

Human Assisted Learning by Evolutionary Multi-Objective Optimization

Dan-Xuan Liu¹, Xin Mu², Chao Qian¹

¹State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China

²Peng Cheng Laboratory, Shenzhen 518000, China
{liudx, qianc}@lamda.nju.edu.cn, mux@pcl.ac.cn

Abstract

Machine learning models have liberated manpower greatly in many real-world tasks, but their predictions are still worse than humans on some specific instances. To improve the performance, it is natural to optimize machine learning models to take decisions for most instances while delivering a few tricky instances to humans, resulting in the problem of Human Assisted Learning (HAL). Previous works mainly formulated HAL as a constrained optimization problem that tries to find a limited subset of instances for human decision such that the sum of model and human errors can be minimized; and employed the greedy algorithms, whose performance, however, may be limited due to the greedy nature. In this paper, we propose a new framework HAL-EMO based on Evolutionary Multi-objective Optimization, which reformulates HAL as a bi-objective optimization problem that minimizes the number of selected instances for human decision and the total errors simultaneously, and employs a Multi-Objective Evolutionary Algorithm (MOEA) to solve it. We implement HAL-EMO using two MOEAs, the popular NSGA-II as well as the theoretically grounded GSEMO. We also propose a specific MOEA, called BSEMO, with biased selection and balanced mutation for HAL-EMO, and prove that for human assisted regression and classification, HAL-EMO using BSEMO can achieve better and same theoretical guarantees than previous greedy algorithms, respectively. Experiments on the tasks of medical diagnosis and content moderation show the superiority of HAL-EMO (with either NSGA-II, GSEMO or BSEMO) over previous algorithms, and that using BSEMO leads to the best performance of HAL-EMO.

Introduction

To solve complex real-world problems, one often needs human experts to make decisions, which, however, has several limitations. For example, some decisions with significant consequences require to be made quickly, while the expert resources may be in short supply. Taking medical diagnosis as an example, patients may need to wait for months to be diagnosed by a specialist. Furthermore, the massive decisions to be made may increase the tiredness of experts, affecting the quality of decision-making. Taking content moderation as an example, reviewers of social network platforms need to review a large number of comments every day to check

whether these comments satisfy some requirements (e.g., no racial or sex discrimination), while their fatigue will probably make some non-compliant comments overlooked.

Machine learning models, which are trained from data, can make decisions automatically, and have achieved significant successes, liberating manpower greatly. Though achieving or even exceeding the average performance of human experts, the machine models may still make worse decisions than humans on some instances. HAL is then naturally introduced (Raghu et al. 2019), whose goal is to develop machine learning models that are optimized to take decisions for most instances, and outsource the remaining ones to humans. By outsourcing a small number of tricky instances to human experts for decision-making, HAL can reduce the difficulty of model training, and lead to the performance improvement without using too many expert resources.

HAL is quite different from active learning (Sabato and Munos 2014; Hashemi et al. 2019) and human-machine collaboration (Tschitschek et al. 2019; Kamalaruban et al. 2019). The goal of active learning is to select a valuable subset of instances for human labeling, such that the trained supervised model can have a good generalization ability. Active learning requires the trained model to perform well over the entire instance space, while HAL only requires that it can perform well on those instances similar to that assigned to the machine during training, while delivering the other instances to human experts. Human-machine collaboration focuses more on the interaction between machine and human, where the model training may need the help of humans, while HAL is mainly to decide which instance is suitable for model prediction and which is suitable for human prediction.

Most previous works usually develop a heuristic policy for deciding which instance should be outsourced to humans (Raghu et al. 2019; Wilder, Horvitz, and Kamar 2020; Mozannar and Sontag 2020; Bordt and von Luxburg 2020). For example, Raghu et al. (2019) proposed two algorithms for human assisted classification, i.e., triage based on algorithmic uncertainty and predicted error. The former trains the classification model on the whole training set and outsources to humans the top k test instances with the highest classification uncertainty of the model. The latter also trains the classification model on the whole training set, but outsources to humans the top k test instances with the highest difference between the prediction of models and humans.

Recently, some HAL algorithms with theoretical guarantees have been proposed (De et al. 2020, 2021). HAL is formulated as a constrained optimization problem that tries to find a limited subset of instances for human decisions such that the sum of model and human errors can be minimized. For human assisted ridge regression, where the model is ridge regression, De et al. (2020) proved that the objective function (i.e., the sum of model and human errors) satisfies the α -approximately submodular property, and then applied the greedy algorithm (Gatmiry and Gomez-Rodriguez 2018) achieving an $(1 + \frac{1}{1-\alpha})^{-1}$ -approximation ratio, where $0 \leq \alpha \leq 1$. For human assisted Support Vector Machine (SVM), De et al. (2021) showed that the objective function can be rewritten as the difference of a monotone γ -approximately submodular function g and a modular function c , and then applied the distorted greedy algorithm as well as its stochastic variant (Harshaw et al. 2019) that can achieve an approximation guarantee of $(1 - e^{-\gamma}) \cdot g(X_{\text{opt}}) - c(X_{\text{opt}})$, where $0 \leq \gamma \leq 1$ and X_{opt} denotes an optimal solution.

The above-mentioned algorithms with theoretical guarantees mainly employ the greedy procedure for optimization, whose performance, however, may be limited due to the greedy nature. In this paper, we propose a new framework based on Evolutionary Multi-objective Optimization (Knowles, Watson, and Corne 2001; Friedrich and Neumann 2015; Qian, Yu, and Zhou 2015), briefly called HAL-EMO, which reformulates HAL as a bi-objective optimization problem that minimizes the size of the selected subset of instances for human decision and an error-related objective simultaneously. That is, HAL-EMO tries to optimize the performance of the human assisted model while requiring as few human resources as possible. HAL-EMO can be equipped with any MOEA to solve this bi-objective problem, and we employ the popular NSGA-II (Deb et al. 2002) as well as the theoretically grounded GSEMO (Laumanns, Thiele, and Zitzler 2004). Empirical results on the tasks of medical diagnosis and content moderation show that using either NSGA-II or GSEMO, HAL-EMO performs better than previous algorithms.

To further improve the performance of HAL-EMO, we design a specific MOEA called BSEMO, employing a biased selection strategy and a balanced mutation operator. For human assisted ridge regression, we prove that HAL-EMO using BSEMO achieves an approximation ratio of $1 - e^{-(1-\alpha)}$, which is better than $(1 + \frac{1}{1-\alpha})^{-1}$ of the greedy algorithm (De et al. 2020). For human assisted SVM, it achieves the same approximation guarantee of $(1 - e^{-\gamma}) \cdot g(X_{\text{opt}}) - c(X_{\text{opt}})$ as the distorted greedy algorithm (De et al. 2021). Experiments on medical diagnosis and content moderation show that using BSEMO can lead to the best performance of HAL-EMO.

Human Assisted Learning

Let $V = \{v_1, v_2, \dots, v_n\}$ denote the training data set, where $v_i = (\mathbf{x}_i, y_i)$, \mathbf{x}_i is the i -th instance and y_i is the corresponding target, which can be continuous or take a finite number of categories, corresponding to regression or classification. We use $c(\mathbf{x}_i, y_i)$ to denote the human error on the instance (\mathbf{x}_i, y_i) , and $\ell(\mathbf{w}, \mathbf{x}_i, y_i)$ to denote the error of the machine

learning model with the parameter vector \mathbf{w} .

As presented in Definition 1, HAL is to select a limited subset X of instances for human decision and deliver the remaining ones for a machine learning model, such that the sum of human and machine learning model errors can be minimized. The human error on X is represented as $\sum_{(\mathbf{x}_i, y_i) \in X} c(\mathbf{x}_i, y_i)$. The error of the machine learning model with a specific parameter vector \mathbf{w} on the remaining instances is represented as $\sum_{(\mathbf{x}_i, y_i) \in V \setminus X} \ell(\mathbf{w}, \mathbf{x}_i, y_i)$. Note that the machine learning model can be optimized simultaneously by adjusting \mathbf{w} , and thus its error is actually $\min_{\mathbf{w}} \sum_{(\mathbf{x}_i, y_i) \in V \setminus X} \ell(\mathbf{w}, \mathbf{x}_i, y_i)$.

Definition 1 (Human Assisted Learning). *Given a training data set V , a human error function c , an error function ℓ of machine learning model (whose parameter vector is denoted as \mathbf{w}), and a budget k , to select a subset $X \subseteq V$ such that*

$$\arg \min_{X \subseteq V} \left(\sum_{(\mathbf{x}_i, y_i) \in X} c(\mathbf{x}_i, y_i) + \min_{\mathbf{w}} \sum_{(\mathbf{x}_i, y_i) \in V \setminus X} \ell(\mathbf{w}, \mathbf{x}_i, y_i) \right) \\ \text{s.t.} \quad |X| \leq k.$$

Next, we will introduce two specific HAL problems, human assisted ridge regression and SVM, where the machine learning models are ridge regression and SVM, respectively.

Human Assisted Ridge Regression

When the machine learning model is ridge regression, the model error $\ell(\mathbf{w}, \mathbf{x}_i, y_i)$ w.r.t. a specific parameter vector \mathbf{w} can be represented by $(y_i - \mathbf{x}_i^T \mathbf{w})^2 + \lambda \cdot \|\mathbf{w}\|_2^2$, which has incorporated the regularization term in model training. Let $X^c = V \setminus X$ denote the complement of X in V . Let \mathbf{y}_{X^c} denote the subvector of $\mathbf{y} = [y_1, \dots, y_n]^T$ indexed by X^c , and \mathbf{X}_{X^c} denote the submatrix formed by the columns of $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ indexed by X^c . The model error on X^c , i.e., $\sum_{(\mathbf{x}_i, y_i) \in X^c} ((y_i - \mathbf{x}_i^T \mathbf{w})^2 + \lambda \cdot \|\mathbf{w}\|_2^2)$, can be minimized by setting the parameter vector to $(\lambda |X^c| \mathbf{I} + \mathbf{X}_{X^c} \mathbf{X}_{X^c}^T)^{-1} \mathbf{X}_{X^c} \mathbf{y}_{X^c}$, where \mathbf{I} is an identity matrix. Thus, the total error (i.e., the sum of human and model errors) of a subset $X \subseteq V$ in Definition 1 can be represented by

$$\text{error}(X) = \sum_{(\mathbf{x}_i, y_i) \in X} c(\mathbf{x}_i, y_i) + \mathbf{y}_{X^c}^T \mathbf{y}_{X^c} \\ - \mathbf{y}_{X^c}^T \mathbf{X}_{X^c}^T \cdot (\lambda |X^c| \mathbf{I} + \mathbf{X}_{X^c} \mathbf{X}_{X^c}^T)^{-1} \mathbf{X}_{X^c} \mathbf{y}_{X^c}.$$

By equivalently maximizing $-\log(\text{error}(X))$, human assisted ridge regression can be defined as follows.

Definition 2 (Human Assisted Ridge Regression (De et al. 2020)). *Given a training data set V , a human error function c , and a budget k , to select a subset $X \subseteq V$ such that*

$$\arg \max_{X \subseteq V} -\log(\text{error}(X)) \quad \text{s.t.} \quad |X| \leq k. \quad (1)$$

Human Assisted Support Vector Machine

When the machine learning model is SVM with parameters \mathbf{w} and b , the model error $\ell(\mathbf{w}, b, \mathbf{x}_i, y_i)$ can be represented by $[1 - y_i(\mathbf{w}^T \Phi(\mathbf{x}_i) + b)]_+ + \lambda \|\mathbf{w}\|_2^2$, where $\Phi(\cdot)$ denotes a given feature transformation, $[\cdot]_+ = \max\{0, \cdot\}$ denotes the hinge loss function, and λ is a given regularization parameter. Given a subset X , the optimal parameters (denoted as

$w^*(X^c)$ and $b^*(X^c)$) for minimizing the model error on X^c , i.e., $\sum_{(\mathbf{x}_i, y_i) \in X^c} ([1 - y_i(w^T \Phi(\mathbf{x}_i) + b)]_+ + \lambda \|w\|_2^2)$, can be found in polynomial time due to the convexity.

Assume that the human error on the i -th instance (\mathbf{x}_i, y_i) can be represented by $[1 - y_i h(\mathbf{x}_i)]_+$, where $h(\cdot) \in [-H, H]$ is the (normalized) score provided by human, and $H > 0$ is a constant. Given a subset $X \subseteq V$, by applying the optimal parameters $w^*(X^c)$ and $b^*(X^c)$, the total error in Definition 1 can be represented by

$$\begin{aligned} \text{error}(X) &= \sum_{(\mathbf{x}_i, y_i) \in X} [1 - y_i h(\mathbf{x}_i)]_+ + \lambda \|w^*(X^c)\|_2^2 \cdot |X^c| \\ &\quad + \sum_{(\mathbf{x}_i, y_i) \in X^c} [1 - y_i(w^*(X^c)^T \Phi(\mathbf{x}_i) + b^*(X^c))]_+. \\ \text{Let } c(X) &= \sum_{(\mathbf{x}_i, y_i) \in X} [1 - y_i h(\mathbf{x}_i)]_+ \text{ and} \\ g(X) &= c(X) - \text{error}(X) + \lambda \|w^*(V)\|_2^2 \cdot |V| \\ &\quad + \sum_{(\mathbf{x}_i, y_i) \in V} [1 - y_i(w^*(V)^T \Phi(\mathbf{x}_i) + b^*(V))]_+, \end{aligned}$$

where $w^*(V)$ and $b^*(V)$ correspond to the optimal parameters when $X^c = V$, i.e., all instances in V are used for model training. As $\lambda \|w^*(V)\|_2^2 \cdot |V| + \sum_{(\mathbf{x}_i, y_i) \in V} [1 - y_i(w^*(V)^T \Phi(\mathbf{x}_i) + b^*(V))]_+$ is a constant w.r.t. X , minimizing $\text{error}(X)$ is equivalently maximizing $g(X) - c(X)$. Human assisted SVM then can be defined as follows.

Definition 3 (Human Assisted SVM (De et al. 2021)). *Given a training data set V , a human score function h , and a budget k , to select a subset $X \subseteq V$ such that*

$$\arg \max_{X \subseteq V} g(X) - c(X) \quad \text{s.t.} \quad |X| \leq k. \quad (2)$$

Approximate Submodularity

We have shown that HAL can be formulated as optimizing a set function (e.g., $-\log(\text{error}(X))$ or $g(X) - c(X)$) under the size constraint $|X| \leq k$. Here we introduce some properties of set functions, which will be used in our analysis.

Let \mathbb{R} denote the set of reals. Given a ground set $V = \{v_1, v_2, \dots, v_n\}$, a set function $f : 2^V \rightarrow \mathbb{R}$ maps any subset of V to a real value. A set function f is monotone if $\forall X \subseteq Y : f(X) \leq f(Y)$, and is submodular (Nemhauser, Wolsey, and Fisher 1978) if $\forall X \subseteq Y, v \notin Y$,

$$f(X \cup \{v\}) - f(X) \geq f(Y \cup \{v\}) - f(Y); \quad (3)$$

or equivalently $\forall X \subseteq Y \subseteq V$,

$$f(Y) - f(X) \leq \sum_{v \in Y \setminus X} (f(X \cup \{v\}) - f(X)). \quad (4)$$

Eq. (3) intuitively represents the diminishing returns property, i.e., the benefit of adding an item to a set will not increase as the set extends. A set function f is modular if Eq. (3) or Eq. (4) holds with equality. For a modular function f , it holds that $\forall X \subseteq V : f(X) = \sum_{v \in X} f(\{v\})$.

For a general set function $f : 2^V \rightarrow \mathbb{R}$, several notions of approximate submodularity have been introduced to measure to what extent f has the submodular property. The α and γ -approximate submodularity in Definitions 4 and 5 are actually defined based on Eqs. (3) and (4), respectively. For a monotone set function f , we have $\alpha, \gamma \in [0, 1]$, and f is submodular if $\alpha = 0$ or $\gamma = 1$.

Definition 4 (α -Approximately Submodular (Zhang and Vorobeychik 2016; Qian et al. 2017a)). *A set function f is α -approximately submodular if for all $X \subseteq Y \subseteq V$ and $v \in V$, $f(X \cup \{v\}) - f(X) \geq (1 - \alpha) \cdot (f(Y \cup \{v\}) - f(Y))$.*

Definition 5 (γ -Approximately Submodular (Das and Kempe 2011)). *A set function f is γ -approximately submodular if for all $X, Y \subseteq V$, $\sum_{v \in Y \setminus X} (f(X \cup \{v\}) - f(X)) \geq \gamma \cdot (f(X \cup Y) - f(X))$.*

Previous Algorithms

For human assisted ridge regression in Definition 2, De et al. (2020) proved that the objective $-\log(\text{error}(X))$ is monotone and α -approximately submodular, and applied the greedy algorithm, which can achieve an $(1 + \frac{1}{1-\alpha})^{-1}$ -approximation ratio (Gatmiry and Gomez-Rodriguez 2018). The greedy algorithm starts from the empty set, and iteratively selects one instance with the largest marginal gain on the objective function until k instances have been selected.

For human assisted SVM in Definition 3, De et al. (2021) proved that the function $g(X)$ is monotone and γ -approximately submodular, and $c(X)$ is modular. They applied the distorted greedy algorithm, which can achieve an approximation guarantee of $(1 - e^{-\gamma}) \cdot g(X_{\text{opt}}) - c(X_{\text{opt}})$ (Harshaw et al. 2019), where X_{opt} denotes an optimal solution. Let X_i denote the subset generated after i iterations. In the $(i + 1)$ -th iteration, rather than maximizing the marginal gain on the original objective function $g - c$, i.e., $(g(X_i \cup \{v\}) - g(X_i)) - c(\{v\})$, it maximizes a distorted one, $(1 - \gamma/k)^{k-(i+1)} (g(X_i \cup \{v\}) - g(X_i)) - c(\{v\})$, which gradually increases the importance of g .

HAL-EMO Framework

Inspired by the excellent performance of MOEAs for solving general subset selection problems (Friedrich and Neumann 2015; Qian, Yu, and Zhou 2015; Qian et al. 2017b, 2019; Roostapour et al. 2022), we propose a new HAL framework based on Evolutionary Multi-objective Optimization, called HAL-EMO. A subset X of V can be naturally represented by a Boolean vector $\mathbf{x} \in \{0, 1\}^n$, where the i -th bit $x_i = 1$ iff the i -th instance in V is contained by X . We will not distinguish $\mathbf{x} \in \{0, 1\}^n$ and its corresponding subset $\{v_i \in V \mid x_i = 1\}$ for notational convenience.

As presented in Algorithm 1, HAL-EMO first reformulates the original HAL problem in Definition 1 as a bi-objective maximization problem

$$\arg \max_{\mathbf{x} \in \{0, 1\}^n} (f_1(\mathbf{x}), f_2(\mathbf{x})), \quad (5)$$

where $f_1(\mathbf{x})$ is related to the original objective function (reflecting the human and model error given \mathbf{x}), and $f_2(\mathbf{x}) = -|\mathbf{x}| = -\sum_{i=1}^n x_i$ is the opposite of the subset size. That is, HAL-EMO tries to optimize an error-related objective and minimize the subset size simultaneously. The setting of f_1 depends on the concrete HAL problem. For human assisted ridge regression in Definition 2, we will use

$$f_1(\mathbf{x}) = -\log(\text{error}(\mathbf{x})). \quad (6)$$

For human assisted SVM in Definition 3, we will use

$$f_1(\mathbf{x}) = (1 - \gamma/k)^{k-|\mathbf{x}|} g(\mathbf{x}) - c(\mathbf{x}) + (|\mathbf{x}|/k) \cdot c(\mathbf{1}), \quad (7)$$

Algorithm 1: HAL-EMO Framework

Input: HAL problem with training set V and budget k **Output:** a subset of V with size at most k **Process:**

- 1: Construct two objective functions $f_1(x)$ and $f_2(x)$ to be maximized, where $f_1(x)$ is related to the objective function of the given HAL problem, and $f_2(x) = -|x|$;
 - 2: Apply an MOEA to solve the bi-objective problem;
 - 3: **return** the best feasible solution in the final population generated by the MOEA
-

where γ is the approximately submodular degree of g , and $\mathbf{1}$ denotes the all-1s vector, i.e., the whole set V . As the two objectives may be conflicting, the domination relationship in Definition 6 is often used for comparing two solutions.

Definition 6 (Domination). For two solutions x and x' ,

1. x weakly dominates x' (i.e., x is better than x' , denoted by $x \succeq x'$) if $\forall i : f_i(x) \geq f_i(x')$;
2. x dominates x' (i.e., x is strictly better than x' , denoted by $x \succ x'$) if $x \succeq x' \wedge \exists i : f_i(x) > f_i(x')$;
3. x and x' are incomparable if neither $x \succeq x'$ nor $x' \succeq x$.

After constructing the bi-objective problem in Eq. (5), HAL-EMO employs an MOEA to solve it, as shown in line 2 of Algorithm 1. EAs, inspired by Darwin's theory of evolution, are general-purpose randomized heuristic optimization algorithms (Bäck 1996), mimicking variational reproduction and natural selection. Starting from an initial population of solutions, EAs iteratively reproduce offspring solutions by crossover and mutation, and select better ones from the parent and offspring solutions to form the next population. The population-based search of EAs matches the requirement of multi-objective optimization, i.e., EAs can generate a set of Pareto optimal solutions by running only once. Thus, EAs have become the most popular tool for multi-objective optimization (Coello et al. 2007; Hong, Yang, and Tang 2021), and the corresponding algorithms are also called MOEAs.

During the evolutionary process of the employed MOEA, the infeasible solutions with size larger than k are excluded. After running a number of iterations, the best solution w.r.t. the original HAL problem will be selected from the final population as the output, as shown in line 3 of Algorithm 1. For human assisted ridge regression in Definition 2, it will return $\arg \max_{x \in P, |x| \leq k} -\log(\text{error}(x))$. For human assisted SVM in Definition 3, it will return $\arg \max_{x \in P, |x| \leq k} g(x) - c(x)$.

After getting the subset X for human decision by HAL-EMO, the training phase of HAL is finished. But before testing, we need to train an additional binary classification model to decide whether a test instance is assigned to the human or machine model. The corresponding training data is constructed by assigning a label d_i to each training instance x_i in V , where $d_i = 1$ if $x_i \in X$, and $d_i = -1$ otherwise.

Note that HAL-EMO can be equipped with any MOEA. In this paper, we will use NSGA-II (Deb et al. 2002) and GSEMO (Giel 2003; Laumanns, Thiele, and Zitzler 2004; Neumann and Wegener 2006). NSGA-II may be the most popular MOEA in practice, which employs binary tour-

Algorithm 2: HAL-BSEMO Algorithm

Input: HAL problem with training set V and budget k **Output:** a subset of V with size at most k **Process:**

- 1: Construct two objective functions $f_1(x)$ and $f_2(x)$ to be maximized, where $f_1(x)$ is related to the objective function of the given HAL problem, and $f_2(x) = -|x|$;
 - 2: Let $P \leftarrow \{\mathbf{0}\}$;
 - 3: **repeat**
 - 4: Choose x from P with prob. $\frac{|x|+1}{\sum_{z \in P} (|z|+1)}$;
 - 5: **if** $|x| = k$ **then**
 - 6: Create x' by choosing a 1-bit and a 0-bit of x uniformly at random and swapping them
 - 7: **else**
 - 8: Create x' by flipping each bit of x with prob. $1/n$
 - 9: **if** $\nexists z \in P$ such that $z \succ x'$ **then**
 - 10: $P \leftarrow (P \setminus \{z \in P \mid x' \succeq z\}) \cup \{x'\}$
 - 11: **until** some criterion is met
 - 12: **return** the best feasible solution in P
-

nament selection, crossover and mutation to generate offspring solutions, and updates the population based on non-dominated sorting and crowding distance. GSEMO is relatively simple, but has shown good theoretical properties in solving many problems (Neumann and Witt 2010; Zhou, Yu, and Qian 2019; Doerr and Neumann 2020). It uses uniform selection and bit-wise mutation to generate an offspring solution in each iteration and keeps the non-dominated solutions generated-so-far in the population. Our experimental results will show the advantage of HAL-EMO using NSGA-II or GSEMO (briefly called HAL-NSGA-II and HAL-GSEMO, respectively) over previous algorithms.

We also design a specific MOEA for HAL-EMO, which is modified from GSEMO by employing Biased selection and incorporating Balanced mutation, briefly called BSEMO. HAL-EMO using BSEMO is called HAL-BSEMO, as presented in Algorithm 2. It starts from the all-0s vector $\mathbf{0}$ (i.e., the empty set) in line 2, and iteratively improves the quality of solutions in the population P (lines 3–11). In each iteration, it first selects a parent solution x from the current population P with probability $\frac{|x|+1}{\sum_{z \in P} (|z|+1)}$ in line 4, which increases with the size of x . Thus, the selection is biased instead of uniform, preferring the solutions with larger sizes. As a solution with a larger size will probably have a better objective value, this biased selection strategy may accelerate the optimization. The bit-wise mutation operator in line 8 flips each bit of x with probability $1/n$ to generate an offspring solution x' . But when the selected parent solution x reaches size k , it will probably generate an infeasible offspring solution with size larger than k . To improve the efficiency, we employ a balanced mutation operator in line 6 when $|x| = k$, which chooses a 1-bit and a 0-bit of x uniformly at random and swaps them, preserving the size k of the generated offspring solution x' . Then, x' is used to update the population P (lines 9–10). If x' is not dominated by any solution in P (line 9), it will be added into P , and

meanwhile those solutions weakly dominated by \mathbf{x}' will be deleted (line 10). This updating procedure makes the population P always contain incomparable solutions. We will show that HAL-BSEMO can achieve good theoretical guarantees in the next section, and perform better than HAL-NSGA-II and HAL-GSEMO in the experiments.

Theoretical Analysis

We prove that for human assisted ridge regression and SVM, HAL-BSEMO can achieve better and same theoretical guarantees than previous greedy algorithms, respectively.

Human Assisted Ridge Regression

For human assisted ridge regression in Definition 2, the objective function $-\log(\text{error}(\mathbf{x}))$ has been proved to be monotone and α -approximately submodular (De et al. 2020). By maximizing $f_1(\mathbf{x}) = -\log(\text{error}(\mathbf{x}))$ in Eq. (6) and $f_2(\mathbf{x}) = -|\mathbf{x}|$ simultaneously, HAL-BSEMO can achieve an approximation ratio of $1 - e^{-(1-\alpha)}$ after running $O(nk^3)$ expected number of iterations, as shown in Theorem 1. This is better (i.e., larger)* than that of the greedy algorithm, which is $(1 + \frac{1}{1-\alpha})^{-1}$ (De et al. 2020). Note that OPT denotes the optimal function value of Eq. (1).

Theorem 1. *For human assisted ridge regression in Definition 2, the expected number of iterations of HAL-BSEMO, until finding a subset $X \subseteq V$ with $|X| \leq k$ and $-\log(\text{error}(X)) \geq (1 - e^{-(1-\alpha)}) \cdot \text{OPT}$, is $O(nk^3)$.*

The proof relies on Lemma 1, which shows that when f is monotone and α -approximately submodular, adding a specific item into a subset X can bring an improvement on $f(X)$ proportional to the current distance to the optimum.

Lemma 1. *Let f be a monotone and α -approximately submodular function. For any $X \subseteq V$ with $|X| < k$, there exists one item $v \notin X$ such that $f(X \cup \{v\}) - f(X) \geq ((1 - \alpha)/k) \cdot (\text{OPT} - f(X))$.*

Proof. Let $O = \{o_1, \dots, o_{|O|}\}$ denote an optimal solution. $\forall X \subseteq V, f(X \cup O) - f(X) = \sum_{i=1}^{|O|} (f(X \cup \{o_1, \dots, o_i\}) - f(X \cup \{o_1, \dots, o_{i-1}\})) \leq \sum_{i=1}^{|O|} (f(X \cup \{o_i\}) - f(X)) / (1 - \alpha)$, where the inequality holds by the α -submodularity of f in Definition 4 and $X \subseteq X \cup \{o_1, \dots, o_{i-1}\}$. Let $v^* \in \arg \max_{v \in O} f(X \cup \{v\})$. Because $|O| \leq k$ and $f(X \cup \{v^*\}) \geq f(X)$ due to the monotonicity of f , we have $f(X \cup \{v^*\}) - f(X) \geq ((1 - \alpha)/k) \cdot (f(X \cup O) - f(X))$. As $f(X \cup O) \geq f(O) = \text{OPT}$, the lemma holds. \square

Proof of Theorem 1. We define a quantity J_{\max} as

$$J_{\max} = \max\{j \in \{0, 1, \dots, k\} \mid \exists \mathbf{x} \in P : |\mathbf{x}| \leq j \wedge f_1(\mathbf{x}) \geq (1 - (1 - (1 - \alpha)/k)^j) \cdot \text{OPT}\}.$$

It can be seen that $J_{\max} = k$ implies that there exists one subset \mathbf{x} in P satisfying that $|\mathbf{x}| \leq k$ and $f_1(\mathbf{x}) \geq (1 - (1 - (1 - \alpha)/k)^k) \cdot \text{OPT} \geq (1 - e^{-(1-\alpha)}) \cdot \text{OPT}$, i.e., the desired approximation guarantee is reached.

* $1 - e^{-(1-\alpha)} \geq 1 - 1/(1 + 1 - \alpha) = (1 + 1/(1 - \alpha))^{-1}$, where the inequality holds by $e^x \geq 1 + x$.

Next, we only need to analyze the expected number of iterations until $J_{\max} = k$. The initial value of J_{\max} is 0. Assume that currently $J_{\max} = i < k$. Let \mathbf{x} be a corresponding solution with the value i , i.e., $|\mathbf{x}| \leq i$ and $f_1(\mathbf{x}) \geq (1 - (1 - (1 - \alpha)/k)^i) \cdot \text{OPT}$. First, J_{\max} will not decrease. If \mathbf{x} is deleted from P in line 10 of Algorithm 2, the newly included solution \mathbf{x}' must weakly dominate \mathbf{x} , implying that $|\mathbf{x}'| \leq |\mathbf{x}|$ and $f_1(\mathbf{x}') \geq f_1(\mathbf{x})$. Because $f_1(\mathbf{x}) = -\log(\text{error}(\mathbf{x}))$ is monotone and α -approximately submodular, we know from Lemma 1 that flipping one specific 0 bit of \mathbf{x} (i.e., adding a specific item into \mathbf{x}) can generate a new solution \mathbf{x}' , satisfying $f_1(\mathbf{x}') - f_1(\mathbf{x}) \geq \frac{1-\alpha}{k}(\text{OPT} - f_1(\mathbf{x}))$. Then, we have

$$f_1(\mathbf{x}') \geq (1 - (1 - (1 - \alpha)/k)^{i+1}) \cdot \text{OPT}.$$

Since $|\mathbf{x}'| = |\mathbf{x}| + 1 \leq i + 1$, \mathbf{x}' will be included into P . Otherwise, by line 9 of Algorithm 2, \mathbf{x}' must be dominated by one solution in P , implying $J_{\max} > i$ and thus leading to a contradiction. After including \mathbf{x}' , $J_{\max} \geq i + 1$.

Now we analyze the expected number of iterations required to increase J_{\max} . We consider such an event in one iteration of Algorithm 2: \mathbf{x} is selected in line 4, and only one specific 0-bit corresponding to the item v in Lemma 1 is flipped in line 8 to generate \mathbf{x}' . This event occurs with prob. $\frac{|\mathbf{x}|+1}{\sum_{\mathbf{z} \in P} (|\mathbf{z}|+1)} \cdot (1/n)(1 - 1/n)^{n-1}$, where $\frac{|\mathbf{x}|+1}{\sum_{\mathbf{z} \in P} (|\mathbf{z}|+1)}$ is the prob. of selecting \mathbf{x} in line 4, and $(1/n)(1 - 1/n)^{n-1}$ is the prob. of flipping a specific bit of \mathbf{x} while keeping the other bits unchanged in line 8. Note that $|\mathbf{x}| \leq i < k$, and thus bit-wise mutation in line 8 instead of biased mutation in line 6 is performed. As the solutions in the population P are incomparable and $f_2(\mathbf{x}) = -|\mathbf{x}|$, P contains at most one solution for each subset size $0, 1, \dots, k$. Note that the infeasible solutions with size larger than k , are excluded during the evolutionary process. Thus, the successful event occurs with prob. at least $\frac{1}{\sum_{i=0}^k (i+1)} \cdot (1/n)(1 - 1/n)^{n-1} \geq \frac{2}{en(k+1)(k+2)}$, implying at most $en(k+1)(k+2)/2$ iterations in expectation to increase J_{\max} . After at most $k \cdot en(k+1)(k+2)/2$ expected number of iterations, J_{\max} must have reached k . Hence, the required number of iterations is $O(nk^3)$ in expectation. \square

Human Assisted Support Vector Machine

For human assisted SVM in Definition 3, the functions $g(X)$ and $c(X)$ have been proved to be monotone γ -approximately submodular and modular, respectively (De et al. 2021). By maximizing $f_1(\mathbf{x})$ in Eq. (7) and $f_2(\mathbf{x}) = -|\mathbf{x}|$ simultaneously, HAL-BSEMO achieves an approximation guarantee of $(1 - e^{-\gamma}) \cdot g(X_{\text{opt}}) - c(X_{\text{opt}})$ after running $O(nk^3)$ iterations in expectation, as shown in Theorem 2. This is as good as that of the distorted greedy algorithm (De et al. 2021). Note that X_{opt} denotes an optimal solution of Eq. (2). Theorem 2 can be proved by following the proof of Theorem 1 in (Qian 2021), which analyzes GSEMO. The differences are only that HAL-BSEMO starts from the empty set $\mathbf{0}$ directly instead of a random solution; and the probability of selecting a specific solution for mutation by HAL-BSEMO is at least $2/((k+1)(k+2))$ due to biased selection (as shown in the proof of Theorem 1), while is $1/|P| \geq 1/(n+1)$ by GSEMO due to uniform selection.

Theorem 2. For human assisted SVM in Definition 3, the expected number of iterations of HAL-BSEMO, until finding a subset $X \subseteq V$ with $|X| \leq k$ and $g(X) - c(X) \geq (1 - e^{-\gamma}) \cdot g(X_{\text{opt}}) - c(X_{\text{opt}})$, is $O(nk^3)$.

Empirical Study

In this section, we empirically examine the performance of HAL-EMO, by comparing its three variants (i.e., HAL-NSGA-II[†], HAL-GSEMO[‡] and HAL-BSEMO) with competitive baselines on human assisted ridge regression and SVM. As HAL-EMO is an anytime algorithm, we set the number of objective evaluations to $40kn$, to make a trade-off between the performance and runtime. Also it is randomized, and thus we run it ten times independently and report the average objective value on the training set as well as the average metrics on the test set, corresponding to the optimization and generalization performance, respectively. After getting the subset for human decision in the training phase, we use logistic regression to build the model deciding whether a test instance is delivered to human.

The experiments are mainly to answer two questions: Whether any variant of HAL-EMO is better than previous algorithms? Among the implemented three variants of HAL-EMO, whether HAL-BSEMO performs the best?

Human Assisted Ridge Regression

We use two competitive baselines, i.e., the iterative heuristic algorithm DS (Iyer and Bilmes 2012) and the greedy algorithm (De et al. 2021), and consider the tasks of medical diagnosis and content moderation. The three real-world data sets Messidor (Decencière et al. 2014), Stare-H and Stare-D (Hoover, Kouznetsova, and Goldbaum 2000) are used for medical diagnosis, all containing about 400 eye images. The response variable y of each instance (i.e., image) is the score given by an expert, measuring the severity of edema, retinal hemorrhage or Drusen disease. The data Hatespeech (Davidson et al. 2017) for content moderation contains about 25000 tweets, and y is the average score of several experts measuring the severity of hate speech. All instances are pre-processed as (De et al. 2020). The human prediction s is sampled from a categorical distribution $Cat(\mathbf{p}_{x,y})$, where $\mathbf{p}_{x,y} \sim \text{Dirichlet}(\mathbf{q}_{x,y})$ are the probabilities of each potential score s for an instance (x, y) , and $\mathbf{q}_{x,y}$ is a parameter vector ensuring that the probability of $s = y$ is the largest. $c(x, y) = \mathbb{E}[(y - s)^2]$ is used to measure the human error. The regularization parameter λ is set to 1 for Messidor and Stare-D, 0.5 for Stare-H and 0.01 for Hatespeech. We use 80% of the instances for training and the rest for testing.

The results by varying the budget k from 0 to $0.2 \cdot |V|$ are shown in Figure 1. We can see from the upper subfigures that HAL-NSGA-II, HAL-GSEMO and HAL-BSEMO all surpass the DS and greedy algorithms, showing the superiority of the HAL-EMO framework. This may be because

[†]The population size is set to 100; the initial population consists of the all-0s vector $\mathbf{0}$ and 99 randomly generated solutions; one-point crossover is performed in each iteration with probability 0.9.

[‡]The initial solution is set to $\mathbf{0}$.

HAL-EMO naturally maintains a population of diverse solutions due to the bi-objective transformation, and the employed bit-wise mutation operator has a good global search ability. These characteristics can lead to a better ability of escaping from local optima. Among the three variants of HAL-EMO, HAL-BSEMO always achieves the largest objective value on the training set. Note that on Messidor and Hate-speech, the curves of these three variants are overlapped. The lower subfigures show the mean squared error (MSE) on the test set. HAL-NSGA-II, HAL-GSEMO and HAL-BSEMO achieve lower values than the baselines in most cases, and HAL-BSEMO achieves the lowest value except for $k/|V| = 0.05$ on Stare-H and 0.1 on Stare-D. The MSE results also imply that the best optimization does not always lead to the best generalization, which is expected due to overfitting. As $k/|V|$ increases, the objective value gets larger while the MSE gets smaller, which is because delivering more tricky instances to human will bring improvement.

Compared with GSEMO using uniform selection and bit-wise mutation, BSEMO employs biased selection and balanced mutation. To examine the utility of these two introduced components more clearly, we run two variants HAL-BSEMO-bs and HAL-BSEMO-bm, which use only biased selection and balanced mutation, respectively. We plot the curve of objective value over runtime on the data set Stare-H with $k/|V| = 0.1$, as shown in Figure 2(a). The greedy algorithm is a fixed-time (nearly kn) algorithm, while others are anytime algorithms, and can get better performance by using more time (less than $5kn$). We can observe that HAL-BSEMO-bs and HAL-BSEMO-bm are better than HAL-GSEMO, and HAL-BSEMO (i.e., using both biased selection and balanced mutation) performs the best. These results show the effectiveness of biased selection and balanced mutation, and the superiority of HAL-BSEMO-bs over HAL-BSEMO-bm also discloses the more important role of biased selection. For HAL-NSGA-II, the relatively bad performance may be because the population will contain some redundant dominated solutions due to the fixed population size 100, and the crossover operator is not very helpful here.

Human Assisted Support Vector Machine

We use four competitive baselines, i.e., triage based on algorithmic uncertainty (Alg Triage) and predicted error (Estimated Triage) (Raghu et al. 2019), distorted greedy (DG) and its stochastic version, SDG (De et al. 2021)[§]. Besides Messidor, another data set Aptos containing 705 retinal images is used; each image is given a score by an expert, measuring the severity of diabetic retinopathy. To allow for classification, for each instance, $y = -1$ if its severity score is the lowest grade of the associated disease and $y = 1$ otherwise.

The human score $h(x)$ is sampled from a categorical distribution $Cat(\mathbf{p}_{x,y})$. After scaling, $h(\cdot) \in [-1, 1]$, and we use $c(x, y) = \mathbb{E}[(1 - yh(x))_+]^2$ to measure the human error. The regularization parameter λ is set to 0.03 for Messidor and 0.6 for Aptos. We use a common value 0.8 for γ in Eq. (7), and 60% of the instances for training and the rest for testing.

[§]We run SDG ten times independently due to its stochasticity.

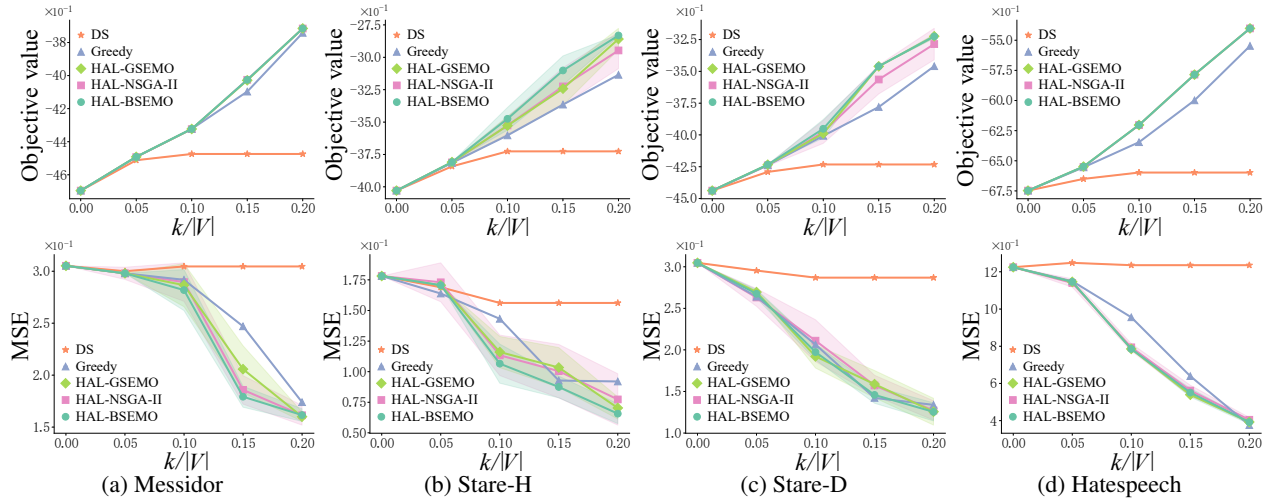


Figure 1: On each data set, the upper and lower subfigures show the objective value (the larger, the better) on the training set and the mean squared error (MSE, the smaller, the better) on the test set, respectively.

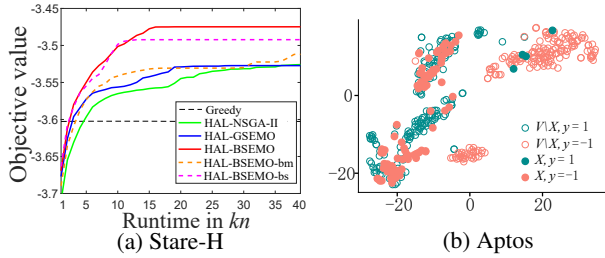


Figure 2: (a) The objective value vs. runtime (i.e., #objective evaluations) on Stare-H, where $k/|V| = 0.1$. (b) The scatter plot of HAL-BSEMO on Aptos, where $k/|V| = 0.2$.

The first row of Figure 3 shows that HAL-NSGA-II, HAL-GSEMO and HAL-BSEMO achieve better objective values than the baselines on the training set, and HAL-BSEMO performs the best, except for $k/|V| = 0.05$ on Messidor. For the classification error rate and F1 score on the test set, the three variants of HAL-EMO also perform better in most cases. Figure 2(b) gives the scatter plot of HAL-BSEMO on the data Aptos, where the green and red colors represent the instances with labels 1 and -1 , respectively, and \circ/\bullet denote the instances assigned to the machine and human, respectively. We can see that the instances (i.e., \bullet) assigned to the human are exactly those tricky instances.

Conclusion

This paper proposes the HAL-EMO framework, employing any MOEA to solve the bi-objective reformulated HAL problem. We use NSGA-II and GSEMO, and also propose the specific MOEA, BSEMO with biased selection and balanced mutation, achieving better and same theoretical guarantees than previous algorithms, respectively, for human assisted ridge regression and SVM. Empirical results on medical diagnosis and content moderation show that HAL-EMO using any of the three MOEAs can achieve good perfor-

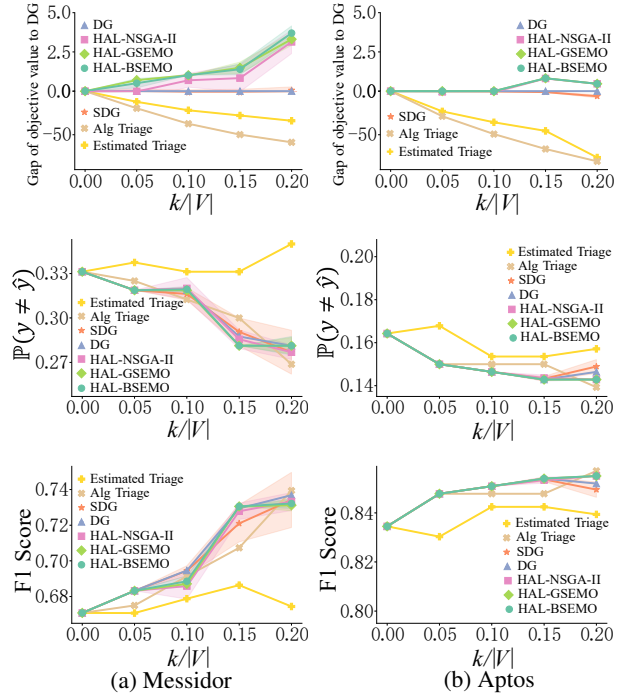


Figure 3: On each data set, the three subfigures show the objective value of each algorithm minus the objective value of DG (the larger, the better) on the training set, the classification error rate (the smaller, the better), and the F1 score (the larger, the better) on the test set, respectively.

mance on both optimization and generalization, and using BSEMO often leads to the best performance. An interesting future work is to design better MOEAs (e.g., using balanced crossover (Friedrich et al. 2022) or surrogate models (Hao et al. 2022; Zhang, He, and Ishibuchi 2022)) for HAL-EMO.

Acknowledgments

This work was supported by the National Science Foundation of China (62022039, 62276124, 62106114, 61921006). Chao Qian is the corresponding author.

References

- Bäck, T. 1996. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford, UK: Oxford University Press.
- Bordt, S.; and von Luxburg, U. 2020. When humans and machines make joint decisions: A non-symmetric bandit model. *CoRR*, abs/2007.04800.
- Coello, C. A. C.; Lamont, G. B.; Van Veldhuizen, D. A.; et al. 2007. *Evolutionary Algorithms for Solving Multi-objective Problems*. New York, NY: Springer.
- Das, A.; and Kempe, D. 2011. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 1057–1064. Bellevue, WA.
- Davidson, T.; Warmley, D.; Macy, M. W.; and Weber, I. 2017. Automated hate speech detection and the problem of offensive language. In *proceedings of the 11th International Conference on Web and Social Media (ICWSM)*, 512–515. Montréal, Canada.
- De, A.; Koley, P.; Ganguly, N.; and Gomez-Rodriguez, M. 2020. Regression under human assistance. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2611–2620. New York, NY.
- De, A.; Okati, N.; Zareade, A.; and Rodriguez, M. G. 2021. Classification under human assistance. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, 5905–5913. Virtual.
- Deb, K.; Agrawal, S.; Pratap, A.; and Meyarivan, T. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on Evolutionary Computation*, 6(2): 182–197.
- Decencière, E.; Zhang, X.; Cazuguel, G.; Lay, B.; Cochener, B.; Trone, C.; Gain, P.; Ordóñez-Varela, J.; Massin, P.; Erginay, A.; Charton, B.; and Klein, J. 2014. Feedback on a publicly distributed image database: the Messidor database. *Image Analysis & Stereology*, 33(3): 231–234.
- Doerr, B.; and Neumann, F., eds. 2020. *Theory of Evolutionary Computation: Recent Developments in Discrete Optimization*. Switzerland: Springer.
- Friedrich, T.; Kötzing, T.; Radhakrishnan, A.; Schiller, L.; Schirneck, M.; Tennigkeit, G.; and Wietheger, S. 2022. Crossover for cardinality constrained optimization. In *Proceedings of the 24th ACM Conference on Genetic and Evolutionary Computation (GECCO)*, 1399–1407. Boston, MA.
- Friedrich, T.; and Neumann, F. 2015. Maximizing submodular functions under matroid constraints by evolutionary algorithms. *Evolutionary Computation*, 23(4): 543–558.
- Gatmiry, K.; and Gomez-Rodriguez, M. 2018. Non-submodular function maximization subject to a matroid constraint, with applications. *CoRR*, abs/1811.07863.
- Giel, O. 2003. Expected runtimes of a simple multi-objective evolutionary algorithm. In *Proceedings of the 2003 IEEE Congress on Evolutionary Computation (CEC)*, 1918–1925. Canberra, Australia.
- Hao, H.; Zhou, A.; Qian, H.; and Zhang, H. 2022. Expensive multi-objective optimization by relation learning and prediction. *IEEE Transactions on Evolutionary Computation*, 26(5): 1157–1170.
- Harshaw, C.; Feldman, M.; Ward, J.; and Karbasi, A. 2019. Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2634–2643. Long Beach, CA.
- Hashemi, A.; Ghasemi, M.; Vikalo, H.; and Topcu, U. 2019. Submodular observation selection and information gathering for quadratic models. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2653–2662. Long Beach, CA.
- Hong, W.-J.; Yang, P.; and Tang, K. 2021. Evolutionary computation for large-scale multi-objective optimization: A decade of progresses. *International Journal of Automation and Computing*, 18(2): 155–169.
- Hoover, A. D.; Kouznetsova, V.; and Goldbaum, M. 2000. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Transactions on Medical imaging*, 19(3): 203–210.
- Iyer, R. K.; and Bilmes, J. A. 2012. Algorithms for approximate minimization of the difference between submodular functions, with applications. In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence (UAI)*, 407–417. Catalina Island, CA.
- Kamalaruban, P.; Devidze, R.; Cevher, V.; and Singla, A. 2019. Interactive teaching algorithms for inverse reinforcement learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, 2692–2700. Macao, China.
- Knowles, J. D.; Watson, R. A.; and Corne, D. W. 2001. Reducing local optima in single-objective problems by multi-objectivization. In *Proceedings of the 1st International Conference on Evolutionary Multi-Criterion Optimization (EMO)*, 269–283. Zurich, Switzerland.
- Laumanns, M.; Thiele, L.; and Zitzler, E. 2004. Running time analysis of multi-objective evolutionary algorithms on pseudo-Boolean functions. *IEEE Transactions on Evolutionary Computation*, 8(2): 170–182.
- Mozannar, H.; and Sontag, D. 2020. Consistent estimators for learning to defer to an expert. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 7076–7087. Virtual.
- Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An analysis of approximations for maximizing submodular set functions – I. *Mathematical Programming*, 14(1): 265–294.
- Neumann, F.; and Wegener, I. 2006. Minimum spanning trees made easier via multi-objective optimization. *Natural Computing*, 5(3): 305–319.

- Neumann, F.; and Witt, C. 2010. *Bioinspired Computation in Combinatorial Optimization: Algorithms and Their Computational Complexity*. Berlin, Germany: Springer-Verlag.
- Qian, C. 2021. Multi-objective evolutionary algorithms are still good: Maximizing monotone approximately submodular minus modular functions. *Evolutionary Computation*, 29(4): 463–490.
- Qian, C.; Shi, J.-C.; Yu, Y.; and Tang, K. 2017a. On subset selection with general cost constraints. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 2613–2619. Melbourne, Australia.
- Qian, C.; Shi, J.-C.; Yu, Y.; Tang, K.; and Zhou, Z.-H. 2017b. Subset selection under noise. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 3562–3572. Long Beach, CA.
- Qian, C.; Yu, Y.; Tang, K.; Yao, X.; and Zhou, Z.-H. 2019. Maximizing submodular or monotone approximately submodular functions by multi-objective evolutionary algorithms. *Artificial Intelligence*, 275: 279–294.
- Qian, C.; Yu, Y.; and Zhou, Z.-H. 2015. Subset selection by Pareto optimization. In *Advances in Neural Information Processing Systems 28 (NeurIPS)*, 1765–1773. Montreal, Canada.
- Raghu, M.; Blumer, K.; Corrado, G.; Kleinberg, J.; Obermeyer, Z.; and Mullainathan, S. 2019. The algorithmic automation problem: Prediction, triage, and human effort. *CoRR*, abs/1903.12220.
- Roostapour, V.; Neumann, A.; Neumann, F.; and Friedrich, T. 2022. Pareto optimization for subset selection with dynamic cost constraints. *Artificial Intelligence*, 302: 103597.
- Sabato, S.; and Munos, R. 2014. Active regression by stratification. In *Advances in Neural Information Processing Systems 27 (NeurIPS)*, 469–477. Montreal, Canada.
- Tschiatschek, S.; Ghosh, A.; Haug, L.; Devidze, R.; and Singla, A. 2019. Learner-aware teaching: Inverse reinforcement learning with preferences and constraints. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, 4147–4157. Vancouver, Canada.
- Wilder, B.; Horvitz, E.; and Kamar, E. 2020. Learning to complement humans. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, 1526–1533. Yokohama, Japan.
- Zhang, H.; and Vorobeychik, Y. 2016. Submodular optimization with routing constraints. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, 819–826. Phoenix, AZ.
- Zhang, J.; He, L.; and Ishibuchi, H. 2022. Dual fuzzy classifier-based evolutionary algorithm for expensive multi-objective optimization. *IEEE Transactions on Evolutionary Computation*, in press.
- Zhou, Z.-H.; Yu, Y.; and Qian, C. 2019. *Evolutionary Learning: Advances in Theories and Algorithms*. Singapore: Springer.