Super Learning hybridized Multi-objective Particle Swarm Optimization for Feature Selection with Imbalanced Health Data in Survey Research

Di Xiong[1] Honghu Liu [1,2,3]

Abstract

Effective screening surveys can assist in detecting early diseases among high-risk children who need treatment intervention. However, it is challenging to optimize the survey protocol for low incidence diseases with large health surveys. This paper proposes a Multi-objective Constrained Binary Particle Swarm Optimization (MCBPSO) method to identify effective and optimal survey items for disease detection. The algorithm balances on dual objectives, minimizing feature redundancy and maximizing partial AUC (Area under the ROC curve) with a constraint sensitivity at 0.8 for training data. Meanwhile, it realizes the variability by controlling velocity and the best performance of the swarm using mutation and resetting operators. Multiple machine learning algorithms were ensembled by a Super Learner to improve prediction performance. The proposed algorithm is applied to a recent oral health survey of children with 192 self-reported items. MCBPSO-based feature selection algorithms can be effectively applied to detect diseases with a low incidence rate. The cost-effective screening toolkit developed can be used in oral health screening for large school-age children in the future.

Key Words: Feature Selection; Partial AUC; Crowding Distance; Multi-objective Optimization; Synthetic Minority Over-sampling Technique (SMOTE).

1. Introduction

Feature selection has been widely used in many research areas with machine learning and pattern recognition. A feature is also called variable or attribute, which describes a property of subjects. Feature selection algorithms aim to improve the prediction performance of the model with more cost-effective subsets of predictors [1]. In the classification problems, irrelevant and redundant features can lead to high dimensional space with a possibly severe bias on the estimation, which is "the curse of dimensionality". The predictive power of a classifier will first increase with the number of features, but it then is followed by steep fall [2]. Besides, when analyzing data in high-dimensional space, data usually becomes much sparser with more bias introduced in the estimation. A condensed feature subset would be more preferable, especially for the survey research where it is infeasible to collect all possible features via a questionnaire with a large population.

Typically, the feature selection algorithms are summarized into three types including filter, embedded, and wrapper methods. The filter method selects the most relevant features

[1]Department of Biostatistics, Jonathan and Karen Fielding School of Public Health, University of California, Los Angeles, 10833 Le Conte Ave, Los Angeles, CA 90095

[2]Division of Public Health & Community Dentistry, School of Dentistry, University of California, Los Angeles, 10833 Le Conte Ave, Los Angeles, CA 90095

[3]Department of Medicine, David Geffen School of Medicine, University of California, Los Angeles, 10833 Le Conte Ave, Los Angeles, CA 90095

by performing independent bivariate analysis with the targeted outcome, such as correlation, χ^2 test, and analysis of variance (ANOVA) while ignoring the relationship among features. Meanwhile, the embedded method, on the other hand, picks the features which contribute the most to the prediction given a specific model [3].

Comparing to the filter and embedded methods, the wrapper method iteratively explores the best subset of features based on the classifier performance directly [4]. Unlike the embedded method, it is flexible to incorporate with various classifiers. However, it still results in more computational difficulties. An exhaustive greedy search can be conceivably performed to guarantee the optimal subset for low-dimensional data. When the dimension getting higher, the number of the potential subset will run into 2^D where D is the dimension of the feature space. It is a classical NP-hard problem (Non-deterministic Polynomial-time) as the number of the subset will increase exponentially with D. Thus, the search may become computationally intensive.

Swarm Intelligent (SI) algorithms have been proved to solve NP-hard computational problems efficiently [5]. It shares good properties on autonomy, self-organization, scalability, and flexibility [6]. The Particle Swarm Optimization (PSO) is one of the evolutionary SI optimization techniques which is inspired by the behavior of birds. Many works have been done to integrating Binary PSO (BPSO) with different Machine Learning algorithms as a wrapper method for feature selection. PSO hybrid Support Vector Machine (SVM) algorithm performed better than Genetics Algorithm on the classification accuracy [7, 8]. Improved BPSO-KNN and BPSO hybrid Decision Tree own higher classification accuracy and a less total number of features comparing with Logistic Regression, Back-propagation Neural Networks (BPNN), SVM and Probabilistic Neural Networks [9–11]. BPSO-Random Forest incorporating a sampler for imbalanced data has a better performance than other re-sampling algorithms [12]. Multi-objective PSO-based filter and wrapper methods for feature selection have been also proposed in situations where multi-objective models outperform the models with a single objective function [13, 14].

Many machine learning algorithms are available to incorporate with BPSO for feature selection. However, no single algorithm could fit for all cases. One possible solution is that we need to train multiple models and compare the results before making the final decision. Another way is to ensemble the performance of multiple learners. Super Learner creates optimal weights for a set of candidate learners for the prediction problem to minimize the cross-validation risk [15]. It has been shown theoretically to perform asymptotically no worse than any of its candidate learners [16]. Despite the predictive power, few works have been done towards feature selection using the Super Learner.

One of the greatest barriers in many state-of-art machine learning algorithms is imbalanced labeled data. Imbalanced data refers to a classification problem where the distribution of classes is extremely skewed. Such imbalance problems are encountered in various fields such as economic, engineering, and public health. However, mainstream learning methods are designed for balanced training data instead. Objective functions that are optimized by these methods are mainly related to cross-entropy, mean squared error, and accuracy [6]. It may generate a frequency bias that emphasizes learning on the dominated class. For real-world data especially in the health field, the class distribution is usually imbalanced. For example, when detecting active cavity among kids and adolescents with a prevalence of 12%, even if the classifier assigns all subjects as being caries-free, it could still attain an error rate merely 12%. Such classification is trivial. The Receiver Operator Characteristic (ROC) curve is one of the common ways to evaluate the classification performance for imbalanced labeled data by presenting the relationship between true-positive rate (sensitivity) and false-positive rate (1 - specificity) under various thresholds. However, the importance of sensitivity and specificity are not always the same. For instance, in an

oral health screening for a large population, it is more essential to identify the kids with high-risk dental problems and ensure they get needed care. A relatively lower specificity with a pre-assigned high sensitivity could be preferred in such a scenario. Therefore, the constraint on the sensitivity or specificity sometimes needs to be considered when tackling the real problems in the health field.

In this work, we utilize the variability of PSO and Super Learner to select the most compact feature subset with a maximum Partial Area under the ROC curve (pAUC) with a constraint on sensitivity. Related algorithm background is introduced in Section 2 including Particle Swarm Optimization, Multi-objective Optimization, and Super Learner. Section 3.1 describes the fitness function along with the Super Learner algorithm modified by Synthetic Minority Over-sampling Technique (SMOTE) sampler for imbalanced labeled data in Section 3.2. A novel Constrained BPSO with mutation and resetting operator algorithm has been proposed in Section 3.3 with a parameter initialization setting. In section 4, the proposed algorithm is applied to a self-reported oral health survey of children to detect active cavity. Some discussions are followed in Section 5.

2. Algorithm Background

2.1 Particle Swarm Optimization

Particle swarm optimization (PSO) was first presented by Eberhart and Kennedy in 1995 to mimic the natural behavior of swarms in an optimization routine [17]. Multiple variations have been developed to work on various fields such as optimal design [18], circle detection [19], medical diagnosis [20], and so on. It releases a swarm of particles to search the available parameter space to find the solution of the optimization problems. Each position of the particle represents one possible solution at that evolution.

PSO initializes population of solutions with size m within the available searching space. In each iteration, as shown in Figure 1, each particle will updated their velocity $\vec{v}_i = (v_{i1}, v_{i2}, \cdots, v_{id})$ and position $\vec{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})$ given formulas (2.1) and (2.2) for a $d-$ dimensional searching space.

$$\vec{v}_i(t+1) = w\vec{v}_i(t) + c_1 r_1(\vec{x}_i^{\text{best}}(t) - \vec{x}_i(t)) + c_2 r_2(\vec{x}^{\text{best}}(t) - \vec{x}_i(t)) \tag{2.1}$$

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \tag{2.2}$$

where $r_1, r_2 \overset{\text{i.i.d.}}{\sim} U[0,1]$ and t represents the iteration times. The best solution is $\vec{x}_i^{\text{best}}$ for each individual particle i across evolution and $\vec{x}^{\text{best}}$ within the entire swarm, marking as personal best solution *pbest* and global best solution *gbest*.

For each iteration, the velocity $\vec{v}_i(t+1)$ and position $\vec{x}_i(t+1)$ for particle i are updated based on $\vec{x}_i^{\text{best}}$ and $\vec{x}^{\text{best}}$ using formulas (2.1) and (2.2). Therefore, the efficiency of the PSO depends on the choice of m, w, c_1 and c_2, which often need to be tuned case by case. w is the inertia weight to control the impact of velocities in the previous iteration on the current one. Inertia weight is a pivotal factor to balance the local and global search. $w > 1$ strengthens the global exploration overexploitation, while $w < 1$ focuses more on the local searching in the favor of current best positions. c_1 and c_2 are individual and social learning rates respectively to represent the relative influence of the best positions of *pbest* and *gbest*.

A Binary version of PSO has been proposed for discrete problems, like feature selection [21]. For feature selection with $d-$ dimensional space, the position for each particle is a set of Boolean lattices with either 0 or 1 to indicate the presence or absence of a feature. Binary PSO (BPSO) aims to restrict each particle to move across the vertices of a $d-$ dimensional

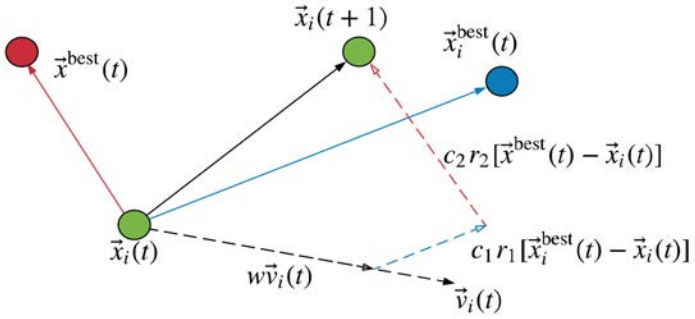

Figure 1: Example of Particle Updates in PSO

hyper-cube by using sigmoid transformation function,

$$S(u) = \frac{1}{1 + \exp(-u)}. \tag{2.3}$$

Instead of using formula (2.2), the position $\vec{x}_i(t)$ could be updated using

$$x_{ij}(t) = \begin{cases} 0, & \text{if } \sigma < S(v_{ij}(t)) \\ 1, & \text{otherwise} \end{cases}. \tag{2.4}$$

in which $\sigma \sim U[0,1]$. $v_{ij}(t)$ is the velocity generated in formula (2.1) for particle i on j-th dimension in $t-$th iteration.

2.2 Multi-objective Optimization

The multi-objective optimization makes the optimal decision by trading off two or more conflicting objectives. The standard form of such problem could be written as

$$\begin{aligned} \min_x F(x) &= \{f_1(x), f_2(x), \cdots, f_k(x)\} \\ \text{s.t. } g_i(x) &\leq 0, i = 1, 2, \cdots, u, \\ h_j(x) &= 0, j = 1, 2, \cdots, v. \end{aligned} \tag{2.5}$$

where $f_r(x), r = 1, 2, \cdots, k$ are various objective functions regarding to the vector of decision variable x. $g_i(x), i = 1, 2, \cdots, u$ and $h_j(x), j = 1, 2, \cdots, v$ are inequality and equality constraints with $u \geq 0, v \geq 0$. Intuitively, the maximization optimization could be treated by taking negative sign on the objective functions in (2.5).

A common way to simplify problem (2.5) is to weight the multi-objectives as a single objective, like classification accuracy and redundancy of the selected feature sets [7, 11]. However, tuning the weight is always tricky and various case by case, which is hard to be generalized and explained in a real case. Multi-objective methods, on the other hand, are designed to optimize all objectives simultaneously.

Let y and z be two possible solutions for the minimization problem (2.5). y is dominated by z (i.e. $y \prec\prec z$) if and only if

$$\forall i : f_i(y) \leq f_i(z) \text{ and } \exists j : f_j(y) < f_j(z), i, j = 1, 2, \cdots, k.$$

Pareto-optimal solutions denote ones which is not dominated by any other solutions in the set. Figure 2, for example, presents a minimisation problem with two objective function

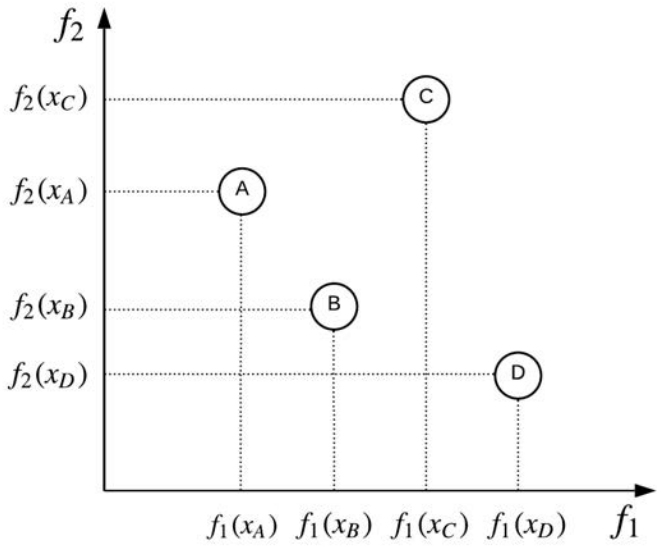

Figure 2: A Multi-objective minimisation problem with two objective functions

f_1 and f_2. Since $f_1(x_A) < f_1(x_B) < f_1(x_C) < f_1(x_D)$ and $f_2(x_D) < f_2(x_B) < f_2(x_A) < f_2(x_C)$, C is not dominated by D but by A and B. A, B and D are all Pareto-optimal solutions, which composing the non-dominated set. Such solution in the non-dominated set is also known as the Pareto front.

For multi-objective PSO, the position and velocity of particles will be updated based on those Pareto fronts in each iteration instead of a single best solution. The crowding distance has been introduced to rank the solutions within the non-dominated set to determine the best solutions $\vec{x}^{\text{best}}$ among the entire swarm and $\vec{x_i}^{\text{best}}$ for each particle i during the evolutionary process [22]. Besides, due to the chance of various scales for objective functions, we applied the Relative Crowding Distance (RCD) instead, which scaling the crowding distance by its maximum distance for each objective metric. The overall relative crowding distance quantifies the sum of the relative distance of its two neighbor solutions corresponding to each objective function.

Multi-objective BPSO using the ideas of crowding, mutation, and dominance to select features can balance accuracy and mutual information [13]. In the next section 3.1, we convert the feature selection into a two-objective optimization problem by trading off the redundancy and classification 'accuracy' for imbalanced labeled data.

2.3 Super Learner

Super Learner aims to ensemble multiple candidate learners for prediction by considering the over-fit problem. The main idea of it is to unify the loss-based estimation in the form of a new learner and pick the optimal weights for a given prediction problem based on cross-validated risk [15]. Theoretical results show that Super Learner is asymptotically efficient as well as or better than any of the candidate learners on finite samples [16].

Figure 3 presents the general steps for Super Learner. The input data is first divided into V folds. For each time, only one fold serves as the validation set, while the remaining folds are used as a training set to train different machine learning algorithms like ML1, ML2, and so on, in the figure. Models are validated by each validation set respectively. The result is weighted to maximize some target metrics. For imbalanced data, α is obtained by optimiz-

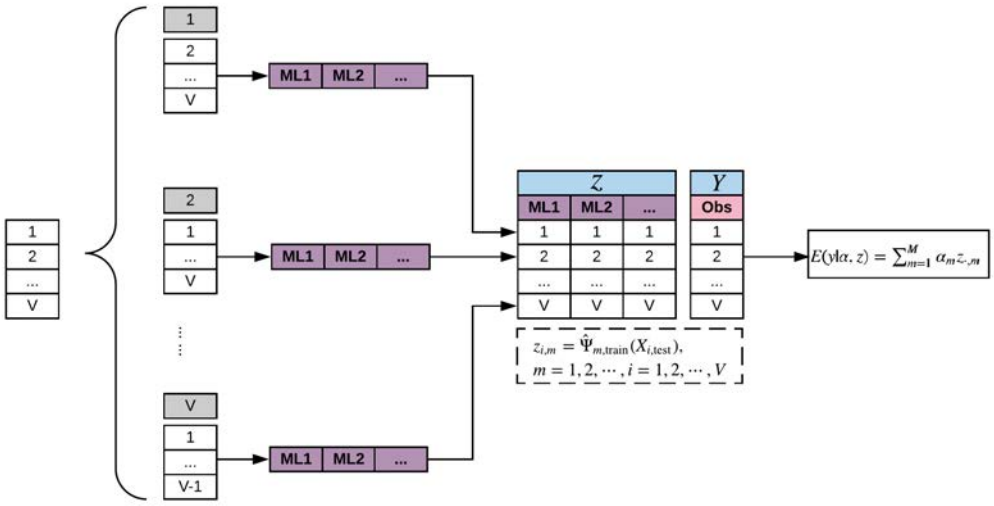

Figure 3: Super Learning (ML: Machine Learning)

ing metric AUC instead of usual unified loss function, $\hat{\alpha} = \arg\max_{\alpha} \sum_{i=1}^{n} (Y_i - S(y_i|\alpha))^2$. Many potential candidate learners are available for Super Learner, such as Least Angle Regression, Logic Regression, D/S/A algorithm (Deletion/ Substitution/ Addition), Classification and Regression Tree (CART), Random Forest, Ridge Regression, and Multiple Adaptive Regression Splines (MARS) and so on. In this paper, we will utilize Super Learning with Logistic Regression, Support Vector Machine (SMV) with Radial Kernel, and K-Nearest Neighborhood (KNN). The results of Super Learner and all candidate learners will be compared in section 4.3.

3. Proposed Algorithm

3.1 Fitness Function

This work aims to select an optimal feature subset by minimizing redundancy of the feature set and maximizing classification performance. The final model should result in the most compact feature subset yielding the largest partial AUC with a specified boundary on sensitivity.

3.1.1 Entropy, Mutual Information, and Redundancy

Entropy (H) is a measure of the uncertainty of a random variable, which is defined as

$$H(X) = -\sum_{j=1}^{n} p(x_j) \log_b p(x_j)$$

where x_j is the j-th possible event for variable X and b is the base of the logarithm used. With the log base $b = 2$, the unit of $H(X)$ is called bits. Intuitively, high entropy implies that each event has about the same probability of occurrence, while events with different probability of occurrence will lead to a low entropy.

Mutual information (MI) is widely used to quantify the statistical independence among

variables [23]. In the case with two random variables,

$$I(X_1, X_2) = H(X_1) + H(X_2) - H(X_1, X_2) = \sum_{x_1 \in X_1, x_2 \in X_2} p(x_1, x_2) \log_b \frac{p(x_1, x_2)}{p(x_1)p(x_2)}$$

Comparing with correlation, MI investigates the distance between two probability distribution without any assumption of linearity, normality, or even monotonicity of random variables. It could be extended to n variables easily as

$$I(X_1, X_2, \cdots, X_n) = \sum_{i=1}^{n} H(X_i) - \sum_{i<j}^{n} H(X_i, X_j) \cdots - (-1)^{n-1} H(X_1, X_2, \cdots, X_n)$$

The sum of MI among two random variables, $\sum I(X_i, X_j)$, was used largely as a criterion for feature selection [13, 24]. However, for multiple categorical variables, MI may not be used to describe the statistically independent relationship [25].

Redundancy (R) also evaluates independence among variables stochastically as MI via log-linear models. It is also known as total correlation and multiinformation [26, 27]. Features might share some information between each other and turn to be redundant. For a given set with d random variables $X = \{X_1, \cdots, X_d\}$, redundancy measures the amount of information sharing among variables. As a generalization of the mutual information, it is in the form as,

$$\begin{aligned} R(X_1, X_2, \cdots, X_d) &= \sum_{i=1}^{k} H(X_i) - H(X_1, X_2, \cdots, X_d) \\ &= I(X_1, X_2, \cdots, X_d) + I(X_2, \cdots, X_d) + \cdots + I(X_{d-1}, X_d) \\ &= \sum_{x_1 \in X_1} \cdots \sum_{x_d \in X_d} p(x_1, x_2, \cdots, x_d) \log_b \frac{p(x_1, x_2, \cdots, x_d)}{p(x_1)p(x_2)\cdots p(x_d)} \end{aligned} \tag{3.1}$$

Noticeably, for $d = 2$, $R(X_1, X_2) = I(X_1, X_2)$. $R(X_1, X_2, \cdots, X_d)$ is non-negative with equation attained if and only if X_is are mutually independent. The property of monotonicity ensures that the amount of redundancy of variables can never decrease with more variables added [28]. In the proposed algorithm, we evaluate the feature redundancy to take consideration on the number of features and their dependence together.

3.1.2 Partial Area under the ROC curve

The Receiver Operating Characteristics (ROC) curves were originally developed for the signal detection theory [29] and later used widely to evaluate machine learning algorithms for binary classification problems in numerous fields like economics [30], medical [31], and health field [32].

	Actual Positive	**Actual Negative**
Predicted Positive	True Positive (TP)	False Positive(FP)
Predicted Negative	False Negative (FN)	True Negative (TN)

TPR = Sensitivity = Recall = $\frac{TP}{TP+FN} = 1-$ FNR
TNR = Specificity = $\frac{TN}{TN+FP} = 1-$ FPR

Table 1: Confusion Matrix for Binary Classification

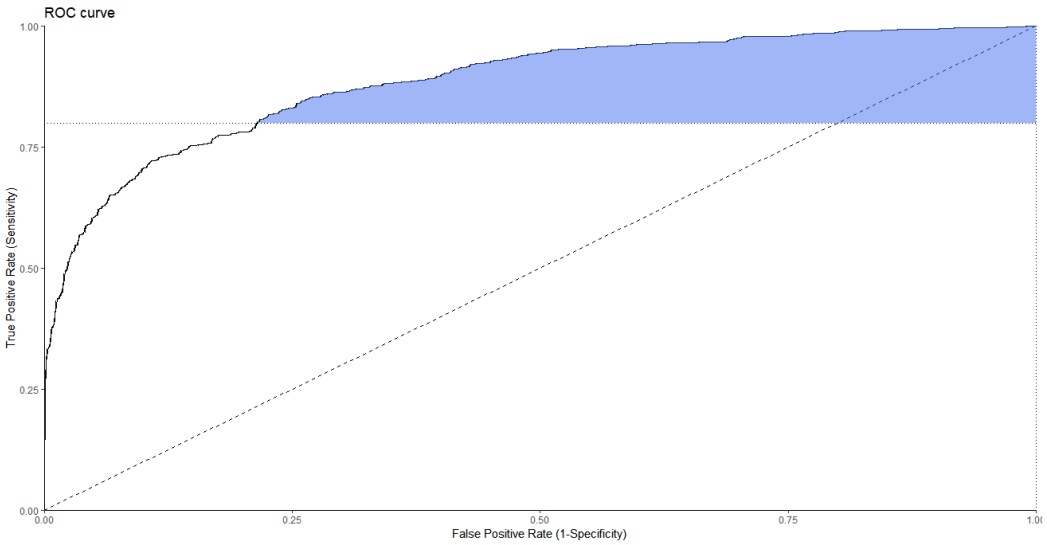

Figure 4: Partial AUC with Minimum Sensitivity Allowance as 0.8

For classification problems with imbalanced labeled data, ROC is one of the gold-standard in the literature to evaluate the classifier's ability. A ROC curve could be presented as the relationship between the True Positive rate (TPR, i.e. sensitivity) and the False Positive rate(FPR, i.e. 1 - specificity) as defined in Table 1 under various thresholds. The perfect classifier has a 100% TPR and 0% FPR with ROC curve passing the upper left corner $(0, 1)$ in Figure 4.

The Area under the ROC curve (AUC) is a natural summary statistic to compare the performance among classifiers. It quantifies the rank power of the positive prediction probabilities exceeding the negative ones in general. However, it is biased to use AUC to select the potential optimal classifiers. In the practice, the two boundary regions on the ROC curve (i.e, with specificity near 1 and sensitivity near 0, or the opposite) are useless. The area under certain regions on the ROC curve is more preferable.

Partial AUC (pAUC) focuses on the area with a specified boundary of sensitivity and specificity [33] . The proposed algorithm aims to select features yielding high specificity with a prerequisite sensitivity level. Let Y denote the continuous outcome of a classifier and X is the feature space. For each threshold c, a subject is labeled as positive if $Y > c$ or negative otherwise. Denote $+$ as actual positive and $-$ as negative. It follows that $\text{TPR}(c) = p(Y > c|+)$ and $\text{FPR}(c) = p(Y > c|-)$. Then, for the classifier with feature space X, the ROC curve in Figure 4 is defined as $\{(s, \text{ROC}(X; s)), s \in (0, 1)\}$ where $\text{ROC}(X; s) = \text{FPR}(\text{TPR}^{-1}(s))$. Given a lower bound of sensitivity $s \in (0, 1)$, the shade area in Figure 4 is

$$\text{pAUC}(X; s) = \int_s^1 \text{ROC}(X; t)dt \tag{3.2}$$

3.2 Super Learning with the SMOTE re-sampler

Class imbalance is a common problem encountered when working with machine learning algorithms. It often leads to a significant impact on classification performance, especially for the minority group. A good performance could be easily achieved by labeling all samples as the majority class. However, such a classifier is worthless in practice. Re-sampling techniques are routinely used to balance the frequencies of classes and convert the data to suit the well-designed machine learning algorithms.

Various under-sampling and over-sampling approaches have been developed. Under-sampling refers to the re-sample procedure to remove some instances from the over-represented (majority) class when the quantity of data is sufficient. On the contrary, over-sampling means to add instances for the under-represented (minority) class. Synthetic Minority Over-Sampling Technique (SMOTE) [34] has shown to perform better on classification and feature selection [12]. In SMOTE, the majority class is under-sampled by removed randomly; while the minority class is over-sampled by creating "synthetic" examples based on its k nearest neighbors instead of bootstrapping with replacement.

Super Learner is an ensemble algorithm that obtains the optimal combination of multiple machine learning algorithms. In the case of cross-validation, we apply the SMOTE technique on the training subset only and leave the validation subset aside to be more consistent with the actual scenario. Algorithm 1 shows the pseudo-code of Super Learner with the SMOTE re-sampler and AUC maximized.

Algorithm 1 Super Learner with the SMOTE re-sampler and AUC maximized (Adapted from [35])

1. Split data set into training and validation sets based on V-fold cross validation scheme:
   - Randomly divide the dataset into V-equal size folds.
   - Let $v-$th group as validation subset $V(v)$ and the rest combined as training subset $T(v)$, $v = 1, 2, \cdots, V$.
   - Applied SMOTE technique on the training subset only.
2. Train each candidate learner $L_m, m = 1, 2, \cdots, M$ using V-cross validation and store the prediction on its corresponding validation subset $V(v)$ as $\hat{\Psi}_{m,T(v)}(X), X \in V(v)$.
3. Propose a linear regression with a vector of weights α, $S(y|\alpha) = \sum_{m=1}^{M} \alpha_m \hat{\Psi}_{m,T(v)}(X)$ where $\alpha_m \geq 0$ and $\sum_{i=1}^{m} \alpha_m = 1$.
4. $\hat{\alpha} = \arg\max_{\alpha}$ AUC.
5. Integrate the final Super Learner as $\hat{\Psi}(X) = S(y|\hat{\alpha})$.

Unlike other machine learning algorithms, Super Learner is designed to find the best-weighted average on performance, instead of tuning for the single best hyper-parameters or model. Classifiers with various tuning parameters will be all included. Although the hyper-parameter tuning helps to improve the performance of a super learner, the impact of it is minor [36].

3.3 Feature Selection Procedure

To select the most compact feature subsect that yields the largest partial AUC with a pre-specified boundary of sensitivity, the dual-objective problem (2.5) targets to minimize the redundancy and maximize the pAUC. Then, it could be written using formulas (3.1) and

(3.2) as

$$\min_X \{R(X), -\text{pAUC}(X; s)\}$$
$$\text{s.t. sensitivity} > s \tag{3.3}$$

where $X = (x_1, x_2, \cdots, x_d)$ is the d dimensional feature space and s is the predefined sensitivity minimum boundary. In this section, a novel constrained BPSO with mutation and resetting operators has been proposed to perform the feature selection on the imbalanced labeled data. Super Learner is the recommended classification algorithm over its candidate learners.

3.3.1 Standard Multi-objective BPSO algorithm with Super Learning

The swarm intelligent algorithms are summarized into 5 main steps: swarm initialization, fitness function evaluation, checking stop conditions, updating the particles or agents, and returning the global best solution.

The swarm is randomly initialized in the $d-$ dimensional feature space. For each particle, it evaluates two metrics, redundancy and pAUC, via a $k-$fold ($k = 5$) Cross-Validated Super Learner with candidate learners including Logistic Regression, SVM with Radial kernel, and KNN. The samples are first divided into training set X_{train} and testing set X_{test} randomly. And it further splits the training set $Z_{\text{train}} = \{X_{\text{train}}, Y_{\text{train}}\}$ into 5 folds with one fold severing as the testing subset $Z_{\text{train, (k)}}$ and the remaining folds as training subset $Z_{\text{train, (-k)}}$ for each time $k = 1, 2, \cdots, K$. Model is trained by the training subset $Z_{\text{train, (-k)}}$ first and then validated by the validation subset. pAUC on the validation sets under the proposed algorithm will be aggregated as

$$pAUC_{\text{CV}} = \sum_{k=1}^{K} \frac{pAUC(Z_{\text{train, (k)}}; s)}{K}. \tag{3.4}$$

The redundancy and the cross-validated pAUC on the training set X_{train} are recorded to identify the non-dominated set A and ranked in the ascending order based on Relative Crowding Distance. A *gbest* is randomly selected from a specified top portion of the sorted non-dominated set (e.g. 10%). Furthermore, for each particle, *pbest* is replaced by the current solution if it is dominated by the current one. After updating the *gbest* and *pbest*, the velocity and position for each particle will be used to calculate the position and velocity of particles for the next iteration based on formulas (2.1) and (2.4). All the procedures will be repeated until the maximum iteration or convergence conditions are reached. The algorithm will return the non-dominated set of solutions A with the performance on the testing set Z_{test} including redundancy, pAUC, threshold, sensitivity, and specificity.

To evaluate the performance and compare different classification methods, the data set has been split into 5 folds in advance. The procedure as shown in Figure 5 has been repeated 5 times with only 4 out of 5 folds serving as the training set and the remaining one as the testing set for each time.

3.3.2 Mechanisms to improve convergence in BPSO

One drawback of the original BPSO in section 2.1 is prematurely convergent to sub-optimal points. Mutation and swarm best-resetting operators are two common modifications to improve the variability of the BPSO, especially for more challenging optimization tasks [37].

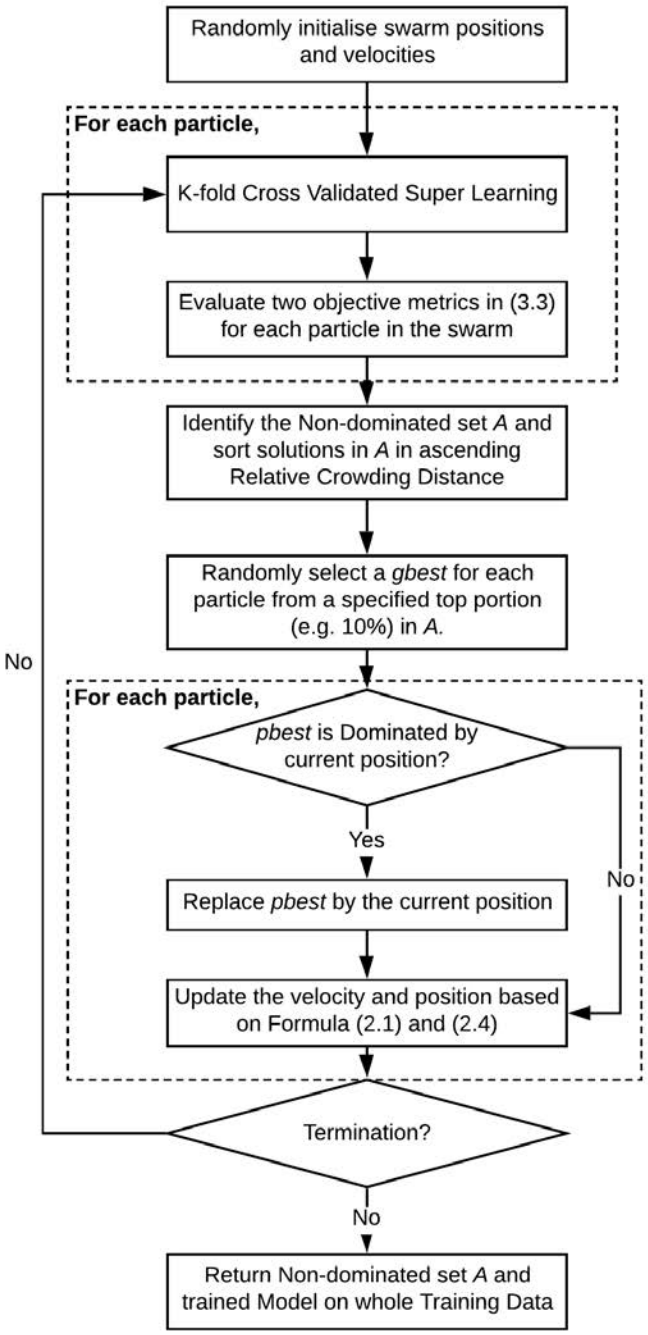

Figure 5: Super Learning hybridized Multi-objective Particle Swarm Optimization

Formula (2.1) update the particle velocity by taking reference on the personal best solution *pbest* and global best solution *gbest*. Therefore, the qualities of *pbest* $\vec{x}_i^{\text{best}}$ and *gbest* $\vec{x}^{\text{best}}$ have a great impact on the performance of PSO. In fact, it shares a high chance to trap in the sub-optimal if one solution happens to be better than any of the previous solution. All the particles might be concentrating on that sub-optimal.

A mutation operator is originally inspired by the Genetic Algorithm, which allows some particles to try unseen areas on the parameter space [38]. Both Multi-objective PSO and BPSO improved their performance when modified by mutation operator [13, 39]. It suggests introducing a trivial mistake on the position as,

$$\begin{cases} x_{ij} = \neg x_{ij} & r \leq R \\ x_{ij} = x_{ij} & \text{otherwise} \end{cases}, i = 1, 2, \cdots, m, j = 1, 2, \cdots, d \tag{3.5}$$

where R is a pre-specified mutation probability and $r \sim U[0, 1]$. In this proposed algorithm, $R = 1/d$ with d as the dimension of feature space. Therefore, after updating the velocities and positions via formulas (2.1) and (2.4), one bit of parameters is expected to flip for each particle. However, high dimensional data might require a higher mutation probability R.

Meanwhile, the personal best solution *pbest* needs reset if it gets trapped in one position for more than $I_{\max}$ iterations. Normally, a solution with all bits 0 except one element as 1 replaces *pbest* in such a scenario. However, since one of our objective redundancy is consistent with the number of features selected, such manual rest might produce a trivial non-dominated solution. Instead, *pbest* is reset as the current position x_i for particle i. $I_{\max}$ is pre-assigned as 3 in this work.

3.3.3 Additional notes for the proposed algorithm

The proposed algorithm for the problem 3.3 as described in Algorithm 2 targets to find the best subset with low redundancy and high partial AUC with constraints on sensitivity, like the minimum allowance as 0.8. However, after obtaining the potentially optimal classifiers, the classification threshold for testing or unseen data should be determined for utilization. In most of the algorithms, the default thresholds are 0.5 which is usually inappropriate for imbalanced classification problems. Some previous works also provide a good insight to use the prediction probability and auto-tuning the threshold for testing data [40].

Also, to avoid over-fitting on the redundancy, the case in the non-dominated set with the lowest-performing on specificity will be deleted in each of the iterations. In multi-objective methods, the best solution for only one of the objectives will always be preserved in the non-dominated set. However, a solution with minimum redundancy but low pAUC is useless.

3.4 Implementation

In this study, we consider dual objectives, minimizing redundancy and maximizing pAUC. An extra sensitivity constraint is taken into consideration when calculating pAUC using formula (3.2) and determining the threshold. Algorithm 2 summarizes a pseudo-code for the proposed constrained BPSO with mutation and resetting operator for various classification algorithms, like Super Learning with SMOTE re-sampler in Algorithm 1. The inertia weight is set as $w = 1.4$ with learning rates as $c_1 = c_2 = 2$ [41]. r_1, r_2, σ, r are four random values independently sampled from (0,1) uniformly. Velocities are restricted within the range of $[-6, 6]$ to utilize the sigmoid transformation function in formula (2.3). A common choice for m is between 20 and 40 [37]. Mutation probability is $R = 1/d$ where d

is the dimension of the feature space. The proposed algorithm with various swarm size $m = 10, 20, 30$ has been tested with $T_{\max} = 60$ iterations in section 4.2 to investigate of swarm size on the performance on the training set. The maximum iteration is set as $T_{\max} = 50$ when comparing across Super Learner and its candidate learners including Logistic Regression, SVM with Radial kernel, and KNN in section 4.3 on the testing set. For each classifier, PSO updates its particles based on the average cross-validated performance on the validation sets.

Algorithm 2 Proposed Multi-objective Constrained Binary PSO Algorithm MCBPSO with Mutation and Resetting Operator

Data: Labelled training set $Z_{\text{train}} = [X_{\text{train}}, Y_{\text{train}}]$, labelled testing set $Z_{\text{test}} = [X_{\text{test}}, Y_{\text{test}}]$, pre-specified lower boundary of sensitivity s.

Result: A set of non-dominated Boolean lattice solutions to indicate selected variables as $X = \arg\min_X \{R(X), -\text{pAUC}(X; s)\}$ on training set Z_{train}.

Initialize the swarm and parameters in section 3.4.

Begin:

for *each particle* $i, i = 1, 2, \cdots, m$ **do**

Assigned classifier (e.g., Super Learner with the SMOTE sampler in Algorithm 1): Conduct 5-fold cross validation (CV) on the training set Z_{train}.

Determine the threshold θ by $\max$(sensitivity + specificity) when sensitivity $> s$.

Record two objective metrics, redundancy and average CV pAUC, in formulas (3.1) and (3.4) on the training set Z_{train}.

end

Denote the positions of particles $\vec{x}_i(t), i = 1, 2, \cdots, m$, in the swarm as set B.

while $t <= T_{max}$ **do**

Identify the non-dominated solutions in B based on the two objective metrics and delete the solution with the lowest specificity.

Calculate the Relative Crowding Distance (RCD) and sort non-dominated set A in the ascending order by its RCD.

for *each particle* $i, i = 1, 2, \cdots, m$ **do**

Randomly selected a solution from top 10% of non-dominated set A as *gbest* $\vec{x}^{\text{best}}$;

if *pbest is dominated by current position or stationary for more than 3 iterations* **then**

Reset *pbest* $\vec{x}_i^{\text{best}}$ by $\vec{x}_i(t)$;

end

Update the velocity $\vec{v}_i(t+1)$ and position $\vec{x}_i(t+1)$ of particle i as formulas (2.1) and (2.4);

Perform mutation operator on position $\vec{x}_i(t+1)$ with mutation probability $R = 1/d$.

end

Add the updated particles in set B.

end

In addition to the cross-validation on the classification algorithms, to evaluate the proposed algorithm, the samples are divided into 5 folds by their labels with one fold serving as the testing set for each time and the remaining 4 folds as the training set in advance. Each fold shares the same percentage of the positive event (disease). Algorithm 2 is conducted on each training set and evaluated on the testing set. The results of average performance on the testing sets are reported including the number of total features selected, redundancy, pAUC, sensitivity, and specificity.

4. Application to Active Cavity Outcome Prediction

4.1 Description of dataset

Oral health is one of the essential components of well-being and overall health. Today, children and adolescents are exposed to diets with high levels of sugar, which leads to dental problems and oral diseases [42]. Most oral health problems, such as active cavities, bad breath, and cold sores, are treatable with proper dental care, especially at an early stage. In the United States, a major challenge is the cost of dental care and access to oral health screenings for children. Especially for large populations, the individual oral health screening is infeasible for everyone. Effective screening programs, including surveys that would assist in identifying early diseases in high-risk children who require intervention, are desired. However, the complexity of the surveys will need to be carefully addressed by identifying the effective and optimal sets of survey items. Therefore, it is critical to design a screening survey toolkit through proper feature selection methods to identify children and adolescents with existing oral health problems.

The *Patient-Reported Outcome Measurement Information System* (PROMIS) was initiated in 2004 to develop and validate a system of highly reliable, precise measures of Patient-reported health status for physical, mental, and social well-being [43]. Liu et al. applied the PROMIS methodology to develop an oral health item bank (OH-PROMIS survey) for administration to children [44].

The survey with 139 items for children with ages from 8-17 covering physical, mental, social, global health. All of those items are categorical data, except child age, grade, and family size. The nominal items are encoded as dichotomous variables by one-hot encoding. The ethnicity of the children, for example, is coded as one category per variable. Meanwhile, the ordinal items, especially the ones on the Likert scale, are treated as continuous variables and re-scaled to a standard normal distribution for classification algorithms. After encoded, it turns to be with 192 items. Two faculty pediatric dentists from the UCLA School of Dentistry examined children. Around 12% of the children are labeled having active cavities, which is imbalanced to work with the traditional classification algorithms.

Field test data were collected from diverse dental clinics and private practices throughout the Greater Los Angeles Area from August 2015 to May 2018. Participating sites cover low-income neighborhoods to high-income communities with diverse racial and ethnic compositions. A total of 380 children participated in the study. To our best knowledge, this database is the unique questionnaire available to focus on the current oral health status of children and adolescents with self-reported outcome labeled by a dental exam result.

The term survey item is equivalent to the feature mentioned above. And therefore, the propose of this study turns to select a condense survey-item subset with good performance on the partial AUC. In the next subsections, we will compare the variability of PSO using different swarm size, prediction performance on the testing sets for various methods, and potential feature subsets for the active cavity.

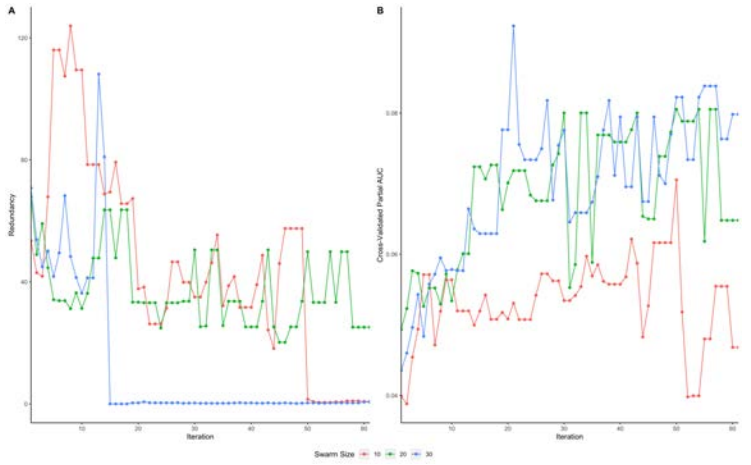

Figure 6: Average Redundancy (A) and Cross-Validated Partial AUC with Sensitivity $>$ 0.8 (B) in Non-Dominated Sets for each Iteration by Different Swarm Size

4.2 Performance with different swarm size

One of the criteria factors for the efficiency of the PSO algorithm is the swarm size of m. Figure 6 presents the average redundancy and cross-validated partial AUC of the validation sets in the non-dominated sets in each iteration with different swarm size. The redundancy tends to decrease among the iteration, while the cross-validated partial AUCs on the training set are increasing. As more particles getting involved in the algorithm for each run, the results become more stable in the target direction. With swarm size $m = 30$, the average redundancy dropped down to about 0 soon after 15 runs and the corresponding cross-validated pAUC is increasing and gradually stays around 0.07. A larger swarm size will provide more solutions in one iteration but takes much more time. Meanwhile, an average unique solution in each runs for the proposed algorithm with different swarm size with m = 30, 40, and 50 are about the same as 94%, 93%, and 92% correspondingly. In the following experiment, we will take swarm size $m = 30$ for illustration.

4.3 Comparison

Super Learner ensembles the performance of multiple machine learning models. In this work, the candidate learners are Logistic Regression, SVM with Radial Kernel, and KNN. To compare the performance of Super Learner and its candidate learners, the samples have been divided into 5 folds with only one fold as the testing set and the other 4 folds as the training set for each time. It will repeat 5 times with different folds been tested only once. Figure 7 shows the minimum redundancy and maximum pAUC among the non-dominated set for the testing set. In general, the Super Learner achieved a higher pAUC with less redundancy of the feature subset selected. In run 1 and 4, KNN performed better on maximum pAUC than other algorithms but with the highest minimum redundancy on the testing set. Logistic Regression obtained a higher pAUC in 3rd run but with double the minimum redundancy. SVM with Radial Kernel works well with the 5th run and owns a relatively small redundancy. Although Super Learner is only the best algorithm for run 2, it ranks almost in the second-best with much smaller redundancy than the best algorithm.

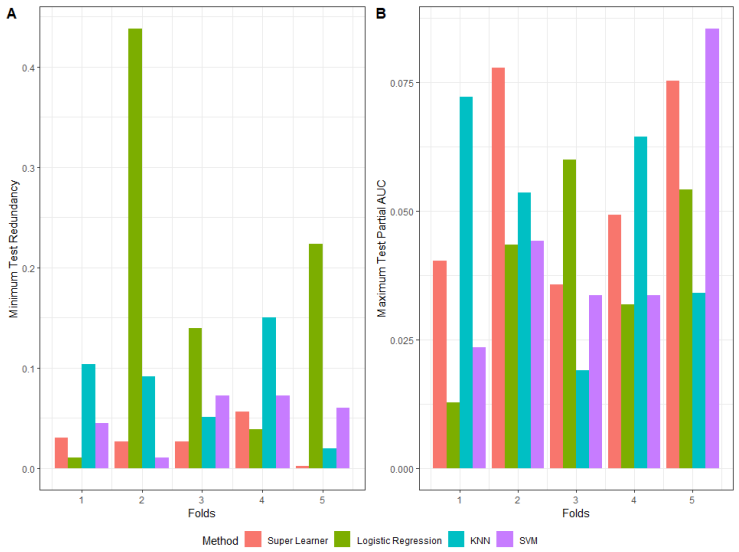

Figure 7: Minimum Test Redundancy (A) and Maximum Test Partial AUC (B) in Non-Dominated Set in each fold by Different Classification Algorithms

The Pareto Fronts for each algorithm in each run is presented in Figure 8 with redundancy less than 10 for the convince of the comparison. To show the non-dominated sets traditionally as a minimization problem in Figure 2, we considered the negative pAUC with the redundancy value of the selected set. All the evaluations are taken on the testing set for each run. The solutions generated by the Super Learner dominated that of other algorithms.

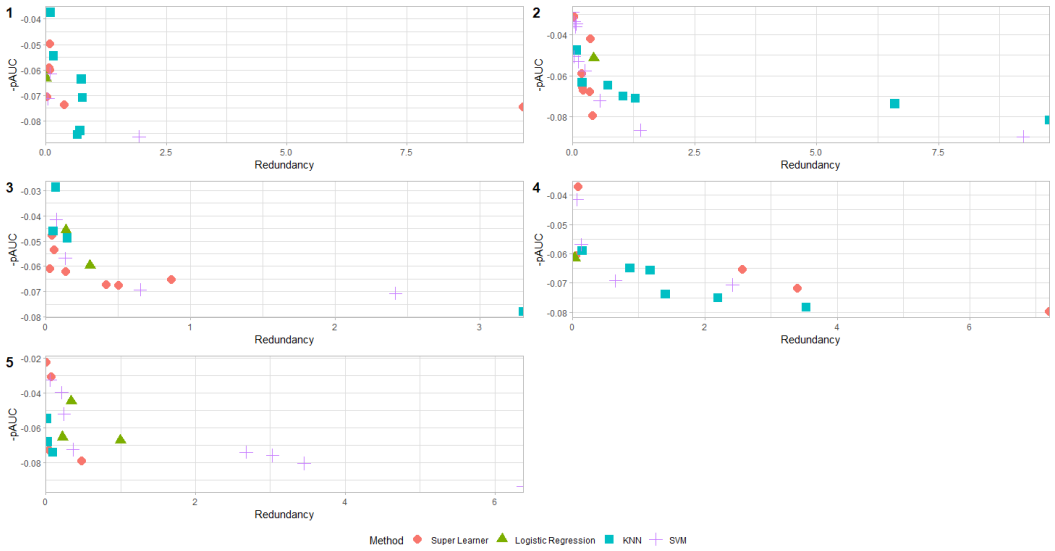

Figure 8: Pareto Fronts in Run 1-5 (1-5) by Different Classification Algorithms

Classifier	Method	#[2]	Redundancy	pAUC	Sensitivity	Specificity
Super Learner	**Baseline**	192	162.54 (1.51)	0.025 (0.020)	0.84 (0.15)	0.16 (0.06)
	MCBPSO	**8 (4)**	**2.70 (3.20)**	**0.029 (0.009)**	0.72 (0.08)	**0.35 (0.04)**
LR[1]	**Baseline**	192	162.54 (1.51)	0.009 (0.004)	1.00 (0.00)	0.00 (0.00)
	MCBPSO	11 (11)	5.49 (9.34)	0.026 (0.009)	0.71 (0.31)	0.28 (0.16)
SVM	**Baseline**	192	162.54 (1.51)	0.018 (0.007)	**0.92 (0.04)**	0.12 (0.02)
	MCBPSO	12 (8)	6.43 (6.51)	0.025 (0.010)	0.73 (0.09)	0.22 (0.06)
KNN	**Baseline**	192	162.54 (1.51)	0.016 (0.007)	0.73 (0.19)	0.33 (0.12)
	MCBPSO	10 (6)	4.56 (4.26)	0.027 (0.008)	0.79 (0.05)	0.23 (0.05)

1 Logistic Regression.
2 Number of Feature Selected.

Table 2: Classification Performance on Testing Sets obtained via Different Classification Methods

There is no single algorithm fitting for all scenarios. Super Learner tunes the weights via cross-validation to combine multiple classifiers to leverage the final performance. Table 2 presents the average classification performance for non-dominated set on testing sets when repeating Algorithm 2 for 5 times independently. The baseline model refers to train the model on all available 192-item features with the same redundancy value as 162.54. Among all baseline models, Super Learner has a higher pAUC of 0.025 than its candidate learners including Logistic Regression, SVM with Radial Kernel, and KNN. MCBPSO improves performance for all classification algorithms, especially for the Super Learner related model. Super Learner selects an average of 8 features with redundancy 2.7, pAUC 0.029, sensitivity 0.72, and specificity 0.35. It obtains a higher partial AUC on the testing set on average but selecting among the same number of features compared with other candidate learners. The standard deviation is a bit large for the average number of features selected and redundancy since the non-dominates set might include solutions with high redundancy but low pAUC comparing with other non-dominated solutions. These solutions also help since they dominate the pAUC metric of other solutions with more or equal redundancy. The non-dominated set will be reviewed by dental experts to select a clinical meaningful solution at the final stage.

A sample of selected items in the non-dominated set for each run is listed in Table 3. They cover physical, mental, and social domains with some demographic background of kids. All 5 runs selected the questions related to reasons that ever keep a child from visiting the dentist. 4 out of 5 runs included language for media from the different sources while primary language and ethnicity also in 3 out of 5 runs. Some questions revoked that a child might have or worry about certain teeth problems, like teeth pain, falling teeth, teeth appearance issues, and gums hurt. The behaviors of oral health care, brushing and flossing, are also essential to decay teeth.

Features	MCBPSO with Super Learner
Did any of the following reasons ever keep you from visiting a dentist?	1,2,3,4,5
In what languages are the TV shows, radio stations, or newspapers that you usually watch, listen to?	1,2,4,5
How much are you afraid to go to a dentist?	1,2,4
Why were you afraid to go to the dentist? I am afraid of feeling sick.	2,4,5
What do you worry about? Pain with my teeth.	1,2,4
What do you worry about? My teeth are falling out.	1,2,4
Do you worry about any problems with your teeth?	1,2,4
Because of the condition of my teeth and mouth, getting a date is difficult.	1,2,4
My gums hurt.	1,2,4
What is your primary language?	1,2,4
People in this neighborhood can be trusted.	1,2,4
If I care for my oral health, I will live longer?	1,2,4
Which of the following do you think caused your pain?	1,2,4
In the past twelve months, have you had any of the following problems? Teeth that hurt when you ate or drank hot or cold liquids or foods?	2,3,4
Do you know what caused the pain?	2,4,5
What is your race/ethnicity?	1,2,4
Brushing my teeth, I can...	1,2,4
Flossing my teeth, I can...	2,4,5
Have you ever avoided laughing or smiling because of the way your teeth look?	2,3,4

Table 3: A List Sample of Features in the Non-dominated Set for Each Run by MCBPSO with Super Learner

5. Discussion

This work provided a multi-objective optimization method for feature selection with the imbalanced labeled data. In this paper, we proposed a novel Multi-objective Constrained BPSO (MCBPSO) algorithm for feature selection. Dual objectives are taken into consideration including minimizing redundancy and maximizing pAUC. The algorithm tended to minimize the redundancy to reduce the complexity of the selected feature subset while trying to increase the pAUC value with a minimum boundary of sensitivity. The 5-fold cross-validated values on objectives guided the feature searching. Non-dominated set preserved the Pareto Fronts on those values in each iteration. All non-dominated solutions are sorted by relative crowding distance to update the global best solution *gbest* and particle best solution *pbest* for PSO algorithm. Mutation and resetting operators are performed to avoid the premature convergence to a sub-optimal solution.

Super Learning with SMOTE sampler for imbalanced labeled data performed better than its candidate learners considerably with higher average pAUC on the non-dominated

set and about the same number of features selected for testing data. It ensembles multiple candidate learners by finding the optimal weights via cross-validation. The candidate learners are Logistic Regression, SVM with Radial Kernel, and KNN, which are widely used for the two-alternative classification problems. It is the first work to apply Super Learner as the wrapper in the feature selection.

The proposed algorithm is applied to a 192-item child-reported survey to predict the active cavity for children ages 8-17 years. It resulted in a non-dominated set with an average of 8 features selected, redundancy 0.27, and partial AUC 0.029, which is much better than the model using all available features. The partial AUC in this application is the area under the ROC curve with a minimum boundary of sensitivity as 0.8. After determining the feature subset, we further traded off sensitivity and specificity by selecting a threshold on training data. The average sensitivity and specificity are 0.72 and 0.35 on the testing data with subsets with 8 items on average. To improve the performance of sensitivity and specificity, we can include more items. Dentists will further review those items and picked a clinical meaningful feature subset based on needs. These findings can serve as the base to inform a more effective survey, so an improved oral health screening tool can be developed with a better combination of sensitivity and specificity to be implemented in large populations of children.

The algorithm could further be tailored to multiple perspectives. For the scenario with the restriction on both range of sensitivity and specificity together, the two-way partial AUC is available [45]. Besides, it may be extended to more objectives or other objectives based on actual needs. However, the increase in the number of objectives will load more burden on searching and the final decision part.

In terms of Swarm Intelligent optimization, the variations of PSO along with some other binary versions of algorithms are also available for feature selection including Artificial Bee Colony Algorithm, Fireflies Algorithm, Glowworm Swarm Optimization, Roach Infestation Optimization, Bat Algorithm, and Grey-wolf Optimization [6]. More work needs to be done to compare different searching algorithms on the variability towards the multi-objective problems for feature selection problems.

References

[1] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[2] Gerard V Trunk. A problem of dimensionality: A simple example. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (3):306–307, 1979.

[3] Thomas Navin Lal, Olivier Chapelle, Jason Weston, and André Elisseeff. Embedded methods. In *Feature extraction*, pages 137–165. Springer, 2006.

[4] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2):273–324, 1997.

[5] Michael R Garey and David S Johnson. *Computers and intractability*, volume 29. wh freeman New York, 2002.

[6] Lucija Brezočnik, Iztok Fister, and Vili Podgorelec. Swarm intelligence algorithms for feature selection: a review. *Applied Sciences*, 8(9):1521, 2018.

[7] Cheng-Lung Huang and Jian-Fan Dun. A distributed pso–svm hybrid system with feature selection and parameter optimization. *Applied soft computing*, 8(4):1381–1391, 2008.

[8] Shih-Wei Lin, Kuo-Ching Ying, Shih-Chieh Chen, and Zne-Jung Lee. Particle swarm optimization for parameter determination and feature selection of support vector machines. *Expert systems with applications*, 35(4):1817–1824, 2008.

[9] Li-Yeh Chuang, Hsueh-Wei Chang, Chung-Jui Tu, and Cheng-Hong Yang. Improved binary pso for feature selection using gene expression data. *Computational Biology and Chemistry*, 32(1):29–38, 2008.

[10] Meng-Chang Tsai, Kun-Huang Chen, Chao-Ton Su, and Hung-Chun Lin. An application of pso algorithm and decision tree for medical problem. In *2nd International Conference on Intelligent Computational Systems (ICS'2012)*, pages 124–126, 2012.

[11] Yudong Zhang, Shuihua Wang, Preetha Phillips, and Genlin Ji. Binary pso with mutation operator for feature selection using decision tree applied to spam detection. *Knowledge-Based Systems*, 64:22–31, 2014.

[12] Li Ma and Suohai Fan. Cure-smote algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests. *BMC bioinformatics*, 18(1):169, 2017.

[13] Bing Xue, Liam Cervante, Lin Shang, Will N Browne, and Mengjie Zhang. A multi-objective particle swarm optimisation for filter-based feature selection in classification problems. *Connection Science*, 24(2-3):91–116, 2012.

[14] Bing Xue, Mengjie Zhang, and Will N Browne. Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE transactions on cybernetics*, 43(6):1656–1671, 2012.

[15] Eric C Polley and Mark J Van Der Laan. Super learner in prediction. 2010.

[16] Mark J Van der Laan, Eric C Polley, and Alan E Hubbard. Super learner. *Statistical applications in genetics and molecular biology*, 6(1), 2007.

[17] Russell Eberhart and James Kennedy. A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. Ieee, 1995.

[18] Weng Kee Wong, Ray-Bing Chen, Chien-Chih Huang, and Weichung Wang. A modified particle swarm optimization technique for finding optimal designs for mixture models. *PloS one*, 10(6):e0124720, 2015.

[19] Na Dong, Chun-Ho Wu, Wai-Hung Ip, Zeng-Qiang Chen, Ching-Yuen Chan, and Kai-Leung Yung. An opposition-based chaotic GA/PSO hybrid algorithm and its application in circle detection. *Computers & Mathematics with Applications*, 64(6):1886–1902, 2012.

[20] H Hannah Inbarani, Ahmad Taher Azar, and G Jothi. Supervised hybrid feature selection based on PSO and rough sets for medical diagnosis. *Computer methods and programs in biomedicine*, 113(1):175–185, 2014.

[21] James Kennedy and Russell C Eberhart. 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. IEEE, 1997.

[22] Margarita Reyes Sierra and Carlos A Coello Coello. Improving pso-based multi-objective optimization using crowding, mutation and-dominance. In *International conference on evolutionary multi-criterion optimization*, pages 505–519. Springer, 2005.

[23] Jorge R Vergara and Pablo A Estévez. A review of feature selection methods based on mutual information. *Neural computing and applications*, 24(1):175–186, 2014.

[24] Emrah Hancer, Bing Xue, Mengjie Zhang, Dervis Karaboga, and Bahriye Akay. A multi-objective artificial bee colony approach to feature selection using fuzzy mutual information. In *2015 IEEE Congress on Evolutionary Computation (CEC)*, pages 2420–2427. IEEE, 2015.

[25] Chong Sun Hong and Beom Jun Kim. Mutual information and redundancy for categorical data. *Statistical Papers*, 52(1):17–31, 2011.

[26] Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 4(1):66–82, 1960.

[27] Milan Studenỳ and Jirina Vejnarová. The multiinformation function as a tool for measuring stochastic dependence. In *Learning in graphical models*, pages 261–297. Springer, 1998.

[28] Patrick Emmanuel Meyer, Colas Schretter, and Gianluca Bontempi. Information-theoretic feature selection in microarray data using variable complementarity. *IEEE Journal of Selected Topics in Signal Processing*, 2(3):261–274, 2008.

[29] James P Egan. *Signal Detection Theory and ROC Analysis Academic Press Series in Cognition and Perception*. London, UK: Academic Press, 1975.

[30] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

[31] Miles N Wernick, Yongyi Yang, Jovan G Brankov, Grigori Yourganov, and Stephen C Strother. Machine learning in medical imaging. *IEEE signal processing magazine*, 27(4):25–38, 2010.

[32] Chintan Parmar, Patrick Grossmann, Johan Bussink, Philippe Lambin, and Hugo JWL Aerts. Machine learning methods for quantitative radiomic biomarkers. *Scientific reports*, 5:13087, 2015.

[33] Zhanfeng Wang and Yuan-Chin Ivan Chang. Marker selection via maximizing the partial area under the roc curve of linear risk scores. *Biostatistics*, 12(2):369–385, 2010.

[34] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[35] Eric Polley, Erin LeDell, Chris Kennedy, Sam Lendle, and Mark van der Laan. Package 'superlearner', 2018.

[36] Jenna Wong, Travis Manderson, Michal Abrahamowicz, David L Buckeridge, and Robyn Tamblyn. Can hyperparameter tuning improve the performance of a super learner?: A case study. *Epidemiology*, 30(4):521–531, 2019.

[37] Susana M Vieira, Luís F Mendonça, Goncalo J Farinha, and João MC Sousa. Modified binary pso for feature selection using svm applied to mortality prediction of septic patients. *Applied Soft Computing*, 13(8):3494–3504, 2013.

[38] Ivan De Falco, Antonio Della Cioppa, and Ernesto Tarantino. Mutation-based genetic algorithm: performance evaluation. *Applied Soft Computing*, 1(4):285–299, 2002.

[39] Carlo R Raquel and Prospero C Naval Jr. An effective use of crowding distance in multiobjective particle swarm optimization. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pages 257–264. ACM, 2005.

[40] Quan Zou, Sifa Xie, Ziyu Lin, Meihong Wu, and Ying Ju. Finding the best classification threshold in imbalanced classification. *Big Data Research*, 5:2–8, 2016.

[41] James Kennedy. Swarm intelligence. In *Handbook of nature-inspired and innovative computing*, pages 187–219. Springer, 2006.

[42] H Kalsbeek and GH Verrips. Consumption of sweet snacks and caries experience of primary school children. *Caries Research*, 28(6):477–483, 1994.

[43] David Cella, Susan Yount, Nan Rothrock, Richard Gershon, Karon Cook, Bryce Reeve, Deborah Ader, James F Fries, Bonnie Bruce, and Mattias Rose. The patient-reported outcomes measurement information system (promis): progress of an nih roadmap cooperative group during its first two years. *Medical care*, 45(5 Suppl 1):S3, 2007.

[44] Honghu Liu, Ron D Hays, Marvin Marcus, Ian Coulter, Carl Maida, Francisco Ramos-Gomez, Jie Shen, Yan Wang, Vladimir Spolsky, Steve Lee, et al. Patient-reported oral health outcome measurement for children and adolescents. *BMC oral health*, 16(1):95, 2016.

[45] Hanfang Yang, Kun Lu, Xiang Lyu, and Feifang Hu. Two-way partial auc and its properties. *Statistical methods in medical research*, 28(1):184–195, 2019.