Neurocomputing 388 (2020) 157–169

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Conformal prediction based active learning by linear regression optimization

Sergio Matiz*, Kenneth E. Barner

Department of Electrical and Computer Engineering, University of Delaware, 140 Evans Hall, Newark, DE 19716, United States

ARTICLE INFO

Article history:
Received 31 October 2018
Revised 2 March 2019
Accepted 10 January 2020
Available online 13 January 2020

Communicated by Dr. Bo Du

Keywords:
Conformal prediction
Active learning
Linear regression
Image classification

ABSTRACT

Conformal prediction uses the degree of strangeness (nonconformity) of data instances to determine the confidence values of new predictions. We propose a conformal prediction based active learning algorithm, referred to as CPAL-LR, to improve the performance of pattern classification algorithms. CPAL-LR uses a novel query function that determines the relevance of unlabeled instances through the solution of a constrained linear regression model, incorporating uncertainty, diversity, and representativeness in the optimization problem. Furthermore, we present a nonconformity measure that produces reliable confidence values. CPAL-LR is implemented in conjunction with support vector machines, sparse coding algorithms, and convolutional networks. Experiments conducted on face and object recognition databases demonstrate that CPAL-LR improves the classification performance of a variety classifiers, outperforming previously proposed active learning techniques, while producing reliable confidence values.

1. Introduction

Conformal prediction (CP) was proposed by Vovk, Shafer and Gammerman [1] based on the principles of algorithmic randomness and transductive inference. CP uses the degree of strangeness (nonconformity) of new data instances to determine the confidence values of new predictions.

The CP framework yields a set of predicted class labels with guaranteed error rate, a property referred to as *validity*. Moreover, unlike Bayesian methods [2], CP is only based on the assumption that the data are independent and identically distributed, *i.e.*, no knowledge on the prior is required. The applications of conformal prediction include: breast cancer diagnosis, clinical diagnosis and prognosis of depression, arrhythmia detection, and robust face recognition [3–6].

Transductive conformal prediction for active learning has been reported in the literature [7,8]. Ho et al. [7] proposed the query by transduction, which sequentially selects the most uncertain instances from an unlabeled pool. The disadvantage of transductive inference is computational inefficiency, which restricts its applicability.

Inductive conformal prediction emerged as an alternative to transductive inference [1,9–11]. The application of inductive conformal predictors (ICP) to decision trees is studied in [10]. Papadopoulos et al. [9] use uncertainty to perform active learning within the CP framework improving the performance of neural networks.

Active learning has been extensively applied in domains like classification, image segmentation and information retrieval [12–15]. Active learning can be roughly divided into two categories: online and pool based. In online active learning, the learner processes data instances sequentially, as they are observed, and the model has to decide whether or not to query the observed instance to update the hypothesis. Pool based active learning is further divided into serial query based active learning and batch mode active learning. In a serial query based active learning system, the classifier is updated after every single query [16–18]. This approach is time consuming since the model needs to be retrained frequently. Batch mode active learning techniques address this issue by selecting multiple instances at a time from the unlabeled pool for annotation [12,19,20]. This work focuses on batch mode active learning with applications to image classification.

Batch mode active learning based on both uncertainty and diversity has been shown to improve the performance of pattern classification algorithms [12,14,20–25], avoiding the selection of similar instances that do not provide additional information. Several approaches based on similarity measures have been proposed to measure diversity [12,26]. For instance, Brinker [12] proposes a diversity criterion based on the cosine angle distance between two different instances. Gu et al. [26] employ the Gaussian kernel to measure the similarity between two instances. Xu et al. [19] apply clustering to measure diversity. Shi et al. [14] combine spatial coherence with clustering to improve the performance of

* Corresponding author.
E-mail address: smatiz@udel.edu (S. Matiz).

https://doi.org/10.1016/j.neucom.2020.01.018
0925-2312/

remote sensing image classification. Chakraborty et al. [24] combine entropy with diversity in a single query function, solving the active leaning problem using quadratic optimization. However, query functions based only on uncertainty and diversity may lead to the selection of outliers that are not representative of the data.

Uncertainty and information density (representativeness) have been combined in a single query function to select instances that are informative and also representative [15,27–31]. Li and Guo [28] propose a systematic way for measuring and combining uncertainty and representativeness of unlabeled instances for active learning. Wang et al. [30] combine clustering with active/semi-supervised learning to select instances that are representative and discriminative. Du et al. [31] derive a robust multi-label active learning algorithm based on the maximum correntropy criterion, merging uncertainty and representativeness in a single optimization problem.

Machine learning algorithms, such as support vector machines (SVMs), sparse coding, and convolutional neural networks (CNNs), have recently gained interest in a variety of problems in image processing and computer vision, including face recognition, classification, and image denoising [32–41]. Support vector machines have received ample treatment being both theoretically well founded and showing excellent generalization performance in practice [32,33]. Sparse coding algorithms incorporating class label information in the objective function have been show to produce state-of-the-art results for image classification [34,42]. Moreover, CNNs have led to a series of breakthroughs in image classification. LeCun et al. [37] developed a multilayer CNN, referred to as LeNet-5, for classification of handwritten digits. Krizhevsky et al. [43] propose a classic CNN architecture, referred to as AlexNet, showing significant improvements upon previous methods for image classification.

Despite these advances, traditional pattern classification algorithms produce simple predictions, without any associated confidence values. Therefore, they require modifications, or additional techniques to be implemented in conjunction with them [44–46] to perform active learning, since confidence values and a measure of uncertainty are required for that purpose. Moreover, as uncertainty measures differ from each other across different types of classifiers, it becomes difficult to implement the same active learning technique over different classification algorithms without performing modifications.

In light of the above, we propose a conformal prediction based active learning algorithm, referred to as CPAL-LR. Different from previous work on active learning, which is mostly based on query functions that linearly combine different selection criteria [12,26], the proposed approach uses a novel query function that determines the relevance of unlabeled instances through the solution of a constrained linear regression model, incorporating uncertainty, diversity, and representativeness in the optimization problem. By using the CP framework, CPAL-LR offers two advantages: (1) it is flexible across different pattern classification algorithms, since CP produces uncertainty measures that are normalized, regardless of the type classifier being used, (2) in addition to performance enhancement, CPAL-LR produces reliable confidence values.

The contributions established in CPAL-LR are threefold: first, we propose a novel query function that determines the relevance of unlabeled instances through the solution of a constrained linear regression model; second, we present a nonconformity measure that produces reliable confidence values. Third, we derive an active learning algorithm within the CP framework.

This paper is organized as follows. First, an introduction to conformal prediction and active learning is provided in Section 2. The CPAL-LR algorithm for active learning is described in Section 3. Furthermore, the proposed query function and nonconformity measures are presented. Experiments conducted on two face databases, the Extended YaleB database [47] and the AR face database [48], and one object recognition database, Caltech101 [49,50], are presented in Section 4. Moreover, the quality of the CPAL-LR confidence values is demonstrated through experimentation.

2. Background

2.1. Conformal prediction

CP uses the nonconformity of new data instances to determine the confidence values of new predictions. For an arbitrary significance level $\epsilon \in [0, 1]$, CP yields a set Ψ^{ϵ} containing the correct class label of a given data instance with probability $(1-\epsilon)$, a property referred to as validity [51]. Define a bag of size $n \in \mathbb{R}$ as a collection of n elements, some of which may be identical with each other. Let that bag be denoted as $\wr z_1, \ldots, z_n \wr$. Define $z_i = (\mathbf{x}_i, h_i)$, where $\mathbf{x}_i$ represents a data instance and h_i its corresponding class label.

A nonconformity measure $A(\wr z_1, \ldots, z_n \wr, z)$ is a function producing a nonconformity score $\alpha \in \mathbb{R}$, representing how different z is from the elements in the bag $\wr z_1, \ldots, z_n \wr$. The nonconformity score of an element z_i in $\wr z_1, \ldots, z_n \wr$ is obtained as $\alpha_i = A(\wr z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n \wr, z_i)$.

In addition, we can measure the conformity of $\mathbf{x}_{n+j}$ to class q using *p-values*, which are defined as [1]:

$$p(\alpha_{n+j}^{(\mathcal{H}_q)}) = \frac{count\{i : \alpha_i > \alpha_{n+j}^{(\mathcal{H}_q)}\}}{n+1}, \tag{1}$$

where $\alpha_{n+j}^{(\mathcal{H}_q)}$ is the nonconformity score of $\mathbf{x}_{n+j}$, under the null hypothesis $\mathcal{H}_q$, and $p(\alpha_{n+j}^{(\mathcal{H}_q)})$ is its p-value. Notice that the p-value is highest when all previous nonconformity scores, $\alpha_1, \ldots \alpha_n$, are higher than that of the new instance, $\alpha_{n+j}^{(\mathcal{H}_q)}$, meaning that $\mathbf{x}_{n+j}$ best conforms to class q. CP uses Eq. (1) to predict the label for $\mathbf{x}_{n+j}$ using the highest p-value. In addition, for each new instance $\mathbf{x}_{n+j}$ and significance level $\epsilon \in [0, 1]$, we form a set of labels $\Psi_{n+j}^{\epsilon} = \{i : p(\alpha_{n+j}^{(\mathcal{H}_i)}) > \epsilon\}$ containing the correct class label for $\mathbf{x}_{n+j}$ with probability $(1-\epsilon)$, according to the validity property.

The p-values are also used to compute the quality of information [8,16]. Ho and Wechsler [16] define the quality of information of instance $\mathbf{x}_{n+j}$ as

$$s(\mathbf{x}_{n+j}) = p_{n+j}^{(1)} - p_{n+j}^{(2)}, \tag{2}$$

where $p_{n+j}^{(1)}$ and $p_{n+j}^{(2)}$ are the largest and second largest p-values for instance $\mathbf{x}_{n+j}$, respectively. The uncertainty of an instance $\mathbf{x}_{n+j}$, within the CP framework, can be defined as:

$$I(\mathbf{x}_{n+j}) = 1 - s(\mathbf{x}_{n+j}). \tag{3}$$

2.1.1. Inductive conformal predictors

Inductive predictors first learn a classification rule, which is then used to make new predictions. Therefore, the underlying algorithm is applied only once, saving significant computation time. For a new instance $\mathbf{x}_{n+j}$, ICPs perform the following steps:

- Split the training set of size n into two smaller sets, the proper training set of size $\ell = n - r$ and the calibration set of size r, where r is a parameter of the algorithm.
- Employ the proper training set $(z_1, \ldots, z_\ell)$ to generate a classification rule C_{prop} using the underlying algorithm.
- Assign a nonconformity score to each one of the instances in the calibration set. This results in the sequence

  $$\alpha_{\ell+1}, \ldots, \alpha_{\ell+r}.$$

– Compute the p-values for $\mathbf{x}_{n+j}$ for all possible null hypotheses $\mathcal{H}_i$ by applying (1) to the sequences

$$\alpha_\ell, \ldots, \alpha_{\ell+r}, \alpha_{n+j}^{(\mathcal{H}_i)} \text{ for } i = 1, \ldots, M.$$

– Predict the classification with the largest p-value and calculate the uncertainty $I(\mathbf{x}_{n+j})$.

2.2. Query functions for active learning

A variety of query functions have been studied in the literature to the select unlabeled instances [12,20,26–28,52,53]. A brief summary of some of the most popular selection criteria is presented below.

2.2.1. Multiclass-level uncertainty (MCLU)

The MCLU criterion selects the unlabeled instances that have maximum uncertainty (minimum confidence) about their correct label among all instances in the unlabeled pool. For instance, let us consider a SVM classifier. The confidence value associated with $\mathbf{x}_j$, denoted as c_j, can be computed as $c_j = d_j^{(1)} - d_j^{(2)}$ [12], where $d_j^{(1)}$ and $d_j^{(2)}$ are the largest and second largest Euclidean distances from an instance $\mathbf{x}_j$ to the separating hyperplanes, respectively.

In the CP framework, the uncertainty given by Eq. (3) is equivalent to the confidence value c_j. Several works, including [7,9,10], have successfully applied active learning to ICPs based on the uncertainty criterion.

2.2.2. Cluster based diversity (CBD)

Clustering techniques group similar instances into the same clusters. Since the instances within the same cluster are correlated and provide similar information, a representative instance is selected for each cluster. In [54], k-means is used to obtain a number of clusters equal to the number of instances to be selected, denoted as N_{AL}. The instance closest to each of the cluster centers is selected.

2.2.3. Combination of uncertainty and diversity

Uncertainty and diversity can be used jointly to enhance the performance of active learning [12,20,26].

The following optimization problem combines uncertainty and diversity in a unique query function

$$\mathbf{x}_t = \underset{\mathbf{x}_i \in T_u/T_s}{\arg\min} \left\{ \rho |c_j| + (1-\rho) \max_{\mathbf{x}_j \in T_s} S_{(\cdot)}(\mathbf{x}_i, \mathbf{x}_j) \right\}, \tag{4}$$

where $S_{(\cdot)}(\mathbf{x}_i, \mathbf{x}_j)$ is a similarity measure, T_s contains the set of selected instances for training (the most uncertain and diverse), T_u denotes the set containing the $L \leq |U|$ most uncertain instances, T_u/T_s represents the set of instances of T_u that are not contained in the current set T_s, $S_{(\cdot)}(\mathbf{x}_i, \mathbf{x}_j)$ represents a similarity measure applied to instances $\mathbf{x}_i$, and $\mathbf{x}_j$, and $\rho \in [0, 1]$ provides the tradeoff between uncertainty and diversity. The first instance of T_d is selected as the most uncertain instance in T_u. The algorithm stops when the number of selected instances in T_d is equal to the number of desired instances N_{AL}.

A variety of similarity measures have been used in the literature for active learning [12,20,26]. Brinker [12] use the cosine angle distance to measure the similarity between instances $\mathbf{x}_i$ and $\mathbf{x}_j$, whereas Gu et al. [26] employ the Gaussian kernel.

3. CPAL-LR: conformal prediction based active learning by linear regression optimization

We propose a conformal prediction based active learning algorithm, referred to as CPAL-LR. The proposed approach uses a novel query function that considers informativeness, diversity, and representativeness as the selection criteria. Furthermore, we present a nonconformity measures that produces reliable confidence values. In the remainder of this section, the proposed query function and nonconformity measures are introduced, and the CPAL-LR algorithm is described.

3.1. CPAL-LR query function

The proposed query function determines the relevance of unlabeled instances through the solution of a constrained linear model, incorporating informativeness, diversity, and representativeness in the optimization problem. Define $U = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L\}$ as the unlabeled pool. Let $\mathbf{Q} \in \mathbb{R}^{L \times L}$ be a kernel distance matrix, containing the distances between each one of the elements in the unlabeled pool. The entries $q_{ij} \in [0, 1]$ in matrix $\mathbf{Q}$ are computed as:

$$\mathcal{K}_\eta(\mathbf{x}_i, \mathbf{x}_j) = q_{ij} = \exp\left(-\frac{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)}{\eta} \right). \tag{5}$$

Let $\mathbf{y} \in \mathbb{R}^L$ be a vector consisting of elements y_i, containing the value of informativeness associated with instances $\mathbf{x}_i \in U$ ($i = 1, \ldots, L$), calculated according to Eq. (3). Let $\mathbf{D} \in \mathbb{R}^{L \times L}$ be a positive diagonal matrix, whose diagonal elements $d_i \in [1, 0]$ provide a measure of the representativeness (information density) of instances $\mathbf{x}_i$. The value d_i decreases when instance $\mathbf{x}_i$ is located in a densely propulated region, otherwise the value d_i increases. The proposed approach obtains a vector $\hat{\mathbf{w}} \in \mathbb{R}^L$, consisting of elements $\hat{w}_i$, containing the relevance values associated with instances $\mathbf{x}_i$ by solving the following optimization problem:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\arg\min} \ ||\mathbf{Q}\mathbf{w} - \mathbf{y}||_2^2 + \lambda ||\mathbf{D}\mathbf{w}||_2^2 \tag{6}$$
$$s.t. \ \mathbf{0} \leq \mathbf{w} \leq \mathbf{1},$$

which is a generalized ridge regression problem, penalized by the diagonal matrix $\mathbf{D}$.

Expanding the first term in Eq. (6) we have

$$(\mathbf{Q}\mathbf{w} - \mathbf{y}) = \underbrace{\begin{bmatrix} q_{11} & q_{12} & \ldots & q_{1L} \\ q_{21} & q_{22} & \ldots & q_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ q_{L1} & q_{L2} & \ldots & q_{LL} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_L \end{bmatrix}}_{\text{diversity}} - \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_L \end{bmatrix}.$$

Notice that the values w_j are weighed by the terms q_{ij}. The weights q_{ij} increase when instances $\mathbf{x}_i$ and $\mathbf{x}_j$ are close to each other ($q_{ij} = 1$, for $i = j$). Since the solution $0 \leq \tilde{w}_j \leq 1$, the instances whose informativeness is high, and are also different from each other, receive a low penalty. Conversely, the instances that are close to each other, *i.e.*, they are not diverse, receive a higher penalty. Therefore, the term $||\mathbf{Q}\mathbf{w} - \mathbf{y}||_2^2$ accounts for diversity, and the parameter η in Eq. (5) provides a tradeoff between informativeness and diversity.

Expanding the second term in (6) we obtain

$$\mathbf{D}\mathbf{w} = \underbrace{\begin{bmatrix} d_1 & 0 & \ldots & 0 \\ 0 & d_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & d_N \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_L \end{bmatrix}}_{\text{representativeness}}.$$

The elements d_i penalize the solution $\hat{\mathbf{w}}_i$ depending on the representativeness of instance $\mathbf{x}_i$. When $\mathbf{x}_i$ is located in a densely populated region, the value d_i decreases (representative, low penalty). Conversely, when x_i is located in a sparsely populated region, d_i increases (not representative, high penalty). Therefore, the parameter λ controls the penalty associated with representativeness, which is used to filter possible outliers.

The expression in (6) can be rewritten as:

$$\begin{aligned}\hat{\mathbf{w}} &= \arg\min_{\mathbf{w}} \ ||\tilde{\mathbf{Q}}\mathbf{w} - \tilde{\mathbf{y}}||_2^2 \\ &= \arg\min_{\mathbf{w}} \ \mathbf{w}^T\tilde{\mathbf{Q}}^T\tilde{\mathbf{Q}}\mathbf{w} - \mathbf{w}^T\tilde{\mathbf{Q}}^T\tilde{\mathbf{y}} \\ &s.t.\ \mathbf{0} \le \mathbf{w} \le \mathbf{1},\end{aligned} \tag{7}$$

where $\tilde{\mathbf{Q}} = \left[\mathbf{Q}\sqrt{\lambda}\mathbf{D}\right]^T \in \mathbb{R}^{2L\times L}$, and $\tilde{\mathbf{y}} = [\mathbf{y}\ \mathbf{0}]^T \in \mathbb{R}^{2L}$. Notice that the expression in (7) is a quadratic programming (QP) optimization problem.

After the optimization problem in (7) is solved, the number of desired instances, N_{AL}, associated with the highest relevance values w_j $(j = 1, \dots, L)$ are selected.

3.2. Incorporating representativeness

The second term in Eq. (6) penalizes the solution $\hat{\mathbf{w}}$ through the weights d_i in matrix $\mathbf{D}$. CPAL-LR computes the weights d_i, associated with instances $\mathbf{x}_i$ in the unlabeled pool, using the distance between $\mathbf{x}_i$ and its k-nearest neighbors, denoted as $\mathbf{z}_i^{(j)}$ [55], for $j = 1, \dots, k$. Define the value $\hat{d}_i$, associated with instance $\mathbf{x}_i$, as:

$$\hat{d}_i = \sum_{n=1}^{k} ||\mathbf{x}_i - \mathbf{z}_i^{(n)}||_2^2. \tag{8}$$

Notice that the value $\hat{d}_i$ will be low if instance $\mathbf{x}_i$ is close to its k-nearest neighbors (densely populated region, low penalty). Conversely, the value $\hat{d}_i$ will be high if instance $\mathbf{x}_i$ is far from its k-nearest neighbors (sparsely populated region, high penalty). Define $d_{\max} = \max\{d_i\}$. CPAL-LR computes the values $\hat{d}_i$ for all instances $\mathbf{x}_i$ in the unlabeled pool $(i = 1, \dots, L)$ as:

$$d_i = \hat{d}_i / d_{max}. \tag{9}$$

3.3. CPAL-LR nonconformity measure

Nonconformity measures produce nonconformity scores, which are then used to compute informativeness, as described in Section 2. Consider a classifier with M outputs, corresponding to M different class labels. Let $\mathbf{x}_j$ be an input instance, and $h_j \in \{1, \dots, M\}$ be its corresponding class label $(j = 1, 2, \dots)$. Define o_j as the jth output of the classifier. Let the estimated class label be obtained as $\max_{i=1,\dots,M} o_j^{(i)}$. The proposed nonconformity measure is given by:

$$A_{CPAL-LR}^{(\mathcal{H}_q)} := -\gamma o_j^{(q)} + (1-\gamma) \max_{i=1,\dots,M, i\neq q} o_j^{(i)}, \tag{10}$$

where $A_{CPAL-LR}^{(\mathcal{H}_q)}$ represents the proposed nonconformity measure under the null hypothesis $\mathcal{H}_q$. Assuming that the classifier is accurate and the null hypothesis $\mathcal{H}_q$ is true, the values of $A_{CPAL-LR}^{(\mathcal{H}_q)}$, will decrease (they may become negative), indicating that $\mathbf{x}$ conforms to class q. Conversely, if the null hypothesis is false, the value of $A_{CPAL-LR}^{(\mathcal{H}_q)}$ will tend increase, indicating that $\mathbf{x}$ does not conform to that particular class. The term $\gamma \in [0, 1]$ is introduced to provide a tradeoff between the importance of the first and second terms.

Notice that, regardless of the type of classifier, the nonconformity scores are normalized through the computation of p-values, which are then used to measure uncertainty, according to (3). For instance, the jth output of a linear classifier to input $\mathbf{x}$ can be defined as $o_j = \mathbf{w}_j\mathbf{x} + b_j \in \mathbb{R}$, whereas the jth output of a CNN is obtained through its forward propagation function, and it is taken directly from its last layer (usually a softmax). In both cases, the value of uncertainty $I(\cdot)$ computed within the CP framework is normalized in the range [0,1], and can be readily used for active learning without further scaling.

3.4. CPAL-LR algorithm

We propose an active learning algorithm within the CP framework. First, we split the training set, $T_{train} = \{z_1, \dots, z_n\}$, into the proper training set, $T_{prop} = \{z_1, \dots, z_\ell\}$, and the calibration set, $T_{cal} = \{z_{\ell+1}, \dots, z_{\ell+r}\}$, where $n = \ell + r$, as described in Section 2. Then, the classification rule, C_{prop}, is obtained through the underlying algorithm employing T_{prop}.

The nonconformity scores of the instances in calibration set, T_{cal}, and the unlabeled pool, U, are computed using Eq. (10) and C_{prop}. The nonconformity scores are used to measure the p-values and the uncertainty of instances in the unlabeled pool, according to Eqs. (1) and (3), respectively.

Matrix $\mathbf{Q}$ is computed using the Gaussian kernel distance as described by (5), and matrix $\mathbf{D}$ is computed using the k-nearest neighbors approach, according to Eq. (8) and (9). Then, the quadratic optimization problem described by Eq. (7) is solved to obtain the relevance $\hat{\mathbf{w}}$ of the instances in the unlabeled pool. The N_{AL} instances $\mathbf{x}_i$ whose relevance is highest are selected.

CPAL-LR returns the training set $T_{AL} = T_{prop} \cup T_s$, where T_s is the set of pairs containing the N_{AL} instances from U, with their corresponding class labels, whose associated relevance $\hat{\mathbf{w}}$ is the highest after solving the optimization problem in (7). The proposed approach is summarized in Algorithm 1.

Algorithm 1 CPAL-LR.

1: **Input:** Proper training set $T_{prop} = \{z_1, \dots, z_\ell\}$, calibration set $T_{cal} = \{z_{\ell+1}, \dots, z_{\ell+r}\}$, unlabeled pool $U = \{\mathbf{x}_{n+1}, \dots, \mathbf{x}_{n+\nu}\}$, classification rule C_{prop}, number of desired instances N_{AL}, and number of class labels M
2: Compute matrix $\mathbf{Q}$ using Eq.(5)
3: Compute the weights d_i, using equations (8), and (9), for all instances in the unlabeled pool U to form $\mathbf{D}$
4: Use Eq.(10) and the classification rule C_{prop} to calculate:
- The nonconformity scores $\{\alpha_{\ell+1}^{\mathcal{H}_{q^{(\ell+1)}}}, \dots, \alpha_{\ell+r}^{\mathcal{H}_{q^{(\ell+r)}}}\}$ corresponding to the instances in the calibration set, where $q^{(\ell+j)}$ is the correct class label of $z_{\ell+j}$, for $j \in \{1, \dots, r\}$
- The nonconformity scores $\{\alpha_{n+1}^{\mathcal{H}_i}, \dots, \alpha_{n+\nu}^{\mathcal{H}_i}\}$ corresponding to the instances in the unlabeled pool, where $i = \{1, \dots, M\}$

5: Use Eq.(1) to calculate the p-values associated with the instances in U, and obtain their uncertainty $I(\mathbf{x}_{n+j})$ through equation(3), where $j \in \{1, \dots, \nu\}$
6: Solve the quadratic optimization problem in (7) and form the set T_s containing the N_{AL} instances from U, with their corresponding class labels, whose associated relevance w is the highest
7: Construct $T_{AL} = T_{prop} \cup T_s$
8: **Output:** T_{AL}

3.5. CPAL-LR As a conformal predictor

The proposed nonconformity measure, described by Eq. (10), can be used to produce confidence values associated with new predictions, during the testing phase. After training the underlying algorithm and obtaining a classification rule, denoted as C_{train}, the nonconformity scores $\alpha_{n+j}^{(\mathcal{H}_q)}$ and p-values $p(\alpha_{n+j}^{(\mathcal{H}_q)})$, associated with a new instance $\mathbf{x}_{n+j}$, are computed according to Eqs. (10) and (1), respectively. Then, for a given significance level $\epsilon \in [0, 1]$, we form a set of labels $\Psi_{n+j}^{\epsilon} = \{i : p(\alpha_{n+j}^{(\mathcal{H}_i)}) > \epsilon\}$ containing the correct class label for $\mathbf{x}_{n+j}$ with probability $(1-\epsilon)$, according to the validity property. CPAL-LR as a conformal predictor is described in Algorithm 2.

Algorithm 2 CPAL-LR (conformal predictor).

1: **Input:** Testing instance $\mathbf{x}_{n+j}$, calibration set nonconformity scores $\{\alpha_{\ell+1}, \ldots, \alpha_{\ell+r}\}$, classification rule C_{train}, significance level ϵ, parameter γ, and number of class labels M

2: Use Eqs. (1) and (10), along with the classification rule C_{train}, to calculate:

- The nonconformity scores $\alpha_{n+j}^{\mathcal{H}_i}$ corresponding to the new instance $\mathbf{x}_{n+j}$, for the different null hypothesis $\mathcal{H}_i$ $(i = \{1, \ldots, M\})$
- The p-values $p(\alpha_{n+j}^{(\mathcal{H}_i)})$, associated with $\alpha_{n+j}^{\mathcal{H}_i}$

3: Construct the set $\Psi_{n+j}^{\epsilon} = \{i : p(\alpha_{n+j}^{(\mathcal{H}_i)}) > \epsilon\}$

4: **Output:** Ψ_{n+j}^{ϵ}

4. Experimental results

The focus of CPAL-LR is twofold: (1) to improve the performance of pattern classification algorithms through active learning; and (2) to produce reliable confidence values. Therefore, our goal is to evaluate CPAL-LR based on the improvement achieved in classification performance and the quality of the produced confidence values. This section is organized as follows. First, we present simulation results obtained on a synthetic database to provide a greater insight into the proposed query function and show its effectiveness. Then, we evaluate the performance of CPAL-LR on face and object recognition databases, providing a comparison between the proposed technique and previous work on active learning. Last, we demonstrate the quality of the confidence values obtained through CPAL-LR.

4.1. Synthetic database experiments

Experiments are conducted on a synthetic database consisting of four two-dimensional clusters, denoted as C_i $(i = 1 \ldots 4)$. The data in C_i is randomly generated following a multivariate gaussian distribution given by $\mathcal{N} \sim (\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, where $\boldsymbol{\mu}_i = (\mu_i^{(1)}, \mu_i^{(2)})$ and $\boldsymbol{\Sigma}_i = \begin{pmatrix} \sigma_i & 0 \\ 0 & \sigma_i \end{pmatrix}$ are the mean and covariance matrix of C_i, respectively. The parameters of the synthetic database are set to $\boldsymbol{\mu}_1 = (-4, 4)$, $\boldsymbol{\mu}_2 = (-4, -4)$, $\boldsymbol{\mu}_3 = (4, 4)$, $\boldsymbol{\mu}_4 = (4, -4)$ and $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = \sigma$ (different values of σ are used). The proper training set T_{prop} consists of five examples per class. The unlabeled pool U and the testing set consist of 200 images per class, each.

SVMs are employed for these experiments, using the one-*vs*-all (OVA) approach. We compare the performance improvement obtained through CPAL-LR with that of the following batch active learning approaches: random sampling, *i.e.*, we take instances from the unlabeled pool at random, active learning based on uncertainty [7,9,11], clustering [20], clustering with uncertainty [20], uncertainty and ABD [12], uncertainty and KBD [26], generalized batch mode active learning [24], BatchRank [25], and active learning by sparse selection [56], which are denoted as (rnd), AL(MCLU), AL(CBD), AL(MCLU-ECBD), AL(MCLU-ABD), AL(MCLU-KBD), AL(GBMAL), AL(BatchRank), and AL(Sparse), respectively. Random sampling is used as the baseline for the experiments.

For the proposed approach, parameter optimization using grid search is performed over the weights η and λ. The grid is formed by values $\eta \in [0.01 \times 10^{-4}, 5.0 \times 10^{-4}]$, and $\lambda \in [0, 10]$. For AL(MCLU-ABD) and AL(MCLU-KBD), the parameter ρ is optimized using the same approach. For random sampling, the training set $T_R = T_{prop} \cup T_{rnd}$ is employed, where T_{rnd} contains N_{AL} randomly selected instances from U with their corresponding class labels, and T_{prop} is the proper training set. The results for active learning are obtained using the training set $T_{AL} = T_{prop} \cup T_s$, where T_s contains N_{AL} instances selected from U using the aforementioned active learning approaches, with their corresponding class labels. Five trials are conducted to compute the classification accuracy. In each trial, the proper, calibration, training and testing sets are selected at random. For each trial, the best results are selected after parameter optimization and the average classification accuracy is presented.

Table 1
Classification accuracy for different query functions and standard deviation σ as a function of the number of selected instances N_{AL}.

Algorithm	Query func.	$\sigma = 1.5$			$\sigma = 2.0$		
		N_{AL}			N_{AL}		
		8	12	16	8	12	16
SVM	(rnd)	92.6	92.6	92.8	90.8	90.8	91.0
	AL(MCLU)	93.6	93.9	93.7	90.7	90.9	91.6
	AL(CBD)	93.8	94.1	94	90.8	91.2	91.8
	AL(MCLU-ECBD)	93.8	94.4	94.2	91	91.5	92.3
	AL(MCLU-ABD)	95.0	95.1	95.1	91.8	92.6	92.9
	AL(MCLU-KBD)	95.4	95.3	95.6	92.1	93.2	93.1
	AL(GBMAL)	94.7	95.1	95.5	91.9	92.1	93.0
	AL(BatchRank)	95.1	95.2	95.7	92	92.2	92.8
	AL(Sparse)	94.9	95.3	95.4	92.2	92.5	92.9
	CPAL-LR	**96.4**	**96.5**	**96.6**	**94.2**	**94.5**	**94.6**

Fig. 1 shows the instances selected by the proposed technique for different parameters η and λ, along with those selected by AL(MCLU-ECBD), and AL(MCLU-ABD). It is observed in Fig. 1(a) that when uncertainty is predominant ($\eta \to 0$, $\lambda = 0$) CPAL-LR selects instances that are concentrated on high uncertainty the regions, *i.e.*, the regions where clusters tend to overlap.

Fig. 1(c) shows the instances selected by CPAL-LR when representativeness is predominant ($\eta \to 0, \lambda = 18$). It is observed that the selected instances are located near the cluster centers, which correspond to densely populated regions. Fig. 1(e) shows the instances selected by CPAL-LR when uncertainty, diversity, and representativeness are considered together ($\eta = 2.5 \times 10^{-5}$, $\lambda = 12$). It is observed that the selected instances are located in high uncertainty regions, and the spread of the selected instances is lower. Different from the selected instances in Fig. 1(g) ($\eta = 5.0 \times 10^{-5}$, $\lambda = 0$), the selected instances in Fig. 1(e) are not located in sparsely populated regions, such as the ones near coordinates (-8,-8) and (-4,8).

Fig. 1(j) shows the instances selected by AL(MCLU-ECBD). It can be seen that the spread of the selected instances is high, and some of them lie in sparsely populated regions, such as (8, −4) and (8, 4). The instances selected by AL(MCLU-ABD) ($\rho = 10^{-3}$) are shown in Fig. 1(l). It is observed that most of the instances are located in high uncertainty regions, with some exceptions lying in sparsely populated regions, around (−4, −8) and (4, 8).

Table 1 shows the classification accuracy obtained on the synthetic database for different query functions and values σ, as a function of the number of selected instances N_{AL}. It is observed that the proposed technique outperforms the considered active learning approaches for all the values of σ and N_{AL}. For instance, when $\sigma = 2.0$ and $N_{AL} = 8$, the performance of (rnd), AL(MCLU), AL(CBD), AL(MCLU-ECBD), AL(MCLU-ABD), AL(MCLU-KBD), AL(GBMAL), AL(BatchRank), and AL(Sparse) is 90.8%, 90.7%, 90.8%, 91.0%, 91.8%, 92.1%, 91.9%, 92.0%, and 92.2%, respectively, whereas that of CPAL-LR is 94.2%.

To visualize the effect of the parameters η and λ on the performance of CPAL-LR we perform a second experiment. In this experiment, we conduct 100 trials. In each trial, the instances in proper, training, and testing sets are selected at random, along with those in the unlabeled pool, and the average classification

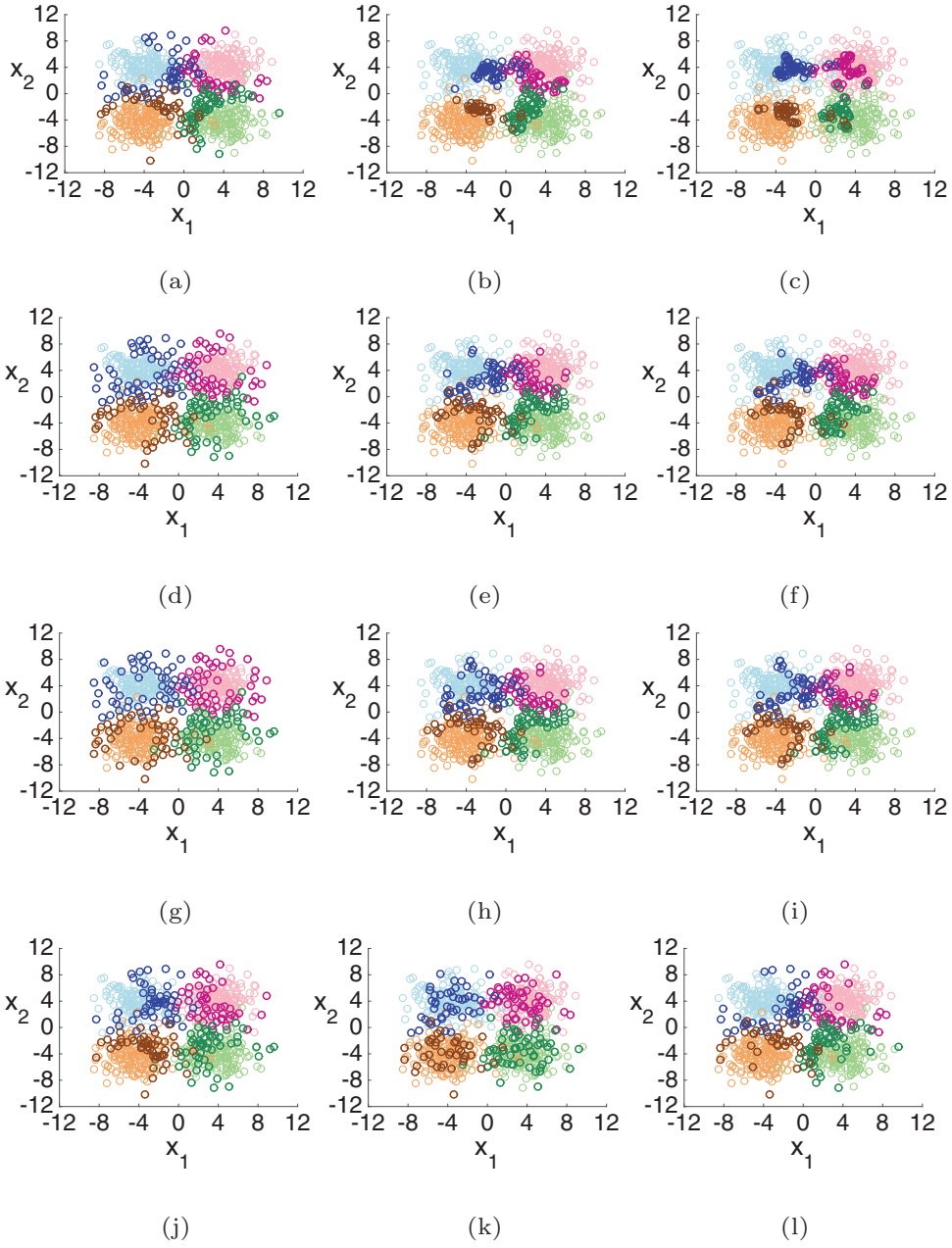


Fig. 1. Synthetic database and selected instances (highlighted) for $\sigma = 2.0$ using CPAL-LR (a) ($\eta = 10^{-9}$, $\lambda = 0$), (b) ($\eta = 10^{-9}$, $\lambda = 12$), (c) ($\eta = 10^{-9}$, $\lambda = 18$), (d) ($\eta = 2.5 \times 10^{-5}$, $\lambda = 0$), (e) ($\eta = 2.5 \times 10^{-5}$, $\lambda = 12$), (f) ($\eta = 2.5 \times 10^{-5}$, $\lambda = 18$), (g) ($\eta = 5.0 \times 10^{-5}$, $\lambda = 0$), (h) ($\eta = 5.0 \times 10^{-5}$, $\lambda = 12$) (i) ($\eta = 5.0 \times 10^{-5}$, $\lambda = 18$), (j) AL (MCLU-ECBD), (k) AL (MCLU-ABD) ($\rho = 0$), (l) AL (MCLU-ABD) ($\rho = 10^{-3}$).

accuracy is presented. The proper training set T_{prop} consists of 10 images, and the number of selected instances from the unlabeled pool is $N_{AL} = 12$. The classification accuracy as a function of η and λ for $\sigma = 1.4, 1.5, 2.0,$ and 2.3 is depicted in Fig. 2. It is observed that in the low variance (low noise) scenario, *i.e.*, Fig. 2(a) and (b) ($\sigma = 1.4$ and 1.5, respectively), the best performance is obtained for low values of λ and a combination of uncertainty and diversity ($\eta > 0$). For instance, when $\sigma = 1.4$, the best performance (99.1%) is obtained for $\eta = 5.0 \times 10^{-4}$ and $\lambda = 0$. As the variance (noise) increases, it is observed that the parameter λ becomes more relevant, as shown in Fig. 2(c) and (d) ($\sigma = 2.0$ and 2.3, respectively). For instance, in Fig. 2(d) the best performance is obtained for high values of λ and low values of η, *i.e.*, the instances that are not representative of the data (noise) are rejected by increasing the parameter λ and reducing the diversity weight η. The maximum performance for $\sigma = 2.3$ (90.2%) is obtained when $\eta = 1.0 \times 10^{-4}$ and $\lambda = 7.0$.

The synthetic database experiments demonstrate that the parameters η and λ effectively control the uncertainty, diversity, and representativeness of the selected instances, providing flexibility to the proposed approach. Moreover, it is observed that CPAL-LR outperforms other existing active learning approaches for classification.

4.2. Face and object recognition

Experiments are conducted on two face databases, the Extended YaleB database [47] and the AR face database [48], and one object recognition database, Caltech101 [49]. CPAL-LR is implemented in conjunction with three different pattern classification

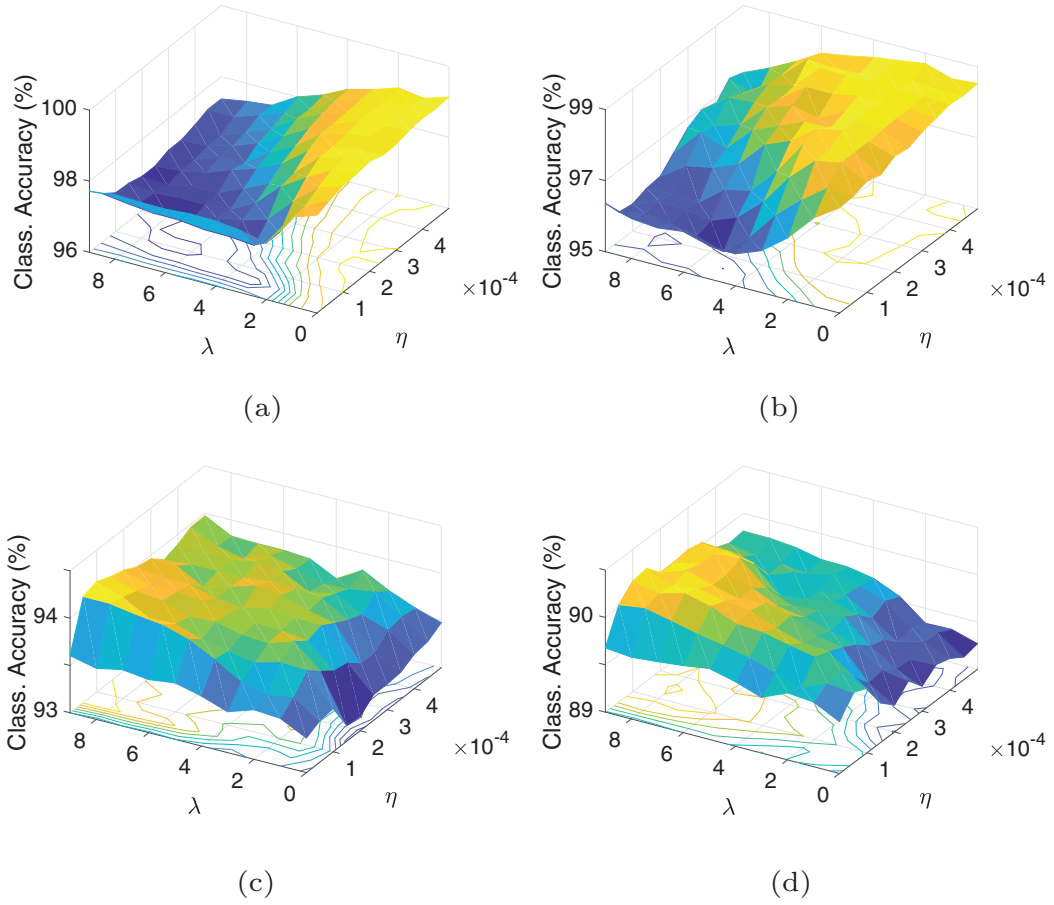


Fig. 2. Classification accuracy (%) obtained through CPAL-LR (SVMs) as a function of η and λ, (a) $\sigma = 1.4$ (b) $\sigma = 1.5$ (c) $\sigma = 2.0$ (d) $\sigma = 2.3$.

algorithms: SVMs, sparse coding (LC-RLSDLA [42]), and CNNs. We compare the performance improvement obtained through CPAL-LR with that of (rnd), AL(MCLU), AL(CBD), AL(MCLU-ECBD), AL(MCLU-ABD), AL(MCLU-KBD), AL(GBMAL), AL(BatchRank), and AL(Sparse). Random sampling is used as the baseline for the experiments, and parameter optimization is performed using grid search, as in the synthetic database experiments. The grid is formed by values $\eta \in [0.01 \times 10^{-3}, 5.0 \times 10^{-3}]$ and $\lambda \in [0, 3.5]$ for the Extended YaleB and AR databases, and $\eta \in [0.01, 0.2]$ and $\lambda \in [0, 3.5]$ for Caltech101.

For each of the experiments in this section, five trials are conducted. In each trial, the proper, calibration, training, and testing sets are selected at random. For each trial, the best results are selected after parameter optimization and the average classification accuracy is presented. The calibration set consists of 199 instances for all the experiments, which results in a resolution of 0.5% in the confidence values calculated, according to Eq. (1). The parameter γ is set to 0.5 in the nonconformity measure given by Eq. (10).

We provide a description of the considered databases and the configuration of the pattern classification algorithms for each one of them below.

The *Extended YaleB database* consists of 2414 frontal-face images of 38 people taken under varying lighting conditions. There are about 64 images for each person. For SVMs, the OVA approach is used. For LC-RLSDLA, the dictionary size is 190 (5 atoms per class). For this database, the proper training set T_{prop} consists of eight images per class, and the size of the unlabeled pool is $|U| = 912$. The feature descriptors used for SVMs and LC-RLSDLA are randomfaces [57] of size $N = 504$. For CNNs, the original images are reshaped to 32×32 pixels. The CNN architecture used for this database is described in Table 2. Example images from the Extended YaleB database are shown in Fig. 3(a).

The *AR database* contains over 4000 frontal-face images of 100 people, including facial variations and also disguises, such as sunglasses and scarves. For SVMs, the OVA approach is used. For LC-RLSDLA, the dictionary size is 400 (4 atoms per class). For this database, the proper training set T_{prop} consists of five images per class, and the size of the unlabeled pool is $|U| = 1200$. The feature descriptors used for SVMs and LC-RLSDLA are randomfaces of size $N = 540$. For CNNs, the original images are converted to greyscale and reshaped to 50×50 pixels. The CNN architecture used for this database is described in Table 3. Example images from the Extended YaleB database are shown in Fig. 3(b).

The *Caltech101* database contains 9144 images from 102 classes (101 object classes and a background class) including animals, vehicles, flowers, etc. The number of images in each category varies from 31 to 800. The samples within the same category display considerable shape variability. A subset of images from 30 different classes is used in our experiments,[1] which accounts for a total of 2475 images. For SVMs, the OVA approach is used. For LC-RLSDLA, the dictionary size is 300 (10 atoms per class). For this database, the proper training set T_{prop} consists of ten images per class, and the size of the unlabeled pool is $|U| = 900$. For SVMs and LC-RLSDLA, the SIFT descriptors are first extracted. Next, spa-

[1] The Caltech101 subset includes the classes: ketch, chandelier, hawksbill, grand piano, brain, butterfly, helicopter, menorah, kangaroo, starfish, trilobit, buddha, ewer, sunflower, scorpion, revolver, laptop, ibis, llama, umbrella, crab, crayfish, cougar face, dragonfly, ferry, flamingo, and lotus.

(a)

(b)

Fig. 3. Images from (a) extended YaleB database and (b) AR database.

Table 2
CNN architecture for the Extended YaleB database.

Layers	Filter size	Stride	Padding	Output $W \times H \times L$
Input	–	–	–	$32 \times 32 \times 1$
Conv-ReLU	5×5	1	0	$14 \times 14 \times 25$
Avg_pool	2×2	2		
Conv-ReLU	5×5	1	0	$10 \times 10 \times 65$
Avg_pool	2×2	2		
FC-ReLU dropout	–	–	–	400
FC-Softmax	–	–	–	38

Table 3
CNN architecture for the AR database.

Layers	Filter size	Stride	Padding	Output $W \times H \times L$
Input	–	–	–	$50 \times 50 \times 1$
Conv-ReLU	7×7	1	0	$22 \times 22 \times 15$
Avg_pool	2×2	2		
Conv-ReLU	7×7	1	0	$8 \times 8 \times 45$
Avg_pool	2×2	2		
FC-ReLU dropout	–	–	–	500
FC-Softmax	–	–	–	100

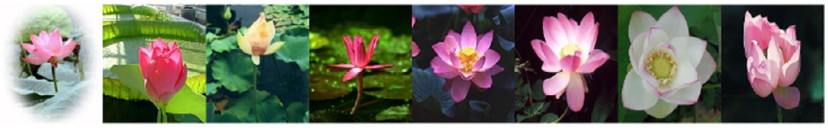
Table 4
CNN architecture for the Caltech101 database.

Layers	Filter size	Stride	Padding	Output $W \times H \times L$
Input	–	–	–	$32 \times 32 \times 1$
Conv-ReLU	5×5	1	0	$14 \times 14 \times 30$
Avg_pool	2×2	2		
Conv-ReLU	5×5	1	0	$10 \times 10 \times 60$
Avg_pool	2×2	2		
FC-ReLU	–	–	–	200
FC-Softmax	–	–	–	30

tial pyramid features are obtained from the SIFT descriptors. Then, the dimensionality of the resulting features is reduced to 3000 through PCA [34]. For CNNs, the original images are converted to greyscale and reshaped to 32×32 pixels. The CNN architecture used for this database is described inTable 4. Example images from the Caltech101 database are shown in Fig. 4.

(a) ketch

(b) llama



(c) lotus

Fig. 4. Images from the Caltech101 database.

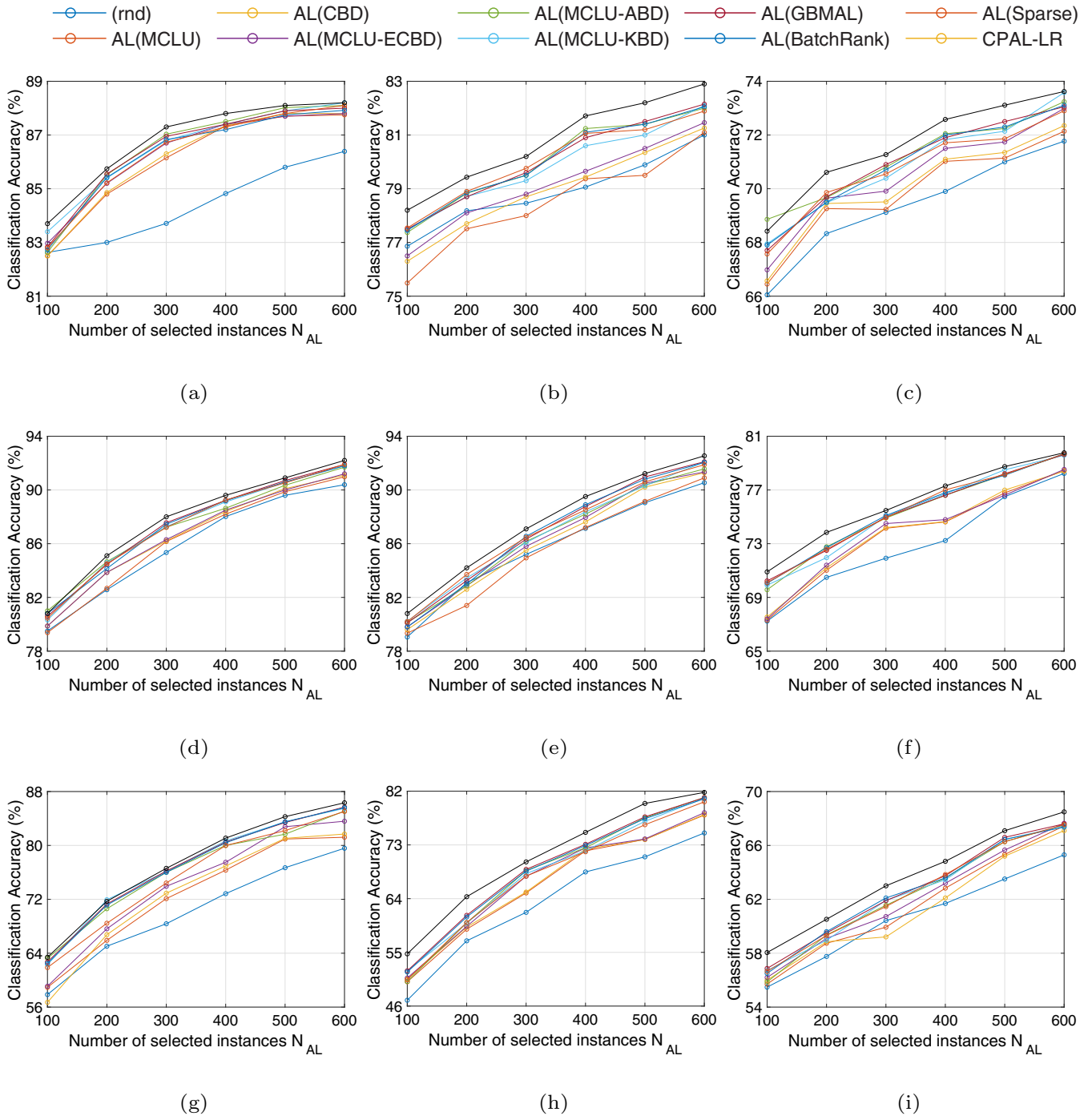


Fig. 5. Classification accuracy (%) using different active learning techniques as a function of the number of selected instances N_{AL} (a) YaleB (LC-RLSDLA) (b) AR (LC-RLSDLA) (c) Caltech101 (LC-RLSDLA) (d) YaleB (SVM) (e) AR (SVM) (f) Caltech101 (SVM) (g) YaleB (CNN) (h) AR (CNN) (i) Caltech101 (CNN).

4.3. Results: CPAL-LR for face and object recognition

The performance of CPAL-LR, along with that of the considered active learning approaches, as a function of the number of selected instances N_{AL}, for the different algorithms and databases, is shown in Fig. 5. It is observed that the performance of the different pattern classification algorithms is significantly improved through active learning, for all the considered databases. Notice that the performance of CPAL-LR compares favorably with that of the other active learning techniques. This demonstrates the effectiveness of the proposed approach.

The results for the Extended YaleB database in Fig. 5(a) (LC-RLSDLA) show that the biggest performance gain is obtained by CPAL-LR for $N_{AL} = 300$. Table 5 shows that for $N_{AL} = 300$ the classification accuracy of (rnd), AL(MCLU), AL(CBD), AL(MCLU-ECBD), AL(MCLU-ABD), AL(MCLU-KBD), AL(GBMAL), AL(BatchRank), and AL(Sparse) is 83.7%, 86.1%, 86.3%, 86.7%, 87.0%, 86.8%, 86.9%, 86.8%,and 86.7%, respectively, whereas that of CPAL-LR is 87.3%.

Table 5
Classification accuracy (%) using different active learning techniques as a function of the number of selected instances N_{AL}.

Algorithm	Query function	YaleB			AR			Caltech101		
		300	400	500	300	400	500	200	300	400
LC-RLSDLA	(rnd)	83.7	84.8	85.8	78.5	79.0	79.9	68.3	69.1	69.9
	AL(MCLU)	86.1	87.3	87.7	78.0	79.4	79.5	69.3	69.2	71.0
	AL(CBD)	86.3	87.3	87.8	78.7	79.4	80.3	69.4	69.5	71.1
	AL(MCLU-ECBD)	86.7	87.4	87.7	78.8	79.6	80.5	69.6	69.9	71.5
	AL(MCLU-ABD)	87.0	87.5	88.0	79.5	81.2	81.4	69.7	70.8	72.0
	AL(MCLU-KBD)	86.8	87.4	87.9	79.3	80.6	81.0	69.5	70.4	71.8
	AL(GBMAL)	86.9	87.4	87.9	79.6	80.9	81.5	69.7	70.9	71.9
	AL(BatchRank)	86.8	87.2	87.7	79.5	81.1	81.4	69.5	70.7	72.0
	AL(Sparse)	86.7	87.3	87.8	79.7	81.1	81.2	69.9	70.5	71.7
	CPAL-LR	**87.3**	**87.8**	**88.1**	**80.2**	**81.7**	**82.2**	**70.6**	**71.3**	**72.6**
SVM	(rnd)	85.3	88.0	89.6	85.2	87.1	89.0	70.5	71.9	73.2
	AL(MCLU)	86.2	88.2	89.9	84.9	87.2	89.1	71.0	74.2	74.6
	AL(CBD)	86.2	88.4	90.1	85.5	87.6	90.2	71.2	74.2	74.6
	AL(MCLU-ECBD)	86.3	88.5	90.0	85.8	87.9	90.5	71.4	74.5	74.8
	AL(MCLU-ABD)	87.2	88.6	90.4	86.2	88.2	90.4	72.8	74.9	76.6
	AL(MCLU-KBD)	87.3	89.1	90.5	86.1	88.4	90.3	71.9	75.0	76.7
	AL(GBMAL)	87.5	89.2	90.7	86.3	88.7	90.9	72.5	75.0	76.6
	AL(BatchRank)	87.4	89.2	90.6	86.5	88.9	90.8	72.7	75.1	76.8
	AL(Sparse)	87.2	89.2	90.5	86.4	88.5	90.6	72.6	74.9	77.0
	CPAL-LR	**88.0**	**89.6**	**90.9**	**87.1**	**89.5**	**91.2**	**73.8**	**75.5**	**77.3**
CNN	(rnd)	68.4	72.8	76.7	61.7	68.5	71.0	57.8	60.4	61.7
	AL(MCLU)	72.1	76.3	80.9	64.9	72	73.9	58.7	59.9	62.8
	AL(CBD)	72.9	76.9	81.0	65.1	72.3	73.9	58.9	59.2	62.1
	AL(MCLU-ECBD)	73.9	77.5	82.8	67.7	72.5	74.0	59.1	60.7	63.2
	AL(MCLU-ABD)	76.0	80.0	81.7	68.7	72.5	77.5	59.4	61.6	63.6
	AL(MCLU-KBD)	76.1	80.7	83.5	68.2	72.0	76.9	59.0	61.9	63.5
	AL(GBMAL)	76.2	80.5	83.4	68.9	73.1	77.7	59.5	61.9	63.7
	AL(BatchRank)	76.0	80.4	83.5	68.6	72.9	77.4	59.6	62.1	63.6
	AL(Sparse)	74.4	80.0	82.2	67.7	71.8	76.4	59.3	61.5	63.9
	CPAL-LR	**76.6**	**81.1**	**84.3**	**70.1**	**75.1**	**79.9**	**60.5**	**63.0**	**64.8**

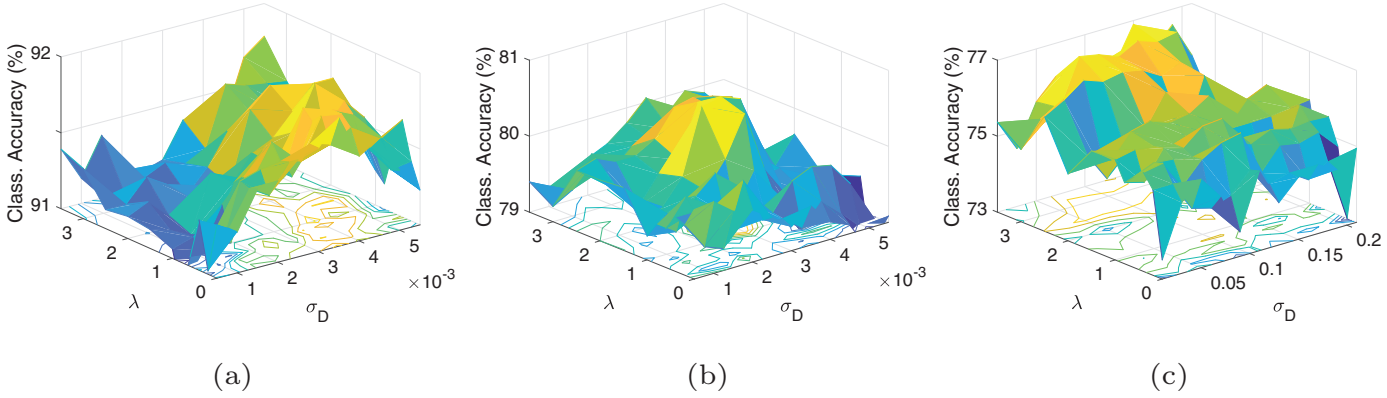


Fig. 6. Classification accuracy (%) obtained through CPAL-LR (SVMs) as a function of η and λ, (a) YaleB ($N_{AL} = 600$), (b) AR ($N_{AL} = 100$), (c) Caltech101 ($N_{AL} = 400$).

The results for the AR database in Fig. 5(h) (CNNs) show that the largest performance gain is obtained when CPAL-LR is applied for $N_{AL} = 500$, which is about 9.0%, with respect to random sampling. It can also be seen that the performance of CPAL-LR is highest among all the considered approaches for the different values of N_{AL}. For instance, for $N_{AL} = 500$, the classification accuracy of (rnd), AL(MCLU), AL(CBD), AL(MCLU-ECBD), AL(MCLU-ABD), AL(MCLU-KBD), AL(GBMAL), AL(BatchRank), and AL(Sparse) is 71.0%, 73.9%, 73.9%, 74.0%, 77.5%, 76.9%, 77.7%, 77.4%, and 76.4%, respectively, whereas that of CPAL-LR is 79.9%.

Similar results are obtained for Caltech101, as shown in Fig. 5(f) (SVMs). For instance, for $N_{AL} = 200$, the largest performance improvement is obtained when CPAL-LR is applied, which is about 3.3%, with respect to random sampling. The classification accuracy of (rnd), AL(MCLU), AL(CBD), AL(MCLU-ECBD), AL(MCLU-ABD), AL(MCLU-KBD), AL(GBMAL), AL(BatchRank), and AL(Sparse) is 70.5%, 71.0%, 71.2%, 71.4%, 72.8%, 71.9%, 72.5%, 72.7%, and 72.6%, respectively, whereas that of CPAL-LR is 73.8%.

Fig. 6 shows the classification accuracy of CPAL-LR (SVMs) as a function of the parameters η and λ for the Extended YaleB, AR, and Caltech101 databases (average over the five trials for the different values of η and λ). It is observed that the best performance is obtained for a combination of uncertainty, diversity, and representativeness, *i.e.*, η, $\lambda \geq 0$, for all the considered databases. For the Extended YaleB database ($N_{AL} = 600$), the best performance is obtained in the region $\lambda \in [0, 2]$ and $\eta \in [3.0 \times 10^{-3}, 5.0 \times 10^{-3}]$, and the classification accuracy peaks when $\eta = 4.0 \times 10^{-3}$ and $\lambda = 1.0$ (91.9%). For the AR database ($N_{AL} = 100$), the best performance is obtained in the region $\lambda \in [1, 2]$ and $\eta \in [2.0 \times 10^{-3}, 4.0 \times 10^{-3}]$, and the classification accuracy peaks when $\eta = 3.0 \times 10^{-3}$ and $\lambda = 1.5$ (80.7%). For Caltech101 ($N_{AL} = 200$), the best performance is obtained in the region $\lambda \in [2.5, 3.5]$ and $\eta \in [5.0 \times$

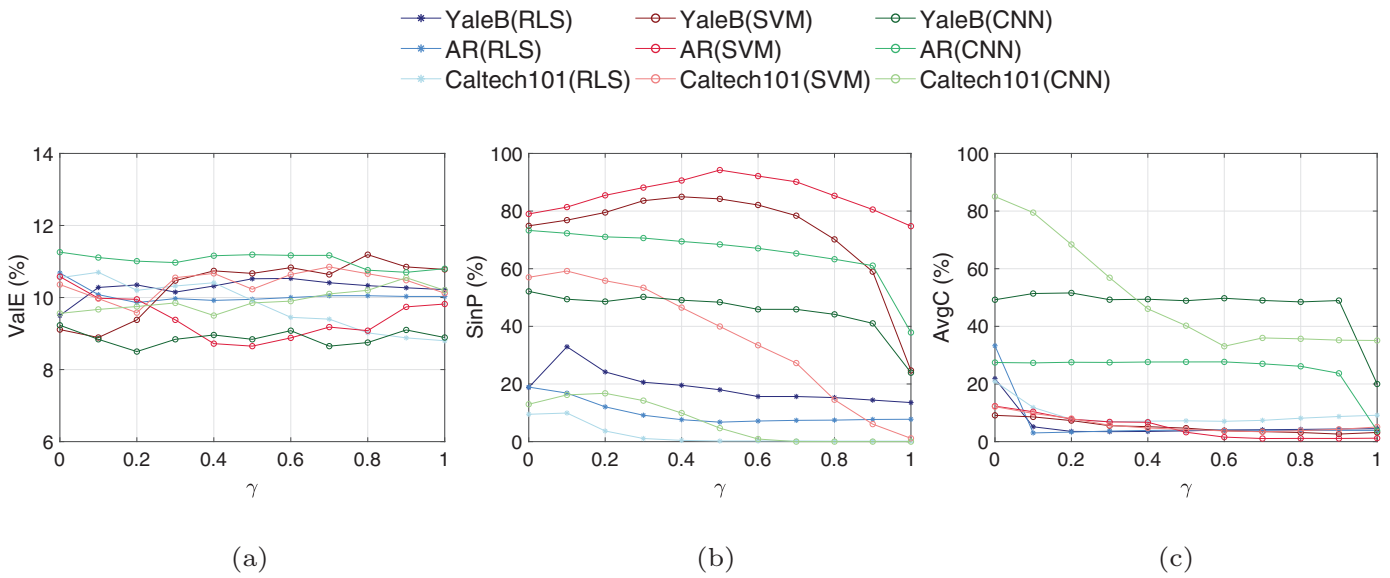


Fig. 7. Performance of the proposed nonconformity measure for $\epsilon = 0.1$ as a function of the parameter $\gamma \in [0, 1]$ using different metrics (a) ValE (b) SinP (c) AvgC.

$10^{-2}, 1.0 \times 10^{-1}]$, and the classification accuracy peaks when $\eta = 7.0 \times 10^{-2}$ and $\lambda = 3.0$ (76.9%). Similar results are obtained for LC-RLSDLA and CNNs.

The results on face and object recognition show that CPAL-LR improves the performance of several pattern classification algorithms across different databases, outperforming other state-of-the-art active learning techniques.

4.4. Results: quality of CPAL-LR confidence values

In this section, the quality of the confidence values produced by CPAL-LR (through Algorithm 2) is compared with that of the confidence values obtained through the hinge and margin nonconformity measures, which are given by $A^{(\mathcal{H}_q)}_{hinge} := 1 - \max_{i=1,\dots,M,} o^{(i)}_j$, and $A^{(\mathcal{H}_q)}_{margin} := -o^{(q)}_j + \max_{i=1,\dots,M, i\neq q} o^{(i)}_j$, respectively. Notice that the hinge and margin nonconformity measures are particular cases of the proposed nonconformity measure described by Eq. (10), when $\gamma = 1.0$ and $\gamma = 0.5$, respectively. Experiments are performed for SVMs, LC-RLSDLA, and CNNs on the Extended YaleB, AR, and Caltech101 databases. Different significance levels, $\epsilon \in [0, 1]$, are used yielding different prediction sets Ψ^{ϵ}_{n+j}, for test instances $\mathbf{x}_{n+j}$. The quality of the CPAL-LR confidence values is demonstrated using three metrics [9,58]:

- *ValE*: The percentage of errors measured as the number of times the correct label for instances $\mathbf{x}_{n+j}$ is not in Ψ^{ϵ}_{n+j}, for a given ϵ, divided by the total number of test instances [9] (ValE $\approx \epsilon$, according to the validity property)
- *SinP*: The proportion of all predictions that are singletons, *i.e.*, instances $\mathbf{x}_{n+j}$ that produce $|\Psi^{\epsilon}_{n+j}| = 1$, for a given $\epsilon \in [0, 1]$. The motivation for this metric is that singleton predictions are the most informative [58] (high SinP is preferable).
- *AvgC*: The average number of class labels in the prediction sets Ψ^{ϵ}_{n+j}, as a percentage of the total number of classes, *i.e.*, a direct measure of how good the model is at rejecting class labels (low AvgC is preferable)

Fig. 7 shows the performance of the proposed nonconformity measure, as a function of the parameter $\gamma \in [0, 1]$, using the three aforementioned metrics, for $\epsilon = 0.1$. It is observed in Fig. 7(a) that ValE fluctuates around 10%, for the different values of λ, across all the considered pattern classification algorithms and databases, which agrees with the validity property (ValE $\approx$ 10%). The parameter γ can be adjusted to obtain the desired performance. For instance, when $\gamma = 0.3$ (LC-RLSDLA, YaleB), ValeE = 10.1%. For SVMs (Caltech101, $\gamma = 0.1$), ValeE=10.0%. For CNNs (AR, $\gamma = 0.9$), ValeE=10.7%. This demonstrates the usefulness of the CPAL-LR confidence values.

Fig. 7(b) shows the behavior of singleton predictions as a function of γ (SinP). It is observed that SinP behaves differently across the considered pattern classification algorithms. The results in Fig. 7(b) show that SVMs obtain the highest number of singleton predictions, followed by CNNs and LC-RLSDLA, respectively. For the Extended YaleB database, the production of singleton predictions peaks when $\gamma = 0.1$, $\gamma = 0.4$, and $\gamma = 0.2$ for LC-RLSDLA (32.9%), SVMs (84.9%), and CNNs (52.1%), respectively.

The average number of class labels in the prediction sets, as a percentage of the total number of classes (AvgC), is shown in Fig. 7(c), for different values of γ. The results show that LC-RLSDLA and SVMs produce more discriminative sets Ψ^{ϵ} than CNNs (low AvgC). For the AR database, AvgC reaches its minimum when $\gamma = 0.1$, $\gamma = 0.7$, and $\gamma = 1.0$ for LC-RLSDLA (3.0%), SVMs (1.0%), and CNNs (3.8%), respectively.

The performance results of the hinge, margin, and CPAL-LR nonconformity measures are summarized in Table 6. For CPAL-LR, the best results are shown (from those obtained using different values of γ). Table 6 shows that CPAL-LR achieves similar or better performance than that obtained through the hinge and margin nonconformity measures, for the considered performance metrics.

5. Conclusion

A conformal prediction based active learning algorithm is presented in this work. The proposed approach uses a novel query function that determines the relevance of unlabeled instances through the solution of a constrained linear regression model, incorporating uncertainty, diversity, and representativeness in the optimization problem.

CPAL-LR is implemented in conjunction with three different pattern classification algorithms: SVMs, sparse coding (LC-RLSDLA), and CNNs. Experiments conducted on face and object recognition databases show that CPAL-LR outperforms previous work on active learning, improving performance across different pattern classification techniques and databases.

Table 6
Performance of hinge, margin, and CPAL-LR nonconformity measures.

Algorithm	Database	Confidence (%)	Performance (%)								
			Hinge			Margin			CPAL-LR		
			ValE	SinP	AvgC	ValE	SinP	AvgC	ValE	SinP	AvgC
LC-RLSDLA	YaleB	98	2.9	0.0	28.5	2.5	0.0	27.5	**2.0**	**3.8**	**24.2**
		95	5.8	0.9	10.8	5.7	2.4	8.6	**5.0**	**11.6**	**7.6**
		90	10.2	13.5	4.5	10.5	18.0	3.7	**10.1**	**32.9**	**3.4**
	AR	98	**2.7**	0.0	**26.0**	2.8	0.0	27.3	**2.7**	**2.0**	**26.0**
		95	5.4	0.0	15.8	5.4	0.0	15.8	**5.3**	**8.7**	**15.3**
		90	**10.0**	7.7	3.9	9.9	6.8	4.0	**10.0**	**18.9**	**3.0**
	Cal101	98	2.4	0.0	21.1	2.7	0.0	20.5	**2.2**	**3.7**	**19.9**
		95	5.3	0.0	15.2	4.9	0.2	**11.9**	**5.0**	**6.4**	**11.9**
		90	8.8	0.2	9.2	**9.9**	0.2	7.2	**9.9**	**9.9**	**6.9**
SVM	YaleB	98	1.8	0.4	22.2	2.5	56.2	13.9	**2.2**	**63.4**	**12.4**
		95	4.9	3.8	**7.7**	5.4	69.4	9.5	**5.0**	**72.5**	**7.7**
		90	10.8	24.7	3.2	10.7	84.2	4.7	**10.5**	**84.9**	**3.1**
	AR	98	2.4	10.2	6.7	1.8	**85.3**	11.4	**2.0**	**85.3**	**6.1**
		95	4.6	21.1	**2.9**	4.5	**88.9**	7.5	**5.0**	**88.9**	**2.9**
		90	9.8	74.7	1.2	8.6	94.2	3.2	**10.0**	**94.2**	**1.0**
	Cal101	98	2.5	0.0	16.3	2.7	25.5	**8.8**	**2.3**	**46.3**	**8.8**
		95	5.1	0.4	12.3	5.5	32.0	**6.3**	**5.0**	**50.1**	**6.3**
		90	10.1	1.2	5.1	10.2	39.9	4.2	**10.0**	**59.2**	**3.4**
CNN	YaleB	98	2.6	8.5	**44.1**	2.2	25.4	73.1	**2.0**	**28.3**	**44.1**
		95	5.1	9.9	**40.8**	**5.0**	42.4	55.8	**5.0**	**47.1**	**40.8**
		90	8.9	23.9	**20.0**	8.8	48.4	48.9	**9.2**	**52.1**	**20.0**
	AR	98	**2.1**	9.5	**16.1**	1.7	36.8	59.5	**2.1**	**39.9**	**16.1**
		95	**4.3**	18.9	**8.8**	6.2	58.5	38.0	**4.3**	**62.1**	**8.8**
		90	10.8	37.9	**3.8**	11.2	68.4	27.6	**10.7**	**73.3**	**3.8**
	Cal101	98	2.8	0.0	71.1	2.5	0.4	**63.0**	**1.7**	**2.9**	**63.0**
		95	6.5	0.0	56.8	5.9	0.8	54.1	**5.2**	**6.3**	**46.8**
		90	10.2	0.0	35.1	9.8	2.0	40.2	**10.1**	**13.3**	**33.1**

In addition to performance enhancement, CPAL-LR produces reliable confidence values that are used to predict class labels with guaranteed error rate. Experimental results show that CPAL-LR achieves similar or better performance than that obtained using previously proposed nonconformity measures.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. 1319598.

References

[1] V. Vovk, A. Gammerman, G. Shafer., Algorithmic Learning in a Random World, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
[2] T. Melluish, S. Craig, I. Nouretdinov, V. Vovk, Comparing the bayes and typicalness frameworks, in: Proceedings of the European Conference on Machine Learning (ECML 2001), 2167, Springer, 2009, pp. 360–371.
[3] A. Gammerman, I. Nouretdinov, B. Burford, A. Chervonenkis, V. Vovk, Z. Luo, Clinical mass spectrometry proteomic diagnosis by conformal predictors, Stat. Appl. Genet. Mol. Biol. 7 (2) (2008).
[4] I. Nouretdinov, S.G. Costafreda, A. Gammerman, A. Chervonenkis, V. Vovk, V. Vapnika, C.H. Fu, Machine learning classification with confidence: application of transductive conformal predictors to MRI-based diagnostic and prognostic markers in depression, Neuroimage 56 (2) (2011) 809–813.
[5] Y. Luo, A.A.-R. Bsoul, K. Najarian, Confidence-based classification with dynamic conformal prediction and its applications in biomedicine, in: Proceedings of the IEEE International Conference Engineering in Medicine and Biology Society (EMBC), 2011, pp. 353–356.
[6] V. Balasubramanian, S.S. Ho, V. Vovk, Conformal prediction for reliable machine learning: Theory, adaptations and applications, Newnes, 2014.
[7] S.-S. Ho, H. Wechsler, Query by transduction, IEEE Trans. Pattern Anal. Mach. Intell. 30 (9) (2008) 1557–1571.
[8] V. Balasubramanian, S. Chakraborty, S. Panchanathan, Generalized query by transduction for online active learning, in: Proceedings of the International Conference Computer Vision Workshops (ICCV Workshops), 2009, pp. 1378–1385.
[9] H. Papadopoulos, V. Vovk, A. Gammerman, Conformal prediction with neural networks, in: Proceedings of the IEEE International Conference Tools with Artificial Intelligence (ICTAI), 2, 2007, pp. 388–395.
[10] H. Papadopoulos, H. Boström, T. Löfström, Conformal prediction using decision trees, in: Proceedings IEEE International Conference Data Mining (ICDM), 2013, pp. 330–339.
[11] U.J. T. Löfström, H. Boström, Effective utilization of data in inductive conformal prediction using ensembles of neural networks, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN), 2, 2013, pp. 1–8.
[12] K. Brinker, Incorporating diversity in active learning with support vector machines, in: Proceedings of the International Conference Machine Learning (ICML), 2003, pp. 59–66.
[13] J. Li, J.M. Bioucas-Dias, A. Plaza, Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning, IEEE Trans. Geosci. Remote Sens. 48 (11) (2010) 4085–4098.
[14] Q. Shi, B. Du, L. Zhang, Spatial coherence-based batch-mode active learning for remote sensing image classification, IEEE Trans. Image Process. 24 (7) (2015) 2037–2050.
[15] Z. Xu, R. Akella, Y. Zhang, Incorporating diversity and density in active learning for relevance feedback, in: Proceedings of the European Conference on Information Retrieval, Springer, 2007, pp. 246–257.
[16] S.-S. Ho, H. Wechsler, Query by transduction, IEEE Trans. Pattern Anal. Mach. Intell. 30 (9) (2008) 1557–1571.
[17] C. Monteleoni, M. Kaariainen, Practical online active learning for classification, in: Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–8.
[18] D. Sculley, Online active learning methods for fast label-efficient spam filtering., in: Proceedings of the CEAS, 7, 2007, p. 143.
[19] Z. Xu, K. Yu, V. Tresp, X. Xu, J. Wang, Representative sampling for text classification using support vector machines, in: Proceedings of the European Conference on Information Retrieval, Springer, 2003, pp. 393–407.
[20] B. Demir, C. Persello, L. Bruzzone, Batch-mode active-learning methods for the interactive classification of remote sensing images, IEEE Trans. Geosci. Remote Sens. 49 (3) (2010) 1014–1031.
[21] R. Wang, S. Kwong, Active learning with multi-criteria decision making systems, Pattern Recognit. 47 (9) (2014) 3106–3119.
[22] S. Chakraborty, V. Balasubramanian, S. Panchanathan, Adaptive batch mode active learning, IEEE Trans. Neural Netw. Learn. Syst. 26 (8) (2015) 1747–1760.

[23] J. Sourati, M. Akcakaya, D. Erdogmus, T.K. Leen, J.G. Dy, A probabilistic active learning algorithm based on fisher information ratio, IEEE Trans. Pattern Anal. Mach. Intell. 40 (8) (2018) 2023–2029.
[24] S. Chakraborty, V. Balasubramanian, S. Panchanathan, Generalized batch mode active learning for face-based biometric recognition, Pattern Recognit. 46 (2) (2013) 497–508.
[25] S. Chakraborty, V. Balasubramanian, Q. Sun, S. Panchanathan, J. Ye, Active batch selection via convex relaxations with guaranteed solution bounds, IEEE Trans. Pattern Anal. Mach. Intell. 37 (10) (2015) 1945–1958.
[26] Y. Gu, Z. Jin, S.C. Chiu, Active learning combining uncertainty and diversity for multi-class image classification, IET Comput. Vis. 9 (3) (2014) 400–407.
[27] B. Settles, M. Craven, An analysis of active learning strategies for sequence labeling tasks, in: Proceedings of the Conference Empirical Methods in Natural Language Processing (EMNLP), 2008, pp. 1070–1079.
[28] X. Li, Y. Guo, Adaptive active learning for image classification, in: Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2013, pp. 859–866.
[29] Q. Li, X. Shi, L. Zhou, Z. Bao, Z. Guo, Active learning via local structure reconstruction, Pattern Recognit. Lett. 92 (2017) 81–88.
[30] Z. Wang, B. Du, L. Zhang, L. Zhang, X. Jia, A novel semisupervised active-learning algorithm for hyperspectral image classification, IEEE Trans. Geosci. Remote Sens. 55 (6) (2017) 3071–3083.
[31] B. Du, Z. Wang, L. Zhang, L. Zhang, D. Tao, Robust and discriminative labeling for multi-label active learning based on maximum correntropy criterion, IEEE Trans. Image Process. 26 (4) (2017) 1694–1707.
[32] O. Chapelle, P. Haffner, V.N. Vapnik, Support vector machines for histogram-based image classification, IEEE Trans. Neural Netw. 10 (5) (1999) 1055–1064.
[33] H. Drucker, D. Wu, V.N. Vapnik, Support vector machines for spam categorization, IEEE Trans. Neural Netw. 10 (5) (1999) 1048–1054.
[34] Z. Jiang, Z. Lin, L.S. Davis, Label consistent K-SVD: learning a discriminative dictionary for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2651–2664.
[35] J. Mairal, M. Elad, G. Sapiro, Sparse representation for color image restoration, IEEE Trans. Image Process. 17 (1) (2008) 53–69.
[36] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, Proceedings of the IEEE Conference Computer Vision and Pattern Recognition (CVPR) (2009) 1794–1801.
[37] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Comput. 1 (4) (1989) 541–551.
[38] M. Yang, H. Chang, W. Luo, Discriminative analysis-synthesis dictionary learning for image classification, Neurocomputing 219 (2017) 404–411.
[39] L. Zhang, L. Zhang, D. Tao, X. Huang, Tensor discriminative locality alignment for hyperspectral image spectral–spatial feature extraction, IEEE Trans. Geosci. Remote Sens. 51 (1) (2013) 242–256.
[40] L. Zhang, Q. Zhang, L. Zhang, D. Tao, X. Huang, B. Du, Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding, Pattern Recognit. 48 (10) (2015) 3102–3112.
[41] Y. Dong, B. Du, L. Zhang, Target detection based on random forest metric learning, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 8 (4) (2015) 1830–1838.
[42] S. Matiz, K.E. Barner, Label consistent recursive least squares dictionary learning for image classification, in: Proceedings of the IEEE International Conference Image Processing (ICIP), 2016, pp. 1888–1892.
[43] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[44] K. Wang, D. Zhang, Y. Li, R. Zhang, L. Lin, Cost-effective active learning for deep image classification, IEEE Trans. Circuits Syst. Video Technol. 27 (12) (2017) 2591–2600.
[45] C. Käding, E. Rodner, A. Freytag, J. Denzler, Active and continuous exploration with deep neural networks and expected model output changes, 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.
[46] S. Otálora, O. Perdomo, F. González, H. Müller, Training deep convolutional neural networks with active learning for exudate classification in eye fundus images, in: Proceedings of the Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis, Springer, 2017, pp. 146–154.
[47] A. Georghiades, P. Belhumeur, D. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 643–660.
[48] A. Martinez, R. Benavente, The AR face database, CVC Tech. Report # 24 (1998). http://www.cat.uab.cat/Public/Publications/1998/MaB1998/CVCReport24.pdf.
[49] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Comput. Vis. Image Underst. 106 (1) (2007) 59–70.
[50] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Comput. Vis. Image Underst. 106 (1) (2007) 59–70.
[51] G. Shafer, V. Vock, A tutorial on conformal prediction, J. Mach. Learn. Res. 9 (2008) 371–421.
[52] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, J. Mach. Learn. Res. 2 (Nov) (2001) 45–66.
[53] G. Schohn, D. Cohn, Less is more: Active learning with support vector machines, in: Proceedings of the International Conference Machine Learning (ICML), 2000, pp. 839–846.
[54] Z. Xu, K. Yu, V. Tresp, X. Xu, J. Wang, Representative sampling for text classification using support vector machines, in: Proceedings of the European Conference on Information Retrieval, Springer, 2003, pp. 393–407.
[55] S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets, in: Proceedings of the ACM SIGMOD Record, 29, ACM, 2000, pp. 427–438.
[56] G. Wang, J. Hwang, C. Rose, F. Wallace, in: Proceedings of the 2017 IEEE Nineteenth International Workshop on Multimedia Signal Processing (MMSP), 2017, pp. 1–6.
[57] W. John, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[58] U. Johansson, H. Linusson, T. Löfström, H. Boström, Model-agnostic nonconformity functions for conformal classification, in: Proceedings of the International Joint Conference on Neural Networks (IJCNN) 2017, 2017, pp. 2072–2079.

Sergio Matiz received the Bachelor of Electronics Engineering degree from the Pontificia Universidad Javeriana, Bogota, Colombia, in 2010, and the M.S. degree in electrical and computer engineering from the University of Delaware, Newark, Delaware, USA, in 2012. He is currently working toward the Ph.D. degree in electrical and computer engineering at the University of Delaware. His research interests include machine learning, signal processing, and embedded systems design.

Kenneth E. Barner received the B.S.E.E. degree (magna cum laude) from Lehigh University, Bethlehem, Pennsylvania, in 1987. He received the M.S.E.E. and Ph.D. degrees from the University of Delaware, Newark, Delaware, USA, in 1989 and 1992, respectively. He was the duPont Teaching Fellow and a Visiting Lecturer with the University of Delaware in 1991 and 1992, respectively. From 1993 to 1997, he was an Assistant Research Professor with the Department of Electrical and Computer Engineering, University of Delaware. He is currently the Charles B. Evans Professor and Chairman with the Department of Electrical and Computer Engineering, University of Delaware.