Cost-Sensitive Active Learning for Incomplete Data

Min Wang, Chunyu Yang, Fei Zhao, Fan Min, *Member, IEEE*, and Xizhao Wang, *Fellow, IEEE*

***Abstract*—Practical data often suffer from missing attribute values and lack of class labels. A reasonable machine learning scenario involves obtaining certain values and labels at cost on request. In this article, we propose the cost-sensitive active learning through unified evaluation and dynamic selection (CALS) algorithm to handle the learning task in this new scenario. For data representation, we consider misclassification cost, label query cost, and attribute query cost. For the cost/benefit estimation, we design a unified assessment of attribute values and labels with softmax regression. For the selection of attribute value and label, we propose an optimal acquisition scheme with permutation and greedy strategies. We perform experiments with synthetic, benchmark, and domain datasets. The results of the significance test verify the effectiveness of CALS and its superiority over cost-sensitive active learning and missing data imputation algorithms.**

***Index Terms*—Active learning, cost sensitive, incomplete data, unified evaluation and dynamic selection.**

I. Introduction

MISSING data exist widely in many domains, such as gene expression [1], electricity distribution systems [2], and speech recognition [3]. Various missing value processing methods have been proposed to improve data availability and learnability. One popular approach is the missing value imputation [4]. Some classic methods, such as regression [5] and correlation analysis [4], are employed for this issue. Frequency distribution [6] and energy dependence [7] are also helpful in designing sophisticated schemes. Another approach is active feature acquisition (AFA) [8], [9], which is the most reliable way when feature values are severely missing [10]. In this case, missing attribute values can be obtained at a cost upon request, such as running additional diagnostic procedures. A common approach is to train the classifier and query the most valuable features of the misclassification instance [8], [9]. Advanced classification methods employed in this process include hidden Markov models (HMMs) [11] and decision trees [12].

Another challenge for task learning is the absence of labeling. According to Settles [13], "labeled instances are often difficult, expensive, or time consuming to obtain, as they require the efforts of experienced human annotators." For information extraction, it may take 1 h or more to determine entities and relationships, even for simple news reports [14]. Semisupervised learning [15], [16] and active learning [17], [18] attempt to break the label bottleneck from different directions. In semisupervised learning [15], the self-training technique trains learners with a few labeled instances to classify the remaining instances. Furthermore, the generative model [19], cluster assumption [15], and manifold assumption [20] are used to construct the relationship between the unlabeled instance distribution and the learning objective. In active learning [17], [18], the fundamental issue is how to select the most critical instances [17]. There are two main criteria: 1) informativeness [17] and 2) representativeness [18]. Relevance and diversity are also incorporated into the process of instance selection [21], [22].

Real-world data may suffer from both attribute value missing and label scarcity. For example, speech utterances are prone to a large number of missing attributes. Meanwhile, speech utterances labeling requires experienced experts, resulting in label scarcity [3]. To classify these complex data, a reasonable scenario is to actively query missing attribute values and labels with cost.

In this article, we propose the cost-sensitive active learning through a unified evaluation and dynamic selection (CALS) algorithm to handle the learning task in the new scenario. First, we define a 5-tuple incomplete decision system data model and propose a cost-sensitive active learning problem. The inputs include an incomplete dataset without initial labels, attribute query cost, and label query cost. The outputs include the critical attribute values and labels for the query, as well as predicted instance labels. The optimization objective is to minimize the total cost.

Second, we develop an optimization method to uniformly evaluate and dynamically select critical attribute values and labels. We use softmax regression to build a probabilistic model to obtain the classification probability. Then, we consider the cost/benefit of attribute values and labels and calculate different types of cost, such as expected misclassification cost. With a greedy search of different missing value filling schemes, we obtain the optimal instance–attribute pairs

Manuscript received January 20, 2022; accepted June 5, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62006200; in part by the Natural Science Foundation of Sichuan Province under Grant 2020YFQ0038 and Grant 2022YFG0179; in part by the Open Fund of State Key Laboratory of Oil and Gas Reservoir Geology and Exploitation (Chengdu University of Technology) under Grant PLC20211104; in part by the Science and Technology Cooperation Project of the CNPC-SWPU Innovation Alliance under Grant 2020CX020000; and in part by the Central Government Funds for Guiding Local Scientific and Technological Development of China under Grant 2021ZYD0003 and Grant 2021ZYD0042. This article was recommended by Associate Editor J. A. Lozano. *(Corresponding author: Fan Min.)*

Min Wang, Chunyu Yang, and Fei Zhao are with the College of Electrical Engineering and Information, Southwest Petroleum University, Chengdu 610500, China (e-mail: wangmin@swpu.edu.cn; yangchunyu@stu.swpu.edu.cn; zhaofei@stu.swpu.edu.cn).

Fan Min is with the School of Computer Science, Southwest Petroleum University, Chengdu 610500, China (e-mail: minfan@swpu.edu.cn).

Xizhao Wang is with the Institute of Big Data, Shenzhen University, Shenzhen 518060, China (e-mail: xizhaowang@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSMC.2022.3182122.

Digital Object Identifier 10.1109/TSMC.2022.3182122

to be queried. By calculating the completeness and significance of each instance, we obtain the label cost/benefit evaluation.

Third, we design a new CALS algorithm. Fig. 1 illustrates the CALS process using a running example. We select the initial training set based on the product of density and distance. Then, we train a probabilistic model to obtain the solution parameter. With the cost minimal method, we will select the critical instance x_{s*} to perform the corresponding prediction or query and incrementally update the training model. Finally, we adjust the algorithm to avoid nonunique solutions and numerical overflows.

Experiments were performed on synthetic, benchmark, and practical logging interpretation datasets. The datasets include continuous, discrete, and mixed attribute types, with the number of instances ranging from 150 to 245 057. We compared the CALS algorithm with state-of-the-art cost-sensitive, cost-sensitive active learning, missing value imputation, and AFA algorithms. We adopt the Friedman test and the Nemenyi post hoc test to verify significant differences between CALS and comparison algorithms. The results indicate CALS outperforms all of these competing algorithms in terms of average cost.

In summary, our contribution is as follows.

1) We propose a cost-sensitive active learning problem that considers a new but meaningful classification scenario. It is dedicated to solving complex data (attribute values missing and label scarcity) classification issue.
2) We present a cost/benefit optimization method that provides the unified evaluation of attribute values and labels. It considers the interaction of attribute values and label cost/benefit.
3) We design an optimal query scheme for obtaining critical attribute values and labels. It guarantees dynamic selection of attribute values and labels, as well as incremental updates to the model.

The remainder of this article is organized as follows. Section II briefly reviews two practical classification scenarios and solutions. Section IV describes the new problem and the CALS algorithm. Section V discusses the experimental results and Section VI makes a conclusion.

II. Related Work

Real-world data are noisy and may suffer from missing attribute values or label scarcity. In this section, we will introduce two typical classification scenarios, including missing data and label scarcity. Meanwhile, we will discuss some related solutions, including missing values imputation, AFA, semisupervised learning, and active learning.

A. Dealing With Missing Data

Missing data are a serious problem that may lead to poor quality of training data and further significantly degrade the model performance. Missing attribute values can be addressed by deleting them directly, such as "complete deletion," "listwise deletion," or "specific deletion." However, simple deletion generally degrades the performance of the model.

Missing data imputation [23], [24] considers the correlation among features to complete missing attribute values. The common imputation method is regression imputation [23], including multiple linear regression, logistic regression, and multinomial logistic regression [24]. These methods explore existing relationships between features to approximate missing attribute values. Improved kNN [25], artificial neural network [24], and genetic programming [26] have also been used to complete missing values. Since a single imputation algorithm may not be able to consider all data distribution characteristics [10], multiple imputation [27] obtains "m" complete datasets, and adopts the weighted average to impute missing attribute values.

AFA [8], [28] assumes that true attribute values can be obtained at a cost. This is a more reliable method when the attribute values are severely missing [29]. It selects the most informative features to obtain, rather than randomly or exhaustively acquire all new features. Zheng and Padmanabhan [8] proposed a "single-pass" approach to acquire an attribute value with the least confidence. Similarly, Melville *et al.* [28] attempted to calculate the expected function for selecting the top feature. Instead of querying one specific feature value, the incremental AFA [30] query some significant feature values at once. For example, Saar-Tsechansky *et al.* [31] took a decision-theoretic approach to acquire a few feature values. In addition, model confidence [9] and cost-sensitive learning [32] are also incorporated into the AFA.

B. Dealing With Label Scarcity

Label scarcity is another challenging classification scenario. In practice, labeled data are costly and difficult to obtain since the labeling procedure may require human experts, expensive devices, or too much time. For example, labeling genes and diseases in biomedical texts often requires a Ph.D.-level biologist.

Semisupervised learning is proposed to address the labeled data scarcity issue [33]. According to Zhou [34], "Semisupervised learning attempts to automatically exploit unlabeled data in addition to labeled data to improve learning performance, where no human intervention is assumed." Miller and Uyar [19] gave the reasons why unlabeled instances can improve learning performance. The key is to determine the relationship between the unlabeled instance distribution and the learning objectives. Common approaches include the generative model [19], self-training [16], and co-training [20]. In addition, multiview learning [35] adopts the ensemble for semisupervised learning, while entropy regularization [36] considers the confidence.

Active learning [18] assumes that the algorithm is able to interactively query an *oracle* to obtain the desired true label. Since obtaining each class label is difficult, it is reasonable to select informative instances whose labels will shrink the version space as fast as possible [37]. The common approaches include query-by-committee [37], uncertainty sampling [38], and optimal experimental design [39]. Another approach is to select the instances that best represent the unlabeled data [18]. This may require a relatively large number of instances to

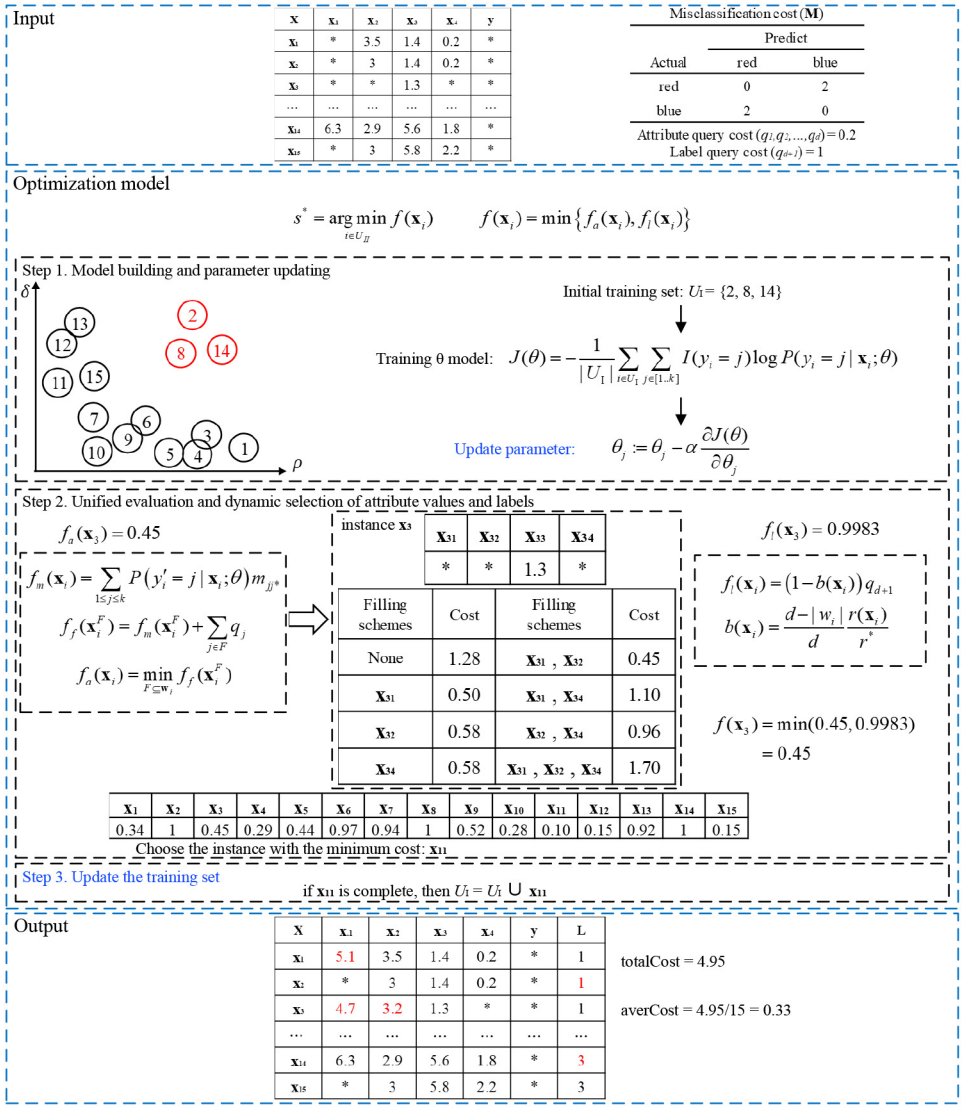


Fig. 1. Running example of the CALS algorithm. The top part is the input with a binary class dataset containing 15 instances, and the cost settings. The middle part is the dynamic evaluation and incremental learning method, which iteratively guides attribute values and instances selection. With the minimal total cost optimization strategy, we obtain the critical attribute values and labels to be queried. In this way, all labels are either queried or predicted. The bottom is the output that calculates the total cost and average cost.

be queried before the optimal decision boundary is found. Deploying one of these two criteria may significantly limit the performance of active learning. Therefore, several active learning algorithms [17], [40] have been developed to find critical instances that are both informative and representative.

For high-dimensional data, because the p-norm is affected by the curse of dimensionality, the distance-based query strategy seems to be ineffective. Therefore, Sinha *et al.* [41] adopted the variational autoencoder to learn the latent space to select critical instances. Additionally, for more practical scenarios, the cost of labeling needs to be considered. Xiao *et al.* [42] proposed a cost-sensitive semisupervised integrated active learning model. Furthermore, active learning has been deeply integrated with related technologies for various application requirements. For example, Ma and Chang [43] integrated active learning with iterative training sampling to obtain the accurate classification of hyperspectral images.

III. Problem Statement

In this section, we discuss the data model, problem definition, and solution framework.

A. Data Model

Active learning becomes cost sensitive [40] when one or more types of costs are considered. Misclassification cost refers to the cost of classifying instances with label i as j. Attribute query cost refers to the expenses paid for obtaining more complete feature information. Label query cost (teacher cost) refers to the expenses paid to experts for labeling data. Considering various costs, we build the data model. Let $\mathbf{w}_i = \{j | x_{ij}$ is missing$\}$ indicate the missing value index of instance $\mathbf{x}_i$.

Definition 1: An incomplete cost-sensitive decision system (ICS-DS) is the 5-tuple

$$S = (\mathbf{X}, \mathbf{y}, \mathbf{W}, \mathbf{M}, \mathbf{q}) \quad (1)$$

where $\mathbf{X} = (x_{ij})_{n\times d}$ is the data matrix, $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{id})$ is the ith instance; n is the number of instances; d is the number of conditional attributes; $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ where $y_i \in [1, \ldots, k]$ is the class label vector; $\mathbf{W} = \{(i, j)|x_{ij} \text{ is missing}\}$ is the set of missing value positions; $\mathbf{M} = (m_{ij})_{k\times k}$ is the misclassification cost matrix; and $\mathbf{q} = (q_1, q_2, \ldots, q_{d+1})$ is the query cost vector, q_i where $1 \le i \le d$ is the query cost for the ith attribute, and q_{d+1} is the label query cost.

We present an example to facilitate an intuitive understanding of the specific scenario (missing attributes and label scarcity) and corresponding costs. Credit card companies predict the most profitable customers. Due to authority reasons, some bank card access transaction data cannot be obtained. The attributes of the instances are missing. Meanwhile, high-quality customers need to be labeled by experienced experts, which results in label scarcity. Therefore, we consider actively querying critical attribute values and instance labels. Missing attribute characteristics are obtained from lease transaction records. The rental fee corresponds to the attribute query cost q_i, $1 \le i \le d$. The selected critical customers are assigned evaluation labels by experienced experts. Experts are paid for each evaluation of the customer, which corresponds to the label query cost q_{d+1}.

For classic active learning, learners interact with experts to actively obtain instance labels [18]. The fundamental issue is to select the most critical instances. We consider high-quality customer classification as an example, in which there are no missing bank card access transaction data. We only consider how to select critical customers, and then hand them over to experts for labeling. Additionally, another scenario is AFA [8]. The assumption is that additional features can be obtained at a cost, such as leasing transaction records from other credit card companies. However, this scenario does not consider obtaining labels through queries.

Finally, we discuss the most recent cost-sensitive learning scenario. Zhang and Zhang [44] proposed an evolutionary cost-sensitive extreme learning machine. The misclassification cost matrix is unknown. The proposed evolutionary cost-sensitive framework guides users to freely and automatically determine task-specific cost matrices. In contrast, we consider the fixed misclassification cost matrix provided by the user.

B. Problem Definition

Now, we present the problem definition.

Problem 1: Cost-Sensitive Active Learning for Incomplete Data

Input: An ICS-DS $S = (\mathbf{X}, \mathbf{y}, \mathbf{W}, \mathbf{M}, \mathbf{q})$.

Output: The set of queries $\mathbf{Q} \subset [1, \ldots, n] \times [1, \ldots, d+1]$, and the predicted/queried label vector $\mathbf{y}' = (y'_1, y'_2, \ldots, y'_n)^T$.

Optimization Objective:

min total cost $= \sum_{(i,j)\in\mathbf{Q}} q_j + \sum_{i=1}^{n} m_{y_i y'_i}$.

Here, the input is an ICS-DS, where missing attribute values and labels can be obtained at a cost. The outputs include the set of queried attribute values $\mathbf{Q}$, and the predicted/queried label vector $\mathbf{y}'$. The optimization objective is to minimize the total cost. Here, $\sum_{(i,j)\in\mathbf{Q}} q_j$ includes the query cost of both attribute values and labels. $m_{y_i y'_i}$ is the misclassification cost of classifying an instance with the label y_i as y'_i. It is 0 when $y_i = y'_i$; hence, we do not distinguish queried instances from the predicted ones.

TABLE I
VARIOUS COST REPRESENTATIONS

Cost	Meaning	Description
q_i	attribute query cost	given by the expert
q_{d+1}	label query cost	given by the expert
m_{ij}	misclassification cost	given by the expert
$f(\mathbf{x}_i)$	instance cost	Eq. (3)
$f_m(\mathbf{x}_i)$	expected misclassification cost	Eq. (13)
$f_f(\mathbf{x}_i^F)$	instance cost with attribute filling	Eq. (14)
$f_a(\mathbf{x}_i)$	minimal instance cost with attribute values filling	Eq. (15)
$f_l(\mathbf{x}_i)$	instance cost with label query	Eq. (18)

IV. PROPOSED ALGORITHM

In this section, we present our CALS algorithm. First, we define a new data model and propose the cost-sensitive active learning problem. Second, we discuss three key technologies of our approach, including the initial training set construction, attribute value and label cost/benefit evaluation, and training set incremental upgrades. Finally, we present the CALS algorithm with complexity analysis.

A. Framework

In the learning process, we should obtain the values of some attributes or labels which are more important. Therefore, the key issue is: how to uniformly evaluate the cost/benefit of attribute values and labels, and dynamically select them for query?

We present a cost minimization method to address the aforementioned issue. In each iteration, the new method selects an instance $\mathbf{x}_{s^*}$ from the unlabeled set U_{II} with the minimal instance cost, that is

$$s^* = \arg\min_{i\in U_{\text{II}}} f(\mathbf{x}_i) \tag{2}$$

where

$$f(\mathbf{x}_i) = \min\{f_a(\mathbf{x}_i), f_l(\mathbf{x}_i)\} \tag{3}$$

is called the instance cost, $f_a(\mathbf{x}_i)$ is the minimal instance cost with attribute query, and $f_l(\mathbf{x}_i)$ is the instance cost with label query. Note that U is the instance set, U_I is the training set, and U_{II} is the unlabeled set.

The cost minimization method solves the proposed issue from the following two aspects. The designed cost functions $f_a(\mathbf{x}_i)$, $f_l(\mathbf{x}_i)$ obtain a unified evaluation of the cost/benefit of the attribute/label query. $f_a(\mathbf{x}_i)$ considers the tradeoff between the misclassification cost and attribute filling cost, and selects the optimal attribute filling scheme. We elaborate on the above steps in Sections IV-C1–IV-C3. $f_l(\mathbf{x}_i)$ considers the label query cost q_{d+1}, and instance completeness and representativeness to achieve an overall evaluation of the label cost/benefit. We describe this technique in detail in Section IV-C4. The general calculation process includes various cost types, as detailed in Table I.

In contrast, (2) and (3) indicate the dynamic selection of attribute values and labels. Equation (3) compares the attribute

and label cost/benefit for each instance and records the minimal cost scheme. Equation (2) selects the minimal cost instance in each iteration and performs the corresponding operation. In subsequent iterations, the model is retrained and the attributes and label cost/benefit are reevaluated. We describe this technique in Section IV-D.

B. Construct the Initial Training Set

According to Problem 1, the extreme case of label scarcity is that there is no class label at all. Therefore, for a more general situation, the first key issue is: how to construct the initial training set? Inspired by density peak clustering [45], we define the significance $\gamma(\mathbf{x}_i)$ of each instance.

The labels of $\min\{\sqrt{n}, 0.1n\}$ instances with the largest γ are queried. Such a setting is adaptive to both small and big datasets. Respective instances form the initial training set U_I, which is represented by their indices. Let $U = [1, \ldots, n]$ indicates the set of all instances. The unlabeled instance set is $U_{II} = U - U_I$. Note that U_I and U_{II} will change in the learning process.

The calculation of instance significance consists of the following three steps. First, the instance density is given by [45]

$$\rho(\mathbf{x}_i) = \sum_{j\in[1..n], i\neq j} e^{-\left(\frac{\text{dist}(\mathbf{x}_i,\mathbf{x}_j)}{d_c}\right)^2} \tag{4}$$

where $\text{dist}(\mathbf{x}_i, \mathbf{x}_j)$ is the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$, and d_c is the cutoff distance.

Second, the instance minimal distance is given by [45]

$$\delta(\mathbf{x}_i) = \begin{cases} \max_{j\in[1,\ldots,n]}(\text{dist}(\mathbf{x}_i, \mathbf{x}_j)), & \rho(\mathbf{x}_i)\text{is maximal} \\ \min_{j:\rho(\mathbf{x}_i)>\rho(\mathbf{x}_j)}(\text{dist}(\mathbf{x}_i, \mathbf{x}_j)), & \text{otherwise.} \end{cases} \tag{5}$$

Third, the instance significance is given by

$$\gamma(\mathbf{x}_i) = \rho(\mathbf{x}_i) \cdot \delta(\mathbf{x}_i). \tag{6}$$

C. Unified Evaluation and Dynamic Selection of Attribute Values and Labels

1) Train a Softmax-Based Probabilistic Model: First, we build a softmax-based probabilistic model [46] to establish the association between attributes and labels, that is

$$P\big(y_i' = j|\mathbf{x}_i; \theta\big) = \frac{e^{\theta_j^T \mathbf{x}_i}}{\sum_{l=1}^{k} e^{\theta_l^T \mathbf{x}_i}} \tag{7}$$

where $\theta = (\theta_{ij})_{k\times(d+1)}$ is the parameter matrix, and $\theta_i = (\theta_{i1}, \ldots, \theta_{i(d+1)})$. Equation (7) obtains the conditional probability of $\mathbf{x}_i$ belonging to class y_i'.

Second, the cost function is given by

$$J(\theta) = -\frac{1}{|U_1|} \sum_{i\in U_1} \sum_{j\in[1..k]} I\big(y_i' = j\big) \log P\big(y_i' = j|\mathbf{x}_i; \theta\big). \tag{8}$$

i represents the ith instance, and j represents the category. The indicator function $I(\cdot) = 1$ when $\cdot$ is true and 0 otherwise. The cost function $J(\theta)$ represents the deviation between the predicted value and the true value. We solve the cost function $J(\theta)$ to obtain the optimal parameter θ.

Finally, we obtain the optimal parameter θ by minimizing $J(\theta)$. Gradient descent [47] is employed for this purpose with the following iteration:

$$\theta_j \coloneqq \theta_j - \alpha \nabla_{\theta_j} J(\theta), j = 1, 2, \ldots, k \tag{9}$$

where α is the step size, and $\nabla_{\theta_j} J(\theta)$ is the gradient. The iteration repeats until θ does not change, or the number of iterations reaches a predefined value.

2) Calculate the Instance Cost With Attribute Value Filling: Different attribute value filling schemes have different cost/benefit. We design a function $f_f(\mathbf{x}_i)$ to consider various attribute value filling schemes.

First, for $(i, j) \in \mathbf{W}$, we adopt the method in [48] to estimate the expected value. In addition, we provide a variety of missing value estimation methods, including average filling, segmented cubic spline interpolation filling, or conformal segmented cubic spline interpolation filling. For example, for the average filling, the expected value of the jth attribute is

$$\overline{x_{\cdot j}} = \frac{\sum_{(i,j)\notin\mathbf{W}} x_{ij}}{|\{(i, j) \notin \mathbf{W}\}|}. \tag{10}$$

For $(i, j) \in \mathbf{W}$, we fill x_{ij} with the expected value $\overline{x_{\cdot j}}$.

Any $F \subseteq \mathbf{w}_i$ corresponds to an attribute filling scheme for instance $\mathbf{x}_i$. Naturally, there are $2^{|\mathbf{w}_i|}$ attribute filling schemes for $\mathbf{x}_i$. An example of attribute filling is presented at the middle of Fig. 1. For instance $\mathbf{x}_3$, $|\mathbf{w}_3| = 3$, we obtain eight attribute filling schemes, i.e., $\emptyset$, $\{1\}$, $\{2\}$, $\{4\}$, $\{1, 2\}$, $\{1, 4\}$, $\{2, 4\}$, $\{1, 2, 4\}$.

Second, we calculate the expected misclassification cost for any given filling method. We compute the hypothesis function $h_\theta(\mathbf{x}_i)$ with the optimal θ obtained through (8) and (9). The hypothesis function is

$$h_\theta(\mathbf{x}_i) = \begin{bmatrix} P\big(y_i' = 1|\mathbf{x}_i; \theta\big) \\ P\big(y_i' = 2|\mathbf{x}_i; \theta\big) \\ \vdots \\ P\big(y_i' = k|\mathbf{x}_i; \theta\big) \end{bmatrix} = \frac{1}{\sum_{l=1}^{k} e^{\theta_l^T \mathbf{x}_i}} \begin{bmatrix} e^{\theta_1^T \mathbf{x}_i} \\ e^{\theta_2^T \mathbf{x}_i} \\ \vdots \\ e^{\theta_k^T \mathbf{x}_i} \end{bmatrix}. \tag{11}$$

Denoting the filled instance by $\mathbf{x}_i^F$, the prediction should maximize the probability. Hence, the predicted label is

$$j^* = \arg\max_{1\le j\le k} P\big(y_i' = j\big|\mathbf{x}_i^F; \theta\big). \tag{12}$$

With this prediction, the expected misclassification cost is

$$f_m\big(\mathbf{x}_i^F\big) = \sum_{1\le j\le k} P\big(y_i' = j\big|\mathbf{x}_i^F; \theta\big) m_{jj^*}. \tag{13}$$

Third, we calculate the instance cost with attribute filling. The instance cost with attribute filling is

$$f_f\big(\mathbf{x}_i^F\big) = f_m\big(\mathbf{x}_i^F\big) + \sum_{j\in F} q_j \tag{14}$$

where $f_m(\mathbf{x}_i^F)$ is the expected misclassification cost with the filled values in F.

3) Obtain the Minimal Instance Cost With Attribute Filling: Which attribute values should be actually queried for various attribute filling schemes? The scheme with a minimum $f_f(\mathbf{x}_i^F)$ is optimal. We obtain the cost function $f_a(\mathbf{x}_i)$ by using a greedy search strategy. Therefore, the minimal instance cost with attribute filling is given by

$$f_a(\mathbf{x}_i) = \min_{F \subseteq \mathbf{w}_i} f_f\left(\mathbf{x}_i^F\right). \tag{15}$$

In the meanwhile, we obtain the optimal missing value index subset

$$F^* = \arg\min_{F \subseteq \mathbf{w}_i} f_f\left(\mathbf{x}_i^F\right). \tag{16}$$

Attribute values in the subset F^* are considered as critical attribute values to be queried. Note that $f_f(\mathbf{x}_i^F)$ is the instance cost with attribute filling, $f_a(\mathbf{x}_i)$ is the minimal instance cost with attribute filling, and F^* is the subset of critical attribute values to be queried.

4) Calculate the Instance Cost With Label Query: The query of a label not only obtains the true class of the current instance but also increases the training set. Hence, there is an additional benefit of the label query. The benefit is influenced by the completeness of the instance, as well as its significance.

Definition 2: Let $\gamma(\mathbf{x}_i)$ be the significance of instance $\mathbf{x}_i$. The benefit of label y_i is

$$b(y_i) = \frac{d - |\mathbf{w}_i|}{d} \frac{\gamma(\mathbf{x}_i)}{\gamma^*} \tag{17}$$

where $(d - |\mathbf{x}_i|/d)$ is the completeness of $\mathbf{x}_i$ and $\gamma^* = \max \gamma(\mathbf{x}_i)$ is the maximal value of γ.

Therefore, the instance cost with label query is given by

$$f_l(\mathbf{x}_i) = (1 - b(y_i))q_{d+1} \tag{18}$$

where the benefit is expressed as a negative cost.

D. Update the Training Set

We select the minimal cost instance through a unified cost/benefit evaluation. Therefore, the third key issue is: considering the impact of the cost/benefit of attributes/labels, how to classify the selected instance to enhance the model?

Here, we classify the selected instance $\mathbf{x}_{s^*}$. If $f_a(\mathbf{x}_{s^*}) < f_l(\mathbf{x}_{s^*})$, we query the critical attribute values according to the corresponding strategy, and calculate the probability $P(y_i' = j|\mathbf{x}_{s^*}; \theta)$ to predict the label. Otherwise, we query the label directly. If $\mathbf{x}_{s^*}$ is completed after querying the instance label, it will be added to the training set U_{I}. With the new training set, we retrain the model and update the parameter θ. The cost and benefit of the attribute/label will change. This process continues until all labels are obtained through query or prediction.

E. Algorithm Design

Algorithm 1 describes the CALS algorithm. It consists of model building, unified evaluation and dynamic selection of attribute values and labels, and model updating.

Line 5 calculates $P(y_i' = j|\mathbf{x}_i; \theta)$. We focus on nonunique solutions and overflow issues. First, according to (19), $\theta_j - \varphi$ is another optimal solution. The solution θ_j is not unique

$$\frac{e^{(\theta_j-\varphi)^T\mathbf{x}_i}}{\sum_{l=1}^k e^{(\theta_l-\varphi)^T\mathbf{x}_i}} = \frac{e^{\theta_j^T\mathbf{x}_i}e^{-\varphi^T\mathbf{x}_i}}{\sum_{l=1}^k e^{\theta_l^T\mathbf{x}_i}e^{-\varphi^T\mathbf{x}_i}} = \frac{e^{\theta_j^T\mathbf{x}_i}}{\sum_{l=1}^k e^{\theta_l^T\mathbf{x}_i}}. \tag{19}$$

To solve the nonunique problem, we use the method in [49] to add the weight attenuation term $(\lambda/2)\sum_{i=1}^k \sum_{j=0}^d \theta_{ij}^2$. Therefore, the improved cost function

$$J(\theta) = -\frac{1}{|U_{\mathrm{I}}|} \sum_{i \in U_{\mathrm{I}}} \sum_{j \in [1..k]} I(y_i = j) \log P(y_i = j|\mathbf{x}_i; \theta) + \frac{\lambda}{2} \sum_{i=1}^k \sum_{j=0}^d \theta_{ij}^2 \tag{20}$$

is strictly convex. We can obtain a unique solution.

In addition, overflow and underflow can be resolved by

$$\frac{e^{\theta_j^T\mathbf{x}_i - A}}{\sum_{l=1}^k e^{\theta_l^T\mathbf{x}_i - A}} = \frac{\frac{e^{\theta_j^T\mathbf{x}_i}}{e^A}}{\frac{\sum_{l=1}^k e^{\theta_l^T\mathbf{x}_i}}{e^A}} = P\left(y_i' = j|\mathbf{x}_i; \theta\right) \tag{21}$$

where A is the largest of all $\theta_j^T\mathbf{x}_i$. The maximum value of $\theta_j^T\mathbf{x}_i - A$ is 0 to avoid overflow. In $([\sum_{l=1}^k e^{\theta_l^{\mathrm{T}}\mathbf{x}_i}]/e^A)$, at least one entry is 1 to avoid underflow. Therefore, no overflow and underflow will occur.

Table II presented the time complexity of CALS. The time complexity of Algorithm 1 is

$$O\left(dn^2\right) + O\left(n^2\right) + \mathrm{O}\left(2^{|\omega_i|}n\right) + O(n) = O\left(dn^2 + 2^{|\omega_i|}n\right) \tag{22}$$

where d and n are the number of attributes and instances, respectively.

Algorithm 1 CALS

Input: An ICS-DS $S = (\mathbf{X}, \mathbf{y}, \mathbf{W}, \mathbf{M}, \mathbf{q})$.
Output: Predicted labels $\mathbf{y}' = [y_i']_{n\times 1}$.

1: $U_{\mathrm{I}} \leftarrow \emptyset$; $U \leftarrow \{1, 2, \ldots, n\}$; $U_{\mathrm{II}} \leftarrow \{1, 2, \ldots, n\}$;
2: $\left[y_i'\right]_{n\times 1} \leftarrow [-1, \ldots, -1]$;
//Step 1. Select the initial training set U_{I}
3: $U_{\mathrm{I}} \leftarrow select(X, \min\{\sqrt{n}, 0.1n\})$; // select critical instances by Eq. (6)
4: $U_{\mathrm{II}} \leftarrow U - U_{\mathrm{I}}$;
//Step 2. Unified evaluation and dynamic selection
5: **repeat**
6: Obtain the parameter $[\theta]_{k\times(d+1)}$ by Eq. (9);
7: **for** ($i \leftarrow 1$ to $|U_{\mathrm{II}}|$) **do**
8: Obtain $f_a(\mathbf{x}_i)$ and F^* by Eqs. (15) and (16);
9: Obtain $f_l(\mathbf{x}_i)$ by Eq. (18);
10: Obtain $f(\mathbf{x}_i)$;
11: **end for**
12: $s^* = \arg\min_{i\in U_{\mathrm{II}}} f(\mathbf{x}_i)$;
//Step 3. Classify instance $\mathbf{x}_{s^*}$ and updating U_{I}
13: $y_{s^*}' \leftarrow classify(\mathbf{x}_{s^*}, F^*)$;
14: $U_{\mathrm{II}} \leftarrow U_{\mathrm{II}} - s^*$;
15: **if** ($\mathbf{x}_{s^*}$ is complete) **then**
16: $U_{\mathrm{I}} \leftarrow U_{\mathrm{I}} \cup s^*$;
17: **end if**
18: **until** ($U_{\mathrm{II}} == \emptyset$)
19: **return** $\mathbf{y}' \leftarrow [y_i']_{n\times 1}$;

TABLE II
COMPUTATIONAL COMPLEXITY OF ALGORITHM 1

Lines	Complexity	Description
Lines 3	$O(dn^2)$	Form the initial training set
Lines 6	$O(n^2)$	Training softmax θ model
Lines 8	$\mathrm{O}(2^{\lvert\omega_i\rvert}n)$	Calculate attribute cost $f_a(\mathbf{x})$
Lines 12	$O(n)$	Search the instance with minimal total cost
Total	$O(dn^2)+O(n^2)+\mathrm{O}(2^{\lvert\omega_i\rvert}n)+O(n)=O(dn^2+2^{\lvert\omega_i\rvert}n)$	

$2^{|w_i|}$ is the number of missing attribute values imputation schemes.

TABLE III
DATASET INFORMATION

ID	Name	Source	Domain	$\lvert U\rvert$	$\lvert C\rvert$	$\lvert V_d\rvert$	Data Type
1	Pathbased	Synthetic	N/A	300	2	3	Continuous
2	Jain	Synthetic	N/A	373	2	2	Continuous
3	Sizes5	Synthetic	N/A	1000	2	4	Continuous
4	Iris	UCI	Life	150	4	3	Continuous
5	Breast	UCI	Life	277	9	2	Mixed
6	Cleveland	UCI	Life	297	13	5	Continuous
7	Led7digit	UCI	Computer	500	7	10	Continuous
8	Australian	UCI	Financial	690	14	2	Continuous
9	Tic-tac-toe	UCI	Game	958	9	2	Discrete
10	MovementAAL	UCI	Computer	13197	4	2	Continuous
11	Skin	UCI	Computer	245057	3	2	Continuous
12	Titanic	KEEL	Social	2201	3	2	Continuous

V. EXPERIMENTS

In this section, we compare CALS with four sets of algorithms, including cost-sensitive learning algorithms, cost-sensitive active learning algorithms, missing value filling algorithms, and AFA algorithms. Test data includes synthetic, benchmark, and field datasets. The CALS source code is available at *https://github.com/FanSmale/CALS*.

A. Datasets and Evaluation

Table III lists 12 datasets, including three synthetic datasets, eight UCI datasets, and one obtained from the literature [50]. Data types include continuous and discrete, as well as mixed types.

The performance of the CALS algorithm is evaluated by the accuracy and the average cost

$$\text{average cost} = \frac{\sum_{(i,j)\in Q} q_j + \sum_{i=1}^{n} m_{y_i y_i'}}{n}. \tag{23}$$

Generally, the cost matrix is set by the user according to the actual scenario [51], [52]. For example, Zhou and Liu [52] provided three cost matrix setting methods considering various scenarios. For balanced data, the cost of misclassification is the same for all categories. For imbalanced data, the cost of misclassification is inversely proportional to the number of instances in its category. Considering the most common scenario, we set the cost $m_{i,j}=2$, and $q_1=q_2=\cdots=q_d=0.2$, $q_{d+1}=1$. We adopt the same cost setting for both CALS and the comparison algorithms.

B. Comparison With Cost-Sensitive Learning Algorithms

For the completeness of the experiment, we first compare the CALS algorithm with nine supervised cost-sensitive learning algorithms: 1) naïve Bayes (NB); 2) sequential minimal optimization (SMO); 3) IBK; 4) attribute selected classifier (ASC); 5) multiclass classifier (MCC); 6) decision table (DT); 7) J48; 8) random forest (RF); and 9) Hoeffding tree (HT). Nine cost-sensitive learning algorithms are tested using Weka's [53] built-in codes. The missing data first perform simple mean filling.

Fig. 2 shows the comparison of the average cost of ten algorithms at 10%, 20%, 30%, 40%, and 50% missing rates. For Jain, Sizes5, Iris, Cleveland, Led7digit, Skin, and Titanic datasets, the average cost of CALS is significantly lower than other algorithms. For example, for Size5, the average cost is 0.3136 at the 50% missing rates. Only for MovementAAL, the average cost is higher than the other comparison algorithms.

C. Comparison With Cost-Sensitive Active Learning Algorithms

Next, we compare the CALS algorithm with six cost-sensitive active learning algorithms: 1) undersampling (US) [54]; 2) SMOTE [55]; 3) oversampling (OS) [56]; 4) threshold moving (TM) [57]; 5) hard ensemble (HE) [58]; and 6) soft ensemble (SE) [58]. First, we run the active learner to obtain the training set. Second, the training set is provided to comparison algorithms to construct a classifier. Third, the results of our algorithm and the comparison classifiers are compared. The missing data perform simple mean filling.

Fig. 3 shows the average cost comparison of CALS and six cost-sensitive active learning algorithms at 10%, 20%, 30%, 40%, and 50% missing rates. For Pathbased, Jain, Sizes5, Iris, Cleveland, Led7digit, Australian, and Skin datasets, the CALS algorithm is significantly better than the other six algorithms. Meanwhile, CALS is more stable than the other six cost-sensitive active learning algorithms. CALS performance fluctuates only on the Pathbased and Breast datasets.

Table IV compares the accuracy and average cost of six algorithms at 50% missing rate. The Friedman and Nemenyi post hoc tests [59] were used to analyze the performance of the algorithms. For average cost, the average ranking obtained through the Friedman test is 3.8750, 2.8750, 4.1250, 4.5417, 5.7917, 5.4583, and 1.3333, respectively. For accuracy, the average ranking obtained through the Friedman test is 3.8570, 3.2917, 4.0417, 4.517, 5.7917, 5.4583, and 1.0000, respectively. In terms of average cost and accuracy, CALS obtains the first place in the ranking computed by Friedman's test. For average cost, CALS has the lowest average cost in nine datasets, such as Jain, Sizes5, etc.

For MovementAAL, SMOTE performance is better than CALS. MovementAAL contains a time stream of measured radio signal strength. This is typical time-series data. CALS considers only one instance at a time, and each instance is considered independently. Therefore, for MovementAAL, CALS cannot achieve the best performance.

Furthermore, we use the Nemenyi post test to analyze whether there are significant differences. Table V presents

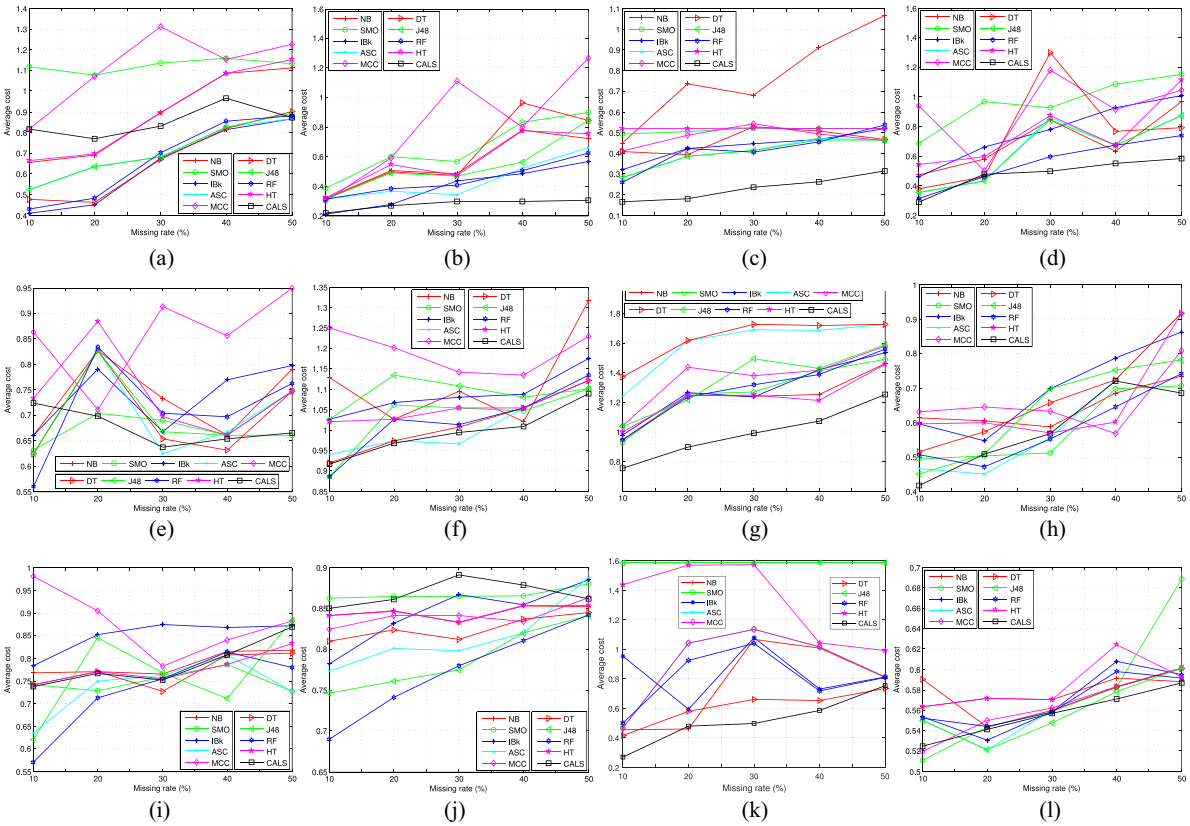


Fig. 2. Comparison of the average cost between CALS and nine supervised cost-sensitive classifier algorithms at 10%–50% missing rate. (a) Pathbased. (b) Jain. (c) Sizes5. (d) Iris. (e) Breast. (f) Cleveland. (g) Led7digit. (h) Australian. (i) Tic-tac-toe. (j) MovementAAL. (k) Skin. (l) Titanic.

TABLE IV
AVERAGE COST BETWEEN CALS AND SIX COST-SENSITIVE ACTIVE LEARNING ALGORITHMS WHEN THE DATASETS ARE MISSING AT 50%. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDFACE

	average cost							accuracy						
	US	SMOTE	OS	TM	HE	SE	CALS	US	SMOTE	OS	TM	HE	SE	CALS
Pathbased	1.2033	1.2167	1.1300	1.2033	1.2767	1.3167	**0.8733**	0.4033	0.3967	0.4400	0.4033	0.3667	0.3467	**0.6357**
Jain	0.8686	0.8740	0.8686	0.8901	0.8740	0.8686	**0.3038**	0.5684	0.5657	0.5684	0.5576	0.5657	0.5684	**0.9285**
Sizes5	1.2960	0.7960	0.8060	0.7880	0.7960	0.8120	**0.3136**	0.3620	0.6120	0.6070	0.6160	0.6120	0.6040	**0.9044**
Iris	1.1000	0.9933	1.1800	1.1000	1.1267	1.1400	**0.5827**	0.4600	0.5133	0.4200	0.4600	0.4467	0.4400	**0.7467**
Breast	0.9819	0.9747	0.9747	0.9892	1.0159	1.0159	**0.7412**	0.5126	0.5162	0.5162	0.5090	0.4632	0.4632	**0.6470**
Cleveland	1.5387	1.5522	1.5859	1.5118	1.5993	1.5724	**1.0892**	0.2391	0.2323	0.2155	0.2525	0.2088	0.2222	**0.4725**
Led7digit	1.6160	1.6080	1.6080	1.6600	1.6560	1.6240	**1.2544**	0.2020	0.2060	0.2060	0.1800	0.1820	0.1980	**0.3911**
Australian	0.9681	0.8116	0.8464	0.8348	0.8870	0.8841	**0.6858**	0.5174	0.5957	0.5783	0.5841	0.5580	0.5594	**0.6973**
Tic-tac-toe	0.9499	0.9499	0.9499	0.9499	0.9549	0.9549	**0.8693**	0.5261	0.5261	0.5261	0.5261	0.5142	0.5142	**0.5921**
MovementAAL	0.9249	**0.8739**	0.9026	0.9646	0.9308	0.9292	0.8868	0.5376	0.5354	0.5488	0.5178	0.5340	0.5348	**0.5629**
Skin	0.7561	**0.7355**	0.9018	0.9617	0.8782	0.8740	0.7524	0.6219	0.6322	0.5491	0.5191	0.5609	0.5630	**0.6745**
Titanic	**0.5797**	0.5866	0.6306	0.6206	0.6488	0.6488	0.5867	0.7601	0.7169	0.6851	0.6901	0.6761	0.6761	**0.7663**
Mean rank	3.8750	2.8750	4.1250	4.5417	5.7917	5.4583	**1.3333**	3.8750	3.2917	4.0417	4.5417	5.7917	5.4583	**1.0000**

TABLE V
COMPARISON OF POST HOC BETWEEN CALS AND SIX COST-SENSITIVE ACTIVE LEARNING ALGORITHMS

Algorithms	$z = (R_0 - R_i)/SE$		p	
	average cost	accuracy	average cost	accuracy
CALS vs. HE	5.055275	5.433239	0.000000	0.000000
CALS vs. SE	4.677310	5.055275	0.000003	0.000000
CALS vs. TM	3.637908	4.015873	0.000275	0.000059
CALS vs. US	2.881979	3.259944	0.003952	0.001114
CALS vs. OS	3.165452	3.448926	0.001548	0.000563
CALS vs. SMOTE	1.748086	2.598506	0.080449	0.009363

the p values obtained for the Nemenyi test. The significance level of $\alpha = 0.05$. In terms of average cost and accuracy, CALS is significantly better than HE, SE, TM, US, and OS. For example, for average cost, for CALS against the HE, $p = 0.000000$; for CALS against the SE, $p = 0.000003$.

D. Comparison With Missing Value Imputation and Active Feature Acquisition Algorithms

Finally, we compare the CALS algorithm with two missing value filling algorithms: 1) BPCA [48] and 2) GESI [60] and two AFA algorithms: 1) CALF [61] and 2) AFASMC[62]. BPCA [48] is a well-known microarray missing value estimation method. GESI [60] is the typical nonparametric neural network ensemble method of multiple imputation. CALF [61] is an active feature value acquisition method. AFASMC [62] is a method of AFA through a supervision matrix.

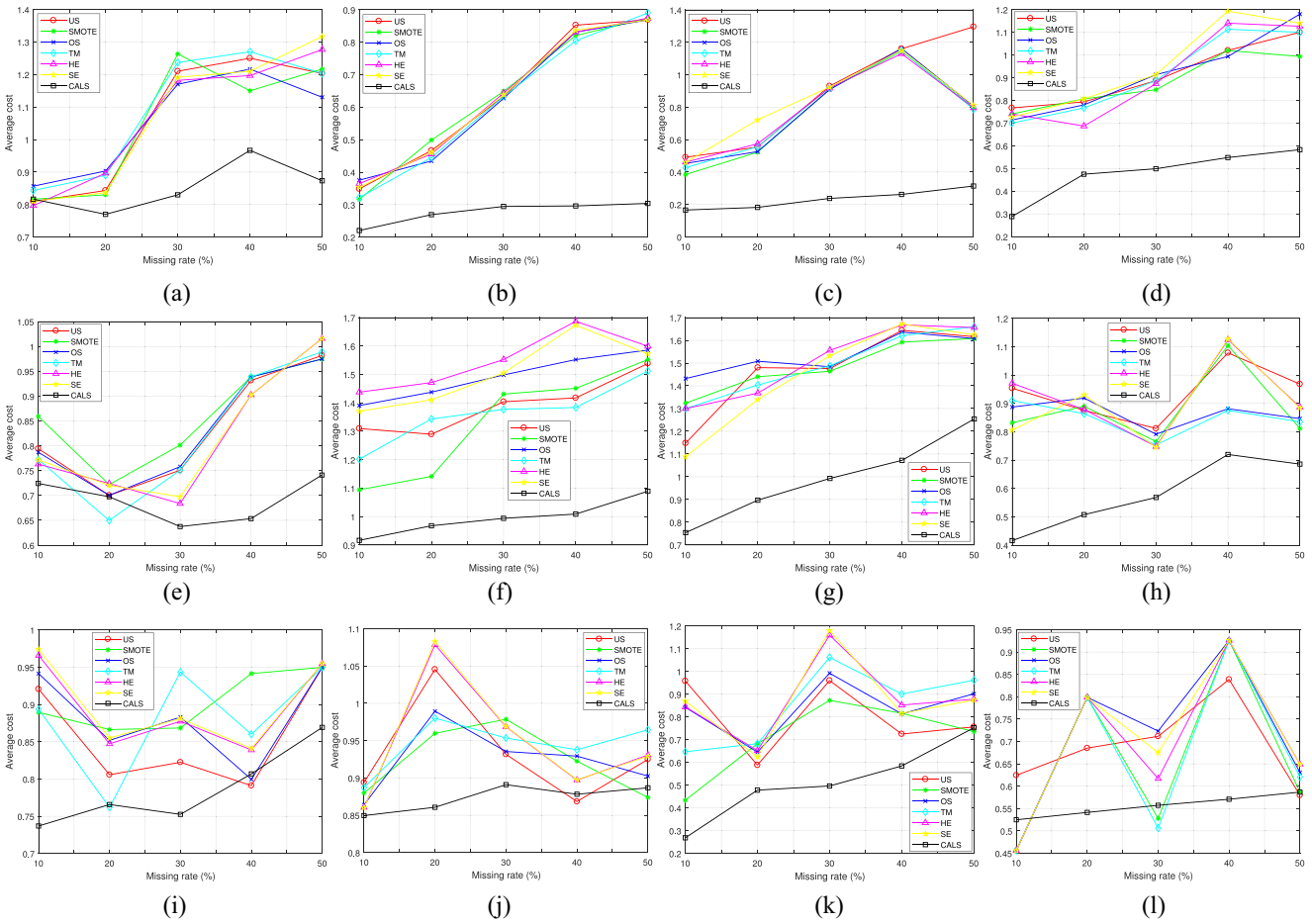


Fig. 3. Comparison of the average cost between CALS and six cost-sensitive active learning algorithms at different missing rates. (a) Pathbased. (b) Jain. (c) Sizes5. (d) Iris. (e) Breast. (f) Cleveland. (g) Led7digit. (h) Australian. (i) Tic-tac-toe. (j) MovementAAL. (k) Skin. (l) Titanic.

TABLE VI
AVERAGE COST OF CALS AND MISSING VALUE FILLING ALGORITHMS WHEN THE DATASETS ARE MISSING AT 50%. THE BEST RESULT IS HIGHLIGHTED IN BOLDFACE

	average cost					accuracy				
	CALF	GESI	BPCA	AFASMC	CALS	CALF	GESI	BPCA	AFASMC	CALS
Pathbased	**0.7813**	0.9567	1.0167	1.0040	0.8733	**0.8100**	0.7067	0.6767	0.4981	0.6357
Jain	0.5657	0.6515	0.4048	1.1223	**0.3038**	0.7828	0.7239	0.8472	0.4833	**0.9285**
Sizes5	0.5388	1.0220	0.3780	0.6996	**0.3136**	0.7970	0.5390	0.8610	0.6912	**0.9044**
Iris	0.6680	0.6600	0.8200	0.6240	**0.5827**	0.7407	0.7200	0.6400	0.7178	**0.7467**
Breast	0.9220	**0.6679**	0.6895	0.7480	0.7412	0.6643	**0.7148**	0.7040	0.6432	0.6470
Cleveland	1.4714	1.2088	1.1616	1.3724	**1.0892**	0.4242	0.4444	0.4680	0.3153	**0.4725**
Led7digit	1.7656	1.4480	1.4880	1.2728	**1.2544**	0.2240	0.3260	0.3060	0.3801	**0.3911**
Australian	1.2142	0.6420	**0.5870**	0.6023	0.6858	0.5609	0.7290	**0.7565**	0.7435	0.6973
Tic-tac-toe	1.0173	0.8382	0.8987	**0.8200**	0.8693	0.6044	**0.6190**	0.5887	0.6184	0.5921
MovementAAL	0.9316	0.9027	0.8988	**0.8712**	0.8868	**0.5773**	0.5586	0.5606	0.5736	0.5629
Skin	1.1033	1.1192	1.2085	0.7615	**0.7524**	0.4735	0.4404	0.3958	0.6302	**0.6745**
Titanic	1.1588	0.9609	0.8201	0.6374	**0.5867**	0.4557	0.5298	0.6002	0.7381	**0.7663**
Mean rank	4.0833	3.25	3.1667	2.75	**1.75**	3.1667	3.1667	3.1667	3.3333	**2.1667**

Fig. 4 shows the average cost comparison of CALS, CALF, GESI, BPCA, and AFASMC algorithms at 10%, 20%, 30%, 40%, and 50% missing rate. For Jain, Sizes5, Iris, Cleveland, Led7digit, Skin, and Titanic, the CALS algorithm is significantly better than GESI, BPCA, and CALF. For example, for Titanic, the average cost of CALS is 0.5867 at 50% missing rate. For Jain, the average cost of CALS is only 0.8733 at 50% missing rate.

Table VI compares the accuracy and average cost at the 50% missing rate. For average cost, the average ranking obtained by the Friedman test is 4.0833, 3.25, 3.1667, 2.7500, and 1.75, respectively. For accuracy, the average ranking obtained by the Friedman test is 3.1667, 3.1667, 3.1667, 3.3333, and 2.1667, respectively. In terms of average cost and accuracy, CALS obtain the first place in the ranking computed by Friedman's test. The CALS algorithm has the lowest average cost in seven datasets, such as Jain, Sizes5, etc.

Furthermore, we adopt the Nemenyi post hoc test to analyze whether there are significant differences. Table VII presents the p values obtained for the Nemenyi test. The significance

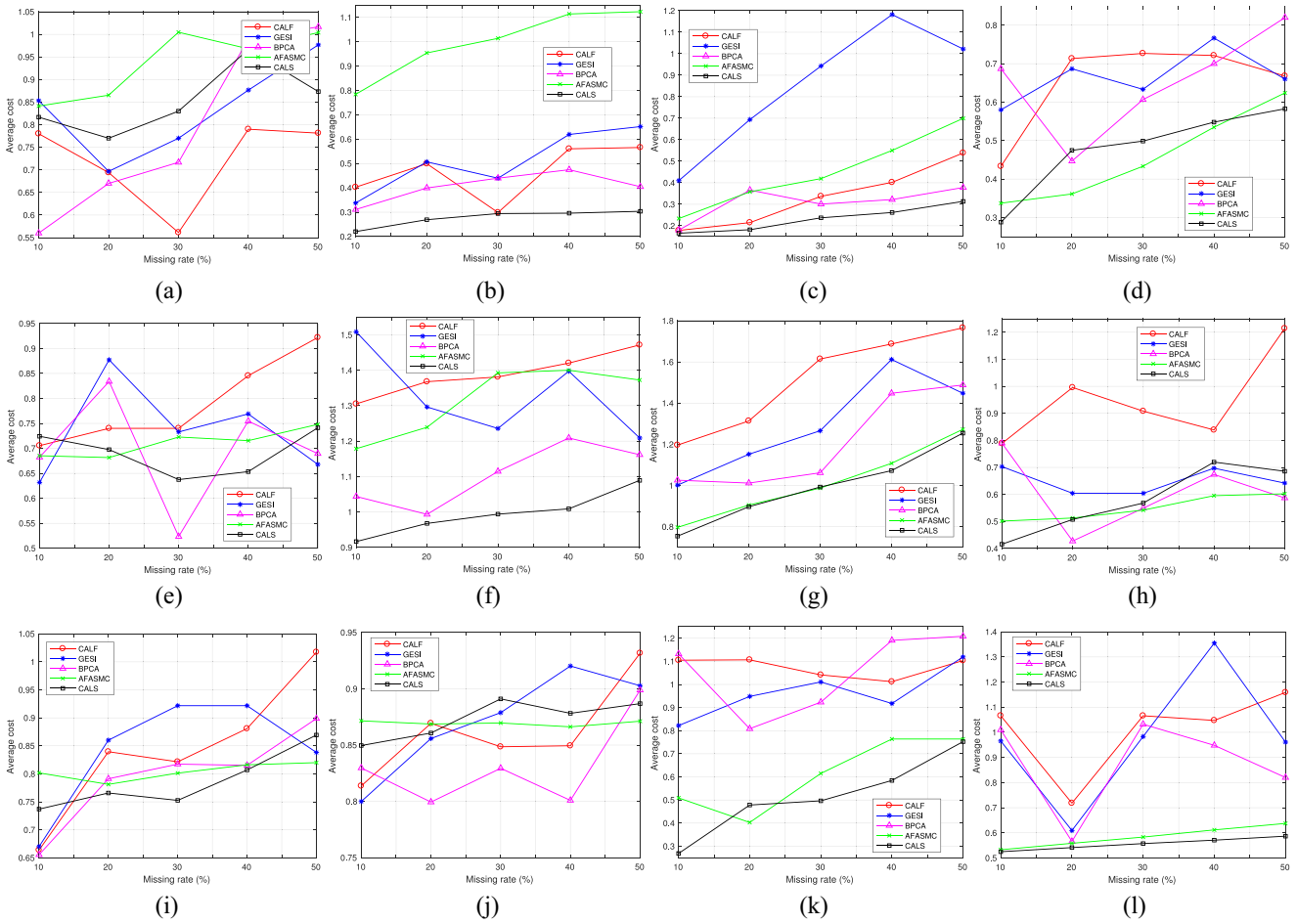


Fig. 4. Comparison of the average cost between CALS and missing value filling algorithms at different missing rates. (a) Pathbased. (b) Jain. (c) Sizes5. (d) Iris. (e) Breast. (f) Cleveland. (g) Led7digit. (h) Australian. (i) Tic-tac-toe. (j) MovementAAL. (k) Skin. (l) Titanic.

TABLE VII
COMPARISON OF POST HOC BETWEEN CALS AND MISSING VALUE FILLING ALGORITHMS

Algorithms	$z = (R_0 - R_i)/SE$		p	
	average cost	accuracy	average cost	accuracy
CALS vs. CALF	3.6148	1.5492	0.0003	0.1213
CALS vs. GESI	2.3238	1.5492	0.0201	0.1213
CALS vs. BPCA	2.1947	1.5492	0.0282	0.1213
CALS vs. AFASMC	1.5492	1.8074	0.1213	0.0707

level of $\alpha = 0.05$. For average cost, for CALS against the CALF, $p = 0.0003$, for CALS against the GESI, $p = 0.0201$; for CALS against BPCA, $p = 0.0282$; and for CALS against AFASMC, $p = 0.1213$. CALS significant better than CALF, GESI, BPCA, and AFASMC.

E. Comparison on Practical Logging Interpretation Data

We adopt the actual logging interpretation data, which contain 4149 instances and seven attributes. The decision attribute is binary: gas layer or not.

The decision attributes are gas and nongas layers. The objective is to predict the geological stratification. The actual missing rate is 28%. For these data, if the traditional NB classifier is used, the prediction accuracy is only 0.6510. A large number of missing attribute values hinder the availability and learnability of data. We adopt CALS, CALF, GESI, BPCA, and AFASMC algorithms to classify the logging interpretation to improve prediction accuracy. Table VIII shows the accuracy and average cost of several comparison algorithms. The CALS algorithm increases the prediction accuracy to 0.8002 and the cost is reduced to 0.4242. CALS has good performance in the actual field data.

TABLE VIII
COMPARISON OF ON LOGGING INTERPRETATION DATA BETWEEN CALS AND MISSING VALUE FILLING ALGORITHMS

Data	NB	CALF	GESI	BPCA	AFASMC	CALS
accuracy	0.6510	0.7397	0.6602	0.7223	0.8139	0.8002
averCost	0.7052	0.5932	0.6869	0.5625	0.7146	0.4242

F. Discussion

We are now able to analyze and summarize the experimental results. CALS was more accurate than popular cost-sensitive, cost-sensitive active learning, missing value imputation, and AFA algorithms. It was effective for most datasets with different distributions and shapes. It obtained excellent classification performance through querying few critical attributes and instances.

However, the performance of CALS was influenced by the selection of the initial instances. The initial training set needed to have instances of all category labels; otherwise, it was impossible to obtain an accurate cost evaluation model. This is the reason that the datasets Skin and Titanic did not achieve the best performance.

VI. Conclusion and Further Works

Accurately classifying data with label acquisition expensive and attributes missing is a challenging but meaningful issue. We proposed the CALS algorithm to resolve this issue. First, we defined the new data model to consider the incomplete data, attribute query cost, and label query cost. Second, we designed the new problem to consider various inputs, outputs, and the optimization objective. Third, we discussed the optimization method to obtain a unified evaluation of attributes/labels. Finally, we presented the CALS algorithm to consider the specific implementation of the method. Extensive empirical studies on both benchmark datasets and real-world applications demonstrated the superiority and robustness of our proposed method.

From the viewpoint of algorithms and applications, the following research problems merit further investigation.

1) *Optimize the Selection of Initial Instances:* The selection of the initial training set affects the quality of the model. Therefore, the selection strategy needs to be further optimized to improve the performance of CALS.
2) *Deal With Incomplete and Noisy Data:* Real data typically contain a certain proportion of noise. Naturally, it is desirable to extend the CALS algorithm to handle more complex classification scenarios, including noise labels.
3) *Consider More Uncertainty:* A large number of applications will be in open scenarios and encounter many uncertainties. An important aspect of future work will be to keep the algorithm in an open environment, consider uncertainty as much as possible, and obtain a set of solutions.

References

[1] T. Aittokallio, "Dealing with missing values in large-scale studies: Microarray data imputation and beyond," *Briefings Bioinform.*, vol. 11, no. 2, pp. 253–264, 2010.

[2] C. Genes, I. Esnaola, S. M. Perlaza, L. F. Ochoa, and D. Coca, "Robust recovery of missing data in electricity distribution systems," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 4057–4067, Jul. 2019.

[3] G. Riccardi and D. Hakkani-Tur, "Active learning: Theory and applications to automatic speech recognition," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 4, pp. 504–511, Jul. 2005.

[4] S.-A. Zahin, C.-F. Ahmed, and T. Alam, "An effective method for classification with missing values," *Appl. Intell.*, vol. 48, no. 10, pp. 3209–3230, 2018.

[5] J. Zhang, M. K. Clayton, and P. Townsend, "Missing data and regression models for spatial images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1574–1582, Mar. 2015.

[6] S. Moreno-Tejera, M. A. Silva-Pérez, I. Lillo-Bravo, and L. Ramírez-Santigosa, "Solar resource assessment in Seville, Spain. Statistical characterisation of solar radiation at different time resolutions," *Solar Energy*, vol. 132, pp. 430–441, Jul. 2016.

[7] P. R. Harvey, B. Stephen, and S. Galloway, "Classification of AMI residential load profiles in the presence of missing data," *IEEE Trans. Smart Grid*, vol. 7, no. 4, pp. 1944–1945, Jul. 2016.

[8] Z.-Q. Zheng and B. Padmanabhan, "On active learning for data acquisition," in *Proc. ICDM*, 2002, pp. 562–569.

[9] P. Melville, M. Saartsechansky, F. Provost, and R. J. Mooney, "Active feature-value acquisition for classifier induction," in *Proc. ICDM*, vol. 19, 2004, pp. 483–486.

[10] O. Kwon and J. M. Sim, "Effects of data set features on the performances of classification algorithms," *Expert Syst. Appl.*, vol. 40, no. 5, pp. 1847–1857, 2013.

[11] S.-H. Ji and L. Carin, "Cost-sensitive feature acquisition and classification," *Pattern Recognit.*, vol. 40, no. 5, pp. 1474–1485, 2007.

[12] S.-C. Zhang, Z.-X. Qin, C. X. Ling, and S.-L. Sheng, "'Missing is useful': Missing values in cost-sensitive decision trees," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 12, pp. 1689–1693, Dec. 2005.

[13] B. Settles, "Active learning literature survey," Dept. Comput. Sci., Univ. Wisconsin–Madison, Madison, WI, USA, Rep. 1648, 2009.

[14] B. Settles, M. Craven, and L. Friedland, "Active learning with real annotation costs," in *Proc. NIPS*, 2008, pp. 1–10.

[15] S. Anand, S. Mittal, O. Tuzel, and P. Meer, "Semi-supervised kernel mean shift clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1201–1215, Jun. 2014.

[16] U. Maulik and D. Chakraborty, "A self-trained ensemble with semisupervised SVM: An application to pixel classification of remote sensing imagery," *Pattern Recognit.*, vol. 44, no. 3, pp. 615–623, 2011.

[17] S.-J. Huang, R. Jin, and Z.-H. Zhou, "Active learning by querying informative and representative examples," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 1936–1949, Oct. 2014.

[18] M. Wang, F. Min, Z.-H. Zhang, and Y.-X. Wu, "Active learning through density clustering," *Expert Syst. Appl.*, vol. 85, pp. 305–317, Nov. 2017.

[19] D. J. Miller and H. Uyar, "A mixture of experts classifier with learning based on both labelled and unlabelled data," in *Advances in Neural Information Processing Systems*, M. Mozer, M. Jordan, and T. Petsche, Eds., vol. 9. MIT Press, Dec. 1996.

[20] B. Geng, D.-C. Tao, C. Xu, L.-J. Yang, and X.-S. Hua, "Ensemble manifold regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1227–1233, Jun. 2012.

[21] R. Wang, X.-Z. Wang, S. Kwong, and C. Xu, "Incorporating diversity and informativeness in multiple-instance active learning," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1460–1475, Dec. 2017.

[22] M. Wang, Y.-Y. Zhang, and F. Min, "Active learning through multi-standard optimization," *IEEE Access*, vol. 7, pp. 56772–56784, 2019.

[23] I. B. Aydilek and A. Arslan, "A hybrid method for imputation of missing values using optimized fuzzy c-means with support vector regression and a genetic algorithm," *Inf. Sci.*, vol. 233, no. 5, pp. 25–35, 2013.

[24] E.-L. Silva-Ramírez, R. Pino-Mejías, M. López-Coello, and M.-D. Cubiles-de-la-Vega, "Missing value imputation on missing completely at random data using multilayer perceptrons," *Neural Netw.*, vol. 24, no. 1, pp. 121–129, 2011.

[25] C. Jiang and Z.-J. Yang, "CKNNI: An improved kNN-based missing value handling technique," in *Advanced Intelligent Computing Theories and Applications*, vol. 9227. Cham, Switzerland: Springer, 2015, pp. 441–452.

[26] C. T. Tran, M.-J. Zhang, and P. Zhang, "A genetic programming-based imputation method for classification with missing data," in *Genetic Programming*, vol. 9594. Cham, Switzerland: Springer, 2016, pp. 149–163.

[27] C. D. Bodt, D. Mulders, M. Verleysen, and J. A. Lee, "Nonlinear dimensionality reduction with missing data using parametric multiple imputations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1166–1179, Apr. 2019.

[28] P. Melville, M. Saartsechansky, F. Provost, and R. J. Mooney, "An expected utility approach to active feature-value acquisition," in *Proc. ICDM*, 2005, pp. 745–748.

[29] W.-P. Loh and C. W. H'ng, "Data treatment effects on classification accuracies of bipedal running and walking motions," in *Proc. SCDM*, 2014, pp. 477–485.

[30] K. Sankaranarayanan and A. Dhurandhar, "Intelligently querying incomplete instances for improving classification performance," in *Proc. CIKM*, 2013, pp. 2169–2178.

[31] M. Saar-Tsechansky, P. Melville, and F. Provost, "Active feature-value acquisition," *Manag. Sci.*, vol. 55, no. 4, pp. 664–684, 2009.

[32] Q. Yang, C. Ling, X.-Y. Chai, and R. Pan, "Test-cost sensitive classification on data with missing values," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 5, pp. 626–637, May 2006.

[33] B. M. Shahshahani and D. A. Landgrebe, "The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon," *IEEE Trans. Geosci. Remoto Sens.*, vol. 32, no. 5, pp. 1087–1095, Sep. 1994.

[34] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 44–53, 2018.

[35] R. Virginia, "Learning classification with unlabeled data," in *Advances in Neural Information Processing Systems*, vol. 6. Red Hook, NY, USA: Curran, 1994, pp. 112–119.

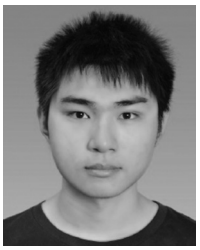
[36] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proc. NIPS*, vol. 17, 2005, pp. 529–536.
[37] S.-L. Sun and D. R. Hardoon, "Active learning with extremely sparse labeled examples," *Neurocomputing*, vol. 73, nos. 16–18, pp. 2980–2988, 2010.
[38] E. Lughofer and M. Pratama, "Online active learning in data stream regression using uncertainty sampling based on evolving generalized fuzzy models," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 1, pp. 292–309, Feb. 2018.
[39] V. Fedorov, "Optimal experimental design," *Wiley Interdiscipl. Rev. Comput. Stat.*, vol. 2, no. 5, pp. 581–589, 2010.
[40] M. Wang, Y. Lin, F. Min, and D. Liu, "Cost-sensitive active learning through statistical methods," *Inf. Sci.*, vol. 501, pp. 460–482, Oct. 2019.
[41] S. Sinha, S. Ebrahimi, and T. Darrell, "Variational adversarial active learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5971–5980.
[42] J. Xiao, X. Zhou, Y. Zhong, L. Xie, X. Gu, and D. Liu, "Cost-sensitive semi-supervised selective ensemble model for customer credit scoring," *Knowl. Based Syst.*, vol. 189, Feb. 2020, Art. no. 105118.
[43] K. Y. Ma and C.-I. Chang, "Iterative training sampling coupled with active learning for semisupervised spectral–spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8672–8692, Oct. 2021.
[44] L. Zhang and D. Zhang, "Evolutionary cost-sensitive extreme learning machine," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 12, pp. 3045–3060, Dec. 2017.
[45] A. Rodriguez and A. Laio, "Machine learning clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.
[46] S. Horiguchi, D. Ikami, and K. Aizawa, "Significance of softmax-based features in comparison to distance metric learning-based features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1279–1285, May 2020.
[47] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, pp. 195–202, Sep. 2017.
[48] F.-C. Meng, C. Cai, and H. Yan, "A bicluster-based Bayesian principal component analysis method for microarray missing value estimation," *IEEE J. Biomed. Health Inform.*, vol. 18, no. 3, pp. 863–871, May 2014.
[49] R. Memisevic, C. Zach, G. Hinton, and M. Pollefeys, "Gated softmax classification," in *Proc. NIPS*, 2010, pp. 1603–1611.
[50] R. J. M. Dawson, "The 'unusual episode' data revisited," *J. Stat. Educ.*, vol. 3, no. 3, pp. 1–7, 1995.
[51] K.-M. Ting, "An instance-weighting method to induce cost-sensitive trees," *IEEE Trans. Knowl. Data Eng.*, vol. 14, no. 3, pp. 659–665, May/Jun. 2002.
[52] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 1, pp. 63–77, Jan. 2006.
[53] G. Holmes, A. Donkin, and I. H. Witten, "WEKA: A machine learning workbench," in *Proc. ANZIIS*, vol. 30, 2002, pp. 357–361.
[54] X.-Y. Liu, J.-X. Wu, and Z.-H. Zhou, "Exploratory under-sampling for class-imbalance learning," in *Proc. ICDM*, vol. 39, 2006, pp. 965–969.
[55] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.
[56] S. Ando and C.-Y. Huang, "Deep over-sampling framework for classifying imbalanced data," in *Machine Learning and Knowledge Discovery in Databases*, vol. 10534. Cham, Switzerland: Springer, 2017, pp. 770–785.
[57] M. A. Maloof, "Learning when data sets are imbalanced and when costs are unequal and unknown," in *Proc. ICML*, vol. 21, 2003, pp. 1–2.
[58] E. Bauer and R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," *Mach. Learn.*, vol. 36, no. 1, pp. 105–139, 1999.
[59] O. Reyes, A. H. Altalhi, and S. Ventura, "Statistical comparisons of active learning strategies over multiple datasets," *Knowl. Based Syst.*, vol. 145, pp. 274–288, Apr. 2018.
[60] I. A. Gheyas and L. S. Smith, "A neural network-based framework for the reconstruction of incomplete data sets," *Neurocomputing*, vol. 73, nos. 16–18, pp. 3039–3065, 2010.
[61] W. Huang, F. Min, and J. Ren, "Missing value imputation with active learning based on collaborative filtering weighted prediction," *J. Nanjing Univ.*, vol. 54, no. 4, pp. 758–765, 2018.
[62] S.-J. Huang, M. Xu, M.-K. Xie, M. Sugiyama, G. Niu, and S. Chen, "Active feature acquisition with supervised matrix completion," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2018, pp. 1571–1579.

Min Wang received the B.S. degrees in electronic and information from the School of Electronic and Information Engineering, Southwest University, Chongqing, China, and the M.S. degree in metering technology from the School of Electrical Engineering and Information, Southwest Petroleum University, Chengdu, China.

She is a Professor of Southwest Petroleum University. She presided over a number of longitudinal tasks including the National Natural Science Foundation of China. She has authored over 10 refereed papers in various journals and conferences, including the *Knowledge-Based Systems* and *Information Science*. She has obtained several patents and software copyrights. Her research interests include data mining and active learning.

Chunyu Yang received the B.S. degree in automation from the Taiyuan University of Science and Technology, Taiyuan, China. He is currently pursuing the postgraduate degree with Southwest Petroleum University, Chengdu, China.

His field of study is active learning and deep learning.



Fei Zhao received the M.S degree in automation control engineering from Southwest Petroleum University, Chengdu, China, where he is currently pursuing the postgraduate degree.

His field of study is active learning and deep learning.

Fan Min (Member, IEEE) received the M.S. and Ph.D. degrees in computer science from the School of Computer Science and Engineering, University of Electronics Science and Technology of China, Chengdu, China, in 2000 and 2003, respectively.

He visited the University of Vermont, Burlington, VT, USA, from 2008 to 2009. He is currently a Professor with Southwest Petroleum University, Chengdu. He has published more than 100 refereed papers in various journals and conferences, including the *Information Sciences*, *International Journal of Approximate Reasoning*, and *Knowledge-Based Systems*. His current research interests include data mining, recommender systems, active learning, and granular computing.

Xizhao Wang (Fellow, IEEE) received the M.S. degree from Shanghai Jiaotong University, Shanghai, China, and the Ph.D. degree in engineering from the Harbin Institute of Technology, Harbin, China.

Before 2014, he served as a Professor and the Dean of the School of Mathematics and Computer Sciences, Hebei University, Baoding, China. Then, he worked as a Professor with the Big Data Institute, Shenzhen University, Shenzhen, China. His major research interests include uncertainty modeling and machine learning for big data. He has edited more than 10 special issues and published three monographs, two textbooks, and over 200 peer-reviewed research papers. As a Principle Investigator (PI) or co-PI, he has completed over 30 research projects. He has supervised more than 100 M.Phil. and Ph.D. students.

Prof. Wang was the recipient of the IEEE SMCS Outstanding Contribution Award in 2004 and the IEEE SMCS Best Associate Editor Award in 2006. He is the Previous BoG Member of IEEE SMC Society, the Chair of IEEE SMC Technical Committee on Computational Intelligence, the Chief Editor of *Machine Learning and Cybernetics* Journal, and an Associate Editor for a couple of journals in the related areas. He is the General Co-Chair of the 2002–2017 International Conferences on Machine Learning and Cybernetics, cosponsored by IEEE SMCS. He was a Distinguished Lecturer of the IEEE SMCS.