Applied Soft Computing, Volume 69, August 2018, pp. 541-553
DOI: 10.1016/j.asoc.2018.04.033

Information Gain Directed Genetic Algorithm Wrapper Feature selection for Credit Rating

Swati Jadhav, Hongmei He and Karl Jenkins

School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, UK

{s.jadhav,h.he, k.w.jenkins@cranfield.ac.uk}

A B S T R A C T

Financial credit scoring is one of the most crucial processes in the finance industry sector to be able to assess the credit-worthiness of individuals and enterprises. Various statistics-based machine learning techniques have been employed for this task. "Curse of Dimensionality" is still a significant challenge in machine learning techniques. Some research has been carried out on Feature Selection (FS) using genetic algorithm as wrapper to improve the performance of credit scoring models. However, the challenge lies in finding an overall best method in credit scoring problems and improving the time-consuming process of feature selection. In this study, the credit scoring problem is investigated through feature selection to improve classification performance. This work proposes a novel approach to feature selection in credit scoring applications, called as Information Gain Directed Feature Selection algorithm (IGDFS), which performs the ranking of features based on information gain, propagates the top m features through the GA wrapper (GAW) algorithm using

Published by Elsevier. This is the Author Accepted Manuscript issued with: Creative Commons Attribution Non-Commercial No Derivatives License (CC:BY:NC:ND 4.0).
The final published version (version of record) is available online at DOI: 10.1016/j.asoc.2018.04.033. Please refer to any applicable publisher terms of use.

three classical machine learning algorithms of KNN, Naïve Bayes and Support Vector Machine (SVM) for credit scoring. The first stage of information gain guided feature selection can help reduce the computing complexity of GA wrapper, and the information gain of features selected with the IGDFS can indicate their importance to decision making.

Regarding the classification accuracy, SVM accuracy is always better than KNN and NB for Baseline techniques, GAW and IGDFS. Also, we can conclude that the IGDFS achieved better performance than generic GAW, and GAW obtained better performance than the corresponding single classifiers (baseline) for almost all cases, except for the German Credit dataset, IGDFS+KNN has worse performance than generic GAW and the single classifier KNN. Removing features with low information gain could produce conflict with the original data structure for KNN, and thus affect the performance of IGDFS+KNN.

Regarding the ROC performance, for the German Credit Dataset, the three classic machine learning algorithms, SVM, KNN and Naïve Bayes in the wrapper of IGDFS GA obtained almost the same performance. For the Australian credit dataset and the Taiwan Credit dataset, the IGDFS+Naive Bayes achieved the largest area under ROC curves.

Keywords: Feature selection; Genetic algorithm in wrapper; Support vector machine; K nearest neighbour clustering; Naive Bayes classifier; Information Gain; Credit scoring; Accuracy; ROC curve

1 Introduction

The survey by Jadhav et.al [1] showed that machine learning techniques have been extensively applied in Credit Scoring, Loan Prediction, Money Laundering and other time series problems (e.g. prediction of earnings per share [2]) in finance industry. In this research, we focus on the Credit Scoring problem. Despite the advances of machine learning techniques, financial institutions continually seek improvements in classifier performance in an attempt to mitigate the credit risk [3].

Many machine learning applications involving large datasets usually exhibit the characteristic of high dimensionality; one such example is financial analytics [4]. To deal with such high-dimensional data, a solution involving dimensionality reduction is required before looking for any insights into the data. Feature selection creates more accurate predictive models in these applications while keeping the cost associated in evaluating the features to a minimum. Credit scoring aims to reduce the probability of a customer defaulting i.e. it predicts the credit risk associated with a customer. This helps decisions-making, maximising the expected profit from the customers for financial institutions.

Feature subset selection removes redundant and irrelevant features from the dataset, thus improving the classification accuracy and reducing the computational cost [5], [6]. The advantage of feature selection is that the information of feature importance in the dataset is not lost [7].

GA wrapper is the most popular method applied in the area of feature selection, and it has shown its efficacy in various areas (medical diagnosis [8], computer vision/image processing [9], text mining [10], bioinformatics [11], industrial applications [12]. Therefore, we explore the approach to solving feature selection in credit scoring problem.

In this study, we apply Information Gain [13] for initial feature selection, and then apply K-nearest neighbour (KNN), Naïve Bayes (NB) and Support Vector Machines (SVM) [14], [15] as the classification algorithms in the Genetic Algorithm Wrapper for credit scoring.

The paper is structured as follows: Section 2 introduces the state of the art in classical wrapper algorithms such as Genetic Algorithms and Particle Swarm Optimisation, the machine learning models used in wrappers for feature selection in solving the credit scoring problem and the challenges of feature selection along with the gaps identified. Section 3 discusses Information Gain, KNN, NB, SVM and the performance measures employed in this study. A genetic algorithm wrapper with the above three models is developed in Section 4. The experiments and evaluation are presented in Section 5. Finally, Section 6 concludes with discussions about the findings and future work.

2 Existing Work

Feature selection techniques have emerged as crucial in the applications where the input space affects the classification algorithm's performance. The process of feature selection searches through the space of all feature subsets while calculating evaluation measure to score the feature subsets. Since an exhaustive search is computationally too expensive, meta-heuristic search techniques, such as Genetic algorithm (GA) [16] and Particle swarm optimisation (PSO) [17] have been favoured for feature selection.

The wrapper-based feature selection approach [18] wraps the feature selection algorithm around a classification/induction algorithm. The performance of this algorithm finally selects a subset of features. The wrapper approach especially is useful to solve the problems for which a fitness function cannot be easily expressed with an exact mathematical equation. This technique has attracted a lot of research attention because of its simplicity in implementation since the induction algorithm acts as a black box in the whole process where knowledge of this algorithm is not mandatory [18]. The accuracy of this algorithm is used as evaluation measure to select the features. Other advantages of wrapper techniques are: Since classification algorithm decides the final selected subset, one gets more control over the whole feature selection process; wrapper techniques can produce

very high accuracy because of this learning capacity rendered by the inner induction algorithm.

A Genetic Algorithm Wrapper (GAW) has been widely applied to feature selection in data mining [19]. An advanced data mining technique of SVM classifier is most popularly used in such wrapper approach [20], [9], [21], [22]. When using SVM in a GA wrapper, SVM parameters optimisation needs consideration. In the literature, a few variants of GA+SVM algorithm have been proposed for feature selection in different application areas. For example, a GA+SVM technique was studied for the classification of hyper spectral images [9]. GA was used as a pre-processing step for the optimisation of SVM by Verbiest et al. [21]. Frohlich & Chapelle [22] minimised existing generalization error bounds on SVMs instead of performing cross-validation for a given feature subset. Anirudha et al. [23] proposed a Genetic Algorithm Wrapper Hybrid Prediction Model for feature selection. In this study, the outliers from the dataset were removed using K-means clustering technique, and then a Genetic Algorithm Wrapper was used to select the optimal features. These selected features were used to build the classifier models of Support Vector Machine, Naive Bayes, Decision Tree, and k-nearest neighbour. A hybrid feature selection method with GA wrapper using mutual information and using SVM is proposed by Huang et al.[24]. Some recent attempts to improve the optimised feature selection process by parallel processing are: [8], [25], [26], [27].

Aside from SVM, other machine learning models used in a wrapper approach include: C4.5 Decision trees [28], [29]; the model tree algorithm M5 [30]; Fuzzy Apriori Classifier [31]; Neural Network [32]; Bayesian Network classifier [33].

Another evolutionary computing method investigated for feature selection apart from Genetic Algorithm is Particle Swarm Optimisation. A PSO wrapper for selecting features, which are the most informative features for classification, was proposed in [34]. Lin et al.

[35] simultaneously determined the parameters and feature subset using PSO with SVM and obtained similar result to GA + SVM.

Credit risk analysis, credit scoring and classification are significant problems in computational finance. “A 2016 Credit Access Survey by the U.S. Federal Reserve Bank of New York indicates that approximately 40% of U.S. credit applications are rejected. Moreover, between 20% and 40% of consumers expect to be rejected depending on the type of credit sought, and many do not even apply. Yet, among these people there may well be qualified customers for the right kind of lender” [36].

The recent rapid growth in credit industry has made huge amounts of data available. Credit scoring datasets often are high dimensional making the classification problem highly complex, computationally intensive and less accurate for prediction [37]. Feature selection becomes necessary to reduce the burden of computing and to improve the prediction accuracy of the classification models for credit scoring [38], [39]. Various supervised wrapper methods have been studied for feature selection due to the classification accuracy entailed by the underlying algorithm although it comes at a cost of flexibility and scalability.

Somol et al. [40] studied filter as well as wrapper-based feature selection for the problem of credit scoring classification. Huang et al.[41] proposed three strategies, which included grid search and F-score calculations, for credit score evaluation. In this study, the authors proposed a hybrid strategy based on GA and SVM for feature selection and parameters optimisation built with relatively few input features. This achieved similar classification accuracy when compared against neural networks, genetic programming, and C4.5 decision tree classifiers. Non-linear approaches such as kernel SVM have seen recent applications in credit scoring since credit scoring data is often not linearly separable. In an attempt to develop wrapper techniques on bankruptcy and credit scoring classification

problems, Liang et al. [42] used GA and PSO wrapper embedded with different machine learning models, such as linear SVM, RBF kernel-SVM, NB, KNN, Classification and Regression Tree, and Multilayer Perceptron Neural Network (MLP) to select features for financial distress prediction. No best combination was found over the four datasets used in the study. This study concluded that performing GA+logistic regression can improve prediction improvements. Waad et al. [43] applied Logistic Regression, Naïve Bayes, MLP, Random Forest trees in wrapper on three credit datasets and showed that feature subsets selected by such fusion methods were equally good or better than those selected by individual methods.

Various traditional methods from statistics, non-parametric methods from computer science, modern methods from data mining and machine learning, and artificial intelligence techniques have been proposed in a bid to move away from manual methods and in search of building complex classification models which yield better accuracies and reliability of credit scorecards. Some of these applications are listed as following:

Application of KNN [44]; a wrapper feature selection with several machine learning models, such as SVM, Rough Set Theory, Decision Tree and Linear Discriminant Analysis (LDA), for credit scoring [45]; an ensemble classifier for feature selection in credit scoring [46], [47]; Logistic Regression, Neural Networks, least square SVMs Gradient Boosting, Decision Trees, and Random Forests for prediction of loan defaults [48]; a corporate credit rating model using multi-class SVMs with an ordinal pairwise partitioning [49]; a weighted least squares SVM which emphasised the importance of different classes [50] with successful acceptable accuracy and less computation time; hybrid models using Rough sets, Naïve Bayes and GA to classify credit risk of customers [51]; combination of Rough set and meta heuristic search for feature selection for the credit scoring problem [52]; wrapper approach with Naïve Bayes, MLP, RBF neural network, SVM, Random Forest,

Linear Discriminant classifier and Nearest Mean classifier for feature selection for credit rating prediction [53]; the combination of a clustering algorithm and GA with Decision Tree for feature selection for credit scoring of customers [29]; a hybrid approach for credit risk assessment using GA and ANN to obtain an optimum set of features to improve the classification accuracy and scalability [54]; a GA with weighted bitmask as alternative of polynomial fitness functions to estimate parameter range for building credit scoring models [54]; parallelisation of Random Forest method and feature selection methods, such as filters (t-test, LR, LDA), wrappers (GA, PSO) in credit scoring models [55].

The computing complexity of a machine learning algorithm is directly affected by problem space, more so in the area of credit analysis due to the complex decision process involved. Because of rapid advances in computing and information technologies, different types of techniques have been studied in combination with each other in many of today's real applications. There is a growing tendency of using hybrid methods for complex problems.

Typically, credit scoring databases are often large and characterised by redundant and irrelevant features [56]. Financial data and credit data in particular usually contain irrelevant and redundant features [57]. The redundancy and the deficiency in data can reduce the classification accuracy and lead to incorrect decision [58], [39]. The ability of interpretation of the predictive power of each feature in the dataset is often a necessity in certain applications. In such cases, a feature selection method such as Information Gain that returns a score is more useful than methods that return only a ranking or a subset of features, where the importance of features is not accounted for [59]. The choice of feature selection method largely depends on the problem, the type of data and the end use of the model. Which methods are most useful for feature subsetting is an open debate.

To fill the gap identified above in the field of credit scoring, we will investigate feature selection problem for credit scoring by proposing an Information gain directed feature

selection method incorporating the GA wrapper with machine learning techniques of SVM, KNN and Naïve Bayes. The literature has shown a few hybrid feature selection studies undertaken using GA as wrapper along with the machine learning classification algorithms. Some of them apply filtering techniques as a preprocessing techniques before the feature selection step. But in the area of credit scoring, such applications are very few.

The novelty of the proposed methodology lies in how the features contributing most towards the classification of credit applicants are propagated through the wrapper process. This is a novel approach specifically in the area of credit scoring. Firstly, the proposed strategy uses information-based ranking of features to reduce the feature set by modifying the initial population pool of GA so that best individuals are picked. Secondly, this measure is used to guide the evolution of GA by modifying the GA parameters of population pool, crossover and mutation. The novelty also lies in the usage of a large credit dataset constituting 30000 credit applicants: the Taiwan credit dataset which is not yet being used in the hybrid feature selection strategies for credit scoring applications.

3 Methodology

To classify the credit applicants, this work first ranks the features in order of importance to decision making/classification by measuring the information gain. The results are incorporated in the information directed wrapper feature selection method using genetic algorithm. Three classic machine learning models are embedded in the wrapper of GA, as a black box of fitness evaluation and these are SVM, KNN and NB.

The SVM hyperparameter selection is done by the method of grid search. The hyperparameter selection for *K*-nearest neighbor method (KNN) is done based on Euclidean distances with cross-validation. KNN calculates a decision boundary (i.e. boundaries for more than 2 classes) and uses it to classify new points. The *K* in KNN is a

hyperparameter that must be selected to get the best possible fit for the dataset. *K* controls the shape of the decision boundary. The best *K* is the one corresponding to the lowest error rate in cross validation. If test set is being used for hyperparameter setting, it may lead to overfitting.

In the rest of this section, the various techniques used for developing the proposed algorithm are discussed briefly. For improving the readability of this article, we describe in brief the basic principles of the KNN, Naïve Bayes, SVM technique, especially for finance industry who are not working in the machine learning area.

3.1 Information Gain of features

There are many ways of scoring the features such as Information entropy, Correlation, Chi squared test and Gini index. Entropy is one of several ways to measure diversity. Impurity of information can be measured by information entropy to quantify the uncertainty of predicting the value of the goal variable.

Let y be a discrete random variable with two possible outcomes. The binary entropy function H, expressed in logarithmic base 2, i.e. Shannon unit is given by Eq. (1):

$$H(y) = -p(+)log_2 p(+) - p(-)log_2(p(-)) \quad (1)$$

where, (+,–) are the classes, $p(+)$ denotes the probability that a sample $y \in (+)$, and $p(-)$ denotes the probability that $y \in (-)$. Entropy quantifies the uncertainty of each feature in the process of decision making. Eq. (2) calculates the conditional entropy of two events *X* and *Y*, when *X* has value *x*:

$$H(Y|X) = \sum_{x \in X} p_x(x) H(Y|X = x)$$

$$= -\sum_{x \in X} p_x(x) \sum_{y \in Y} p(y|x) log_2 p_y(y|x)$$

$$= -\sum_{x \in X} \sum_{y \in Y} p_{xy}(x, y) log_2 p_y(y|x) \quad (2)$$

Note: $\lim_{x \to 0} x \ log_2(x) = 0$.

The smaller the degree of impurity, the more skewed the class distribution. Entropy and misclassification error are highest when class distribution is uniform. The minimum value of entropy is attained when all the samples belong to the same class.

Information Gain (IG) is widely used on high dimensional data to measure the effectiveness of features in classification. It is the expected amount of information, i.e. reduction in entropy.

Namely, the information gain (IG) from a feature *x* is given by Eq. (3):

$$IG(y|x) = H(y) - H(y|x) \qquad (3)$$

Higher information gain means better discriminative power for decision making. Information gain is a good measure to determine the relevance of feature for classification.

The importance of features towards decision making in our model is done by evaluating them with the information gain measurement. Not all data attributes are created equally and not all of them contribute equally to the decision making. Hence the attributes can be sorted in the order of their contribution in decision making by listing the features in decreasing order of information gain scores.

3.2 K-Nearest Neighbour (KNN) Algorithm

KNN algorithm is a simple clustering algorithm, which produces highly competitive and easily interpreted results, is faster and comes with good predictive power. It is one of the most effective nonparametric methods, is simple to understand and easy to implement since only one parameter - *K* (the number of nearest neighbors) - needs tuning. The number *K* of nearest neighbors is key to the performance of the clustering process. The input to KNN

are *K* closest samples from training data and a new testing sample is classified based on the minimum Euclidean distance as in Eq. (4).

$$d(X,Z) = \sqrt{\sum_{i=1}^{n}(Z_i - X_i)^2} \quad (4)$$

where, X and Z are *n*-dimensional vectors in the feature space.

If a sample is in the *K* nearest neighbors, then it is assigned class membership of most common *K* neighbours. The main task of KNN is to search the nearest neighbors for each sample. The parameter *K* must be tuned for each dataset for enhancing the classification accuracies. To choose the parameter *K* we use 10-Fold-cross validation to validate KNN for various quantities of neighbors near rule-of-thumb values. Cross validation leads to the highest classification generalizability. If employing KNN with different values of *K* on a dataset, we obtain different accuracy at each round. The optimum *K* achieving the best accuracy is used in the feature selection.

3.3 Naïve Bayes

The Naïve Bayes (NB) classifier uses Bayes' Theorem which counts the frequency of 'attribute value - class' combinations in the historical data to calculate probability of class label C_i.

As stated by Twala [60], the basic principle of NB is the Bayes rule. The probability of each class is calculated, given all attributes, and the class with the highest posterior probability is the estimated class. Given an instance *X* for *n* observations, the probability of a class value C_i can be calculated with Eq. (5).

$$p(C_i|X) = \prod_{j=1}^{n} p\left(A_j\middle|C_i\right).p(C_i) \quad (5)$$

Let a training set of samples and corresponding class labels be given by *D*. Each sample *X* includes *n* independent attributes $(x_1, x_2, \ldots, x_n)$. If there are *m* class labels such as $C_1, C_2, \ldots, C_m$, then classification is to derive the maximal posteriori, $P(C_i|X)$:

$$P(C_i|X) = \frac{P(X|C_i).\ P(C_i)}{P(X)} \quad (6)$$

$P(X)$, which is prior probability, is fixed for all classes in a data set; hence $P(C_i|X)$ can be represented with Eq. (7).

$$P(C_i|X) = P(X|C_i).P(C_i) \quad (7)$$

Naïve Bayes algorithm assumes the conditional independence of attributes. Hence, the class assignments of the test samples are given by Eq. (8) and (9):

$$p(X|C_i) = \prod_{s=1}^{n} p(X_s|C_i) \quad (8)$$

$$C = \ argmax\ \{p(X \mid C_i).p(C_i)\} \quad (9)$$

If for a new sample, the posterior probability $P(C_2|X)$ is the highest for all the *s* classes, then this sample belongs to class C_2 according to the NB classifier.

3.4 The RBF-SVM classifier

SVM, a popular binary classifier, is used in the wrapper algorithm as a fitness evaluator since it is able to deal with high dimension space [61]. The hyperplane supported by a small number of vectors can be adaptable to various applications and yields good classification performance [62]. SVMs are robust against local minima, offer good generalization performance to new data, and are easily represented by few parameters [63]. But the SVM method cannot directly show how important each feature is to decision making [64].

The credit scoring problem is modelled as mapping of input feature-set into the decision variable (taking value as creditworthy or non-creditworthy), represented as $y=f(F)$, where

y is the decision variable and F is the feature vector. Identifying creditworthy applicants from non-creditworthy ones is not a linearly separable problem. Non-linear machines which map the data to higher dimensions can be used to find a SVM hyperplane minimising the number of errors for the training set.

RBF-kernel SVM, equivalent to a specific three-layer feed-forward neural network, is powerful for non-linear binary classification problems. This kernel SVM maps the problem space to higher dimension, i.e. makes the data linearly separable. Consequently, the linear SVM could be applied to solve the non-linear problem, mapped to the newly generated space with higher dimension. A RBF-SVM is good for solving very high dimensional problems, even if number of features is larger than number of samples [65]. Let $\Phi(F)$ be a mapping function which maps feature vector F to the kernel function $G\left(F_j, F_i\right) = \Phi(F_j)^T\ \Phi(F_i)$. The kernel SVM is expressed by Eq. (10):

$$f(F) = \left(\sum_{i=1}^{N} \alpha_i y_i\ g\left(F_j, F_i\right) + b\right) \qquad (10)$$

where α_i*'s* are dual variables and $g\left(F_j, F_i\right)$ is the kernel function replacing the inner product of the corresponding two feature vectors, performing the nonlinear mapping into feature space.

Correspondingly, learning to maximise the Eq. (11):

$$\sum \alpha_i - \frac{1}{2}\left(\sum_{ji} \alpha_j\ \alpha_i\ y_j\ y_i\ g\left(F_j, F_i\right)\right)$$

$$\textit{Subject to } C \geq \alpha_i \geq 0, \forall_i \textit{ and } \sum \alpha_i\ y_i = 0 \qquad (4)$$

where C -an upper bound on α_i- is the penalty parameter and is determined by the user.

In this study, the kernel of the SVM is set to (Gaussian) Radial-based function (RBF) (Eq. 12).

$$g(F, \tilde{F}) = exp\left(\frac{-\|F-\tilde{F}\|^2}{2\sigma^2}\right) \quad (52)$$

The RBF-kernel SVM is given in Eq. (13):

$$f(F) = \sum_{i=1}^{N} \alpha_i y_i \, exp\left(\frac{-\|F-\tilde{F}\|^2}{2\sigma^2}\right) + b \quad (63)$$

The radial basis function kernel has an additional kernel parameter γ i.e. kernel bandwidth to be optimised, where $\gamma = \frac{1}{2\sigma^2}$. As γ increases, the fit becomes more and more non-linear.

3.5 Performance Assessment Methods

The most commonly used measure of classifier performance is accuracy: the percent of correct classifications predicted. Comparing performance of different classifiers is easy with accuracy as a performance measure. But it is not possible to observe the performance for each class, especially for those datasets where the classes are not balanced.

Accuracy is the number of correct predictions divided by the total number of observations, and can be calculated with the confusion matrices by Eq. (14):

$$\text{Accuracy} = (TP+TN) / (TP+FN+TN+FP) \quad (14)$$

where,

- TP is the True Positives, when an applicant is creditworthy and is correctly classified as creditworthy.
- TN is the True Negatives, when an applicant is non-creditworthy and is correctly classified as non-creditworthy.
- FP is the False Positives, when an applicant is wrongly detected as being creditworthy.
- FN is the False Negatives, when an applicant is wrongly detected as being non-creditworthy.

For a highly-unbalanced problem, we do not want to overfit to a single class, and the receiver operating characteristic (ROC) is a good performance measure. ROC is a graphical plot showing the trade-off between the rates of correct predictions of creditworthy applicants with the rate of incorrect predictions of creditworthy applicants. The value of Area Under the Curve (AUC) of ROC ranges from 0.50 to 1.00, and the values above 0.80 can be viewed as a good discrimination between the two categories of the target variable.

3.6 k-fold Cross Validation

We use *k*-fold cross-validation technique to validate our models for assessing how the results generalize to an independent new dataset and to estimate prediction error. The training data were randomly split into equal-sized *k* mutually exclusive subsets before training the SVM classifier on each subset of data. Each time one of the *k* subsets is used for testing and the remaining *k-1* subsets are used for training. Accuracy computation is performed *k* times based on the *k* tests, an average of the *k* resultant accuracies gives a prediction of the classification accuracy. Cross validation uses all observations in the available data for testing and all the test sets are independent of each other, hence the reliability of the results is improved.

This study used $k = 10$, randomly dividing the data into 10 equal-sized parts, of which, one part is used as a test dataset, and nine parts as training sets. The results of the 10 iterations are averaged.

4 Genetic Algorithm Wrapper(GAW) for the Reduction of Feature Space

The GAW is used to obtain the optimal set from the original attributes, thus to reduce the feature space. Meta-heuristic algorithms have played important role in optimisation, as exhaustive search is too expensive. This section discusses the Genetic Algorithm wrapper technique used in this study.

4.1 The Wrapper approach of Feature Selection

Figure 1 illustrates the wrapper approach, where the feature subset is selected by using a classification algorithm, i.e., the classification algorithm acts as a black box without the requirement of knowledge of the algorithm, and the results produced by the classifier are evaluated with classification accuracy or other performance measures.

The feature selection process proceeds with the data being partitioned into training sets and validation sets in a specific training/test rate (e.g. 90% for training and 10% for test in *k*-fold cross-validation), then the classifier is run on the selected features of dataset. The optimal feature subset is the one with highest classification accuracy.

For every feature subset taken into consideration, the wrapper method trains the classifier and evaluates the feature subset by estimating the generalisation performance i.e. the accuracy of the machine trained with this feature subset on the original data. The search space is full feature space with *n* dimensions, where *n* is the number of full features. Hence, a *n* bit string can be used to represent the selected status of *n* features. Namely, each bit indicates whether a feature is selected (1) or unselected (0).

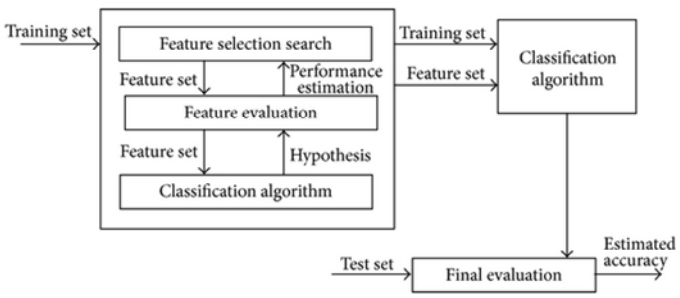


Figure 1: The Framework of Wrapper Approach for Feature Selection [18].

4.2 The Improved Genetic Algorithm Wrapper-IGDFS

Genetic algorithm(GA) is an adaptive mutation technique performing a heuristic search, inspired by the evolution process of genetics. A population, comprising of competing solutions, is maintained, which undergoes selection, crossover, and mutation to evolve and converge to the best solution. Parallel search is performed on the solution space to find an optimal solution without getting stuck in a local optimum. GA can produce promising solutions for feature selection over a high-dimension space due to its robustness to the underlying search space size and multivariate distributions [61].

To apply this algorithm to solve the credit scoring, two essential issues need to be solved: fitness function and classifier choice. The classifier should be able to handle very high dimension feature space given a limited sample space. SVM has the capacity [66] of treating a high-dimensional data, avoid overfitting and offering nonlinear modelling. In this study, we first apply the information gain to rank the features of dataset, then propagate the top n features through the GAW process of feature selection using NB, KNN and SVM as underlying classifiers.

Generally, the requirements for searching an optimal solution in the whole feature space include a search engine with an initial state, a state space and a termination condition [18]. Given n number of features, the size of search space is 2^n-1. As every feature has two possible states: “1” or “0”, an n bit string will have 2^n possible combinations. Assume τ features, which are not important to decision making in terms of the values of their information gains, be removed. The length of a binary string becomes n-τ. Even in the reduced search space ($2^{n-\tau}$), a brute-force search for a large space of $2^{n-\tau}$ is still infeasible. Of course, such space reduction is worthy for GA Wrapper search.

The ingredients of a Genetic Algorithm are:

(1) Chromosome: GA maintains a diverse population $x_{1...n} = <x_1,..., x_n>$ of n individuals x_i, the candidate solutions. The fitness of these individuals is evaluated by calculating an objective function $F(x_i)$ that is to be optimised for a given problem. These individual solutions are represented as 'chromosomes', which cover the entire range of possible solutions.

In this study, binary bit string is used to represent a chromosome. The bit strings representing the genotype (abstract representation) need to be transformed to phenotype (physical make-up), namely, feature index representation. The number n of bits represents the number of features. If the *i-th* bit is 1, the feature x_i is selected in the subset and if it is 0, feature x_i is not selected.

(2) Selection operator: Selection is the process of evaluating the fitness of the individuals and selecting them for reproduction. There are several ways to perform selection. Some commonly used methods include Elitist Selection, Hierarchical Selection, Rank Selection, Roulette-Wheel Selection and Tournament Selection. This work has used Tournament selection to select sufficiently good individuals for mating.

(3) Crossover operator: Crossover operator creates two offspring from the two selected parent chromosomes by exchanging part of their genomes. Crossover is the process of extracting the best genes from parents and reassembling them into potentially superior offspring. The simplest form of crossover is known as Single-point crossover. Other types are Two-Point Crossover, Uniform crossover. This work has used single point crossover.

(4) Mutation operator: Mutation maintains genetic diversity of population from one generation of chromosomes to the next and increases the prospect of the algorithm to generate more fit individuals. Using a small mutation probability, at each position in the string, a character at this position is changed randomly. Mutation of bit strings flips the bits at random positions with a small probability. This work has used uniform mutation.

(5) Elitism: Elitism guarantees that the best fit members are passed on to the next generation. The best individual or a set percentage of fittest members survives to the next generation. Small elitism compared to the population size yields a good balance between diversity and non-overfitting situation. High elitism makes the fittest individuals dominate the population resulting in ineffective search. This work guarantees that 2 elite offspring survive to the next generation.

(6) Diversity: Diversity of the population is an important factor influencing the performance of the genetic search. Diversity ensures that the solution space is adequately explored, especially in the earlier stages of the optimisation process. Very little diversity results into the GA converging prematurely. The initial range of the population and the amount of mutation affect the diversity of the population. Here tournament selection and uniform mutation are used in the evolutionary process of GA.

(7) Termination criteria: Three possible termination criteria could be used for the GA: A satisfying solution has been obtained, a predefined maximum number of generations has been reached, the population has converged to a certain level of genetic variation [67]. The algorithm convergence is sensitive to the mutation probability: a very high mutation rate prevents the search from converging, whereas a very low rate results in premature convergence of the search. The termination criteria for this work is maximum number of generations = 20 to 50.

(8) Blackbox with fitness function: A fitness function evaluates the goodness of each individual in the population in each generation against the optimisation criterion. To create the next generation, the fittest individuals obtained are allowed to reproduce using the set crossover and mutation rate. In this study, SVM, KNN and NB are used as the induction algorithms for fitness evaluation.

Assume $g(x)$ is the mapping function of machine learning model. Given an x, the state of the goal variable can be estimated, i.e. $y = g(x)$.

Assume A is the accuracy obtained by the classifier. It can be calculated by the function: $A=\varphi(\tilde{Y},Y)$, where Y is the list of goal states, and $\tilde{Y}$ is the list of estimated goal states for all test points.

We use the classification accuracy as the fitness value f, then

$$f = (g(x)|_D, Y) \tag{7}$$

where D is the test set.

The three GA wrapper techniques with the SVM, KNN and NB are denoted as GA-SVM, GA-KNN, and GA-NB respectively.

Algorithm 1 provides the operational steps of the proposed method of Information Gain Directed Feature Selection (IGDFS), where Algorithm 2 is one of KNN, NB and SVM classifiers.

Algorithm 1 Information Gain Directed Feature Selection

```
Measure Information Gain of individual features from the dataset
Rank the features in the dataset according to their importance: F = (f1 > f2 > f3, ....)
Input: Top m feature set Fr and class label C
Output: S
S ← null
procedure GA
    Input: PopSize Ps, GenSize, GenomeLength N, ProbMutation Pm
    Output: The Best individual in all generations
    Initialize: Population: Ps*N
    Retain f1 from Fr
    Ps ← random binary chromosomes
    for each chromosome do
        Compute fitness according to Algorithm2;
    end for
    repeat
        Select parents p1, p2 from population based on the fitness;
        for all new children do
            retain f1 from Fr;
            Crossover p1, p2;
            Mutate each gene in new child chromosome with probability Pm;
        end for
        Evaluate fitness of new individuals according to Algorithm2
        Replace least-fit population with new best individuals
    until Stopping Criteria
end procedure
```

4.3 Experimental setup

In this work, three publicly available credit datasets are used to build and test the performance of the proposed IGDFS algorithm. In the literature, these benchmark datasets are frequently employed to compare performance of different classification methods. Table 1 describes these datasets. To ensure validity of the model to make predictions on new data, *k*-fold cross validation method is implemented.

Our implementation of algorithms was carried out on Intel Pentium IV CPU running at 1.6 GHz and 256 MB RAM, in Matlab 2016 mathematical development environment and the LibSvm toolbox developed by Chang & Lin [63].

For the proposed IGDFS approach, the parameters for the SVM classifier were obtained using the Grid Search algorithm. The grid search algorithm is widely used in the literature for model selection to obtain the best penalty parameter C and the kernel parameter γ [64].

4.4 The Datasets

This section details the characteristics of the datasets used in this study.

Table 1: Characteristics of all the datasets

Dataset	N	n	N_n	N_p
German Credit	1000	20	700	300
Australian Credit	690	14	307	383
Taiwan Credit	30000	24	23364	6636

In above table,

N = number of total samples present in the dataset,

n = number of features in the dataset,

N_n = number of good credit samples,

N_p = number of bad credit samples.

4.4.1 The German Credit Dataset

The German Credit dataset [70] contains observations for 1000 past credit applicants on 20 variables. The applicants are rated as ‘good credit’ or ‘bad credit’. The two target

classes are distributed as: 700 samples (70%) for ‘good credit’ class and 300 samples (30%) for ‘bad credit’ class.

4.4.2 The Australian Credit Dataset

The Australian Credit dataset [71] contains data from credit card applications. The distribution of two target classes is fair, with 307 cases ($\approx$ 44.5%) of ‘good credit’ and 383 cases ($\approx$ 55.5%) of ‘bad credit’.

4.4.3 The Taiwan Credit Dataset

The Taiwan Credit dataset [72] contains data about customers’ default payment in Taiwan. This is the largest dataset used in this study. The two target classes have 23364 cases (77.88%) of ‘good credit’ and 6636 cases (22.12%) of ‘bad credit’.

4.5 Attribute normalisation

Often the attributes in the data have varying scales i.e. attributes with larger numeric ranges may dominate those with smaller numeric ranges. One way to overcome this is by using attribute normalisation. Kernel values are calculated by inner products of feature vectors where greater-numeric-range attributes might cause numerical problems and normalisation avoids these numerical difficulties [69]. We performed linear normalisation on each attribute to the range [0, +1] using following formula:

$$x' = \frac{x - min(x)}{max(x) - min(x)} \quad (8)$$

where *x'* is the normalised value of feature *x*, *x* is the original value of feature *x*, *min(x)* and *max(x)* are the minimum and maximum values of feature *x*.

The rest of this section details the parameter selection method for SVM and KNN techniques.

4.5.1 SVM parameters selection

C is the cost of classification and γ is the kernel parameter for a nonlinear support vector machine (SVM) with a Gaussian radial basis function kernel.

The general procedure in developing an SVM is to optimise both C and γ for a dataset. The problem of optimising these parameter values is called model selection, and the selection results strongly influence the performance of the classifier. Accuracy is used to evaluate the performance of a model on the datasets. To achieve good performance, some preliminary experiments were conducted to determine the optimal model parameters using exhaustive grid search approach [69] in finding the best C and γ for each dataset.

Both C and γ are scale parameters, so the grid is on a logarithmic scale. Doubling/halving C and γ on adjacent grid points is a tried and tested process, as a complete grid-search is a time-consuming process. If the search grid too fine, we may end up over-fitting the model selection criterion, so a fairly coarse grid turns out to be good for generalisation as well as computational expense. We exponentially increase the values of C and γ to identify best parameters [69]. A coarse grid is used first to identify promising region on the grid and then a finer grid search is conducted on that region to obtain a better cross-validation rate.

The grid search is described below:

Step 1: Set up a grid in decision space of (C, γ) with $\log_2 C \in \{-5,\ldots 15\}$ and $\log_2 \gamma \in \{-15, ...,3\}$.

Step 2: Train SVM on each pair of (C, γ) in the model space, with k-fold cross validation.

Step 3: Experiment with various pairs of (C, γ) values and choose the one that leads to the highest accuracy in cross validation.

Step 4: These best parameters are used to build a predictive model.

4.5.2 KNN parameter selection

The optimal *K* (number of neighbours) for KNN is the parameter that corresponds to the lowest test error rate. We want to choose the tuning parameters, which best generalize the data and which leads to the highest classification generalizability. In a better approach, the test error rate is estimated by taking a subset from the training set in the fitting process [73], [74]. We used *k*-fold cross validation as performance testing algorithm along with KNN. Various quantities of *K* were used as near rule-of-thumb values. On each dataset, we employed KNN with different values for *K* and obtained different accuracy for each *K*. The *K* which leads to achieving the best accuracy is the optimum *K*.

4.6 Genetic Algorithm parameters

The general approach in determining the appropriate parameter set of genetic algorithm for a given dataset is to conduct a number of trials of different combinations and choose the best combination that produces good results for the particular problem [75]. In this study, the parameters of GA are selected, referring to the ones [41], [76]. We tried various values of population size (20–100), mutation rate (0.001–0.3), and number of generations (20-100) to compare and obtain the best parameter combination. The final values of GA parameters obtained after these comparisons which are used to train the GA system are summarised in Table 2.

Table 2: The main GA parameters.

Parameter	**Value**
Objective function	Fitness value = Average accuracy
Population Size	50-70
Number of generations	20-50
Parent Selection	Tournament selection

Tournament Size	2
Crossover Type	Single point
Mutation Rate	0.1
Mutation Type	Uniform mutation
Stop Condition	Max number of generations

4.7 Experimental Results and Discussion

4.7.1 Information Gain based Ranking

Tables 3-5 show the information gain ranking for the features of all three datasets. The ranking directly reflects the contribution of the features towards classification. Considering these rankings, we devised the information gain directed feature selection (IGDFS) algorithm. From the table below, the feature 'Credit amount' is the most informative among all features and 'Number of people being liable to provide maintenance for' is the least informative in the German credit dataset.

Table 3: The order of features based on Information Gain for the German Credit Dataset

Rank No.	**Feature name**	**Rank No.**	**Feature name**
1	Credit amount	11	Other instalment plans
2	Status of existing checking account	12	Personal status and sex
3	Duration in months	13	Foreign worker
4	Age in years	14	Other debtors / guarantors

5	Credit history	15	Instalment rate in percentage of disposable income
6	Savings account/bonds	16	Number of existing credits at this bank
7	Purpose	17	Job
8	Property	18	Telephone
9	Present employment since	19	Present residence since
10	Housing	20	Number of people being liable to provide maintenance for

Table 4: The order of features based on Information Gain for the Australian Credit Dataset

Rank No.	**Feature name**	**Rank No.**	**Feature name**
1	X_2	8	X_9
2	X_{14}	9	X_5
3	X_8	10	X_6
4	X_3	11	X_4
5	X_{13}	12	X_{12}
6	X_7	13	X_{11}
7	X_{10}	14	X_1

Table 4 shows the ranking of features for Australian Credit dataset. This dataset does not name the features but identifies them with the labels X_1, X_2,….X_{14}. As per the information gain ranking, feature X_2 is the most informative and X_1 is the least informative.

Table 5: The order of features based on Information Gain for the Taiwan Credit Dataset

Rank No.	**Feature name**	**Rank No.**	**Feature name**
1	*BILL_AMT_1*	13	*PAY_0*
2	*BILL_AMT_2*	14	*PAY_2*
3	*BILL_AMT_3*	15	*PAY_3*
4	*BILL_AMT_4*	16	*PAY_4*
5	*BILL_AMT_5*	17	*PAY_5*
6	*BILL_AMT_6*	18	*PAY_6*
7	*PAY_AMT_1*	19	*SEX*
8	*PAY_AMT_2*	20	*EDUCATION*
9	*PAY_AMT_3*	21	*MARRIAGE*
10	*PAY_AMT_6*	22	*LIMIT_BAL*
11	*PAY_AMT_4*	23	*AGE*
12	*PAY_AMT_5*		

Table 5 shows the ranking of features for Taiwan Credit dataset. As per the information gain ranking, the feature *BILL_AMT_1(Amount of bill statement in September, 2005 (NT dollar))* is the most informative and *AGE* is the least informative.

4.7.2 Parameter selection for SVM by Grid-Search method

A grid search was employed to search the SVM parameter space using a logarithmic scale. A coarse search is first performed with a step $\Delta_{C\ \text{coarse}}$ for parameter C *in* the range of $[2^{-5}, 2^{15}]$ and a step $\Delta_{\gamma\ \text{coarse}}$ for γ in the range $[2^{-15}, 2^{3}]$, where $\Delta_{C\ \text{coarse}} = \Delta_{\gamma\ \text{coarse}} = 2$. Then a finer search with step size $\Delta_{C\ \text{fine}} = \Delta_{\gamma\ \text{fine}} = 0.0625$ is carried out in the promising region obtained on the coarse grid. The prediction accuracy (10-fold) showed the best value at (C, γ) = (2.1810, 0.0423) for German credit dataset. Thus, the optimal values of C and γ for this dataset are 2.1810 and 0.0423, respectively (Figure 2).

Figures 2-4 below show the contour plot of grid search results for optimum values of SVM parameters C and γ. The two parameters are shown in logarithmic axes x and y in the graphs, the lines indicating the area where the deeper grid search was performed. The colours of the lines indicate the graphical bounds of the searched space in the graph. The parameter values obtained are used for training RBF-SVM.

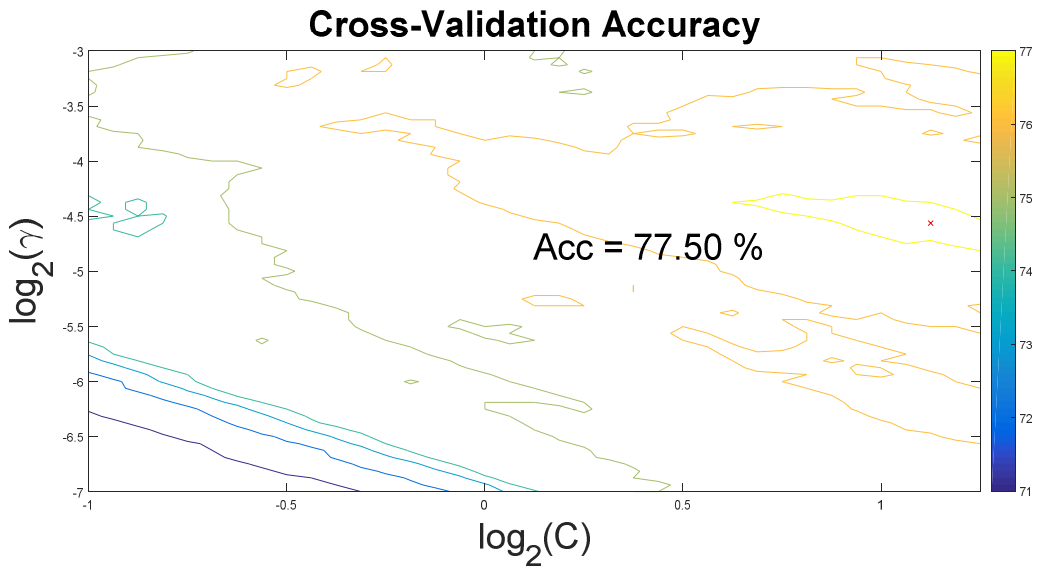


Figure 2: Grid search trace for optimised parameter values for German credit dataset. The model peaks at Accuracy=77.50%; (C=2.1810, γ=0.0423)

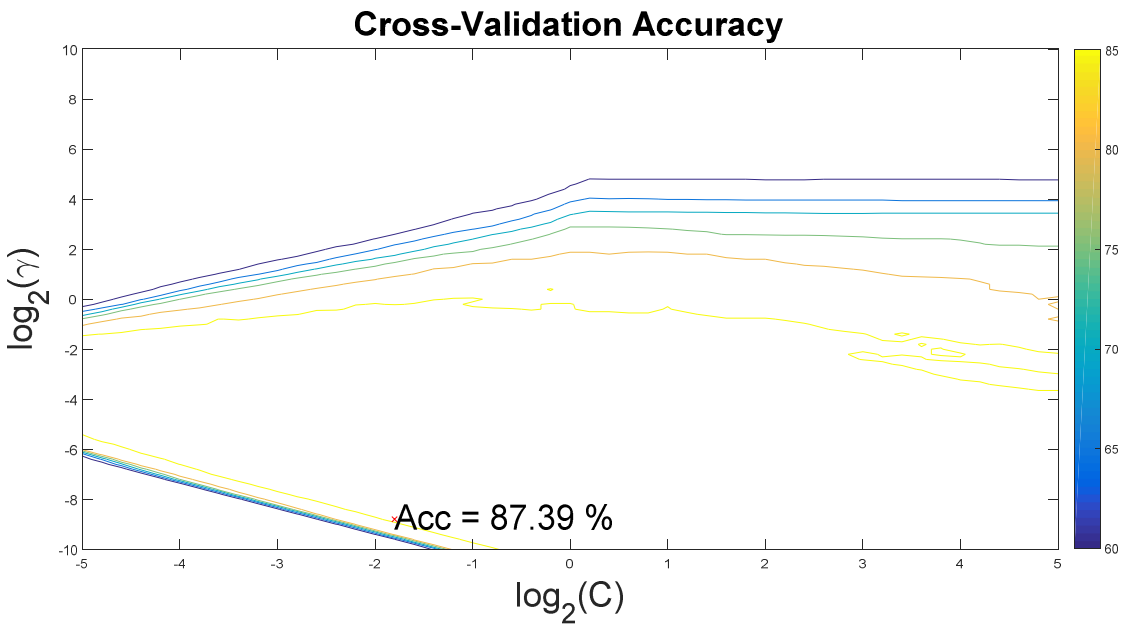


Figure 3: Grid search trace for optimised parameter values for Australian credit dataset. The model peaks at Accuracy=87.39%; (C=0.2872, γ=0.0022)

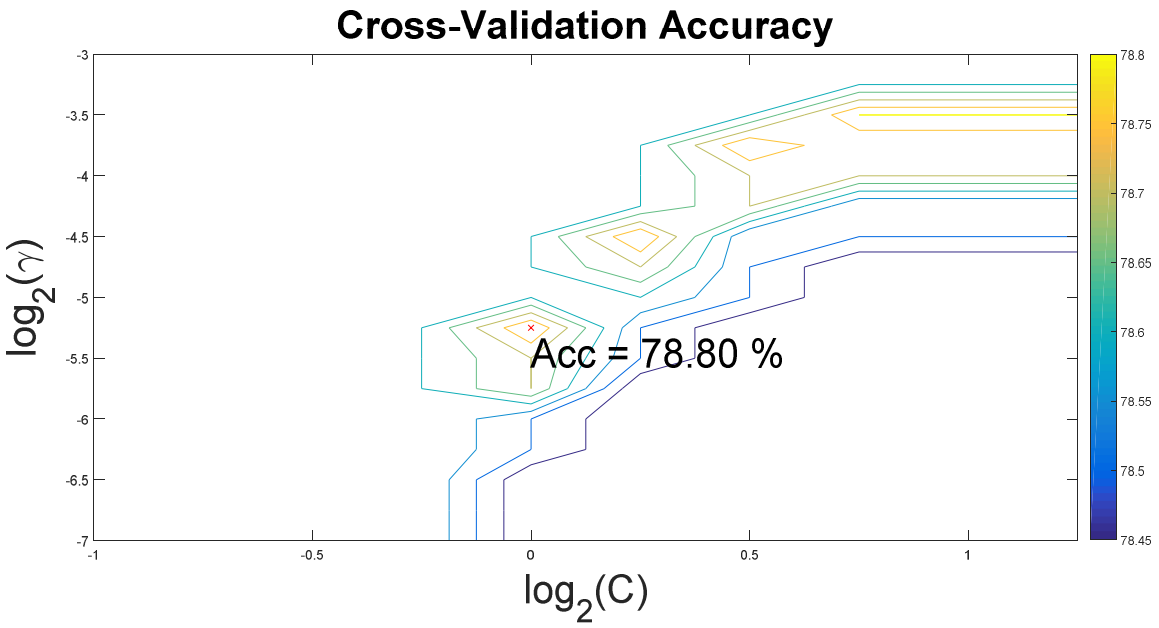


Figure 4: Grid search trace for optimised parameter values for Taiwan credit dataset. The model peaks at Accuracy=78.80%; (C=1, γ=0.0263)

Above grid search shows how the SVM classifier is optimised by cross validation using accuracy score. There are no rules of thumb for grid search parameter optimisation. The parameters are found at the best accuracy score of 77.5%, 87.39% and 78.80% for the German credit, Australian credit and Taiwan credit datasets respectively. The parameter values obtained are used for the experiments in next sections.

4.7.3 Accuracies for best solutions

To strengthen the significance of feature selection, we first ran experiments on baseline classifiers with all features before applying GAW and then IGDFS using the three classical classifiers (Table 6).

In GAW, Genetic algorithm acts as a wrapper technique with performance of three classical machine learning algorithms used to obtain the best fitness function. In the IGDFS algorithm, the top-ranking features obtained from information gain ranking are propagated through the wrapper process as shown in Algorithm 1 in previous section.

The results of 10-fold cross validation on GAW and IGDFS for all the datasets are shown in table below. The best average classification results are printed in bold.

It is seen that the GAW and IGDFS algorithms have performed better than the baseline classifier algorithms. Hence, feature selection improves the performance of classification compared to baseline methods. Compared with GAW, IGDFS gives improved accuracy in most of the classifiers except KNN (German credit data) and NB (Taiwan credit data).

Table 6: Accuracy Performance of different classifiers over three datasets. (Best performance in bold italics)

Method	**German Credit data**	**Australian Credit data**	**Taiwan Credit data**

SVM	Baseline	76.4	85.7	81.9
	GAW	80.4	89.0173	81.2097
	IGDFS	***82.8***	***90.7514***	***82.5733***
KNN	Baseline	75.2	85.7	80.8
	GAW	***75.8***	85.6522	80.9833
	IGDFS	70.2	***86.75***	***81.1733***
NB	Baseline	73.70	80.43	71.36
	GAW	76.8	86.79131	***82.0267***
	IGDFS	***77.3***	***87.971***	81.98

4.7.4 ROC curves for the best solutions

ROC curves allow for a detailed analysis of the differences. Figure 5 shows the ROC curves obtained with IGDFS for the three classifier algorithms on the German credit dataset.

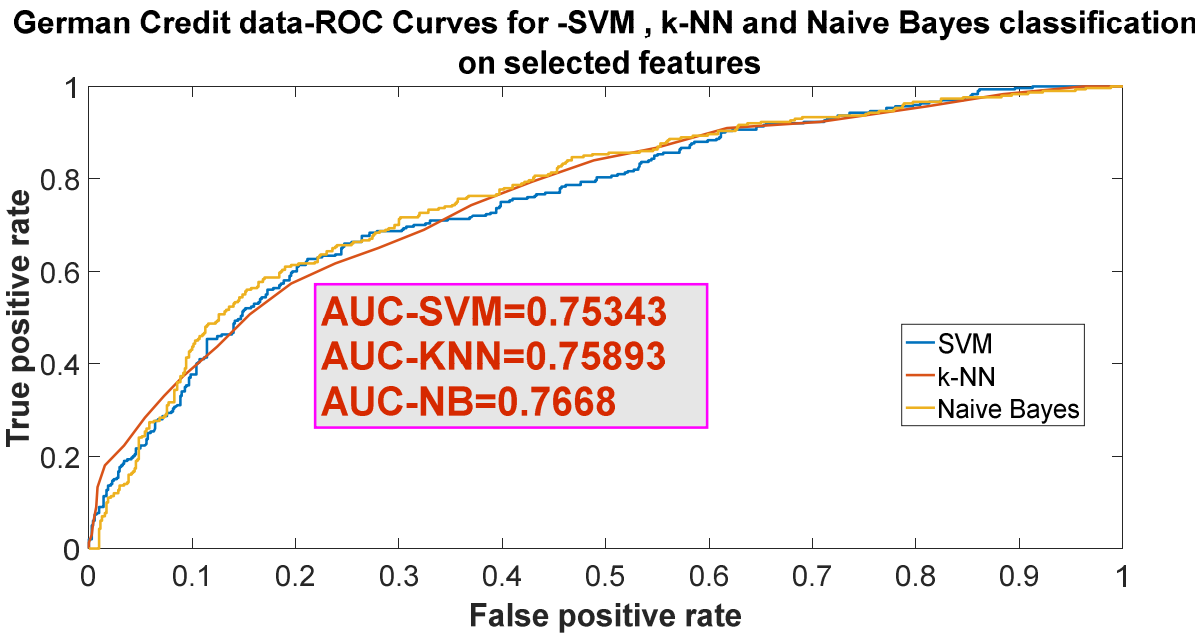


Figure 5: ROC results of IGDFS on German credit dataset

This figure shows that the three classifiers in the wrapper of the GA with IGDFS obtained almost the same performance for this dataset. The perfect close to the top left corner have a better performance level than the ones closer to the baseline. Comparisons of all the classifiers shows that the ROC curves are crossing each other. FPR (=1-Specificity) defines how many samples are classified as bad even if they were good credit. For smaller false positive rates, i.e. for early retrieval area (a region with high specificity values in the ROC space- FPR between 0 and 0.1), IGDFS+KNN classifier (red curve) seems to perform better. For middle FPR (between 0.1 and 0.75), IGDFS+NB (yellow curve) gives good results. As the FPR increases beyond 0.75, IGDFS+SVM (blue curve) performs best.

Figure 6 shows the performance of all three classifiers on Australian credit dataset. IGDFS + NB, which has the largest area under ROC curve, performs best in classifying the credit applicants in Australian Credit dataset. Next best performance is shown by IGDFS+KNN, followed by IGDFS+SVM.

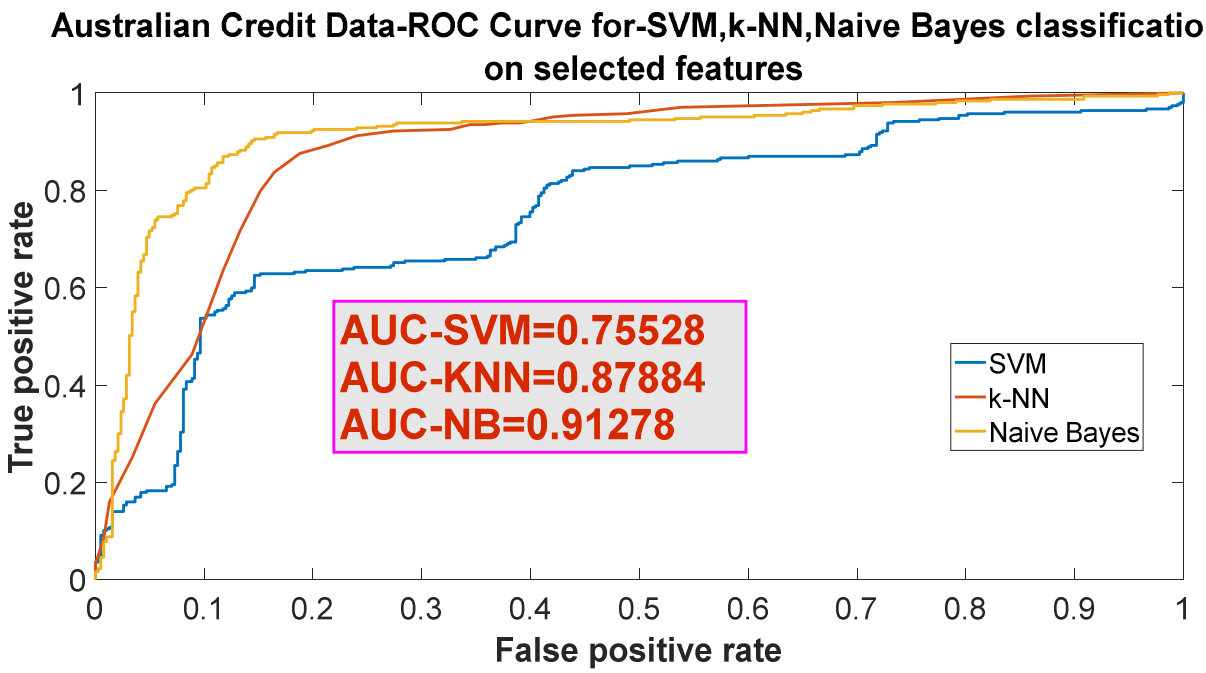


Figure 6: ROC results of IGDFS on Australian credit dataset

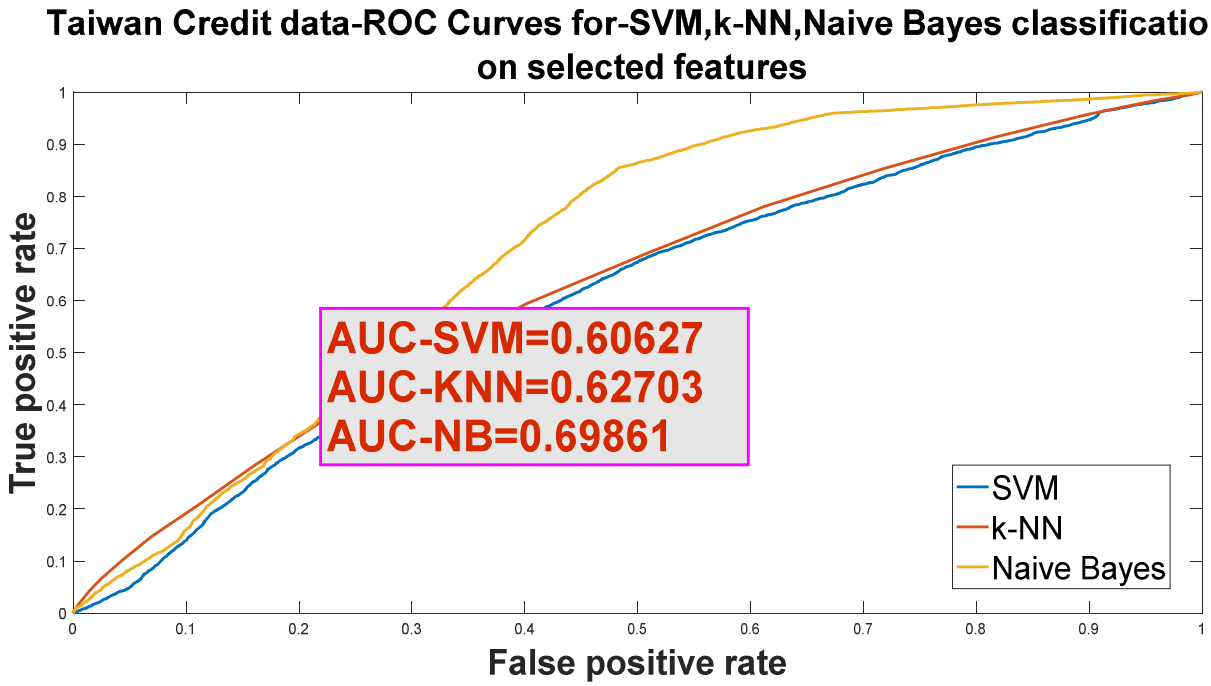


Figure 7: ROC results of IGDFS on Taiwan credit dataset

For the Taiwan credit dataset IGDFS+NB gives best results followed by IGDFS+KNN, followed by IGDFS+SVM (Figure 7).

Observing the performance in Figures 5-7, the classifier and IGDFS combination giving best ROC performance for all three datasets is IGDFS+NB.

4.7.5 Comparison of GAW and IGDFS for all datasets

Here, the performance of IGDFS is compared with the GAW algorithm based feature selection method and the baseline classifiers in terms of prediction accuracy made by three different classifiers KNN, and Naïve Bayes and RBF-SVM (Table 6). The findings are:

- GAW+SVM performed better than the baseline SVM for all the datasets. This implies that selected features have positive support to the decision making with the RBF-SVM for the datasets. There is an improvement in the performance compared with the work done by Liang et al. [42]. But the best accuracy results are obtained by the improved IGDFS algorithm, where we identified important features and propagated them throughout the whole wrapper process.
- GAW+KNN performed slightly better than baseline KNN over the German and Taiwan credit dataset, but not for Australian credit dataset. Our proposed IGDFS with KNN has performed best in Australian and Taiwan credit dataset but not the German credit dataset.
- GAW+NB significantly outperformed the baseline NB in all three datasets. This finding was consistent with similar work by Chen et al. [77], who found that NB classifier was highly sensitive to feature selection and the work done by Liang et al [42]. The IGDFS again has proved to be the best method and it gives the highest prediction accuracy for NB in all the datasets.

For the German credit data with 7 numerical and 13 categorical features:

- There is much less variation in the classification accuracy of IGDFS for all three classifiers;
- Wrapper methods (GAW) clearly outperform baseline methods;
- These GAW methods have shown very high-performance improvement when used with SVM and NB as underlying classifier and acceptable classification accuracy improvement on the German dataset than with KNN.
- IGDFS performance is best with SVM and NB. GAW performs best with KNN for this dataset.

For the Australian credit data (6 numerical and 8 categorical features):

- There is a lot of variation in the classification accuracy of IGDFS for all three classifiers;
- Wrapper methods (GAW) outperform baseline methods except for the KNN method;
- IGDFS performs best for all the three classifiers.

For the Taiwan credit data (16 numeric and 8 categorical features):

- There is not much variation in the results for all the three techniques;
- IGDFS performs better than GAW and baseline for all three classifiers except in NB.

5 Conclusion

Credit scoring is one of the significant problems in computational finance. In this work, we developed the IGDFS, based on Information Gain and Wrapper technique, using three different classical decision-making models of KNN, NB and SVM to select features for the credit scoring problem. The average prediction performance by IGDFS, Genetic Algorithm Wrapper and Baseline models are compared.

The intuition behind this work is that not all features are equally important and retaining the top contributing feature into the final selected subset may improve the results of classification, since the features not important to decision making could affect the performance of decision making.

Looking at experimental results for all the datasets investigated, it is observed that the classification accuracy achieved with different strategies is highly sensitive to the type of data, size of data set and the rate of positive and negative samples in the dataset.

Among the three machine learning algorithms investigated, accuracies for the SVM with baseline, GAW and IGDFS are consistently higher across all the datasets compared with those for KNN and NB. This provides an evidence for the claim that SVMs may indeed suffer in high dimensional spaces where many features are irrelevant and feature selection may result in significant improvement in their performance [78].

GAW+KNN and IGDFS+KNN have shown very little improvement in the accuracy of classification on the selected feature sets for all the datasets, compared with the baseline KNN on the full features; for German credit dataset, the accuracy obtained by IGDFS has in fact dropped. This might be because KNN is sensitive to the local structure of the data, and the data structure is decided by Euclidean distance. When we remove some features with low information gain in the process of decision making, the reduction of features could affect the structure. Namely, the information gain of features could produce conflict with the original data structure for KNN.

Wrapper feature selection is a costly method due to its comprehensive search on the feature space. To reduce its computational cost, we used Information Gain to guide the feature selection initially. This step removes features with low information gain, so that the wrapper method is carried out on a smaller space, and the time complexity is reduced. This can be seen by the results on all three credit datasets used in the study. We can conclude

that there is a potential for improvement in the models' performances if the feature selection method is chosen carefully.

In future studies, the results with other combinations of parameters for genetic algorithms could be studied. The method of convergence as a stopping criterion for the GA will also be investigated. The performance of IGDFS algorithm could be investigated with other high dimensional datasets. Because of the nature of the credit scoring problem and its real application domain, computing complexity is important concern when generating credit scoring models [79]. Reducing the cost of credit analysis and aiding faster credit evaluation are among top objectives of credit scoring models. The computational complexity of the proposed algorithm, both in training and at runtime needs to be assessed to make it robust. Also, the combination of other evolutionary algorithms and other machine learning algorithms could be explored in future. Lastly, we aim to develop a soft package based on the technique for public use in future.

References

[1] S. Jadhav, H. He, K.W. Jenkins, An Academic Review: Applications of Data Mining Techniques in Finance Industry, Int. J. Soft Comput. Artif. Intell. 4 (2017) 79–95.

[2] S. Jadhav, H. He, K. Jenkins, Prediction of Earnings per Share for Industry, in: Knowl. Discov. Knowl. Eng. Knowl. Manag. (IC3K), 2015 7th Int. Jt. Conf., 2015: pp. 425–432.

[3] T. Harris, Credit scoring using the clustered support vector machine, Expert Syst. Appl. (2015).

[4] D. Roobaert, G. Karakoulas, N. V Chawla, Information Gain , Correlation and Support Vector Machines, Featur. Extr. Found. Appl. 470 (2006) 463–470.

[5] A.L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artif. Intell. 97 (1997) 245–271. doi:10.1016/S0004-3702(97)00063-5.

[6] D. Koller, M. Sahami, Toward optimal feature selection, Stanford InfoLab. (1996).

[7] A. Janecek, W. Gansterer, M. Demel, G. Ecker, On the relationship between feature selection and classification accuracy, New Challenges Featur. Sel. Data Min. Knowl. Discov. (2008) 90–105.

[8] O. Soufan, D. Kleftogiannis, P. Kalnis, V.B. Bajic, DWFS: A Wrapper Feature Selection Tool Based on a Parallel Genetic Algorithm, PLoS One. 10 (2015) e0117988. doi:10.1371/journal.pone.0117988.

[9] L. Zhuo, J. Zheng, X. Li, F. Wang, B. Ai, J. Qian, A genetic algorithm based wrapper feature selection method for classification of hyperspectral images using support vector machine, in: Geoinformatics 2008 Jt. Conf. GIS Built Environ. Classif. Remote Sens. Images, International Society for Optics and Photonics, 2008: p. 71471J–71471J. doi:10.1117/12.813256.

[10] H. Chen, W. Jiang, C. Li, R.L.-M. problems in Engineering, U. 2013, A heuristic feature selection approach for text categorization by using chaos optimization and genetic algorithm, Hindawi.com. (2013).

[11] M. Naseriparsa, A.-M. Bidgoli, T. Varaee, A Hybrid Feature Selection Method to Improve Performance of a Group of Classification Algorithms, (2014). doi:10.5120/12065-8172.

[12] C. Liu, D. Jiang, W.Y.-E.S. with Applications, U. 2014, Global geometric similarity scheme for feature selection in fault diagnosis, Elsevier. 41 (2014) 3585–3595.

[13] T. Mitchell, Machine learning, McGraw Hill Ser. Comput. Sci. (1997).

[14] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, in: Proc. Fifth Annu. Work. Comput. Learn. Theory - COLT '92, ACM Press, New York, New York, USA, 1992: pp. 144–152. doi:10.1145/130385.130401.

[15] C. Cortes, V. Vapnik, Soft margin classifier, US Pat. 5,640,492. (1997).

[16] M. Mitchell, An introduction to genetic algorithms, MIT press, 1998.

[17] J. Kennedy, Particle swarm optimization, Encycl. Mach. Learn. (2011) 760–766.

[18] R. Kohavi, G.H. John, The Wrapper Approach, in: Featur. Extr. Constr. Sel., Springer US, Boston, MA, 1998: pp. 33–50. doi:10.1007/978-1-4615-5725-8_3.

[19] L. Jourdan, C. Dhaenens, E. Talbi, A genetic algorithm for feature selection in data-mining for genetics, Proc. 4th Metaheuristics Int. Conf. (2001) 29–34.

[20] S. Maldonado, J. Pérez, C. Bravo, Cost-based feature selection for Support Vector Machines: An application in credit scoring, Eur. J. Oper. Res. 261 (2017) 656–665. doi:10.1016/j.ejor.2017.02.037.

[21] N. Verbiest, J. Derrac, C. Cornelis, S. García, F. Herrera, Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis, Appl. Soft Comput. 38 (2016) 10–22. doi:10.1016/j.asoc.2015.09.006.

[22] H. Frohlich, O. Chapelle, B. Schölkopf, "Feature selection for support vector machines by means of genetic algorithm" In Tools with artificial intelligence, in: 15th

Ieee Int. Conf., IEEE, 2003: pp. 142–148.

[23] R.C. Anirudha, R. Kannan, N. Patil, Genetic algorithm based wrapper feature selection on hybrid prediction model for analysis of high dimensional data, in: 2014 9th Int. Conf. Ind. Inf. Syst., IEEE, 2014: pp. 1–6. doi:10.1109/ICIINFS.2014.7036522.

[24] J. Huang, Y. Cai, X. Xu, A hybrid genetic algorithm for feature selection wrapper based on mutual information, Pattern Recognit. Lett. 28 (2007) 1825–1844. doi:10.1016/j.patrec.2007.05.011.

[25] D. Kimovski, J. Ortega, A. Ortiz, R. Baños, Parallel alternatives for evolutionary multi-objective optimization in unsupervised feature selection, Expert Syst. Appl. 42 (2015) 4239–4252. doi:10.1016/j.eswa.2015.01.061.

[26] E.-S.M. El-Alfy, M.A. Alshammari, Towards scalable rough set based attribute subset selection for intrusion detection using parallel genetic algorithm in MapReduce, Simul. Model. Pract. Theory. 64 (2016) 18–29. doi:10.1016/j.simpat.2016.01.010.

[27] Z. Chen, T. Lin, N. Tang, X. Xia, A parallel genetic algorithm based feature selection and parameter optimization for support vector machine, Sci. Program. (2016).

[28] H. Sabzevari, M. Soleymani, E. Noorbakhsh, A comparison between statistical and data mining methods for credit scoring in case of limited available data, Proc. Credit Scoring Conf. UK. (2007) 1–8.

[29] M. Khanbabaei, M. Alborzi, The use of genetic algorithm, clustering and feature selection techniques in construction of decision tree models for credit scoring, Int. J.

Manag. Inf. Technol. 5 (2013) 13–31.

[30] Y. Liu, M. Schumann, Data mining feature selection for credit scoring models, J. Oper. Res. Soc. (2005).

[31] S. Sadatrasoul, M. Gholamian, Combination of feature selection and optimized fuzzy apriori rules: the case of credit scoring., Int. Arab J. Inf. Technol. 12 (2015) 138–145.

[32] R. Allami, A. Stranieri, A genetic algorithm-neural network wrapper approach for bundle branch block detection, Comput. Cardiol. Conf. (2016) 461–464.

[33] A. Özçift, A. Gülten, Genetic algorithm wrapped Bayesian network feature selection applied to differential diagnosis of erythemato-squamous diseases, Digit. Signal Process. 23 (2013) 230–237. doi:10.1016/j.dsp.2012.07.008.

[34] A. Daamouche, F. Melgani, N. Alajlan, Swarm optimization of structuring elements for VHR image classification, IEEE Geosci. Remote Sens. Lett. 10 (2013) 1334–1338.

[35] S. Lin, K. Ying, S. Chen, Z. Lee, Particle swarm optimization for parameter determination and feature selection of support vector machines, Expert Syst. Appl. 35 (2008) 1817–1824.

[36] A. Milne, M. Rounds, P. Goddard, Optimal feature selection in credit scoring and classification using a quantum annealer, 1QB Inf. Technol. (2017).

[37] Y. Liu, M. Schumann, Data mining feature selection for credit scoring models, J. Oper. Res. Soc. 56 (2005) 1099–1108. doi:10.1057/palgrave.jors.2601976.

[38] H. Liu, H. Motoda, Feature selection for knowledge discovery and data mining (Vol. 454), Springer Science & Business Media, 2012.

[39] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.

[40] P. Somol, B. Baesens, P. Pudil, Filter-versus wrapper-based feature selection for credit scoring, Int. J. Intell. Syst. 20 (2005) 985–999.

[41] C.-L. Huang, M.-C. Chen, C.-J. Wang, Credit scoring with a data mining approach based on support vector machines, Expert Syst. Appl. 33 (2007) 847–856. doi:10.1016/j.eswa.2006.07.007.

[42] D. Liang, C.F. Tsai, H.T. Wu, The effect of feature selection on financial distress prediction, Knowledge-Based Syst. 73 (2014) 289–297. doi:10.1016/j.knosys.2014.10.010.

[43] B. Waad, B.M. Ghazi, L. Mohamed, A three-stage feature selection using quadratic programming for credit scoring, Appl. Artif. Intell. 27 (2013) 721–742.

[44] F. Li, The hybrid credit scoring strategies based on knn classifier, in: Fuzzy Syst. Knowl. Discov. 2009. FSKD'09. Sixth Int. Conf., IEEE, 2009: pp. 330–334.

[45] F.-L. Chen, F.-C. Li, Combination of feature selection approaches with SVM in credit scoring, Expert Syst. Appl. 37 (2010) 4902–4909. doi:10.1016/j.eswa.2009.12.025.

[46] N.-C. Hsieh, L.-P. Hung, A data driven ensemble classifier for credit scoring analysis, Expert Syst. Appl. 37 (2010) 534. doi:http://dx.doi.org/10.1016/j.eswa.2009.05.059".

[47] F. Koutanaei, H. Sajedi, M. Khanbabaei, A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring, J. Retail. (2015).

[48] I. Brown, C. Mues, An experimental comparison of classification algorithms for imbalanced credit scoring data sets, Expert Syst. Appl. 39 (2012) 3446. doi:http://dx.doi.org/10.1016/j.eswa.2011.09.033.

[49] K. Kim, H. Ahn, A corporate credit rating model using multi-class support vector machines with an ordinal pairwise partitioning approach, Comput. Oper. Res. 39 (2012) 1800. doi:http://dx.doi.org/10.1016/j.cor.2011.06.023.

[50] L. Yu, X. Yao, S. Wang, K.K. Lai, Credit risk evaluation using a weighted least squares SVM classifier with design of experiment for parameter selection, Expert Syst. Appl. 38 (2011) 15392–15399. doi:10.1016/j.eswa.2011.06.023.

[51] A.Z. Hamadani, A. Shalbafzadeh, T. Rezvan, A. Moghadam, An Integrated Genetic-Based Model of Naive Bayes Networks for Credit Scoring, Int. J. Artif. Intell. Appl. 4 (2013) 85–103. doi:10.5121/ijaia.2013.4107.

[52] J. Wang, A.-R. Hedar, S. Wang, J. Ma, Rough set and scatter search metaheuristic based feature selection for credit scoring, Expert Syst. Appl. 39 (2012) 6123–6128. doi:10.1016/j.eswa.2011.11.011.

[53] P. Hajek, K. Michalak, Feature selection in corporate credit rating prediction, (2013). doi:10.1016/j.knosys.2013.07.008.

[54] S. Oreski, G. Oreski, Genetic algorithm-based heuristic for feature selection in credit risk assessment, Expert Syst. Appl. 41 (2014) 2052–2064.

doi:10.1016/j.eswa.2013.09.004.

[55] H. Van Sang, N. Nam, N. Nhan, A novel credit scoring prediction model based on Feature Selection approach and parallel random forest, Indian J. Sci. (2016).

[56] W. Bouaguel, On Feature Selection Methods for Credit Scoring, Ph. D. thesis, Institut Superieur de Gestion de Tunis, 2015.

[57] V.-S. Ha, H.-N. Nguyen, FRFE: Fast Recursive Feature Elimination for Credit Scoring, in: 2016: pp. 133–142. doi:10.1007/978-3-319-46909-6_13.

[58] H. Liu, H. Motoda, Feature extraction, construction and selection: A data mining perspective, 1998.

[59] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Recent advances and emerging challenges of feature selection in the context of big data, Knowledge-Based Syst. 86 (2015) 33–45. doi:10.1016/j.knosys.2015.05.014.

[60] B. Twala, Multiple classifier application to credit risk assessment, Expert Syst. Appl. 37 (2010) 3326–3336. doi:10.1016/j.eswa.2009.10.018.

[61] L. Li, W. Jiang, X. Li, K.L. Moser, Z. Guo, L. Du, Q. Wang, E.J. Topol, Q. Wang, S. Rao, A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset, Genomics. 85 (2005) 16–23. doi:10.1016/j.ygeno.2004.09.007.

[62] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T.S. Furey, M. Ares, D. Haussler, Knowledge-based analysis of microarray gene expression data by using support vector machines, Knowledge-Based Anal. Microarray Gene Expr. Data

by Using Support Vector Mach. 97 (2000) 262–267.

[63] N. Cristianini, J. Shawe-Taylor, An introduction to support vector machines, Cambridge University Press, Cambridge, UK, 2000.

[64] S. Maldonado, R. Weber, A wrapper method for feature selection using Support Vector Machines, Inf. Sci. (Ny). 179 (2009) 2208–2217. doi:10.1016/j.ins.2009.02.014.

[65] H. He, A. Tiwari, J. Mehnen, T. Watson, C. Maple, Y. Jin, B. Gabrys, Incremental information gain analysis of input attribute impact on RBF-kernel SVM spam detection, in: 2016 IEEE Congr. Evol. Comput. CEC 2016, IEEE, 2016: pp. 1022–1029. doi:10.1109/CEC.2016.7743901.

[66] L. Wang, Y. Jin, Fuzzy Systems and Knowledge Discovery: Second International Conference, FSKD 2005, Changsha, China, August 27-29, 2005, Proceedings, Springer Science & Business Media, 2005.

[67] M. Lankhorst, Genetic algorithms in data analysis, [University Library Groningen][Host], 1996.

[68] C.-C. Chang, C.-J. Lin, LIBSVM: A Library for Support Vector Machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 27.

[69] C.-W. Hsu, C.-C. Chang, C.-J. Lin, others, A practical guide to support vector classification, (2003).

[70] M. Lichman, Statlog (German Credit Data) Data Set, Irvine, CA Univ. California, Sch. Inf. Comput. Sci. (2013).

https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data).

[71] M. Lichman, Statlog (Australian Credit Approval) Data Set, Irvine, CA Univ. California, Sch. Inf. Comput. Sci. (2013). http://archive.ics.uci.edu/ml/datasets/statlog+(australian+credit+approval).

[72] M. Lichman, Default of credit card clients Data Set, Irvine, CA Univ. California, Sch. Inf. Comput. Sci. (2013). http://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients.

[73] A. Statnikov, C. Aliferis, I. Tsamardinos, D. Hardin, A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis, Bioinformatics. 21 (2004) 631–643.

[74] F. Pedregosa, G. Varoquaux, A. Gramfort, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[75] I. Kucukkoc, A. Karaoglan, R. Yaman, Using response surface design to determine the optimal parameters of genetic algorithm and a case study, Int. J. Prod. Res. 51 (2013) 5039–5054.

[76] M. Srinivas, L.M. Patnaik, Genetic Algorithms: A Survey, Computer (Long. Beach. Calif). 27 (1994) 17–26. doi:10.1109/2.294849.

[77] J. Chen, H. Huang, S. Tian, Y. Qu, Feature selection for text classification with Naive Bayes, Expert Syst. Appl. 36 (2009) 5432–5435. doi:10.1016/j.eswa.2008.06.054.

[78] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, V. Vapnik, B. BioInformaticscom, Feature Selection for SVMs, Adv. Neural Inf. Process. Syst.

(2001) 668–674.

[79] A.B. Hens, M.K. Tiwari, Computational time reduction for credit scoring: An integrated approach based on support vector machine and stratified sampling method, Expert Syst. Appl. 39 (2012) 6774. doi:http://dx.doi.org/10.1016/j.eswa.2011.12.057.

Cranfield University

CERES https://dspace.lib.cranfield.ac.uk

School of Aerospace, Transport and Manufacturing (SATM) Staff publications (SATM)

2018-04-22

Information gain directed genetic algorithm wrapper feature selection for credit rating

Jadhav, Swati

Elsevier

Jadhav S, Hongmei H, Jenkins K. (2018) Information gain directed genetic algorithm wrapper feature selection for credit rating. Applied Soft Computing, Volume 69, August 2018, pp. 541-553
https://doi.org/10.1016/j.asoc.2018.04.033
Downloaded from Cranfield Library Services E-Repository