Hybrid PSO-Based Rule Classifier for Disease Detection

Cecilia Mariciuc[1,2][a] and Madalina Raschip[1][b]

[1] *Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iasi, General Berthelot 16, Iasi, Romania*

[2] *RomSoft, Bulevardul Chimiei 2bis, Iasi, Romania*

Keywords: Data Mining, Classification Rules, Particle Swarm Optimization, Disease Detection.

Abstract: The application of data mining techniques in healthcare is common because the decision-making process for the diagnosis of medical conditions could benefit from the information extracted. A decision system must not only be accurate but also provide understandable explanations for its reasoning. Rule-based models seek to find a small set of rules that can effectively categorize data while providing great human readability. Rule discovery is a complex optimization problem, making it a good candidate for the application of PSO, a versatile, intuitive search algorithm. In this paper, a particle swarm optimization algorithm is used for learning classification rules as part of a Covering-based rule classifier. The proposed PSO is hybridized with the Iterated Local Search metaheuristic, and association rules are used as part of the initialization step. The classifier is tested on several unbalanced medical disease datasets with different types of attributes to more faithfully reflect real-world data. When compared with state-of-the-art rule-based classifiers, the studied algorithm shows good results and is highly interpretable.

1 INTRODUCTION

As more day-to-day activities become digitalized, medical databases continue to accumulate substantial information. Despite having more extensive data for patients, which has been shown to benefit the healthcare industry, it becomes more challenging to make sense of the stored knowledge. There is an increasing need to develop accurate methods that can discover valuable information hidden in the growingly complex data. Data mining uses powerful tools from statistics, machine learning, and databases to identify patterns. It has been shown to offer a lot of potential for developing illness prediction models, assessing patient risk, and assisting doctors in making treatment choices (Abdulkadium et al., 2022).

The process of categorizing data into predefined classes is known as classification and is done through an algorithm that produces a classification predictive model. In a rule-based model, predictions are made using a set of classification rules, which can be useful in the medical field for diagnosis and prognosis (Yoo et al., 2012). The objective of the classification rule discovery task is to identify a small set of rules from a training dataset that can optimally predict the classes.

[a] https://orcid.org/0000-0003-0826-832X

[b] https://orcid.org/0000-0003-0020-636X

Although many machine learning and deep learning methods produce accurate classification results, their decision-making processes often function as black boxes, offering limited insight into the mechanisms of how the outcomes were reached (Cristani et al., 2022). This makes it difficult for both professionals and patients to interpret any classification findings. Designing models that are interpretable rather than explaining black box models is generally preferable in situations with high-stakes decisions (Rudin, 2019). Furthermore, in the context of medical diagnostic tasks, an effective prediction system should not only demonstrate high performance but also possess the ability to explain decisions, operate efficiently with a limited amount of data, and produce transparent knowledge (Kononenko, 2001). Hence, we opted to construct a rule-based model, which has the advantage of being highly interpretable while having adequate performance.

Recently, in addition to data mining techniques, metaheuristics have been employed successfully in the medical field. Metaheuristic techniques are promising due to their versatility, tolerable accuracy, and easy perception of results. Within the metaheuristic paradigm, Particle Swarm Optimization (PSO) has found widespread use in a variety of fields, including medicine, because of its potential to adapt to spe-

Mariciuc, C. and Raschip, M.
Hybrid PSO-Based Rule Classifier for Disease Detection.
DOI: 10.5220/0012417900003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 3*, pages 906-914
ISBN: 978-989-758-680-4; ISSN: 2184-433X

cific scenarios and because it can achieve high performance through hybridization with other methods (Gad, 2022). Hybridizing various metaheuristics can lead to more robust solutions because of the combination of their different advantages. Particularly for a complex task such as medical classification, hybrid metaheuristics constitute a feasible approach for obtaining approximate yet satisfactory solutions (Al-Muhaideb and El Bachir Menai, 2013).

This paper introduces a rule-based classification technique inspired by the Covering technique (Fürnkranz, 1999), where the rule discovery process is done using an improved hybrid PSO algorithm. This method is applied for disease detection to obtain not only satisfactory accuracy but, more crucially, an understandable explanation of its predictions.

The rest of the paper is organized as follows: Section 2 provides a literature review of metaheuristics and rule-based methods for disease detection. A brief report of the PSO algorithm is given in Section 3. In Section 4, we describe the steps and the design of the proposed solution, and in Section 5, the experimental settings and results are given. We conclude with a summary and future improvements in Section 6.

2 RELATED WORK

Finding effective methods for decision-making in the healthcare sector is crucial given the volume of data generated (Alkeshuosh et al., 2017). A variety of data mining methods developed for disease detection make use of rule-based techniques or pattern recognition. One potential strategy is to incorporate association rules in various ways into the categorization task. In the study (Alaiad et al., 2020), the Apriori algorithm for mining association rules and different machine learning techniques have been used to create an effective decision system for the diagnosis of chronic kidney disease. Other approaches are predictive models expressed as decision lists or decision trees that rely on the recursive induction of rules. (Thabtah and Peebles, 2020) propose a method called Rules-Machine Learning for autism detection. It is based on Covering approaches and utilizes two metrics called Minimum Frequency and Rule Strength to find and extract the rules.

In addition to rule-based classifiers, there are many well studied state-of-the-art machine learning methods used for the classification of diseases, such as K-Nearest Neighbor, Naive Bayes, Support Vector Machines and others (Al-Hashem et al., 2021).

Metaheuristics, such as Ant Colony Optimization (ACO), have gained attention in rule discovery. The study (Hossain et al., 2022) provides an extensive review of ACO variants for rule-based classification, which emphasizes their high interpreatibility. Even though using swarm intelligence in data mining is challenging due to the heavy computational load and the complex nature of the domain, many recent works have obtained encouraging results. In (Zomorodi-moghadam et al., 2021) a hybrid Particle Swarm Optimization algorithm was used to find classification IF-THEN rules for the diagnosis of coronary artery disease. It is referred to as a hybrid algorithm due to the hybrid method used in updating the particles in the swarm and not to the hybridization with another metaheuristic such as the one proposed in the current study. A similar method was used in (Alkeshuosh et al., 2017), but for heart disease classification. In (Mangat and Vig, 2014) a PSO algorithm for rule mining is described. It was used as an intermediate step to build a rule classifier. The approach is similar to the one employed in the current study. Both classifiers are inspired by Covering approaches, so they share the idea of removing the examples that satisfy the best rule mined, but the rule discovery PSO algorithms differ because of the different representations that affect each step of the general scheme.

Hybrid metaheuristics have also produced promising results in medical classification. Hybrid metaheuristics can be used to enhance classification performance by solving tasks such as model selection and feature selection or to represent learning classifier systems, i.e., rule-based systems (Al-Muhaideb and El Bachir Menai, 2013). A hybrid PSO-ACO method that finds classification rules for both continuous and nominal data is described in (Holden and Freitas, 2008). The study conducted by (Al-behadili et al., 2020) introduces a rule-based algorithm that integrates ACO and Iterated Local Search (ILS) and achieves good classification performance across various datasets, including some in the medical domain.

The present research brings several contributions to the literature. First, the initialization step has been modified to use both random rules and altered association rules to provide good starting points without the need for a large swarm. The proposed PSO has been hybridized with the Iterated Local Search metaheuristic to deal with the potential premature convergence and the tendency to get stuck in a local optimum. The hybridization of PSO and ILS has not been previously employed for the classification rule discovery problem. Two different variations for the selection of the best rule were analyzed. We also focused on the interpretability of the model, not only on its predictive performance, as both are crucial for the disease detection task.

3 PARTICLE SWARM OPTIMIZATION

Particle swarm optimization is a stochastic optimization technique that is based on the social behavior of individuals that cooperate to form swarms to serve a common goal (Houssein et al., 2021). A swarm composed of potential solutions called particles is used to search the space of the optimization problem. A particle knows the best position of the swarm in the solution space, its own personal best, and a velocity to adjust its future position. The particle position is updated according to the velocity in each generation in order to obtain a new position potentially that is closer to the optimal solution.

Assume the following notations: $X_i = (x_{i,1}, x_{i,2}, ..., x_{i,N})$ denotes the particle's position vector, the velocity of a particle is $V_i = (v_{i,1}, v_{i,2}, ..., v_{i,N})$, and $P_i = (p_{i,1}, p_{i,2}, ..., p_{i,N})$ represents the best known individual position of X_i. The optimal position achieved so far is also known and is denoted by $P_g = (p_{g,1}, p_{g,2}, ..., p_{g,N})$. The standard formula for adjusting the velocity of a particle is depicted below. The particles in the current iteration t are updated using the newly computed velocity.

$$v_{i,d}^{t+1} = w * v_{i,d}^{t} + c_1 * r_1 * (p_{i,d}^{t} - x_{i,d}^{t}) + c_2 * r_2 * (p_{g,d} - x_{i,d}^{t})$$
$$x_{i,d}^{t+1} = x_{i,d}^{t} + v_{i,d}^{t+1} \tag{1}$$

$$\begin{aligned} where:\ w &= \text{inertia weight} \\ c_1 &= \text{cognitive acceleration factor} \\ c_2 &= \text{social acceleration factor} \\ r_1, r_2 &= \text{uniformly distributed random numbers} \end{aligned}$$

4 PROPOSED METHOD

This section describes the steps required to build a rule-based classifier using Particle Swarm Optimization for the rule discovery process. First, we outline the hybrid PSO-ILS algorithm, which outputs the best classification rule according to the proposed fitness function for a given dataset and one binary label class attribute. Then, we discuss two variations of a Covering rule-based classifier that utilize the output of the PSO algorithm.

4.1 PSO-Based Rule Discovery

In this subsection, we explain how the general PSO schema is adapted for finding optimal classification rules. We also discuss the integration of ILS into the main algorithm.

4.1.1 Particle Representation

In PSO, a particle represents a potential solution to the problem. The problem of classification rule discovery involves finding one or more classification rules. Consequently, two separate types of representations—the Michigan approach and the Pittsburgh approach—have been devised.

In the Pittsburgh approach, a particle encodes an entire set of rules. Despite the fact that this seems like a promising representation, evaluating the solutions is quite computationally expensive, and technically, this encoding leads to a difficult rule extraction technique (Telikani et al., 2020). The Michigan approach considers a fraction of a candidate solution, where each particle encodes only a single rule. The advantage of this representation method is that it requires a more plain implementation, and it has been demonstrated that it can extract strong prediction rules and identify rare events. The proposed algorithm will also use the Michigan representation, which is preferable for the task of classification (Alkeshuosh et al., 2017).

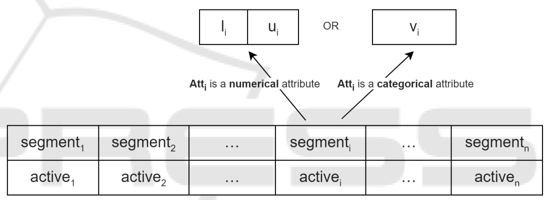

Figure 1: Particle representation for a rule.

A classification rule is expressed as 'IF A THEN C', where A represents the rule antecedent, which is a conjunction of conditions on the input variables, and C is the consequent of the rule, corresponding to the predicted class.

However, only the antecedent of the rule will be encoded in the particle, and we will execute the PSO algorithm separately for each class to generate rules for every possible outcome. This approach is particularly well-suited for tasks involving a small number of classes, as is the case for binary disease detection.

Consider a classification dataset with a set of n input attributes $\{Att_1, Att_2, ..., Att_n\}$ and a predictor class attribute Att_c. To encode the rule's antecedent, the particle will be split into n segments. Each pair of neighboring segments is also considered to be interconnected by an indirect AND operator. As can be seen in Figure 1, each segment "$segment_i$" is associated with a field called "$active_i$", that indicates whether or not that specific variable is used in the rule.

The segment's structure may vary based on the type of variable. We used two additional fields for numerical variables called l_i and u_i, which stand for the lower and upper boundaries of the attribute, and only

Att_1	Att_2		Att_3		Att_4
0	2	4	20.5	40	3
1	0		1		1

Figure 2: Sample encoding of a particle.

one field, v_i, that indicates the value of the attribute for categorical variables.

An example of a rule encoding is given in Figure 2. It defines a rule whose antecedent is $(Att_1 = 0)$ AND $(20.5 < Att_3 < 40)$ AND $(Att_4 = 3)$. Notice that Att_2 is not present in the rule because the *active* bit is set to 0.

4.1.2 Swarm Initialization

The particles are first initialized randomly from a uniform distribution. Still, the rules must be valid, which implies that the values are within the range of the attributes and of the correct types.

There is, however, a high risk of having poor starting points for exploration when using only random rules. Therefore, we additionally set a percent of the initial swarm's particles to some externally provided rules, randomizing only the remaining particles. An inexpensive way to obtain some rules that can perform better than a random rule is to consider the best association rules (AR). Association rules (Kumbhare and Chobe, 2014) are IF/THEN clauses that can be helpful in identifying unknown correlations and interrelations between seemingly unrelated pieces of data. They are expressed as $X => Y$ (IF X THEN Y), where X and Y are some sets of attributes, called itemsets. Association rules are required to satisfy two fundamental user-specified thresholds called minimum support and minimum confidence.

We will consider only association rules with a single variable in the consequent, i.e., the variable class to be predicted. Rules are generated using the Apriori algorithm. Additionally, these rules have been converted to the required particle structure in the initialization phase. These rules should not fill too much of the swarm to still allow the proper diversity for the exploration of the solution space.

4.1.3 Rule Evaluation

Each particle in the swarm is evaluated over the course of each iteration using a fitness function. The fitness should measure the classification performance of the predictions of a rule by using one or more classification-specific metrics. Some of the most commonly used metrics include accuracy, precision, recall, F-score, ROC, etc. (Tharwat, 2020).

Most studies employ a fixed fitness function, which may not be suitable for the medical field. Despite the fact that medical data share a number of common characteristics, different evaluation metrics are more helpful than others, depending on the dataset (Al-Muhaideb and El Bachir Menai, 2013). Thus, because the fitness function is problem-specific, we represent the fitness as a linear combination of four of the previously mentioned metrics. Each metric is multiplied by a coefficient ranging from 0 to 1, allowing for the absence of some metrics from the function.

$$\begin{aligned} Fitness = k_1 * Accuracy + k_2 * Precision + \\ + k_3 * Recall + k_4 * F1, \\ 0 \le k_1, k_2, k_3, k_4 \le 1 \end{aligned} \tag{2}$$

4.1.4 Updating the Particle Velocity and Position

When updating the particles, we have to consider a strategy for each different type of attribute. In a particle, the following types of attributes can be present: numerical, which includes real and integer, and categorical, which also includes binary. However, a conversion to integer values is carried out as part of the preprocessing for categorical variables.

In (Zomorodi-moghadam et al., 2021), a scheme for updating a real-binary hybrid particle encoding is presented. The same scheme was employed in the current study. The velocity and particle update for real and integer attributes is presented in Equation 3.

$$\begin{gathered} V_i^{(t+1)} = w * V_i^t + c_1 * (P_i - X_i^t) + c_2 * (G - X_i^t) \\ X_i^{(t+1)} = X_i^t + V_i^t \end{gathered} \tag{3}$$

After computing the new positions for the real or integer fields within a particle, it is essential to check that they respect the minimum and maximum boundaries of the corresponding attribute. Values that fall below the minimum boundary are replaced with the minimum, and the ones that go over the maximum boundary are updated analogously.

To update binary fields, the formulas are adjusted with logical operators, and the parameters w, c_1, and c_2 are replaced with a random number r ranging from 0 to 1, approximated to the nearest bit. The $\oplus$ operator defines the logical xor. Equation 4 illustrates the derived formulas.

$$\begin{gathered} V_i^{(t+1)} = (r \wedge V_i^t) \vee (r \wedge (P_i \oplus X_i^t)) \vee (r \wedge (G \oplus X_i^t)) \\ X_i^{(t+1)} = X_i^t \vee V_i^t \end{gathered} \tag{4}$$

Since the active fields take binary values, updating them is analogous to updating the binary attribute values.

4.1.5 Refine with Iterated Local Search

Since PSO is a heuristic method, the global best particle obtained at the end of the algorithm, i.e., the best explored solution, is not guaranteed to be the actual global optimum. In order to improve the global best found by the PSO algorithm, we used Iterated Local Search to assist the search when the swarm gets trapped in a local optimum after the last iteration.

Iterated Local Search is a metaheuristic that can search an area around some predetermined solutions rather than the entire space of candidate solutions. The predetermined solutions are usually provided by a local search heuristic (Lourenço et al., 2003).

We define the neighborhood of a particle having the position X_i as comprising all the particles resulted from altering a single field in the position array X_i, but with some restrictions. A single alteration to the solution is done only to one segment of the particle, i.e., to a variable of the rule, and can be one of the following:

1. If the variable was not present in the rule, the *active* field of the segment can be switched to 1.
2. If the variable is present in the rule, the value of the variable will be changed. Any changes to the values should still keep the particle a valid rule.

Notice that a change of a segment is not considered an alteration of the particle if the segment was not active.

The global best solution found by the PSO algorithm is the starting solution for the ILS metaheuristic. Then at each iteration, a local search is performed in the previously described neighbourhood and the acceptance criterion is to only accept better or equal-quality solutions. Before performing a local search in the neighborhood, the current solution is modified through a perturbation procedure to produce a new candidate solution (Jabba, 2021). Since the perturbation modifications should not exactly match the ones in the local search, we use a perturbation of size two, meaning two alterations are applied to a particle on two random segments. The alterations are similar to the ones described above. The algorithm stops when the user-defined number of iterations was reached.

4.1.6 Outline of the Hybrid PSO Algorithm

Algorithm 1 describes the main process of the hybrid PSO-ILS algorithm. The PSO starts with an initial swarm that is randomly generated. If any rules have been provided for initialization, they replace some random rules in the initial swarm. Then, for each iteration, the particles and velocities are updated to search for the rule that has the best fitness.

Algorithm 1: Hybrid PSO-ILS for classification rule discovery.

```
Input  : Train data, PSO parameters,
         Particles for initializations, class c
Output: The global best rule for class c
Initiliaze_random_particles();
if Particles_for_init is not empty then
    partially overwrite random particles with
      Particles_for_init ;
end
Initialize global best and personal bests;
for iteration ← 0 to iteration_no do
    for i ← 0 to swarm_size do
        update velocity V[i] and position X[i]
          of particle i using Equation 3 and 4;
        fitness[i] ← Fitness(particle[i], c);
        if fitness[i] is better than
          p_best_fitness[i] then
            p_best_fitness[i] ← fitness[i];
            p_best[i] ← X[i];
            if fitness[i] is better than
              g_best_fitness then
                g_best_fitness ← fitness[i];
                g_best ← X[i];
            end
        end
    end
    return
     Iterated_Local_Search(global_best)
end
```

Finally, the ILS algorithm is employed in an effort to improve the solution provided by the PSO algorithm. The algorithm iteratively generates new candidate solutions and searches within their neighborhood.

4.2 Rule-Based Classifier

The covering of a rule IF A THEN C is the set of samples that satisfy all the constraints in the antecedent A of the rule. The idea of a sequential covering algorithm, also called a separate-and-conquer method, is to repeatedly learn a single rule and remove all the samples covered by that rule during the training phase (Mangat and Vig, 2014). This step is continued until there is no data left to be removed. Each rule is added to a list in the particular order it was found, obtaining an ordered rule set less likely to contain redundant rules. When testing a new sample, the predicted class is given by the consequent of the first rule that matches the sample. For samples that no rule covers, the predicted class is given a default value.

In this study, a covering rule-based classifier is proposed, and the step of finding a predictive rule is done using the described PSO algorithm, which outputs the best rule for the given data. Because the PSO actually searches only for the best antecedent of a rule, assuming the consequent is known, we devise two different approaches when learning a new rule:

1. Search for the best rule for the majority class. Only one run of PSO is required; the consequent of the class is assumed to be the majority class.
2. Search for the best rule for both classes, and then choose between the two results the one with the higher fitness. This approach requires two runs for learning a rule, one for each class.

5 EXPERIMENTS

5.1 Experimental Setup

5.1.1 Datasets

For evaluating the Covering-PSO algorithm, three different medical datasets meant for binary classification were considered. The Z-Alizadeh Sani dataset, from the UCI machine learning repository, offers information that can be used to determine if a patient has coronary artery disease (CAD). The Pima Indians Diabetes dataset is also originally from the UCI machine learning repository and is intended for the classification of diabetes in females of Pima Indian heritage who are at least 21 years old. The Appendicitis dataset, sourced from the KEEL repository, indicates whether patients have appendicitis or not. More information about the datasets is given in Table 1.

Table 1: Datasets particularities.

Dataset	Instances	Attributes	Attribute types	Classes
CAD	303	56	Real, integer, categ	2
Pima	768	8	Real, integer	2
Append	106	7	Real	2

When it comes to medical diagnosis, the majority of real-world medical databases show some degree of imbalance, with healthy patients typically outnumbering sick ones. The performance of classification algorithms is greatly impacted by this imbalance due to the high degree of misclassification caused by the strong bias towards the majority class (Mienye and Sun, 2021).

The chosen datasets also contain a slight to moderate degree of imbalance to accurately represent real-world circumstances, as illustrated in Table 2. The degree of imbalance in data with two classes can be measured through the imbalance ratio (IR), calculated as the ratio of the number of samples in the majority class to the number of samples in the minority class.

Table 2: Datasets classes distribution.

Dataset	Majority class	Minority class	IR
CAD	216	87	2.48
Pima	500	267	1.87
Append	85	21	4.04

The three datasets undergo minimal preprocessing, which consists of the conversion of categorical data to integers. Additionally, feature selection is done for the CAD dataset to reduce the number of attributes, resulting in only using 14 features out of 56. The algorithm used was Lasso Regression, and features with coefficient 0 were discarded.

5.1.2 Parameter Tuning

PSO is influenced by a number of control parameters, namely the number of particles, acceleration coefficients, inertia weight, and number of iterations. Beside the standard parameters, the number of iterations required for the ILS and the percent of particles in the initial swarm that are set with association rules must also be provided. We observed that the fitness function has a great impact on the results, so the fitness function must be properly selected for each dataset.

The parameters are selected using trial and error by testing 5 to 10 runs of the algorithm with different configurations. The selected values shown in Table 3 also take into account research findings reported in the literature on the impact of the listed parameters.

Table 3: Parameters of the hybrid PSO algorithm.

Parameter	Value
Number of iterations (PSO)	60
Number of particles	120
Inertia (w)	0.8
Cognition factor (c_1)	2
Social factor (c_2)	2
Number of iterations (Local Search)	100
Number of particles initialized with ARs	20-30

The fitness functions selected for each dataset are presented in Table 4 and are formulated according to Equation 2. The ratio of each metric has a direct impact on the resulting rules, and having a flexible fitness function allows for establishing what is more important for that specific dataset and task. A greater ratio for precision may lead to more specific rules, while when only using accuracy, we might obtain rules that are too general. For the Z-Alizadeh Sani (CAD) dataset, better results were obtained when slightly more importance was given to precision, while for the other two datasets, the F1 score was used, replacing the need to include precision and recall.

Table 4: Fitness functions for different datasets.

Dataset	Fitness function
CAD	$0.3*accuracy+0.5*precison+0.2*recall$
Pima	$0.3*accuracy+0.7*f1score$
Append	$0.3*accuracy+0.7*f1score$

5.2 Extracted Rules

The rules mined by the PSO algorithm are not only useful for building a classifier but also on their own, as they provide useful information about attributes. They can reveal primary risk factors or signs of the disease.

Table 5: Rules obtained with the proposed PSO for the Z-Alizadeh Sani dataset.

Rule Antecedent	Rule Consequent
$(30 < Age < 59)$ AND $(Typical\ Chest\ Pain = 0)$ AND $(51 < EF\text{-}TTE < 57)$ AND $(Region\ RWMA = 0)$	Normal
$(30 < Age < 65)$ AND $(Typical\ Chest\ Pain = 0)$ AND $(6 < BUN < 51)$ AND $(Region\ RWMA = 0)$	Normal
$(43 < Age < 86)$ AND $(Typical\ Chest\ Pain = 1)$	CAD
$(38 < Age < 86)$ AND $(144 < Length < 187)$ AND $(Typical\ Chest\ Pain = 1)$	CAD

Table 5 displays some examples of the best individual rules that were found for the Z-Alizadeh Sani dataset. The rules for the *CAD* class, which is for sick patients, show that older age and typical chest pain are risk factors for coronary artery disease.

5.3 Covering-PSO Algorithm Results

The proposed classifier is trained and tested using 10-fold cross-validation and compared with other rule-based machine learning algorithms. The accuracy is used to evaluate the classification's performance. For the proposed algorithm, it is computed as an average over 10 runs.

First, we conducted some independent experiments with the following classifiers: C4.5, CBA, and Random Forest, with the same minimally processed datasets, as more complex preprocessing was not the main focus of the current study.

Additionally, we contrast the two variants of the covering classifier: the first searches solely for the best rules for the majority class, whereas the second searches for rules for both classes. We also added the average accuracy of the optimum rule obtained by the hybrid PSO algorithm for each class, i.e., a classifier composed only of a single rule. The results given in Table 6 indicate that the Covering-PSO algorithm can potentially compete with other common rule-based classifiers. Additionally, the algorithm shows better performance when searching for rules exclusively for the majority class. This variant also offers the advantage of reduced computational time, as the best rule is determined with a single run rather than two.

Table 6: Performance of different rule-based classifiers.

Model \ Dataset	CAD	Pima	Append
C4.5	79.19%	73.82%	85.84%
Random Forest	84.81%	76.43%	86.72%
CBA	77.55%	64.84%	73.27%
Hybrid-PSO individual rules	79.6%	70.95%	80.4%
Covering-PSO (only majority class rule)	85.11%	74.72%	89.12%
Covering-PSO (best rule out of both classes)	84.67%	74.63%	84.9%

We further analyze the results by looking at other studies, noting that different performance validation techniques and preprocessing, such as balancing the classes, complex feature selection, or scaling, are factors that directly impact the accuracy. Although there are other machine learning techniques that are more accurate, the proposed algorithm performs well when compared to other PSO and rule-based techniques, as demonstrated in Table 7.

Table 7: Comparison of the proposed algorithm with other results available in literature.

Dataset	Method	Accuracy
CAD	Hybrid PSO (Zomorodi-moghadam et al., 2021)	84.25%
	Fuzzy rule-based system (Singh and Singh, 2021)	89%
	Covering-PSO	85.11%
Pima	Dynamic PSO (DP-AC) (Mangat and Vig, 2014)	74.11%
	Adaptive rule classifier (Farid et al., 2016)	75.65%
	CT+ASA+NB (Changpetch et al., 2021)	81.25%
	Covering-PSO	74.72%
Append	Adaptive rule classifier (Farid et al., 2016)	87.73%
	CT+ASA+NB (Changpetch et al., 2021)	95.28%
	Covering-PSO	89.12%

5.4 Ablation Study

To demonstrate the contributions of each of the proposed additions to the PSO algorithm, we carried out an ablation study for the CAD dataset. The average accuracy and the standard deviation over 10 runs are reported in Table 8. We notice that both the use of association rules and the ILS hybridization help stabilize the algorithm, which also leads to overall better accuracy.

Table 8: Performance of Covering-PSO algorithm with and without proposed improvements for the CAD dataset.

Variant	Average accuracy	Std. dev.
Without AR, Without ILS	81.71%	1.44
Without AR, With ILS	82.72%	1.10
With AR, Without ILS	83.31%	0.945
With AR, With ILS	85.11%	0.56

The Mann-Whitney U test with a significance level of 0.05 is employed to determine if there is a statistically significant difference in performance between the proposed algorithm (with association rules and with ILS) and the other three variants. The analysis shows that the proposed method brings improvements over the other variants.

5.5 Interpretability Analysis

Predictive accuracy is the sole criterion for evaluation classification models in a significant number of studies. However, in medical applications, the accuracy is insufficient for users to trust in the predictions if they are not supported by an understandable explanation, so the model's interpretability is equally crucial (Freitas, 2014).

The average rule size and the average number of rules in a model are two standard metrics to assess a rule classifier's interpretability. These two metrics do not account for the number of rules that are actually needed to make class predictions; hence, they might not be sufficient to demonstrate how comprehensible a model is. Therefore, a third metric called prediction-explanation size is employed. It is defined as the average number of attributes that the model evaluates to be able to predict the class value of an instance, where the average is calculated over all instances in the test set (Otero and Freitas, 2016). For example, if a model contains four ordered rules and a match is found at the third rule for some sample, the prediction-explanation size is the sum of the first three rule sizes. The third rule directly impacts the prediction, while the first two rules are indirectly involved.

Table 9 shows the values of the discussed metrics for each dataset. The model generally contains up to

Table 9: Interpretability measures for the PSO-Covering classifier.

Dataset / Metric	CAD	Pima	Append
Average number of rules	4.5	3.2	2.75
Average rule size	2.1	1.933	1.84
Prediction-explanation size	3.53	2.58	2.69

five rules, and the average number of attributes used in a rule is around two. Further, when looking at the prediction-explanation size, a prediction is given after evaluating a small number of attributes.

For the Pima Indian Diabetes dataset, additional results can be found in (Otero and Freitas, 2016), where the described Ant-Colony algorithm, specifically designed for better interpretability, has a prediction-explanation size of 2.06. This value is close to our result of 2.58. However, the predictive accuracy was slightly lower, at 74.55%, compared to our 74.72%. Additionally, the prediction-explanation size reported for some standard rule-based methods is much higher.

Overall, the model's decision-making is done with a very limited number of short rules, leading to a simple textual representation. This proves the interpretability of the classifier and also shows some insight related to its accuracy. It is a challenging task for a small linear model to compete with complex machine learning methods when making predictions.

6 CONCLUSIONS

In this paper, a hybridized version of PSO with ILS is proposed for the purpose of discovering classification rules. Additionally, the PSO algorithm incorporates a hybrid swarm initialization method, as it combines both random rules and association rules. The discovered classification rules are subsequently employed to construct an ordered rule-based classifier, obtained by iteratively removing samples covered by the optimum rule. The classifier is tested on a few medical benchmark datasets. The results indicate that the proposed algorithm is better in terms of accuracy than certain rule-based classifiers and PSO-based techniques. Furthermore, the algorithm demonstrates increased stability with the introduced adjustments. The predictive performance, while potentially not reaching the levels of some machine learning and deep learning methods, is compensated by a high degree of interpretability.

For future work, our aim is to enhance the algorithm so that it can compete against other complex related methods. The rule discovery process could be improved by taking the interactions between the rules into account. Particle competitiveness and variety could be increased by incorporating concepts such as dynamic neighborhoods and repulsive forces. The impact of more optimal feature selection can also be studied more closely. Moreover, we intend to optimize and test the algorithm on more large datasets.

REFERENCES

Abdulkadium, A. M., Shekan, R. A. A., and Hussain, H. A. (2022). Application of data mining and knowledge discovery in medical databases. *Webology*, 19(1):4912–4924.

Al-behadili, H. N. K., Ku-Mahamud, K., and Sagban, R. (2020). Hybrid ant colony optimization and iterated local search for rules-based classification. *J. Theor. Appl. Inf. Technol*, 98(04):657–671.

Al-Hashem, M. A., Alqudah, A. M., and Qananwah, Q. (2021). Performance evaluation of different machine learning classification algorithms for disease diagnosis. *International Journal of E-Health and Medical Communications (IJEHMC)*, 12(6):1–28.

Al-Muhaideb, S. and El Bachir Menai, M. (2013). Hybrid metaheuristics for medical data classification. *Hybrid metaheuristics*, pages 187–217.

Alaiad, A., Najadat, H., Mohsen, B., and Balhaf, K. (2020). Classification and association rule mining technique for predicting chronic kidney disease. *Journal of Information & Knowledge Management*, 19(01):2040015.

Alkeshuosh, A. H., Moghadam, M. Z., Al Mansoori, I., and Abdar, M. (2017). Using pso algorithm for producing best rules in diagnosis of heart disease. In *2017 international conference on computer and applications (ICCA)*, pages 306–311. IEEE.

Changpetch, P., Pitpeng, A., Hiriote, S., and Yuangyai, C. (2021). Integrating data mining techniques for naïve bayes classification: Applications to medical datasets. *Computation*, 9(9):99.

Cristani, M., Olvieri, F., Workneh, T. C., Pasetto, L., and Tomazzoli, C. (2022). Classification rules explain machine learning. In *ICAART (3)*, pages 897–904.

Farid, D. M., Al-Mamun, M. A., Manderick, B., and Nowe, A. (2016). An adaptive rule-based classifier for mining big biological data. *Expert Systems with Applications*, 64:305–316.

Freitas, A. A. (2014). Comprehensible classification models: a position paper. *ACM SIGKDD explorations newsletter*, 15(1):1–10.

Fürnkranz, J. (1999). Separate-and-conquer rule learning. *Artificial Intelligence Review*, 13:3–54.

Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. *Archives of computational methods in engineering*, 29(5):2531–2561.

Holden, N. and Freitas, A. A. (2008). A hybrid pso/aco algorithm for discovering classification rules in data mining. *Journal of Artificial evolution and Applications*, 2008.

Hossain, S. K. M., Ema, S. A., Sohn, H., et al. (2022). Rule-based classification based on ant colony optimization: a comprehensive review. *Applied Computational Intelligence and Soft Computing*, 2022.

Houssein, E. H., Gad, A. G., Hussain, K., and Suganthan, P. N. (2021). Major advances in particle swarm optimization: theory, analysis, and application. *Swarm and Evolutionary Computation*, 63:100868.

Jabba, A. M. (2021). Rule induction with iterated local search. *International Journal of Intelligent Engineering & Systems*, 14(4).

Kononenko, I. (2001). Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine*, 23(1):89–109.

Kumbhare, T. A. and Chobe, S. V. (2014). An overview of association rule mining algorithms. *International Journal of Computer Science and Information Technologies*, 5(1):927–930.

Lourenço, H. R., Martin, O. C., and Stützle, T. (2003). Iterated local search. In *Handbook of metaheuristics*, pages 320–353. Springer.

Mangat, V. and Vig, R. (2014). Dynamic pso-based associative classifier for medical datasets. *IETE Technical Review*, 31(4):258–265.

Mienye, I. D. and Sun, Y. (2021). Performance analysis of cost-sensitive learning methods with application to imbalanced medical data. *Informatics in Medicine Unlocked*, 25:100690.

Otero, F. E. and Freitas, A. A. (2016). Improving the interpretability of classification rules discovered by an ant colony algorithm: extended results. *Evolutionary computation*, 24(3):385–409.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215.

Singh, N. and Singh, P. (2021). Medical diagnosis of coronary artery disease using fuzzy rule-based classification approach. In *Advances in Biomedical Engineering and Technology: Select Proceedings of ICBEST 2018*, pages 321–330. Springer.

Telikani, A., Gandomi, A. H., and Shahbahrami, A. (2020). A survey of evolutionary computation for association rule mining. *Information Sciences*, 524:318–352.

Thabtah, F. and Peebles, D. (2020). A new machine learning model based on induction of rules for autism detection. *Health informatics journal*, 26(1):264–286.

Tharwat, A. (2020). Classification assessment methods. *Applied computing and informatics*, 17(1):168–192.

Yoo, I., Alafaireet, P., Marinov, M., Pena-Hernandez, K., Gopidi, R., Chang, J.-F., and Hua, L. (2012). Data mining in healthcare and biomedicine: a survey of the literature. *Journal of medical systems*, 36:2431–2448.

Zomorodi-moghadam, M., Abdar, M., Davarzani, Z., Zhou, X., Pławiak, P., and Acharya, U. R. (2021). Hybrid particle swarm optimization for rule discovery in the diagnosis of coronary artery disease. *Expert Systems*, 38(1):e12485.