

Stable Feature Selection for Gene Expression using Enhanced Binary Particle Swarm Optimization

Hassen Dhrif¹ and Stefan Wuchty

Computer Science, University of Miami, Coral Gables, FL, U.S.A.


Keywords: Feature Selection, Stability, Scalability, Particle Swarm Optimization, Evolutionary Computation, Gene Discovery.

Abstract: Feature subset selection (FSS) is an intractable optimization problem in high-dimensional gene expression datasets, leading to an explosion of local minima. While binary variants of particle swarm optimization (BPSO) have been applied to solve the FSS problem, increasing dimensionality of the feature space pose additional challenges to these techniques impairing their ability to select most relevant feature subsets in the massive presence of uninformative features. Most FSS optimization techniques focus on maximizing classification performance while minimizing subset size but usually fail to account for solution stability or feature relevance in their optimization process. In particular, stability in FSS is interpreted differently compared to PSO. Although a large volume of published studies on each stability issue separately exists, wrapper models that tackle both stability problems at the same time are still missing. Specifically, we introduce a novel approach COMBPSO (COMBINatorial PSO) that features a novel fitness function, integrating feature relevance and solution stability measures with classification performance and subset size as well as PSO adaptations to enhance the algorithm's convergence abilities. Applying our approach to real disease-specific gene expression data, we found that COMBPSO has similar classification performance compared to BPSO, but provides reliable classification with considerably smaller and more stable gene subsets.

1 INTRODUCTION

Gene expression profiles have long been used to discover small numbers of features (biomarkers) that are important for patient stratification, drug discovery and the development of personalized medicine strategies. However, genes that govern biological processes are usually co-expressed, aggravating the differentiation between features that are (ir)relevant for the corresponding classification task (Perthame et al., 2016). Therefore, the identification of independent genes (features) whose expression patterns point to a meaningful phenotype as well as the scaling to high dimensional search spaces is a challenge for many feature selection methods. Particle swarm optimization (PSO) approaches (Eberhart and Kennedy, 1995), its binary, single objective variant BPSO (Kennedy and Eberhart, 1997) and its multi-objective variant MOPSO (Zhang et al., 2017; Yong et al., 2016; Xue et al., 2013; Chandra Sekhara Rao Annavarapu and Banka, 2016) are evolutionary computation techniques, that have been combined with different clas-

sification methods to select informative markers from gene expression data (Han et al., 2017; Han et al., 2014). In particular, such approaches aim to maximize classification performance, while keeping the size of the gene subset as small as possible. As a consequence, such approaches are supposed to find all genes that are strongly relevant for the classification process and ignore all irrelevant ones. While the above-mentioned PSO methods select subsets of predictive genes that allow reliable sample classification, the size of the obtained gene subsets is usually large. Furthermore, subsets tend to be highly variant, limiting the stability of obtained results, a necessary condition to scale to high dimensional feature space. Such characteristics are putatively rooted in the tendency of PSO algorithms to usually lose the diversity of the swarm, leading to premature convergence and leaving many areas of the search space unexplored. While many solutions were suggested (Babu et al., 2014) to tackle these drawback, previous work left the integration of the relevance and stability of selected feature subset and the stability of the PSO algorithm untouched.

¹  <https://orcid.org/0000-0002-3842-1762>

Here, we introduce an extension of the BPSO variant proposed in (Dhrif et al., 2019b; Dhrif, 2019) by introducing multiple algorithm modifications, that maximizes (i) the performance of sample classification, (ii) minimizes the underlying set size of informative (relevant) genes, and (iii) maintains stability of the size of gene subsets in the massive presence of uninformative (*i.e.* irrelevant) genes. As a consequence, we expect that our algorithm scales with datasets that have tens of thousands of genes, while selecting relatively small subsets of informative (*i.e.* relevant) genes. In more detail, we introduce (i) a novel multi-objective optimization fitness function integration not only classification performance and subset size but also the relevance of features to the class label and the stability of the selected features subset (subsection 3.1), (ii) an encoding technique that enhances the diversity of the swarm to solve binary optimization problems (subsection 3.2), (iii) a novel adaptive function that governs the inertia weight and the acceleration coefficients, allowing the swarm to explore and exploit the search space more thoroughly (subsection 3.3), (iv) a dynamic population strategy to faster discover new global best solutions (gbest) and a turbulence operator, enabling the swarm to escape a local optimum (subsection 3.3), (v) and asymmetric position boundaries that control the divergence of the swarm and increase the probability of sampling candidate solutions with smallest number of selected genes (subsection 3.4).

Applying our approach on real disease-specific gene expression data, we observe that COMBPSO has similar classification performance compared to BPSO through considerably smaller and more stable gene subsets.

2 BACKGROUND

2.1 Stability of Feature Subset Selection

The FSS problem revolves around the minimization of selected feature subsets, while optimizing a given performance measure. Generally, the solution to a FSS problem features three steps (Kumar and Minz, 2014). In the *Subset discovery* step, approaches determine a subset of features that are subsequently evaluated. While many strategies to select feature subsets have been proposed, we focus on the Particle Swarm Optimization (PSO) algorithm. In the *Subset evaluation* step, the performance of feature subsets is tested according to given evaluation criteria. While subsets are usually evaluated through diverse machine learning procedures, we focus on supervised

learning only (*i.e.* classification), where a-priori class labels are known. Furthermore, we evaluate the relevance of features for the classification process (Kumar and Minz, 2014) through metrics that consider consistency, dependency, distance and information of feature subsets. In (John et al., 1994), features were classified as *strongly relevant*, *weakly relevant*, and *irrelevant*. As a consequence, an optimal subset must include all strongly relevant features, may account for some weakly relevant ones, but no irrelevant features. *Subset discovery* and subsequent *subset evaluation* are repeated until a stopping condition such as a predefined maximum number of iterations or a minimum classification error rate is met. In the *Result validation* step the optimal feature subset is validated using n-fold cross-validation.

Stability, defined as sensitivity of a FSS algorithm to a small perturbation in the training data is as important as high classification performance when evaluating FSS performance (Khair and Dhanalakshmi, 2019). Strong correlation between features frequently lead to multiple equally performing feature subsets, reducing the stability of traditional FSS methods and our confidence in selected feature subsets.

2.2 Stability of Particle Swarm Optimization

PSO, a simple mathematical model developed by Kennedy and Eberhart in 1995 (Eberhart and Kennedy, 1995), is a meta-heuristic algorithm that uses a streamlined model of social conduct to solve an optimization problem in a cooperative framework. In particular, PSO has been combined with different classification methods to select informative feature subsets. However, PSO algorithms are limited in their abilities to converge (Bonyadi and Michalewicz, 2017). In particular, the *convergence to a point* problem the velocity vector of particles grows to infinity for some values of the acceleration and inertia coefficients, an issue that is also known as *swarm explosion*. *Stability analysis* focuses on the particles' behavior to find the reasons why the sequence of generated solutions does not converge. In particular, *First-order* stability analysis investigates the expectation of the position of particles to ensure that this expectation converges. *Second-order* stability analysis focuses on the variance of the particle's position to ensure convergence to zero. In (Cleghorn and Engelbrecht, 2014), first-order analysis has been conducted where it was assumed that the personal best and global best vectors can occupy an arbitrarily large finite number of unique locations in the search space. In (Poli, 2009), second-order analysis showed that particles do

not stop moving (convergence of the variance of the particles' positions) until their personal bests coincide with the global best of the swarm. Furthermore, PSO is not locally convergent (Bonyadi and Michalewicz, 2017). To solve local convergence issues, a mutation operator replaces global/personal best vectors by a randomly selected point around the global best vector (Bonyadi and Michalewicz, 2014; Van den Bergh and Engelbrecht, 2010). Furthermore, regeneration of velocity vectors prevent particles from stagnating, solutions that considerably slow the search process. While the stability features of standard PSO have been thoroughly investigated convergence behavior of BPSO remains unknown.

3 PROPOSED METHODS

3.1 Objective Function

Objective functions to solve a FSS problem usually feature two conflicting objectives, maximizing the classification performance and minimizing the size of the selected feature subset. However, discarding the features which are highly associated with the response variable is one of the main causes of instability. Therefore, we introduce a novel fitness function that integrates feature relevance, subset stability and classification performance in a weighted-sum multi-objective optimization model.

To eliminate noise we introduce a measure of non-linear correlation between features and response variables. In particular, we adopted the Randomized Dependence Coefficient (RDC) (Lopez-Paz et al., 2013) that we implemented in (Dhrif et al., 2019a). RDC is an empirical estimator of the Hirschfeld-Gebelein-Rényi (HGR) maximum correlation coefficient that measures non-linear dependencies between random variables $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$, defined as

$$RDC(X, Y) = \max_{\alpha \in \mathbb{R}^k, \beta \in \mathbb{R}^l} \bar{\rho}(\alpha^T \Phi_X, \beta^T \Psi_Y). \quad (1)$$

Given a dataset of m samples with n features and q response variables, the individual association between any feature $f \in \mathbb{R}^{m \times n}$ and the class $C \in \mathbb{R}^{m \times q}$ is defined by $RDC(f, C) \in [0, 1]$ where 1 indicates that the feature f is strongly relevant. Indicating the relevance of a subset S we average the RDC score over all features by

$$\mathcal{R}(S) = \frac{1}{|S|} \sum_{f \in S} RDC(f, C), \quad (2)$$

suggesting that $\mathcal{R} = 1$ when all features in S are strongly relevant.

Accounting for subset stability in the FSS evaluation step, we calculate the amount of overlap between subsets of selected features (Mohammadi et al., 2016). As multiple iterations of the algorithm provide differing feature subsets, we define the consistency $\mathcal{C}(f)$ of a feature f as

$$\mathcal{C}(f) = \frac{F_f - F_{min}}{F_{max} - F_{min}}, \quad (3)$$

where F_f is the number of occurrences of feature f in subsets obtained at iteration t , $F_{min} = 1$ is the global minimum number of occurrences and F_{max} is the global maximum number of occurrences of any feature at iteration t . Furthermore, we defined the average consistency of the whole subset S by

$$\mathcal{C}(S) = \frac{1}{|S|} \sum_{f \in S} \mathcal{C}(f). \quad (4)$$

$\mathcal{C}(S)$ tends toward 1 if features appear repeatedly in obtained feature subsets, indicating high stability.

Measuring classification performance of a feature subset S , we considered recall defined as $\mathcal{P}(S) = \frac{tp}{tp+fn}$, where tp and fn refer to the number of true positive and false negative predictions. Recall is considered a measure of a classifiers completeness, where a low recall rate points to the presence of many false negatives.

Based on the relevance $\mathcal{R}(S)$, consistency $\mathcal{C}(S)$ and the classification performance $\mathcal{P}(S)$ that account for the size of the feature subset S we define the fitness function as

$$\begin{aligned} \max \quad \mathcal{F}(S) &= \alpha_1 \mathcal{P}(S) + \alpha_2 \mathcal{R}(S) + \alpha_3 \mathcal{C}(S) \\ \text{subject to} \quad &\mathcal{P}(S) \geq \mathcal{P}(D), \end{aligned} \quad (5)$$

where D is the set of all features, and $\alpha_1 + \alpha_2 + \alpha_3 = 1$ are weight factors, balancing classification performance, feature relevance and subset stability.

3.2 Improving Exploration and Exploitation Capabilities

As our objective is the selection of a limited set of features, we consider each particle as a binary vector where the presence (absence) of a feature is represented by a binary digit. BPSO handles this type of representation by mapping particle positions to a binary space where a particle moves by flipping its bits. However, such a movement does not provide an intuitive notion of velocity, direction and momentum in a binary feature space. While BPSO underperforms compared to PSO, Saberi et al. (Mohamad et al., 2011) indicated that modelling velocity as a

sigmoid function reduces the number of attributes to roughly half the total number of features. As BPSO suffers from poor scaling behavior Lee et al. (Lee et al., 2008) introduced a velocity update that is based on a binary encoding mechanism of the underlying position. Here, we propose a novel encoding scheme (Fig. 1) that maps particle positions to probabilities, sustaining search in continuous space. In contrast to (Lee et al., 2008), velocity vector \vec{v} and position vector \vec{x} are represented in continuous form by

$$\begin{aligned} \vec{v}_i^{t+1} &= \omega \vec{v}_i^t + r_1 c_1 (\vec{p}_i - \vec{x}_i^t) + r_2 c_2 (\vec{g} - \vec{x}_i^t) \\ \vec{x}_i^{t+1} &= \vec{x}_i^t + \vec{v}_i^t. \end{aligned} \quad (6)$$

where i indicates the i^{th} particle, and t indicates the t^{th} iteration. Furthermore, we utilize a binary vector \vec{b} that maps the continuous space position to binary digits by

$$b_{ij} = \begin{cases} 1, & \text{if } \text{rand}() < S(x_{ij}) \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

where

$$S(x_{ij}) = \frac{1}{(1 + e^{-x_{ij}})}, \quad (8)$$

indicating that feature (i.e. gene) j in particle i is accounted for in a feature subset if $b_{ij} = 1$.

Furthermore, dynamics of the particles in PSO must be carefully controlled to avoid premature convergence in the early stages of the search and enhance convergence to the global optimum solution during later stages of the search. Specifically, a high value of the inertia component, $\omega \vec{v}_i(t)$ and cognitive component, $r_1 c_1 (\vec{p}_i - \vec{x}_i(t))$ in Eq. (6), where \vec{p}_i is the particle specific best solution encountered so far will result in particles *explore* the search space. In turn, a high value of the social component, $r_2 c_2 (\vec{g} - \vec{x}_i(t))$, where \vec{g} is the global best solution, rushes particles prematurely toward a local optimum. In the early stages of a population-based optimization process, particles are supposed to explore the search space thoroughly, without being limited to local optima. In later stages, particles are supposed to converge toward the global optimum. Bansal et al. (Bansal et al., 2011) compared multiple inertia weight functions for parameters ω , c_1 and c_2 , concluding that, despite its popularity, a linear time-variant function does not secure best performance. Here, we propose to model coefficients c_1 , c_2 and ω as sigmoid functions that allow fast transitions between search phases and extends the particles time in the exploration and exploitation phase by

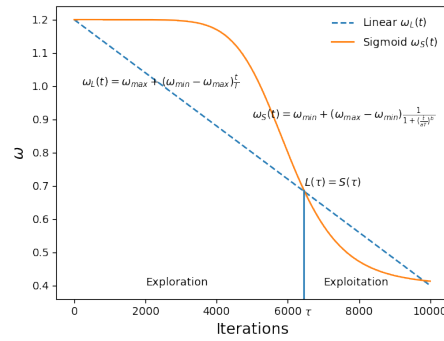


Figure 1: Example of inertia weight ω as a sigmoid function. Instead of a linear function COMBPSO applies a sigmoid function to establish inertia weights, where $\omega_{min} = 0.6$, $\omega_{max} = 0.9$, $a = 0.5$ and $b = 4$. Compared to a linearly decreasing function, our function maintains longer exploration and exploitation phases.

$$\begin{aligned} \omega &= \omega_{min} + (\omega_{max} - \omega_{min}) \frac{1}{1 + (\frac{t}{aT})^b} \\ c_1 &= c_{min} + (c_{max} - c_{min}) \frac{1}{1 + (\frac{t}{aT})^b} \\ c_2 &= c_{max} + (c_{min} - c_{max}) \frac{1}{1 + (\frac{t}{aT})^b}, \end{aligned} \quad (9)$$

where t is the iteration number, and T is the maximum number of iterations. Such a function is shown in Fig. 1 where a governs the transition point, while b determines the length of the exploration and exploitation phase of the particles. Compared to a linearly decreasing function, our proposed sigmoid function maintains longer exploration and exploitation phases and avoids premature convergence of the swarm.

3.3 Improving Convergence Rate and Avoiding Premature Convergence

To solve the local convergence problem in PSO, we introduce a dynamic population strategy. While popular approaches update global best solutions in each step, our new strategy uses a heap data structure to heapify all the previously identified best positions. Each time a new global best position is found, the one being replaced is pushed onto the heap. Then, every time a particle's best position stagnates, the heap is checked for better solution which is eventually popped from the heap.

Furthermore, avoiding premature convergence, a turbulence operator re-initializes the velocity of a fraction $\gamma \in [0, 1]$ of particles, if the global best solution was not updated after θ iterations.

3.4 Reducing the Size of Gene Dubsets

Engelbrecht (Engelbrecht, 2012) indicated that particles tend to leave the boundaries of the search space irrespective of the initialization approach. As such characteristics result in wasted search efforts, particles therefore should be controlled by boundary constraints. However, the choice of the boundary values of x_{min} and x_{max} is critical, affecting the balance between exploration and exploitation and the size of the generated subsets. If x_{max} is too large, many genes that are irrelevant for the underlying classification task will be selected. In turn, some critical genes will be missed in the selection process if x_{min} is too small. While a vast majority of approaches adopts symmetric boundaries, i.e. $x_{max} = -x_{min}$, we introduce an asymmetric velocity boundary coefficient λ by

$$v_{max} = -\lambda v_{min}, \lambda \in [0, 1]. \quad (10)$$

As a consequence, an elevated value of λ increases the probability to obtain additional genes.

3.5 Design of COMBPSO

As outlined in Algorithm 1, COMBPSO first initializes the particle population and subsequently solves the optimization objective. Given a dataset of gene expression profiles, the algorithm searches for the most stable subset of genes with the highest prediction performance. The identified subset S_i is represented by a binary vector \vec{b}_i where each element points to a gene, such that $S_i = 1$ when gene i is selected, and 0 otherwise. Iteratively, velocities and positions of particles are updated according to Eqs. 6 and 7. In each step, the fitness function \mathcal{F} is evaluated while the personal best solution p_i of each particle and global best solution g are updated accordingly. The search and evaluation process keeps iterating until a maximum number of iterations, T_{max} , is reached. After R independent runs (Alg. 1 Line 38) the final result is the most performing subset S^* out of R subsets thus obtained.

4 EXPERIMENTAL RESULTS

4.1 Experimental Datasets

To verify the effectiveness and efficiency of the proposed method for gene selection, we considered public disease specific gene expression datasets (Table II). The *Leukemia* (Armstrong et al., 2001) data set contains 28 Acute Myeloid Leukemia (AML), 24 Acute Lymphoblastic Leukemia (ALL) and 20

Algorithm 1: The COMBPSO algorithm.

```

1: procedure COMBPSO
2:   Initialize swarm  $sw$ 
3:   for  $t \leftarrow 1, T_{max}$  do
4:      $\omega \leftarrow \omega_{min} + (\omega_{max} - \omega_{min}) \frac{1}{1 + (\frac{t}{T_{max}})^b}$ 
5:      $c_1 \leftarrow c_{min} + (c_{max} - c_{min}) \frac{1}{1 + (\frac{t}{T_{max}})^b}$ 
6:      $c_2 \leftarrow c_{max} + (c_{min} - c_{max}) \frac{1}{1 + (\frac{t}{T_{max}})^b}$ 
7:     for  $i \leftarrow 1, |sw|$  do  $\triangleright |sw|$  swarm size
8:        $v_i \leftarrow wv_i + c_1 r_1 (p_i - x_i) + c_2 r_2 (g - x_i)$ 
9:       Clip  $v_i$  to velocity boundaries
10:       $x_i \leftarrow x_i + v_i$ 
11:      Clip  $x_i$  to position boundaries
12:       $b_i \leftarrow \text{sigmoid}(x_i)$   $\triangleright b_i$  is feature subset  $i$ 
13:       $F_i \leftarrow \alpha_1 \mathcal{R}(b_i) + \alpha_2 \mathcal{C}(b_i) + \alpha_3 \mathcal{P}(b_i)$ 
14:      if  $F_i > p_i$  then
15:        Push  $p_i$  into heap
16:         $p_i \leftarrow F_i$ 
17:      else
18:         $p_i \leftarrow \text{Pop from heap}$ 
19:      end if
20:      if  $F_i > g$  then
21:        Push  $g$  into heap
22:         $g \leftarrow F_i$ 
23:      else
24:        if  $g$  stagnates then
25:          Partial velocities reinitialized
26:        end if
27:      end if
28:    end for
29:  end for
30:   $S \leftarrow g_{best}$ 
31:  return  $S$ 
32: end procedure

33: procedure MAIN
34:   Load dataset  $D$  into  $X, y$ 
35:   for all  $f \in D$  do
36:      $R(f) \leftarrow RDC(f, y)$ 
37:   end for
38:   for  $k \leftarrow 1, R$  do
39:      $S_k \leftarrow \text{COMBPSO}(D)$ 
40:   end for
41:    $S^* \leftarrow \text{Best of all } S_k$ 
42:   return  $S^*$ 
43: end procedure

```

Mixed-Lineage Leukemia (MLL) samples, referring to expression profiles of 11,225 human genes. The *Prostate Tumor* (Singh et al., 2002) data set has 52 tumor and 50 non-disease control samples, each consisting of expression profiles of 10,509 human genes. The *DLBCL* data (Shipp et al., 2002) contains 58 patient samples with Diffuse Large B-Cell Lymphomas (DLBCL) and 14 patient samples with Follicular Lymphomas where each sample has 5,469 human genes.

Table 1: Characteristics of Cancer data sets.

	#samples	#features	#classes
Leukemia	72	11,225	3
Prostate	102	10,509	2
Lymphoma	72	5,469	2

Table 2: Hyper parameters used in the experimental set-up. Symbol ω is the inertia weight, c_1 and c_2 are the velocity coefficients, and λ is the velocity boundary coefficient.

Parameters	BPSO		COMBPSO	
	MIN	MAX	MIN	MAX
ω	0.4	0.9	0.4	0.9
c_1, c_2 (a, b) in Eq. 9	2.05		1.7	2.1 (0.6, 8)
velocity λ in Eq. 10 (θ, γ) in Subsec.3.3	-6.0	6.0	-6.0	0.25 1/32 (5, 20%)
α_1	0.8		0.8	
α_2	0.1		0.1	
α_3	0.1		0.1	
swarm size	100		100	
# iterations	300		300	

4.2 Experimental Setup

The choices of appropriate values of hyperparameters for metaheuristic algorithms have been strongly debated (Ye, 2017; Rezaee Jordehi and Jasni, 2013). Here, simulations are carried out with numerical benchmarks to find the best range of values. Given the dynamic nature of c_1 and c_2 as introduced in subsection 3.3, we allow c_1, c_2 to vary between $c_{min} = 1.7$ and $c_{max} = 2.1$ while the transition coefficients are set to $a = 0.6$ and $b = 8$. Furthermore, we introduce an asymmetric boundaries coefficient, as defined in Eq. 10, that we empirically set to $\lambda = 1/32$. To control premature convergence by avoiding stagnation of the swarm, we introduce both a stagnation coefficient θ , representing the number of iterations the globally best solution, $gbest$, did not change before firing the turbulence operator. Moreover, the turbulence coefficient $\gamma, \gamma \in [0, 1]$, indicates the fraction of particles in the swarm that reset their velocities (subsection 3.3). These two coefficients are empirically set to $\theta = 5$ and $\gamma = 0.2$, respectively. Furthermore, swarm size impacts the performance of PSO as a smaller swarm leads to particles trapped in local optima while a larger swarm slows the performance of the algorithm. In Eq. 5 we set $\alpha_1 = 0.8$, $\alpha_2 = 0.1$, and $\alpha_3 = 0.1$. Finally, we set the swarm size to 100 particles while the number of iterations is set to 300. All parameters are presented in Table 2.

We use a wrapper approach, requiring a machine learning estimator to evaluate the classification

performance of the selected features. Here, we use Random Forest (RF), that shows excellent performance when most predictive variables are noisy, and when the number of variables is much larger than the number of observations. Furthermore, RFs can handle problems with more than two classes and returns measures of variable importance (Breiman, 2001).

During the search process, we randomly sample 70% of instances as the training set and 30% as the test set. A 10-fold cross-validation is employed to evaluate the classification performance of the selected feature subset on the training set, while the selected features are evaluated on the test set to obtain testing classification performance.

4.3 Results

The performance of COMBPSO is examined by considering three cancer specific gene expression sets. In particular, we determined the performance of classification of COMBPSO and BPSO averaging over 10 independent executions (1,000 iterations each) for each disease specific dataset individually. Considering all genes in the underlying data sets, we obtained a 84.88% performance in the Leukemia dataset, while we observed classification performance of 81.36% and 84.58% in the Prostate tumor and Lymphoma datasets. In Table 3, we observe that both BPSO and COMBPSO provide similar classification performance, outperforming the all-gene classification benchmark. Compared to BPSO, however, COMBPSO provides significantly smaller gene subsets that allow a reliable classification.

5 CONCLUSION

Combining stability of FSS with stability of PSO and suggesting solutions to tackle both issues within a wrapper model are the main contributions of our work. Integrating feature relevance, we introduced a non linear correlation measure between features and response variables. Accounting for feature subset stability, we integrated a consistency measure as part of the fitness function. Enhancing PSO stability, we introduced a variant of BPSO, called COMBPSO, that allowed us to find feature subsets that boosted classification performance when implemented on datasets with tens of thousands of features. In particular, we improved PSO's stability and scalability characteristics by introducing (i) a new encoding scheme in the continuous space, (ii) fast varying inertia weight and acceleration coefficients as well as

Table 3: Performance of COMBPSO and BPSO obtained with disease specific gene expression datasets. First column R: represents run number. G in columns 2,4,6,8,10,12 represent the number of genes selected. R.(%) in columns 3,5,7,9,11,13 represents classification recall. μ, σ represent mean and standard deviation. Bold typeface represents the best average values of performance (Recall) and size of gene subsets.

R	Leukemia				Prostate				Lymphoma			
	COMBPSO		BPSO		COMBPSO		BPSO		COMBPSO		BPSO	
	G	A.(%)	G	A.(%)	G	A.(%)	G	A.(%)	G	A.(%)	G	A.(%)
1	15	97.82	198	95.65	8	92.09	169	92.09	9	94.25	91	93.75
2	16	97.80	202	95.71	13	91.18	181	92.09	9	96.25	92	92.50
3	18	98.21	208	96.07	15	95.18	193	93.18	10	96.25	93	92.50
4	18	98.89	211	97.50	16	91.27	193	91.18	12	95.50	95	93.75
5	11	98.23	212	96.90	17	92.18	196	93.09	13	96.75	95	94.58
6	11	97.64	214	95.89	17	95.18	200	90.27	13	96.07	97	92.08
7	12	97.89	217	97.32	17	91.18	204	93.09	14	96.32	97	93.75
8	12	97.57	217	98.57	19	94	204	92.09	8	96.32	99	93.75
9	13	97.85	221	96.07	19	94	205	93.09	9	95.90	100	93.57
10	14	97.75	221	98.57	19	95.09	205	93.09	9	95.50	100	95.00
μ	14	97.96	212	96.83	16	93.14	195	92.33	11	95.91	96	93.52
σ	± 2.90	± 0.01	± 7.27	± 0.01	± 3.2	± 1.4	± 11.3	± 0.01	± 2.64	± 0.01	± 3.08	± 0.01

(iii) a novel diversity strategy. Notably, such algorithmic changes allowed us to identify subsets of considerably smaller size and low classification error when we compared their performance to the standard binary variant BPSO.

Although our approach did not explicitly consider redundancy of features, our method selected strongly relevant features as indicated by an average SRF cover close to 100% in most cases. As we applied COMBPSO to cancer specific gene expression profiles, such a characteristic indicates the ability of our approach to select smallest, yet robust gene subsets that are highly relevant for the underlying disease system. As a consequence of their stability, such small gene subsets may well serve as consistent biomarkers that allow a reliable diagnostic call, may point to disease relevant genes as well as drug targets.

Although our wrapper method that uses a random forest classifier is highly cost effective at obtaining high classification performance subsets, computational costs may not scale with large datasets. Therefore, further research may need to focus on mitigating computation costs in the presence of ultra-high dimensional search space with millions of features.

REFERENCES

- Armstrong, S. A., Staunton, J. E., Silverman, L. B., Pieters, R., den Boer, M. L., Minden, M. D., Sallan, S. E., Lander, E. S., Golub, T. R., and Korsmeyer, S. J. (2001). Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nature Genetics*, 30:41 EP -. Article.
- Babu, S. H., Birajdhara, S. A., and Tambad, S. (2014). Face recognition using entropy based face segregation as a pre-processing technique and conservative bps based

feature selection. In *Proceedings of the 2014 Indian Conference on Computer Vision Graphics and Image Processing*, page 46. ACM.

- Bansal, J. C., Singh, P., Saraswat, M., Verma, A., Jadon, S. S., and Abraham, A. (2011). Inertia weight strategies in particle swarm optimization. In *Nature and Biologically Inspired Computing (NaBIC), 2011 Third World Congress on*, pages 633–640. IEEE.
- Bonyadi, M. R. and Michalewicz, Z. (2014). A locally convergent rotationally invariant particle swarm optimization algorithm. *Swarm intelligence*, 8(3):159–198.
- Bonyadi, M. R. and Michalewicz, Z. (2017). Particle swarm optimization for single objective continuous space problems: a review.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.
- Chandra Sekhara Rao Annavarapu, S. D. and Banka, H. (2016). Cancer microarray data feature selection using multi-objective binary particle swarm optimization algorithm. *EXCLI journal*, 15:460.
- Cleghorn, C. W. and Engelbrecht, A. P. (2014). A generalized theoretical deterministic particle swarm model. *Swarm intelligence*, 8(1):35–59.
- Dhrif, H. (2019). *Stability and Scalability of Feature Subset Selection using Particle Swarm Optimization in Bioinformatics*. PhD thesis, University of Miami, FL.
- Dhrif, H., Giraldo, L. G., Kubat, M., and Wuchty, S. (2019a). A stable hybrid method for feature subset selection using particle swarm optimization with local search. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 13–21. ACM.
- Dhrif, H., Giraldo, L. G. S., Kubat, M., and Wuchty, S. (2019b). A stable combinatorial particle swarm optimization for scalable feature selection in gene expression data. *arXiv preprint arXiv:1901.08619*.
- Eberhart, R. and Kennedy, J. (1995). Particle swarm optimization, proceeding of ieee international conference on neural network. *Perth, Australia*, pages 1942–1948.

- Engelbrecht, A. (2012). Particle swarm optimization: Velocity initialization. In *Evolutionary Computation (CEC), 2012 IEEE Congress on*, pages 1–8. IEEE.
- Han, F., Sun, W., and Ling, Q.-H. (2014). A novel strategy for gene selection of microarray data based on gene-to-class sensitivity information. *PLoS one*, 9(5):e97530.
- Han, F., Yang, C., Wu, Y.-Q., Zhu, J.-S., Ling, Q.-H., Song, Y.-Q., and Huang, D.-S. (2017). A gene selection method for microarray data based on binary pso encoding gene-to-class sensitivity information. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 14(1):85–96.
- John, G. H., Kohavi, R., and Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *Machine Learning Proceedings 1994*, pages 121–129. Elsevier.
- Kennedy, J. and Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 5, pages 4104–4108 vol.5.
- Khaire, U. M. and Dhanalakshmi, R. (2019). Stability of feature selection algorithm: A review. *Journal of King Saud University-Computer and Information Sciences*.
- Kumar, V. and Minz, S. (2014). Feature selection. *SmartCR*, 4(3):211–229.
- Lee, S., Soak, S., Oh, S., Pedrycz, W., and Jeon, M. (2008). Modified binary particle swarm optimization. *Progress in Natural Science*, 18(9):1161–1166.
- Lopez-Paz, D., Hennig, P., and Schölkopf, B. (2013). The randomized dependence coefficient. In *Advances in neural information processing systems*, pages 1–9.
- Mohamad, M. S., Omatu, S., Deris, S., and Yoshioka, M. (2011). A modified binary particle swarm optimization for selecting the small subset of informative genes from gene expression data. *IEEE Transactions on Information Technology in Biomedicine*, 15(6):813–822.
- Mohammadi, M., Noghabi, H. S., Hodtani, G. A., and Mashhadi, H. R. (2016). Robust and stable gene selection via maximum–minimum correntropy criterion. *Genomics*, 107(2-3):83–87.
- Perthame, E., Friguet, C., and Causeur, D. (2016). Stability of feature selection in classification issues for high-dimensional correlated data. *Statistics and Computing*, 26(4):783–796.
- Poli, R. (2009). Mean and variance of the sampling distribution of particle swarm optimizers during stagnation. *IEEE Transactions on Evolutionary Computation*, 13(4):712–721.
- Rezaee Jordehi, A. and Jasni, J. (2013). Parameter selection in particle swarm optimisation: a survey. *Journal of Experimental & Theoretical Artificial Intelligence*, 25(4):527–542.
- Shipp, M. A., Ross, K. N., Tamayo, P., Weng, A. P., Kutok, J. L., Aguiar, R. C. T., Gaasenbeek, M., Angelo, M., Reich, M., Pinkus, G. S., Ray, T. S., Koval, M. A., Last, K. W., Norton, A., Lister, T. A., Mesirov, J., Neubergh, D. S., Lander, E. S., Aster, J. C., and Golub, T. R. (2002). Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine*, 8:68 EP – Article.
- Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A. A., D’Amico, A. V., Richie, J. P., Lander, E. S., Loda, M., Kantoff, P. W., Golub, T. R., and Sellers, W. R. (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2):203 – 209.
- Van den Bergh, F. and Engelbrecht, A. P. (2010). A convergence proof for the particle swarm optimiser. *Fundamenta Informaticae*, 105(4):341–374.
- Xue, B., Zhang, M., and Browne, W. N. (2013). Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE transactions on cybernetics*, 43(6):1656–1671.
- Ye, F. (2017). Particle swarm optimization-based automatic parameter selection for deep neural networks and its applications in large-scale and high-dimensional data. *PLoS one*, 12(12):e0188746.
- Yong, Z., Dun-wei, G., and Wan-qiu, Z. (2016). Feature selection of unreliable data using an improved multi-objective pso algorithm. *Neurocomputing*, 171:1281–1290.
- Zhang, Y., Gong, D.-w., and Cheng, J. (2017). Multi-objective particle swarm optimization approach for cost-based feature selection in classification. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 14(1):64–75.