation metrics.

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

$$Precision = \frac{TP}{TP + FP} \tag{18}$$

$$F1 - measure = 2 \times \frac{Recall \times Precision}{Recall + Precision} \tag{19}$$

4.2. Experiment Results

Details of the experimental results are presented in Table 3. We compared our proposed method with several unsupervised algorithms and K-means based on SVM. As shown in Table 3, FUS-QSSVM performs better than other algorithms in accuracy and F1-measure except on the Credit approval dataset. First, the fuzzy membership function weakens the influence of outliers and noise points on the decision plane. Secondly, our objective function (5) is designed to balance the distance between the two types of samples to stabilize the separation plane. In addition, linear classifiers (UAVIC) are less efficient when applied to real-world credit datasets. Due to the linear indivisibility of most real-world credit datasets, it is worth noting that our method still achieves reasonable performance on large credit datasets, namely Prosper. The application of our method to credit risk management demonstrates its practical value.

Table 3. The results of all the tested datasets.

		Accuracy	Recall	Precision	F1-Measure
Bankruptcy	FUS-QSSVM	0.7760	0.7560	0.7876	0.7715
	US-QSSVM	0.5860	0.5240	0.6366	0.5748
	US-QSSVM-GOLD	0.6480	0.4600	0.7801	0.5787
	UAVIC	0.6000	0.5600	0.6087	0.5833
	K-means	0.6270	0.2540	1.000	0.4051
Japanese	FUS-QSSVM	0.6749	0.6418	0.6400	0.6409
	US-QSSVM	0.5777	0.5348	0.5308	0.5328
	US-QSSVM-GOLD	0.5841	0.5945	0.5362	0.5638
	UAVIC	0.6190	0.8299	0.5520	0.6630
	K-means	0.6410	0.7377	0.5809	0.6500

Table 3. Cont.

		Accuracy	Recall	Precision	F1-Measure
Australian	FUS-QSSVM	0.6891	0.8313	0.5754	0.6801
	US-QSSVM	0.5675	0.4442	0.4675	0.4556
	US-QSSVM-GOLD	0.5684	0.4471	0.4486	0.4478
	UAVIC	0.6255	0.0640	0.9286	0.1197
	K-means	0.6720	0.6318	0.5846	0.6073
Credit approval	FUS-QSSVM	0.5531	0.5389	0.4987	0.5180
	US-QSSVM	0.5499	0.5009	0.4970	0.4990
	US-QSSVM-GOLD	0.5428	0.4860	0.4885	0.4860
	UAVIC	0.5580	0.0052	1	0.0103
	K-means	0.6814	0.6908	0.6275	0.6576
Prosper	FUS-QSSVM	0.6463	0.8322	0.5003	0.6250
	US-QSSVM	0.5716	0.4169	0.4282	0.4225
	US-QSSVM-GOLD	0.5975	0.3308	0.4358	0.3761
	UAVIC	0.5608	0.4824	0.4399	0.4602
	K-means	0.5406	0.3507	0.3781	0.3638

The bolds mean the most effective performance methods in this dataset.

In all datasets, the recall and accuracy of US-QSSVM, US-QSSVM-GOLD, and UAVIC are significantly smaller than accuracy. In other words, these algorithms are inefficient at classifying minority classes. The reason for this is that the techniques used to construct the nearest hyperplanes to contain data points achieved more flexible and accurate data descriptions in some datasets.

However, there is a significant problem with noisy points and outliers in these techniques. Because even a small number of points can lead the separation plane to a certain class, as shown in Table 3. Moreover, the objective function (5) of our algorithm helps balance the two classes of sample points, which is one reason why our algorithm is efficient when judging minorities.

To demonstrate the robustness of our algorithm, we present the results obtained using the Bankruptcy dataset under different hyperparameters η_1 . The experimental hyperparameters were selected by calculating the experimental results on half of the sample points, following the literature [8]. For all the methods tested in this paper, the same grid method was used to select the best penalty parameters. However, the reported results in Figure 2 are tested in all samples. As a result, we demonstrated that the hyperparameters of FUS-QSSVM, US-QSSVM, and US-QSSVM-GOLD are robust within a suitable range. However, our algorithm is more effective in all hyperparameters than the others.

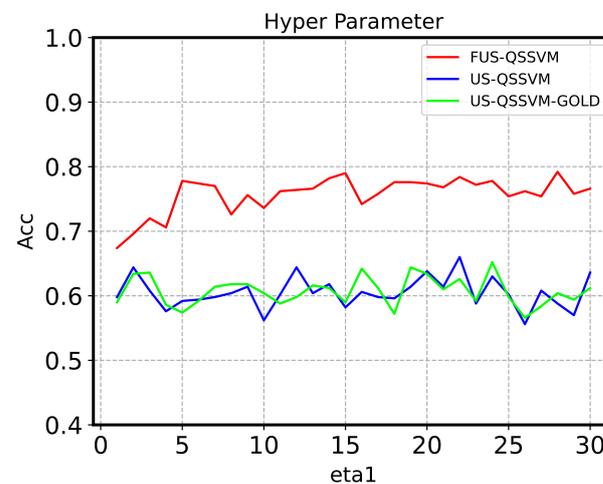


Figure 2. The results in different hyper-parameters η_1 .

4.3. Fuzzy Membership

To further prove that our fuzzy membership algorithm has favorable results when dealing with outliers and noise points, this paper uses a similar technique to add $p\%$ sample points to the Japanese dataset [7]. We assume that the mean and variance of the Japanese dataset are (μ, σ) . The new samples generated from a Gaussian distribution are defined by $(3\mu, \sigma)$, and the proportion of outliers p is equal to 5 or 10.

As before, the other experimental details remain the same. The results of the experiments are shown in Table 4.

Table 4. The results of the Japanese dataset, which includes outliers and noise points.

		Accuracy	Recall	Precision	F1-Measure
Japanese	FUS-QSSVM	0.6749	0.6418	0.6400	0.6409
	US-QSSVM	0.5777	0.5348	0.5308	0.5328
	US-QSSVM-GOLD	0.5841	0.5945	0.5362	0.5638
	UAVIC	0.6190	0.8299	0.5520	0.6630
	K-means	0.6410	0.7377	0.5809	0.6500
Japanese-5 ¹	FUS-QSSVM	0.6962	0.6855	0.6578	0.6714
	US-QSSVM	0.5560	0.4343	0.5136	0.4706
	US-QSSVM-GOLD	0.5647	0.5727	0.5214	0.5458
	UAVIC	0.5837	0.3946	0.5550	0.4613
	K-means	0.5453	0.2112	0.2591	0.2327
Japanese-10 ²	FUS-QSSVM	0.6846	0.7034	0.6368	0.6684
	US-QSSVM	0.5759	0.5329	0.4583	0.4928
	US-QSSVM-GOLD	0.5536	0.4719	0.4788	0.4753
	UAVIC	0.6175	0.3537	0.6380	0.4551
	K-means	0.5716	0.1668	0.2053	0.1840

¹ Japanese-5 includes 5% artificial noise samples; ² Japanese-10 includes 10% artificial noise samples. The bolds mean the most effective performance methods in this dataset.

In Table 4, FUS-QSSVM produces good results when applied to artificial noise datasets. However, there is a significant decrease in the performance of other algorithms. Because fuzzy membership functions (4) weaken the noise points and outliers, our proposed method shows a balance between the two classes. This makes the classification more reliable and accurate. In addition, it reduces the risk of misclassification and improves model performance. On the other hand, noisy points and outliers negatively affect the F1-measurement of other algorithms, as shown in Table 4. In this case, our algorithm shows better application prospects in the field of credit risk assessment, especially where classification will result in inconsistent losses.

In addition, the results can be improved by simply adjusting the position of the separation plane, such as in US-QSSVM-GOLD. However, the impact of noise points on US-QSSVM-GOLD is greater than that on US-QSSVM. Moreover, UAVIC is not suitable for datasets with too many noisy samples.

5. Conclusions

In this study, we introduced a novel kernel-free fuzzy unsupervised QSSVM technique designed specifically for the direct classification of unlabeled data that is nonlinearly separable for credit risk assessment. In addition, the novel fuzzy membership function reduces noise points and outliers in line with the direction of variation in sample density. To perform the proposed model efficiently and effectively, we designed a convergent DCA for the FUS-QSSVM model. A number of numerical experiments were carried out to study the performance of the proposed method. Our main findings are summarized below.

- Drawing on comprehensive numerical findings, the proposed FUS-QSSVM approach is in strong competition with other well-known classification techniques in credit datasets, including three SVM-based methods and K-means. A key advantage of

the proposed method is its ability to benefit from the high classification accuracy of SVM models. In addition, it minimizes the shortcomings associated with classification algorithms based on SVMs. These include factors such as unstable results due to random initialization, reduced effectiveness due to kernel dependencies, and noise points and outliers;

- The fuzzy join function optimizes the hyperplane position and shape by reducing noise points and outliers. Therefore, this enhancement bolsters both the accuracy and robustness of the model. Application of this technique is particularly useful when the dataset contains outliers or uneven distributions between two classes. By minimizing the impact of such data points, the model captures the underlying structure of the data. As a result, more robust predictions of the proposed model can be made about default applicants;
- As a result of the DCA design, the nonconvex FUS-QSSVM model was significantly more efficient in solving. In addition, this reduces reliance on initial values and tuning parameters, thus improving calculation efficiency.

Despite the fact that our algorithm performs well under fixed initial values, it suffers from the local optimal, which is determined by the initial values. In addition, parameter tuning requires domain-specific knowledge. For future research, we are interested in designing a global convergence of such a classification algorithm. Another area of research is the development of multi-classification models or the extension of this method to feature selection or semi-supervised learning.

Author Contributions: Methodology, T.Y.; Software, X.T.; Data curation, T.Y. and X.T.; Writing—original draft, T.Y.; Writing—review & editing, X.T.; Supervision, W.H.; Funding acquisition, W.H. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to acknowledge the partial grant support to the research (Grant ID: 72061127002). This research is also supported by DeFin research center of National Center for Applied Mathematics Shenzhen, Shenzhen Key Research Base in Arts & Social Sciences (Intelligent Management & Innovation Research Center, SUSTech, Shenzhen), and the National Laboratory of Mechanical Manufacture System, XJTU.

Data Availability Statement: Publicly available datasets were analyzed in this study. These data can be found here: UCI, <https://archive.ics.uci.edu/>, Kaggle, <https://www.kaggle.com/> and StatLib, <https://lib.stat.cmu.edu/datasets/>.

Acknowledgments: The authors would like to thank the editors and anonymous reviewers for their helpful comments and suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Proof.

1. By the definition, $g(x)$ is non-negative and $\|v\|_\infty \leq \hat{\delta}$. Moreover, there exist a constant κ_1 such that $\sum_{i=1}^n \|H_i v\|_2^2 \leq \kappa_1 \|H_i v\|_\infty^2$. Therefore, $g(x) = \frac{1}{n} v^T G v = \frac{1}{n} \sum_{i=1}^n \|H_i v\|_2^2 \leq \frac{\kappa_1}{n} \sum_{i=1}^n \|H_i v\|_\infty^2 \leq \frac{\kappa_1}{n} \sum_{i=1}^n \|H_i\|_\infty^2 \|v\|_\infty^2 \leq \frac{\kappa_1 \hat{\delta}}{n} \sum_{i=1}^n \|H_i\|_\infty^2$. Hence, the minimum of $g(x) - h(x)$ is finite.
2. The sequence $\{x^k\} = (v^k, h^k, c^k)$ is a solution with $\arg \min \{g(x) - \langle x, y^{k-1} \rangle\}$; hence, the sequence $\{v^k\}$ is bounded by the constraint $\|v\|_\infty \leq \hat{\delta}$. Furthermore, the sequence $\{y^k\} = (2\hat{\eta}_2 (v^k)^T G, 0, 0)^T$ is bounded. Then, we can prove that the sequences $\{h^k\}$ and $\{c^k\}$ are bounded. It is simple to verify that the objective value of the problem $\min \{g(x) - \langle x, y^{k-1} \rangle\}$ is finite. There exists a constant κ_2 for all k such

that $\min\{g(x) - \langle x, y^{k-1} \rangle\} \leq \kappa_2$. Since $\|v\|_\infty \leq \hat{\delta}$, $\sum_{i=1}^n \|H_i v^k\|_2^2 \leq \kappa_1^k \|H_i v^k\|_\infty^2$, $\zeta_i^k \geq 0$ and $0 \leq \hat{r}_i \leq 1$ for all $k \in N$, then

$$\begin{aligned} h^k &\leq \hat{\eta}_2 v^{kT} G v^{k-1} - \hat{\eta}_3 \left| \sum_{i=1}^n (s_i^T v^k + c^k) \hat{r}_i \right| - \hat{\eta}_1 \sum_{i=1}^n \hat{r}_i \zeta_i^k + \kappa_2 \\ &\leq \hat{\eta}_2 v^{kT} G v^{k-1} + \kappa_2 \tag{A1} \\ &\leq \hat{\eta}_2 \sum_{i=1}^n (H_i v^k)^T H_i v^{k-1} + \kappa_2 \\ &\leq \hat{\eta}_2 \sum_{i=1}^n \|(H_i v^k)^T\|_2 \|H_i v^{k-1}\|_2 + \kappa_2 \\ &\leq \hat{\eta}_2 \kappa_1^k \kappa_1^{k-1} \sum_{i=1}^n \|(H_i v^k)^T\|_\infty \|H_i v^{k-1}\|_\infty + \kappa_2 \\ &\leq \hat{\eta}_2 \kappa_1^k \kappa_1^{k-1} \sum_{i=1}^n \|H_i\|_\infty \|v^k\|_\infty \|H_i\|_\infty \|v^{k-1}\|_\infty + \kappa_2 \\ &\leq \hat{\eta}_2 \kappa_1^k \kappa_1^{k-1} \hat{\delta}^2 \sum_{i=1}^n \|H_i\|_\infty^2 + \kappa_2. \end{aligned}$$

Moreover, since $s_i^T v + c \leq h + \zeta_i$ and $s_i^T v + c \geq -h - \zeta_i$ for all $k \in N$, it is obvious that the sequence c^k is bounded. Overall, the sequences x^k and y^k in the algorithm are bounded.

3. From a result of 1 or 2, the $x^* = (v^*, h^*, c^*)$ is a local optimal solution in the problem in (11) according the DCA's convergence properties [18,23].

□

References

1. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
2. Almotairi, S.; Badr, E.; Abdul, S.M.; Ahmed, H. Breast Cancer Diagnosis Using a Novel Parallel Support Vector Machine with Harris Hawks Optimization. *Mathematics* **2023**, *11*, 3251. [CrossRef]
3. Jovanovic, D.; Antonijevic, M.; Stankovic, M.; Zivkovic, M.; Tanaskovic, M.; Bacanin, N. Tuning Machine Learning Models Using a Group Search Firefly Algorithm for Credit Card Fraud Detection. *Mathematics* **2022**, *10*, 2272. [CrossRef]
4. Shen F.; Yang, Z.Y.; Zhao, X.Z.; Lan, D. Reject inference in credit scoring using a three-way decision and safe semi-supervised support vector machine. *Inf. Sci.* **2022**, *606*, 614–627. [CrossRef]
5. Endri, E.; Kasmir, K.; Syarif, A. Delisting sharia stock prediction model based on financial information: Support Vector Machine. *Decis. Sci. Lett.* **2020**, *9*, 207–214. [CrossRef]
6. Harris, T. Quantitative credit risk assessment using support vector machines: Broad versus Narrow default definitions. *Expert Syst. Appl.* **2013**, *40*, 4404–4413. [CrossRef]
7. Twala, B. Multiple classifier application to credit risk assessment. *Expert Syst. Appl.* **2010**, *37*, 3326–3336. [CrossRef]
8. Luo, J.; Yan, X.; Tian, Y. Unsupervised quadratic surface support vector machine with application to credit risk assessment. *Eur. J. Oper. Res.* **2020**, *280*, 1008–1017. [CrossRef]
9. Camastra, F.; Verri, A. A novel kernel method for clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 801–805. [CrossRef]
10. Bicego, M.; Figueiredo, M.A.T. Soft clustering using weighted one-class support vector machines. *Pattern Recognit.* **2009**, *42*, 27–32. [CrossRef]
11. Luo, J.; Tian, Y.; Yan, X. Clustering via fuzzy one-class quadratic surface support vector machine. *Soft Comput.* **2017**, *21*, 5859–5865. [CrossRef]
12. Fung, G.M.; Mangasarian, O.L. Unsupervised and semisupervised classification via absolute value inequalities. *J. Optim. Theory Appl.* **2016**, *168*, 551–558. [CrossRef]
13. Lin, C.F.; Wang, S.D. Fuzzy support vector machines. *IEEE Trans. Neural Netw.* **2002**, *13*, 464–471.
14. Manh, C.N.; Van, T.N. A method for reducing the number of support vectors in fuzzy support vector machine. In *Advanced Computational Methods for Knowledge Engineering: Proceedings of the 4th International Conference on Computer Science, Applied Mathematics and Applications, ICCSAMA 2016, Vienna, Austria, 2–3 May 2016*; Springer: Berlin/Heidelberg, Germany, 2016; p. 17.
15. Lessmann, S.; Baesens, B.; Seow, H.V.; Thomas, L.C. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *Eur. J. Oper. Res.* **2015**, *247*, 124–136. [CrossRef]
16. Hartman, P. On functions representable as a difference of convex functions. *Pac. J. Math.* **1959**, *9*, 707–713. [CrossRef]

17. Tao, P.D. Algorithms for solving a class of nonconvex optimization problems. Methods of subgradients. *North-Holl. Math. Stud.* **1986**, *129*, 249–271.
18. Le Thi, H.A.; Pham Dinh, T. DC programming and DCA: thirty years of developments. *Math. Program.* **2018**, *169*, 5–68. [[CrossRef](#)]
19. Yogendran D.; Punniyamoorthy M. Improved bias value and new membership function to enhance the performance of fuzzy support vector Machine. *Expert Syst. Appl.* **2022**, *208*, 118003.
20. Luo, J.; Fang, S.C.; Bai, Y.Q.; Deng, Z.B. Fuzzy quadratic surface support vector machine based on fisher discriminant analysis. *J. Ind. Manag. Optim.* **2017**, *12*, 357–373. [[CrossRef](#)]
21. Tomek, I. Two Modifications of CNN. *IEEE Trans. Syst. Man Cybern.* **1976**, *6*, 769–772.
22. Schaible, S.; Shi, J.M. Fractional programming: the sum-of-ratios case. *Optim. Methods Softw.* **2003**, *18*, 219–229. [[CrossRef](#)]
23. Pham Dinh, T.; Le Thi, H.A. A DC optimization algorithm for solving the trust-region subproblem. *SIAM J. Optim.* **1998**, *8*, 476–505.
24. Liu, C.; Xie, J.; Zhao, Q.; Xie, Q.W.; Liu, C.Q. Novel evolutionary multi-objective soft subspace clustering algorithm for credit risk assessment. *Expert Syst. Appl.* **2019**, *138*, 112827. [[CrossRef](#)]
25. Obermann, L.; Waack, S. Demonstrating non-inferiority of easy interpretable methods for insolvency prediction. *Expert Syst. Appl.* **2015**, *42*, 9117–9128. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.