Divide and Conquer: Learning Label Distribution with Subtasks

Haitao Wu [1 2] **Weiwei Li** [2] **Xiuyi Jia** [1]

Abstract

Label distribution learning (LDL) is a novel learning paradigm that emulates label polysemy by assigning label distributions over the label space. However, recent LDL work seems to exhibit a notable contradiction: 1) existing LDL methods employ auxiliary tasks to enhance performance, which narrows their focus to specific applications, thereby lacking generalizability; 2) conversely, LDL methods without auxiliary tasks rely on losses tailored solely to the primary task, lacking beneficial data to guide the learning process. In this paper, we propose $\mathcal{S}$-LDL, a novel and minimalist solution that generates subtask label distributions, i.e., a form of extra supervised information, to reconcile the above contradiction. $\mathcal{S}$-LDL encompasses two key aspects: 1) an algorithm capable of generating subtasks without any prior/expert knowledge; and 2) a plug-and-play framework seamlessly compatible with existing LDL methods, and even adaptable to derivative tasks of LDL. Our analysis and experiments demonstrate that $\mathcal{S}$-LDL is effective and efficient. To the best of our knowledge, this paper represents the first endeavor to address LDL via subtasks.

1. Introduction

Multi-label learning (MLL) (Zhang & Zhou, 2013) handles label polysemy in a binary manner, whereas label distribution learning (LDL) (Geng, 2016) offers a more nuanced perspective by answering: "*How much does each label y describe the instance $\boldsymbol{x}$?*". This is accomplished through the concept of a *label distribution* $\boldsymbol{d}$, which is a form of probability simplex that assigns a real value (i.e., *description degree* $d_{\boldsymbol{x}}^{y}$) to each label of each instance. This form introduces a quantitative manner to address label polysemy and extends LDL's practical applications to a wider range, e.g., counting (or grading) (Geng et al., 2013; Wu et al., 2019), sentiment analysis (Chen et al., 2020; Le et al., 2023), segmentation (Gao et al., 2017; Li et al., 2023b), etc.

However, LDL encounters a spectrum of challenges: 1) $d_{\boldsymbol{x}}^{y}$s interfere with each other since $\boldsymbol{d}$ is subject to two constraints, non-negativity (i.e., $d_{\boldsymbol{x}}^{y} \geq 0$) and sum-to-one (i.e., $\sum_{y \in \mathcal{Y}} d_{\boldsymbol{x}}^{y} = 1$), and $\boldsymbol{d}$ is often formed from mixture distributions, posing significant hurdles for fitting, particularly when employing a maximum entropy model (Shen et al., 2017); 2) $\boldsymbol{d}$s are usually obtained via crowdsourcing, which is time-consuming and labor-intensive, so one often copes with scarce and low-quality datasets (Wang et al., 2023). These *two key issues* stand as formidable barriers to performance improvement in LDL.

With the widespread use of multi-task learning, some LDL work tries to compensate for performance from the perspective of *auxiliary tasks*, which are learned concurrently alongside the primary task, thereby refining its representations and ultimately boosting performance. Unfortunately, though these methods can exploit additional supervised information, they 1) neglect the first key issue mentioned above; and 2) require prior/expert knowledge (e.g., facial characteristics (Chen et al., 2020), pathology criteria (Wu et al., 2019), emotion wheel theory in psychology (Yang et al., 2017a), etc.), limiting their generalizability to those corresponding specific applications. Conversely, LDL methods that do not take advantage of auxiliary tasks, despite their efforts in loss function engineering and network structure design, they 1) neglect the second key issue mentioned above; and 2) focus solely on one aspect of label correlations (e.g., correlation of local instances (Jia et al., 2019), ranking relation (Jia et al., 2023), suboptimal label (Wang, Jing and Geng, Xin, 2019), etc.), each with its own set of limitations.

The generalizability appears to conflict with the ability to exploit additional data, so benefiting from both simultaneously seems elusive. However, we can still see the light from some MLL methods, which partition the label space and apply operations on the subspaces (Tsoumakas et al., 2008; 2010). These methods construct *subtasks* without involving extra knowledge. Intuitively, in the context of LDL, reliable su-

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China [2]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China. Correspondence to: Xiuyi Jia <jiaxy@njust.edu.cn>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

pervised information can be generated from these subtasks, which can eventually be aggregated and reconstructed to the information of the primary task via ensemble strategies. In this paper, we introduce $\mathcal{S}$-LDL, a novel and minimalist LDL algorithm that constructs and exploits subtasks, to reconcile the contradiction between the generalizability across various domains and the ability to exploit extra data. Serving as auxiliary tasks, subtasks of LDL can 1) provide different views of the primary task distribution, rendering the mixture of distributions more traceable (i.e., the key issue one); 2) furnish constructed supervised data to mitigate the scarcity and ambiguity inherent in LDL datasets (i.e., the key issue two); 3) require no prior/expert knowledge from specific applications; and 4) emphasize various label correlations via partitioning of the label space.

The main contributions of this paper are outlined below: 1) we propose $\mathcal{S}$-LDL, which is considered the first endeavor to address LDL via subtasks; 2) our analysis shows the validity and reconstructability of these subtasks; 3) we present a plug-and-play framework seamlessly compatible with existing LDL methods, and adaptable to derivative tasks of LDL; and 4) the code will be available on GitHub soon, facilitating reproducible research endeavors.

2. Related work

LDL Our work is mainly related to LDL. Initially employed to tackle age estimation (Geng et al., 2013), LDL has evolved into a novel machine learning paradigm (Geng, 2016), which is supported by theoretical underpinnings (Wang & Geng, 2019). Concurrently, more and more derivative tasks of LDL (González et al., 2021b; Lu & Jia, 2022; Wang, Jing and Geng, Xin, 2019; Xu & Zhou, 2017; Xu et al., 2019) are emerging to offer assistance in various real-world dilemmas. Most methods focus on improving performance via loss function engineering (Jia et al., 2019; 2023; Ren et al., 2019; Wen et al., 2023) or efficient model structures (González et al., 2021a; Jin et al., 2024; Shen et al., 2017; Yang et al., 2017b), while some work is dedicated to practical application scenarios (Gao et al., 2017; Li et al., 2023a; Shirani et al., 2019). However, the scarcity of LDL datasets and the complexity of the label distribution itself make it difficult to further improve performance, at which point one may think of leveraging auxiliary tasks.

LDL with auxiliary tasks While there are LDL methods that leverage auxiliary tasks to enhance performance, they often rely on prior/expert knowledge from specific applications, extending beyond the scope of the LDL task. For example, LDL-ALSG (Chen et al., 2020) designs auxiliary tasks dedicated to facial emotion recognition, necessitating the use of external tools to extract facial points and action units from human faces. Wu et al. (2019) exploit the Hayashi criterion, a rule for counting and grading in acne lesions, which results in their method being only applicable in a small branch of the dermatology field. Yang et al. (2017a) employ a multi-task framework for image emotion classification, designing constraints inspired by Mikel's wheel, a psychological emotion model. It also suffers from similar limitations. The need for prior/expert knowledge significantly narrows the application scenarios of these methods.

MLL with partitioning of the label space For reference, there exist MLL methods based on partitioning of the label space, which can construct multi-label subtasks without involving additional knowledge and can be widely used in various domains. The most classic related work is that of HOMER (Tsoumakas et al., 2008) and RAkEL (Tsoumakas et al., 2010), the former forms a hierarchy of label subspaces while the latter randomly selects label subspaces. Many subsequent papers have been inspired by them (Prabhu et al., 2018; Read et al., 2013; Wang et al., 2021). Read et al. (2014) present a general framework of label subspaces and provide theoretical justification for it. Since label distribution contains rich knowledge, we can follow the patterns of these methods to construct label distribution subtasks.

LDL with ensemble strategy It is imperative to aggregate the output of subtasks. Fortunately, ensemble-based LDL methods have demonstrated promising performance. For instance, LDLFs (Shen et al., 2017) learns different label distributions on the leaf nodes of differentiable decision trees and learns weights that aggregate these label distributions. DF-LDL (González et al., 2021a) aggregates the label distribution of output of multiple base models by simple averaging, while Zhai et al. (2018) focus on aggregating the results of various neural networks via a combining learner. However, 1) the above methods are not suitable for incomplete label spaces (i.e., subtask label spaces); and 2) *none* of them involve label space partitioning, therefore no extra supervised information of label distributions is constructed.

3. Subtask construction

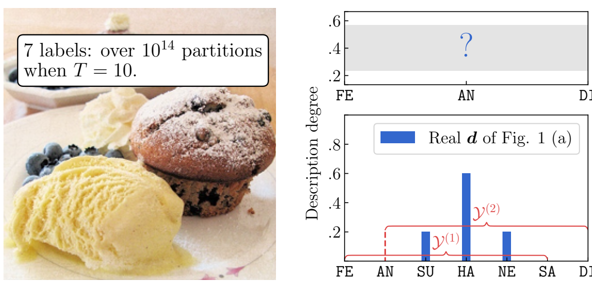
3.1. Preliminary

Notation Vectors are denoted by lowercase bold letters, e.g., $\boldsymbol{v}$, and the corresponding regular letter with subscript i, i.e., v_i, indicates its i-th element. Matrices are denoted by uppercase bold letters, e.g., $\boldsymbol{A}$, with $\boldsymbol{a}_i$ as the i-th row and $\boldsymbol{a}_{\bullet j}$ as the j-th column. A_{ij} is the element in i-th row and j-th column of $\boldsymbol{A}$. The superscript (t) indicates that a symbol corresponds to the t-th subtask. Table 1 outlines the key notation in this paper.

[1] `HA`, `SA`, `SU`, `AN`, `DI`, `FE`, and `NE` represent 7 common emotions in sentiment analysis datasets, namely `happiness`, `sadness`, `surprise`, `anger`, `disgust`, `fear`, and `neutral`, respectively.

Table 1. Key notation and terminology in this paper

Symbol	Description	Example
$\mathcal{Y} = \{y_j\}_{j=1}^{L}$	Label space (with L labels)	$\mathcal{Y} = \{$HA, SA, SU, AN, DI, FE$\}$
$\mathcal{Y}^{\circ}$	Subtask label spaces	$\mathcal{Y}^{\circ} = \{\cdots, \mathcal{Y}^{(t)} = \{HA, SA, SU, FE\}, \cdots\}$
$d_{\boldsymbol{x}_i}^{y_j}$	Description degree of $\boldsymbol{x}_i$ about y_j	$d_{\boldsymbol{x}_i}^{y_0} = 0.4$, i.e., HA describes $\boldsymbol{x}_i$ by 0.4
$\boldsymbol{d}_i = (d_{\boldsymbol{x}_i}^{y_j})_{j=1}^{L}$	Label distribution of $\boldsymbol{x}_i$	$\boldsymbol{d}_i = (0.4,\ 0.05,\ 0.3,\ 0.1,\ 0.1,\ 0.05)$
$\boldsymbol{D} = (\boldsymbol{d}_i)_{i=1}^{N} = (\boldsymbol{d}_{\bullet j})_{j=1}^{L}$	Label distribution matrix (with N samples)	$\boldsymbol{D} = (\cdots, \boldsymbol{d}_i, \cdots)$
$\boldsymbol{d}_i^{(t)} = (d_{\boldsymbol{x}_i}^{(t)y_j})_{j=1}^{\lvert\mathcal{Y}^{(t)}\rvert}$	Subtask label distribution	$\boldsymbol{d}_i^{(t)} = (0.5,\ 0.0625,\ 0.375,\ 0.0625)$
$\mathcal{D}^{\circ}$	Subtask label distribution matrices	$\mathcal{D}^{\circ} = \{\cdots, \boldsymbol{D}^{(t)}, \cdots\}$
$\boldsymbol{M} = (\boldsymbol{m}_t)_{t=1}^{T} = (M_{tj})$	Mask matrix (with T anticipated tasks)	$\boldsymbol{M} = (\cdots, \boldsymbol{m}_t = (1,\ 1,\ 1,\ 0,\ 0,\ 1), \cdots)$



(a) Huge number of partitions (b) Ignorance & homogeneity [1]

Figure 1. **(a)** is sourced from the emotion6 (Yang et al., 2017b) dataset, which has only 7 labels, but the number of potential partitions is huge. (The upper part of) **(b)** exemplifies a subtask label space $\{$FE, AN, DI$\}$, which is challenging to describe (a). (The lower part of) **(b)**'s two subtask label spaces encompass all descriptive information, meaning no new knowledge is generated about (a). This example vividly illustrates the limitations that may result from local ignorance and lack of diversity in subtask label spaces.

Problem definition Let $\boldsymbol{x} \in \mathcal{X} = \mathbb{R}^P$ denote the feature of the instance and $\boldsymbol{d} \in \Delta^{L-1}$ denote the label distribution, where $\Delta^{K-1} \triangleq \{\boldsymbol{v} \in \mathbb{R}^K \,|\, \boldsymbol{1}\boldsymbol{v}^{\top} = 1,\ \boldsymbol{v} \geq 0\}$ is the $(K-1)$-dimensional probability simplex. LDL's goal is to find a mapping $\zeta : \mathcal{X} \mapsto \Delta^{L-1}$. In this paper, we partition the label space $\mathcal{Y}$ to obtain the subtask label space set $\mathcal{Y}^{\circ}$, then accordingly generate extra supervised information, i.e., subtask distribution matrix set $\mathcal{D}^{\circ}$, to guide ζ's learning.

Technical challenges Our first challenge arises from *the exponential growth in partitions* as the number of labels increases (Tsoumakas et al., 2010). When generating T tasks from a label space with L labels, the number of unique partitions is given by ${(2^L-L-2)!}/{(T!(2^L-L-2-T)!)}$. This makes it impractical to calculate metric for each case to select subtasks. We tackle this challenge in a mask matrix learning manner. The second challenge lies in *discerning reasonable partitions*. Since the label distribution matrix is usually imbalanced in average description degree (Zhao et al., 2023), some partitions exhibit unreasonable local ignorance. As a result, the corresponding spaces struggle to handle the majority of instances, because 1) theoretically, there is no objective standard for the degree of negative correlation; and 2) empirically, weakly or negatively correlated information is easily overlooked by human annotators in crowdsourced datasets. To mitigate this, we incorporate the description degree as a metric for the reliability of supervised information in guiding the generation of subtask masks. The third challenge is *avoiding analogous extra supervised information*, i.e., to generate label distributions containing new knowledge (González et al., 2021b). This necessitates fostering richness and diversity in both subtask label spaces and distributions. To achieve this objective, we 1) minimize pairwise similarity among subtask masks; and 2) normalize each subtask label distribution to yield brand new insights distinct from the label distribution of the primary task. Fig. 1 portraits an illustrative example of these challenges.

3.2. Learning subtask masks

Let $\boldsymbol{M} \in \{0,1\}^{T\times L}$ be the subtask mask matrix, where T represents the number of anticipated tasks. To ensure that the subtask label spaces contain as reliable information as possible, the learning of the subtask mask matrix can be converted into this problem: $\max_{\boldsymbol{M}} \sum_{t\in[T]} \bar{\boldsymbol{d}}\boldsymbol{m}_t^{\top}$, where $\bar{\boldsymbol{d}} = {(\sum_{i\in[N]} \boldsymbol{d}_i)}/{N}$. Obviously, a senseless solution is $\boldsymbol{m}_t = \boldsymbol{1}$ for all $t \in [T]$, i.e., all extra supervised information is equivalent to the primary task information. Therefore, solving the above problem alone is inappropriate. To address this, we consider pairwise similarity among subtask masks. We also employ exponential tricks to convert maximization into minimization. Finally, $\boldsymbol{M}$ is calculated as

$$\boldsymbol{M}^{*} = \operatorname*{arg\,min}_{\boldsymbol{M}} \left(\frac{1}{T} \sum_{t\in[T]} \exp\left(-\bar{\boldsymbol{d}}\boldsymbol{m}_t^{\top}\right) + \frac{2\lambda}{(T(T-1))} \sum_{i,\,j\in[T],\,i>j} \frac{\boldsymbol{m}_i\boldsymbol{m}_j^{\top}}{\|\boldsymbol{m}_i\|\,\|\boldsymbol{m}_j\|} \right), \quad \text{s.t. } M_{tj} \in \{0,1\};\ t\in[T];\ j\in[L], \tag{1}$$

where λ is a trade-off parameter. Eq. (1) is slightly more

Algorithm 1 Subtask construction

Input: Input matrix $\boldsymbol{D}$, trade-off parameter λ, anticipated number of subtasks T.
Output: Subtask distribution matrices $\mathcal{D}^{\circ}$ (with corresponding subtask label spaces $\mathcal{Y}^{\circ}$).

Initialization: $\mathcal{Y}^{\circ} \leftarrow \{\varnothing\}, \mathcal{D}^{\circ} \leftarrow \{\varnothing\}$;
Calculate $\boldsymbol{M}$ using L-BFGS; ▷ (Eq. (1))
for $t \leftarrow 1$ **to** T **do**
  $\mathcal{Y}^{(t)} \leftarrow \{\varnothing\}$;
  for $j \leftarrow 1$ to L **do**
    if $M_{tj} = 1$ **then**
      $\mathcal{Y}^{(t)} \leftarrow \mathcal{Y}^{(t)} \cap \{y_j\}$;
    end if
  end for
  if $|\mathcal{Y}^{(t)}| = L$ **or** $|\mathcal{Y}^{(t)}| \leq 1$ **then**
    continue; /* Ignore invalid subtask masks. */
  end if
  /* The clip$(x,\, a,\, b)$ function restricts x to $[a,\, b]_{\mathbb{R}}$. */
  $\boldsymbol{D}^{(t)} \leftarrow []$;
  for $j \leftarrow 1$ **to** L **do**
    if $y_j \in \mathcal{Y}^{(t)}$ **then**
      $\boldsymbol{D}^{(t)} \leftarrow$ concatenate$(\boldsymbol{D}^{(t)},\, \text{clip}(\boldsymbol{d}_{\bullet j}, \varepsilon, 1))$;
      /* ε is a very small positive number. */
    end if
  end for
  for $i \leftarrow 1$ **to** N **do**
    Normalization: $\boldsymbol{d}_i^{(t)} \leftarrow \mathcal{N}_{\text{SUM}}(\boldsymbol{d}_i^{(t)})$; ▷ (Eq. (2))
  end for
  $\mathcal{Y}^{\circ} \leftarrow \mathcal{Y}^{(t)} \cup \mathcal{Y}^{\circ}, \mathcal{D}^{\circ} \leftarrow \boldsymbol{D}^{(t)} \cup \mathcal{D}^{\circ}$;
end for

complicated than conventional integer programming. For convenience, we solve it using the L-BFGS method, with its constraint enforced via sigmoid (a conversion threshold is set, where outputs greater than it are set to 1, while those below are set to 0). Refer to Section 4.1 for an analysis of the validity of Eq. (1).

3.3. Generating subtask distributions

We slice $\boldsymbol{D}$ according to $\mathcal{Y}^{(t)}$s. To generate diversified $\boldsymbol{d}^{(t)}$s, we perform normalization on each slice with

$$\mathcal{N}_{\text{SUM}}(\boldsymbol{v})_j = \frac{v_j}{\sum_{i=1}^{|\boldsymbol{v}|} v_i}. \tag{2}$$

The rationale for utilizing $\mathcal{N}_{\text{SUM}}$ as the normalization function can be found in Section 4.2. The overall subtask construction process, denoted by SC, is illustrated in Alg. 1. Then, one can naturally come up with an adaptive LDL pipeline based on the shallow regime, as depicted in Alg. 2.

4. Analysis about subtask construction

In this section, we analyze the subtask construction algorithm SC by studying the following questions:

- $\mathcal{Q}_1$: Are $\mathcal{Y}^{(t)}$s provided by Eq. (1) valid and meet expectations? How do they compare to those generated by random selection?
- $\mathcal{Q}_2$: Are $\boldsymbol{D}^{(t)}$s provided by Eq. (2) reconstructable? Can one replace it with other normalization functions?
- $\mathcal{Q}_3$: What is the overall time complexity of SC? Is it practical for large-scale datasets?

Algorithm 2 $\mathcal{S}$-LDL (shallow regime)

Input: Feature matrix $\boldsymbol{X}$, label distribution matrix $\boldsymbol{D}$, testing instance $\boldsymbol{x}'$.
Output: Predicted label distribution $\boldsymbol{d}'$ for instance $\boldsymbol{x}'$.

Initialize parameter of each estimator;
$\mathcal{D}^{\circ} \leftarrow \texttt{SC}(\boldsymbol{D})$;
for $t \leftarrow 1$ **to** $|\mathcal{D}^{\circ}|$ **do**
  Fit an estimator $f^{(t)}$ on dataset $\{\boldsymbol{X},\, \boldsymbol{D}^{(t)}\}$;
  $\boldsymbol{d}^{(t)\prime} \leftarrow f^{(t)}(\boldsymbol{x}')$;
end for
Concatenate $\boldsymbol{X}$ and all $\boldsymbol{D}^{(t)}$s to get $\boldsymbol{Z}$, where $t \in [|\mathcal{D}^{\circ}|]$;
Fit an estimator f on dataset $\{\boldsymbol{Z},\, \boldsymbol{D}\}$;
Concatenate $\boldsymbol{x}'$ and all $\boldsymbol{d}^{(t)\prime}$s to get $\boldsymbol{z}'$, where $t \in [|\mathcal{D}^{\circ}|]$;
$\boldsymbol{d}' \leftarrow f(\boldsymbol{z}')$;

The validity, reconstructability, and complexity analysis are conducted for $\mathcal{Q}_1$, $\mathcal{Q}_2$, and $\mathcal{Q}_3$, respectively.

4.1. Validity analysis

Eq. (1) manages the intricate task of selecting $\mathcal{Y}^{(t)}$s via λ and T, and the main idea is to suppress local ignorance and increase diversity of each $\mathcal{Y}^{(t)}$ simultaneously. To check whether the subtasks are valid and meet expectations, we design two metrics, the information index and the diversity index, both of which are based on the mask valid rate.

Definition 4.1 (Mask valid rate)**.** For all $t \in [T]$, the following are considered invalid masks: 1) $\boldsymbol{m}_t = \boldsymbol{1}$, or 2) $|\boldsymbol{m}_t|! = 1$, or, 3) excluding masks in cases 1) and 2), for any remaining mask index i, $\boldsymbol{m}_i = \boldsymbol{m}_j$ where $j \in [i]$.[2] The mask valid rate is defined as:

$$\rho(\boldsymbol{M}) = \frac{1}{T} \sum_{t \in [T]} \begin{cases} 0, & \boldsymbol{m}_t = \boldsymbol{1}, \\ 0, & |\boldsymbol{m}_t|! = 1, \\ \begin{cases} 0, & \varrho_t(\boldsymbol{m}_t) > 1, \\ 1, & \text{o/w}, \end{cases} & \text{o/w}, \end{cases} \tag{3}$$

where $\varrho_i(\boldsymbol{m}) = \sum_{j=i}^{T} \mathbb{I}(\boldsymbol{m} = \boldsymbol{m}_j)$ and $\mathbb{I}$ is the indicator function.

Definition 4.2 (Information index)**.** We call it informative if $M_{tj} = 1$ where $t \in [T]$ and $j \in [L]$. The information

[2]These three cases correspond to 1) masks that are exactly the same as the primary task; 2) masks that fail to form label distributions; and 3) duplicate masks among the remaining masks, respectively.

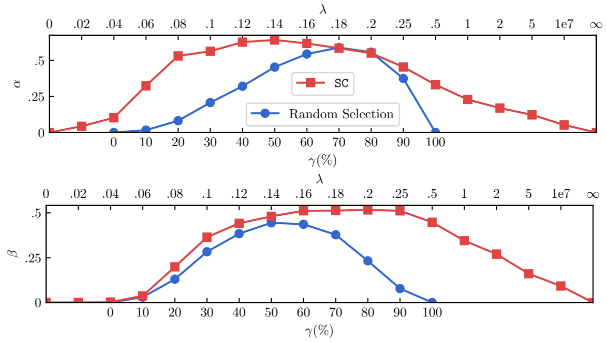


Figure 2. Information/diversity index w.r.t. λ on `emotion6`.

index is defined as:

$$\alpha(\boldsymbol{M}) = \frac{\rho}{T} \sum_{t \in [T]} \bar{\boldsymbol{d}} \boldsymbol{m}_t^\top. \tag{4}$$

Definition 4.3 (Diversity index)**.** We call it diverse if two $\boldsymbol{m}_t$s reflect disjoint labels. The diversity index is defined as:

$$\beta(\boldsymbol{M}) = \frac{2\rho}{T(T-1)} \sum_{i=1}^{T-1} \sum_{j=i+1}^{T} \bar{\boldsymbol{d}} (\boldsymbol{m}_i \oplus \boldsymbol{m}_j)^\top. \tag{5}$$

We calculate the two metrics of $\boldsymbol{M}$ generated by `SC` with varying λ, results on the `emotion6` dataset are shown in Fig. 2 (red curve). For comparison, we also study these two metrics of masks generated by random selection with varying probability γ (blue curve). Results on other datasets are provided in the appendix. All results indicate that the red curve is always higher than the blue curve, suggesting that the subtasks generated by `SC` are more informative and diverse than those generated by random selection, as long as the parameters are appropriate. Parameter selection is discussed in Section 6.

4.2. Reconstructability analysis

Reconstructability ensures the effectiveness of the path from $\boldsymbol{d}^{(t)}$s to $\boldsymbol{d}$. We strive to choose a normalization function so that $\boldsymbol{d}^{(t)}$s retain more information, even efficacious enough to reconstruct $\boldsymbol{d}$. Theorem 4.4 illustrates that Eq. (2) is the only possibility.

Theorem 4.4. *Let each subtask label space form a connected graph with its each label as a node. Then merge these graphs according to their respective labels to form $\mathcal{G}$. If and only if $\mathcal{N}_{SUM}$ is used for normalization, the primary label distribution can be reconstructed from these subtask label distributions, when the following conditions are satisfied: 1) $\mathcal{G}$ is connected; 2) $\mathcal{G}$ covers all labels in the label space, and 3) corresponding description degrees of all cut vertices of $\mathcal{G}$ are not zero.*[3]

Proof. We solely discuss the extreme case where two subtask label spaces overlap with just one label. Further specialized cases can be deduced by the reader via induction. With a little bit of symbol abuse, let the general normalization function be defined as $\mathcal{N}(\boldsymbol{v}) \triangleq {}^{p(\boldsymbol{v})}/_{q(\boldsymbol{v})}$. Assume that there is a label distribution $\boldsymbol{d} = (d_1, \cdots, d_L)$ and its corresponding label space is $\mathcal{Y} = \{y_1, \cdots, y_L\}$. The two decompositions of $\mathcal{Y}$ are $\mathcal{Y}_{\boldsymbol{a}} = \{y_1, \cdots, y_k\}$ and $\mathcal{Y}_{\boldsymbol{b}} = \{y_k, \cdots, y_L\}$, respectively. It is obvious that $\mathcal{Y}_{\boldsymbol{a}} \cup \mathcal{Y}_{\boldsymbol{b}} = \mathcal{Y}$ and $\mathcal{Y}_{\boldsymbol{a}} \cap \mathcal{Y}_{\boldsymbol{b}} = \{y_k\}$. Let the subspace label distribution corresponding to these two decompositions be $\boldsymbol{a} = (a_1, \cdots, a_k)$ and $\boldsymbol{b} = (b_k, \cdots, b_L)$. According to our assumptions, $d_k \neq 0$. Then, for any integer $j \in [1, k]_{\mathbb{Z}}$, we have

$$\frac{a_j}{a_k} = \frac{\mathcal{N}(\boldsymbol{d})_j}{\mathcal{N}(\boldsymbol{d})_k} = \frac{p(\boldsymbol{d})_j}{q(\boldsymbol{d})_j} \frac{q(\boldsymbol{d})_k}{p(\boldsymbol{d})_k}. \tag{6}$$

Typically, for most normalization functions, $q(\cdot)$ is a normalizing constant, i.e., $q(\boldsymbol{d})_j = q(\boldsymbol{d})_k$. Thus Eq. (6) can be rewritten into $a_j p(\boldsymbol{d})_k = a_k p(\boldsymbol{d})_j$. Plug it into $\sum_{j=1}^{k} a_j = 1$, and do the same for $\boldsymbol{b}$ as well, and get

$$\frac{a_k \sum_{j=1}^{k} p(\boldsymbol{d})_j}{p(\boldsymbol{d})_k} = 1, \quad \frac{b_k \sum_{j=k}^{L} p(\boldsymbol{d})_j}{p(\boldsymbol{d})_k} = 1. \tag{7}$$

Add these two equations together, we have

$$p(\boldsymbol{d})_k + \sum_{j=1}^{L} p(\boldsymbol{d})_j = \frac{p(\boldsymbol{d})_k}{a_k} + \frac{p(\boldsymbol{d})_k}{b_k}. \tag{8}$$

Eq. (8) implies that $\sum_{j=1}^{L} p(\boldsymbol{d})_j$ must be given *a priori*, and $p(\boldsymbol{d})_k$ is related to d_k, and only d_k. To make it possible, the only thing we can exploit is the sum-to-one constraint of $\boldsymbol{d}$, i.e., $\sum_{j=1}^{L} d_j = 1$. Note that any nonlinear operations here will lead to inconsistencies with the problem constraints. Therefore $p(\boldsymbol{v})_j = v_j$ for all j is the simplest nontrivial solution. Since $\sum_{i}^{|\boldsymbol{v}|} \mathcal{N}(\boldsymbol{v})_i = 1$, we have $q(\boldsymbol{v}) = \sum_{i}^{|\boldsymbol{v}|} v_i$, i.e., the finally deduced normalization function is Eq. (2). In this case, for any integer $j \in [1, L]_{\mathbb{Z}}$, we have

$$d_j = \begin{cases} \dfrac{a_j b_k}{a_k + b_k - a_k b_k}, & j = 1, \cdots, k, \\ \dfrac{a_k b_j}{a_k + b_k - a_k b_k}, & j = k+1, \cdots, L, \end{cases} \tag{9}$$

which illustrates that the original label distribution $\boldsymbol{d}$ can be reconstructed by subtask label distributions $\boldsymbol{a}$ and $\boldsymbol{b}$. This is possible thanks to the use of $\mathcal{N}_{\text{SUM}}$. □

Theorem 4.4 also states that it is not appropriate to replace Eq. (2) with the min-max or softmax function because doing so destroys the reconstruction information.

[3]Maintaining non-zero description degrees for all cut vertices is essential for preserving the proportional information.

4.3. Complexity analysis

The overall time cost of SC is primarily influenced by the calculation of $\boldsymbol{M}$ and the normalization process. The time complexity of computing and updating $\boldsymbol{M}$ are $\mathcal{O}(L(TN+T^2))$ and $\mathcal{O}(LT)$, respectively. The time complexity of the normalization process is $\mathcal{O}(LTN)$. The overall time complexity of each iteration of SC is $\mathcal{O}(L(TN+T^2))$, which is linear with respect to the number of instances and labels. Therefore, it is clear that SC can be applied to large-scale datasets.

5. $\mathcal{S}$-LDL of the deep regime

The aforementioned analysis has exposed the problems of the shallow regime: 1) shallow methods as base estimators have low potential in themselves; 2) there is a training gap between the primary task and subtasks, i.e., no representation learning is involved. Therefore, it is necessary to introduce our proposed $\mathcal{S}$-LDL of the deep regime, the overview of which is illustrated in Fig. 3. We illustrate our framework by introducing the learnable parts one by one.

- $\varphi(\cdot)$ is guided by subtasks to learn a powerful representation, i.e., $\boldsymbol{R} = \varphi(\boldsymbol{X})$.

- $\psi(\cdot)$ is responsible for predicting subtask label distributions, i.e., $(\tilde{\boldsymbol{D}}^{(1)}, \cdots) = \psi(\boldsymbol{R})$. To ensure the precise prediction of subtask label distributions for reconstruction, we employ the mean absolute error function for subtask learning. The loss is weighted by the summation of the description degrees corresponding to the primary tasks, allowing more reliable label spaces to receive more attention. The subtask learning loss has the following form:

$$\ell_{\mathrm{SUB}}\left(\mathcal{D}^{\circ}; \tilde{\mathcal{D}}^{\circ}\right) = \frac{1}{N\,|\mathcal{Y}^{\circ}|}\sum_{\mathcal{Y}^{(t)}\in\mathcal{Y}^{\circ}}\sum_{i\in[N]} \left(\sum_{y_k\in\mathcal{Y}^{(t)}} d_{\boldsymbol{x}_i}^{y_k}\right)\sum_{j\in\left[|\mathcal{Y}^{(t)}|\right]}\left|d_{\boldsymbol{x}_i}^{(t)y_j} - \tilde{d}_{\boldsymbol{x}_i}^{(t)y_j}\right|. \tag{10}$$

- $\omega(\cdot)$ can be any existing method that can be expressed as a network structure theoretically. Since the concatenation of the representation and subtask label distributions, we have $\boldsymbol{Z} = (\boldsymbol{R}, \psi(\boldsymbol{R}))$ and $\tilde{\boldsymbol{D}} = \omega(\boldsymbol{Z})$. In the case of the primary task being vanilla LDL, the primary task loss ℓ_{PRI} can be

$$\ell_{\mathrm{KL}}\left(\boldsymbol{D}, \tilde{\boldsymbol{D}}\right) = \frac{1}{N}\sum_{i\in[N]}\sum_{j\in[L]} d_{\boldsymbol{x}_i}^{y_j}\ln\frac{d_{\boldsymbol{x}_i}^{y_j}}{\tilde{d}_{\boldsymbol{x}_i}^{y_j}},\ \ell_{\mathrm{KL}}\in\mathcal{L}_{\mathrm{LDL}}. \tag{11}$$

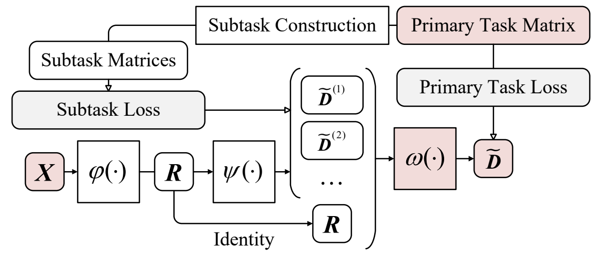


Figure 3. The overview of $\mathcal{S}$-LDL (deep regime). White, red and gray highlight our proposed, existing methods, and loss functions, respectively.

Note that ℓ_{PRI} changes as the primary task changes. Finally, we can learn the model parameters $\boldsymbol{\Theta}$ by

$$\boldsymbol{\Theta}^{*} = \arg\min_{\boldsymbol{\Theta}}(\ell_{\mathrm{PRI}} + \mu\ell_{\mathrm{SUB}}), \tag{12}$$

where μ is a trade-off parameter. Compared with the shallow regime, $\mathcal{S}$-LDL of the deep regime has the following advantages: 1) There is no two-stage training gap, which makes the representation contain insights from both the primary task and the subtasks; 2) the framework not only serves LDL, but can also be directly applied to derivative tasks of LDL, e.g., LDL for classification (LDL4C) (Wang, Jing and Geng, Xin, 2019), incomplete LDL (IncomLDL) (Xu & Zhou, 2017), label enhancement (LE) (Xu et al., 2019). The modifications involved are shown in Table 2, where $\mathcal{L}_{\mathrm{X}}$ indicates the set of losses for adaptable methods in the task of type "X". Special mathematical procedures of LDL4C and IncomLDL are defined as

$$\left(\bar{\boldsymbol{D}}^{(t)}\right)_{ij} \triangleq \begin{cases}1, & \text{if } y_j = \arg\max_{\bar{y}\in\mathcal{Y}^{(t)}} d_{\boldsymbol{x}_i}^{\bar{y}},\\ 0, & \text{o/w},\end{cases} \tag{13}$$

$$\left(\mathcal{R}_{\Omega}\left(\boldsymbol{D}\right)\right)_{ij} \triangleq \begin{cases}(\boldsymbol{D})_{ij}, & \text{if } (i, y_j)\in\Omega,\\ 0, & \text{o/w},\end{cases} \tag{14}$$

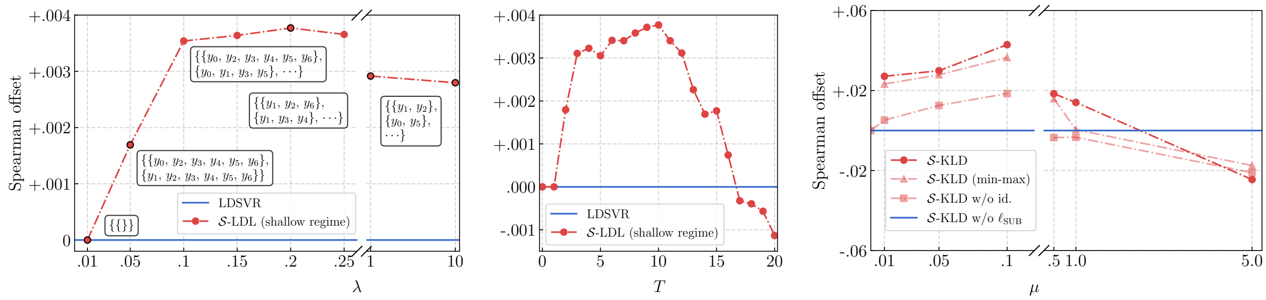
respectively, where $(\cdot)_{ij}$ represents the element in i-th row of the matrix corresponding to label y_j, and Ω represents observed elements sampled uniformly at random from $\boldsymbol{D}$ in IncomLDL. Such modifications are rational since: 1) targets of LDL4C and IncomLDL, i.e., $\bar{\boldsymbol{D}}$ and $\mathcal{R}_{\Omega}(\boldsymbol{D})$, are essentially different forms of degradation of the label distribution matrix; and 2) the target of LE is a logical label matrix $\boldsymbol{L}$, the same as the target of MLL, which is actually a special case of LDL.

6. Experiments

In this section, we evaluate $\mathcal{S}$-LDL via a series of experiments. Due to page limitations, datasets, comparison methods, parameter settings, and full experimental results are

Table 2. Modifications of different task adaptations

Type	Subtask construction	ℓ_{PRI}	ℓ_{SUB}
Vanilla LDL	$(\boldsymbol{D}^{(1)}, \cdots) \leftarrow \texttt{SC}(\boldsymbol{D})$	$\ell(\boldsymbol{D}, \tilde{\boldsymbol{D}}) \in \mathcal{L}_{\text{LDL}}$	$\ell_{\text{SUB}}(\boldsymbol{D}^{(1)}, \cdots; \tilde{\boldsymbol{D}}^{(1)}, \cdots)$
LDL4C	$\bar{\boldsymbol{D}}^{(1)}, \cdots \leftarrow \texttt{SC}(\bar{\boldsymbol{D}})$	$\ell(\boldsymbol{D}, \bar{\boldsymbol{D}}, \tilde{\boldsymbol{D}}) \in \mathcal{L}_{\text{LDL4C}}$	$\ell_{\text{SUB}}(\bar{\boldsymbol{D}}^{(1)}, \cdots; \tilde{\boldsymbol{D}}^{(1)}, \cdots)$
IncomLDL	$(\boldsymbol{D}_{\Omega}^{(1)}, \cdots) \leftarrow \texttt{SC}(\mathcal{R}_{\Omega}(\boldsymbol{D}))$	$\ell(\mathcal{R}_{\Omega}(\boldsymbol{D}), \mathcal{R}_{\Omega}(\tilde{\boldsymbol{D}})) \in \mathcal{L}_{\text{IncomLDL}}$	$\ell_{\text{SUB}}(\boldsymbol{D}_{\Omega}^{(1)}, \cdots; \mathcal{R}_{\Omega}(\tilde{\boldsymbol{D}}^{(1)}), \cdots)$
LE	$(\boldsymbol{L}^{(1)}, \cdots) \leftarrow \texttt{SC}(\boldsymbol{L})$	$\ell(\boldsymbol{L}, \tilde{\boldsymbol{D}}) \in \mathcal{L}_{\text{LE}}$	$\ell_{\text{SUB}}(\boldsymbol{L}^{(1)}, \cdots; \tilde{\boldsymbol{D}}^{(1)}, \cdots)$



(a) `Spear.` offset w.r.t. λ (b) `Spear.` offset w.r.t. T (c) `Spear.` offset w.r.t. μ

Figure 4. **(a)** and **(b)** are visualized results of the parameter sensitivity analysis of `SC`, which is on the `emotion6` dataset. The blue lines in (a) and (b) represent performance without auxiliary tasks. **(c)** is visualized results of the ablation study and the parameter sensitivity analysis of deep regime $\mathcal{S}$-LDL on the `Natural_Scene` dataset.

introduced in the appendix. Details of all implementations are openly accessible at GitHub.[4]

Metrics For LDL, we use the same metrics suggested by Jia et al. (2023). Due to page limitations, we mainly present results on `Cheby.` ↓ (Chebyshev distance), `Clark` ↓ (Clark distance), `Cosine` ↑ (cosine similarity), and `Spear.` ↑ (Spearman's coefficient) in the main paper, where ↓ (↑) indicates "the lower (higher) the better". Note that these metrics are *not* as intuitive as accuracy or error rate, i.e., *small changes can mean large performance differences.* For LDL4C, objective of which is different from LDL, we use `0/1 loss` ↓ (zero one loss) and `Err. prob.` ↓ (error probability) as metrics (Wang, Jing and Geng, Xin, 2019).

Results and discussion We apply $\mathcal{S}$-LDL to existing methods to demonstrate performance improvements. For each dataset we conduct ten-fold experiments repeated 10 times, and the average performance is recorded. Tables 3 to 6 show representative results of the shallow/deep regime $\mathcal{S}$-LDL and the remainder are in the appendix, where • (◦) indicates that more than half of the metrics support that "$\mathcal{S}$-X" is statistically superior (inferior) to the corresponding methods "X" (pairwise t-test at 0.05 significance level); there is no significant if neither • nor ◦ is shown.

For $\mathcal{S}$-LDL of the shallow regime, $f^{(t)}$s and f are implemented by a representative LDL method, LDSVR (Geng & Hou, 2015). It is important to highlight that $\mathcal{S}$-LDL of the shallow regime (Alg. 2) relies on naive concatenation operations and is not tied to representation learning. Consequently, any performance improvement over f is solely attributed to the effects of $\boldsymbol{d}^{(t)}$s and the original performance of f. The results in Tables 3 and 4 show that $\mathcal{S}$-LDL significantly improves the performance of the base estimator LDSVR.

In Tables 5 to 6, LRR focuses on the label ranking relationship, which is also emphasized by each subtask. We believe this is why $\mathcal{S}$-LDL and LRR fit so well. Note that our method has the least improvement in SCL, which may be attributed to its reliance on shallow regime methods in the prediction phase. It is also worth noting that the improvement in vanilla KLD is considerable, which just illustrates the limitations of loss function engineering that considers label correlation one-sidedly. QFD^2 and CJS are tailored for ordinary LDL, and may have better results than LRR on this regard. Powered by $\mathcal{S}$-LDL, these methods can all achieve better level. For LDL4C, $\mathcal{S}$-LDL significantly improves both HR and LDLM. However, it can be observed that $\mathcal{S}$-LDL4C is unstable on the `Flickr` dataset, which is not surprising since LDL4C itself fails on it. This is caused by the combined effect of the sparsity of the dataset and the information entropy operation involved in LDL4C.

Parameter sensitivity analysis First, we employ $\mathcal{S}$-LDL of the shallow regime for parameter sensitivity analysis of `SC` (Alg. 1). With λ varying and T fixed at 10, we conduct

[4] https://github.com/SpriteMisaka/PyLDL

Table 3. Experimental results of LDL on the `JAFFE` dataset formatted as ($\mathtt{mean} \pm \mathtt{std}$)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR •	.0959 $\pm$.013	.3280 $\pm$.027	.6778 $\pm$.058	.0476 $\pm$.011	.9549 $\pm$.010	.8838 $\pm$.012	.5175 $\pm$.102	.4508 $\pm$.086
$\mathcal{S}$-LDSVR (shallow)	**.0859** $\pm$.012	**.3024** $\pm$.033	**.6143** $\pm$.066	**.0449** $\pm$.012	**.9580** $\pm$.011	**.8941** $\pm$.012	**.5237** $\pm$.091	**.4580** $\pm$.079

Table 4. Experimental results of LDL on the `Natural_Scene` dataset formatted as ($\mathtt{mean} \pm \mathtt{std}$)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR •	.4899 $\pm$.016	2.0831 $\pm$.025	5.7724 $\pm$.092	2.0862 $\pm$.085	.5740 $\pm$.017	.4430 $\pm$.015	.4997 $\pm$.015	.3695 $\pm$.012
$\mathcal{S}$-LDSVR (shallow)	**.4692** $\pm$.017	**1.9231** $\pm$.034	**5.2307** $\pm$.113	**2.2261** $\pm$.121	**.5893** $\pm$.018	**.4602** $\pm$.018	**.5236** $\pm$.017	**.3888** $\pm$.013

Table 5. Experimental results of LDL on `JAFFE` and `Yeast_diau` formatted as ($\mathtt{mean} \pm \mathtt{std(rank)}$)

Algorithms	`JAFFE` (Lyons et al., 1998)		Algorithms	`Yeast_diau` (Geng, 2016)	
	`Clark` ↓	`Cosine` ↑		`Cheby.` ↓	`Spear.` ↑
LDSVR (Geng & Hou, 2015)	.3280 $\pm$.027 (6)	.9549 $\pm$.010 (7)	CPNN (Geng et al., 2013)	.0385 $\pm$.001 (9)	.2962 $\pm$.034 (10)
AA-kNN (Geng, 2016)	.3483 $\pm$.032 (8)	.9497 $\pm$.010 (9)	AA-kNN	.0385 $\pm$.001 (9)	.3674 $\pm$.029 (9)
LDLFs (Shen et al., 2017)	.3637 $\pm$.032 (10)	.9494 $\pm$.009 (10)	LDLFs	.0371 $\pm$.001 (8)	.4088 $\pm$.021 (8)
DF-BFGS (González et al., 2021a)	.3062 $\pm$.025 (3)	.9633 $\pm$.007 (2)	DF-BFGS	.0368 $\pm$.001 (5)	.4161 $\pm$.027 (5)
KLD (Geng, 2016) •	.3608 $\pm$.031 (9)	.9538 $\pm$.008 (8)	LRR •	.0370 $\pm$.001 (7)	.4154 $\pm$.023 (6)
$\mathcal{S}$-KLD	.3007 $\pm$.032 (2)	.9625 $\pm$.009 (3)	$\mathcal{S}$-LRR	**.0366** $\pm$.001 (1)	.4198 $\pm$.023 (2)
SCL (Jia et al., 2019) •	.3358 $\pm$.024 (7)	.9592 $\pm$.006 (6)	QFD^2 (Wen et al., 2023) •	.0369 $\pm$.001 (6)	.4118 $\pm$.025 (7)
$\mathcal{S}$-SCL	.3184 $\pm$.025 (4)	.9604 $\pm$.008 (5)	$\mathcal{S}$-QFD^2	**.0366** $\pm$.001 (1)	**.4203** $\pm$.021 (1)
LRR (Jia et al., 2023) •	.3230 $\pm$.027 (5)	.9616 $\pm$.008 (4)	CJS (Wen et al., 2023)	.0367 $\pm$.001 (4)	.4164 $\pm$.025 (4)
$\mathcal{S}$-LRR	**.2934** $\pm$.028 (1)	**.9635** $\pm$.008 (1)	$\mathcal{S}$-CJS	**.0366** $\pm$.001 (1)	.4198 $\pm$.024 (2)

Table 6. Experimental results of LDL4C on `sBU_3DFE` and `Flickr` formatted as ($\mathtt{mean} \pm \mathtt{std(rank)}$)

Algorithms	`sBU_3DFE` (Geng, 2016)		Algorithms	`Flickr` (Yang et al., 2017b)	
	`0/1 loss` ↓	`Err. prob.` ↓		`0/1 loss` ↓	`Err. prob.` ↓
LDL4C (Wang, Jing and Geng, Xin, 2019)	.5578 $\pm$.028 (6)	.7671 $\pm$.007 (5)	LDL4C	.8971 $\pm$.008 (6)	.8884 $\pm$.004 (6)
$\mathcal{S}$-LDL4C	.5526 $\pm$.025 (5)	.7686 $\pm$.006 (6)	$\mathcal{S}$-LDL4C	.8705 $\pm$.138 (5)	.8702 $\pm$.100 (5)
HR (Wang & Geng, 2021a) •	.5167 $\pm$.027 (3)	.7596 $\pm$.006 (2)	HR •	.4513 $\pm$.015 (4)	.5823 $\pm$.007 (4)
$\mathcal{S}$-HR	.5069 $\pm$.025 (2)	.7598 $\pm$.006 (3)	$\mathcal{S}$-HR	**.4219** $\pm$.015 (1)	**.5639** $\pm$.007 (1)
LDLM (Wang & Geng, 2021b) •	.5258 $\pm$.034 (4)	.7619 $\pm$.009 (4)	LDLM •	.4384 $\pm$.014 (3)	.5740 $\pm$.007 (3)
$\mathcal{S}$-LDLM	**.4809** $\pm$.024 (1)	**.7524** $\pm$.005 (1)	$\mathcal{S}$-LDLM	.4321 $\pm$.016 (2)	.5667 $\pm$.007 (2)

ten-fold experiments repeated 10 times on the `emotion6` dataset and record the average Spearman's coefficient. Results are shown in Fig. 4(a). The panels from left to right display examples of $\mathcal{Y}^{(t)}$s when the λ is 0.01, 0.05, 0.2, 1, and 10, respectively. When λ is suitable, $\mathcal{Y}^{(t)}$s are diverse and do not have excessive local ignorance; as λ decreases, $\mathcal{Y}^{(t)}$s tend to be homogeneous, and invalid masks account for the majority; as λ increases, the local ignorance of each $\mathcal{Y}^{(t)}$ becomes significant.

Besides, we also study the parameter sensitivity of T with λ fixed at 0.2. Results are shown in Fig. 4(b), illustrating that having a plethora of auxiliary tasks are detrimental to performance, which may be due to overfitting.

For $\mathcal{S}$-LDL of the deep regime, we check the sensitivity of the trade-off parameter μ on the LDL task with the `Natural_Scene` dataset by varying the parameter in $\{0.01, 0.05, 0.1, 0.5, 1, 5\}$. Results are shown in Fig. 4(c). Spearman's coefficient of $\mathcal{S}$-LDL first increases and then decreases as μ varies, demonstrating a desirable bell-shaped curve. This justifies our motivation of jointly learning the primary task and subtasks, as a good trade-off between them can enhance the performance.

Ablation study Here we are interested in the importance of each part of $\mathcal{S}$-LDL, thus an ablation study is performed with $\mathcal{S}$-KLD: 1) we replace $\mathcal{N}_{\text{SUM}}$ in `SC` with the min-max function to examine the importance of the subtask distribution reconstruction, and this model is denoted as $\mathcal{S}$-KLD (min-max); 2) we remove the identity mapping in Fig. 3 to examine the importance of the prediction via subtask representation, and this model is denoted as $\mathcal{S}$-KLD w/o id.; 3) we train without the term of ℓ_{SUB} (i.e., setting $\mu = 0$) to examine the importance of subtask learning, and this model is denoted as $\mathcal{S}$-KLD w/o ℓ_{SUB}. Results are also shown in Fig. 4(c), which confirms that each part of $\mathcal{S}$-LDL contributes as long as there is a good trade-off.

7. Limitations and conclusion

Limitations First, $\mathcal{S}$-LDL of the shallow regime is proposed out of intuition, and in Section 5, we have discussed its limitations, which are addressed via the designing of $\mathcal{S}$-LDL of the deep regime. Second, when the label space is large, especially when labels are continuous and result in unimodal label distributions (e.g., age estimation), our proposed cannot be rationally applied. Fortunately, one possible workaround is to use a binning tricks for preprocessing, and then construct subtasks.

Conclusion We propose $\mathcal{S}$-LDL, a subtask learning framework nested into LDL. $\mathcal{S}$-LDL is generic: it generates extra supervised information via subtask construction without any extra knowledge; $\mathcal{S}$-LDL is minimalist: it can be attached to existing methods and handle derivative tasks; $\mathcal{S}$-LDL is efficient: it captures a wide variety of label correlations. The analysis shows the validity and reconstructability of subtasks, and experiments show the superiority of our framework.

Impact statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowledgements

This research is partially supported by Shanghai Central Guidance Science and Technology Development Fund (YDZX20233100004008), the Natural Science Foundation of Jiangsu Province (BK20242045), and the National Natural Science Foundation of China (62176123, 62476130).

References

Chen, S., Wang, J., Chen, Y., Shi, Z., Geng, X., and Rui, Y. Label distribution learning on auxiliary label space graphs for facial expression recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13984–13993, 2020.

Gao, B.-B., Xing, C., Xie, C.-W., Wu, J., and Geng, X. Deep label distribution learning with label ambiguity. *IEEE Transactions on Image Processing*, 26(6):2825–2838, 2017.

Geng, X. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, 2016.

Geng, X. and Hou, P. Pre-release prediction of crowd opinion on movies by label distribution learning. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pp. 3511–3517, 2015.

Geng, X., Yin, C., and Zhou, Z.-H. Facial age estimation by learning from label distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10): 2401–2412, 2013.

González, M., González-Almagro, G., Triguero, I., Cano, J.-R., and García, S. Decomposition-fusion for label distribution learning. *Information Fusion*, 66:64–75, 2021a.

González, M., Luengo, J., Cano, J.-R., and García, S. Synthetic sample generation for label distribution learning. *Information Sciences*, 544:197–213, 2021b.

Jia, X., Li, Z., Zheng, X., Li, W., and Huang, S.-J. Label distribution learning with label correlations on local samples. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1619–1631, 2019.

Jia, X., Shen, X., Li, W., Lu, Y., and Zhu, J. Label distribution learning by maintaining label ranking relation. *IEEE Transactions on Knowledge and Data Engineering*, 35 (02):1695–1707, 2023.

Jin, Y., Gao, R., He, Y., and Zhu, X. Gldl: Graph label distribution learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 12965–12974, 2024.

Kingma, D. P. and Ba, J. L. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

Le, N., Nguyen, K., Tran, Q., Tjiputra, E., Le, B., and Nguyen, A. Uncertainty-aware label distribution learning for facial expression recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6088–6097, 2023.

Li, X., Liang, X., Luo, G., Wang, W., Wang, K., and Li, S. Ambiguity-aware breast tumor cellularity estimation via self-ensemble label distribution learning. *Medical Image Analysis*, 90:102944, 2023a.

Li, X., Luo, G., Wang, W., Wang, K., and Li, S. Curriculum label distribution learning for imbalanced medical image segmentation. *Medical Image Analysis*, 89:102911, 2023b.

Liang, L., Lin, L., Jin, L., Xie, D., and Li, M. Scut-fbp5500: A diverse benchmark dataset for multi-paradigm facial beauty prediction. In *Proceedings of the 24th International Conference on Pattern Recognition*, pp. 1598–1603, 2018.

Lu, Y. and Jia, X. Predicting label distribution from multi-label ranking. In *Proceedings of the 36th Annual Conference on Neural Information Processing Systems*, pp. 36931–36943, 2022.

Lyons, M., Akamatsu, S., Kamachi, M., and Gyoba, J. Coding facial expressions with gabor wavelets. In *Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 200–205, 1998.

Prabhu, Y., Kag, A., Harsola, S., Agrawal, R., and Varma, M. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In *Proceedings of the 2018 World Wide Web Conference*, pp. 993–1002, 2018.

Read, J., Bielza, C., and Larrañaga, P. Multi-dimensional classification with super-classes. *IEEE Transactions on Knowledge and Data Engineering*, 26(7):1720–1733, 2013.

Read, J., Puurula, A., and Bifet, A. Multi-label classification with meta-labels. In *Proceedings of the 2014 IEEE International Conference on Data Mining*, pp. 941–946, 2014.

Ren, T., Jia, X., Li, W., Chen, L., and Li, Z. Label distribution learning with label-specific features. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 3318–3324, 2019.

Shen, W., Zhao, K., Guo, Y., and Yuille, A. Label distribution learning forests. In *Proceedings of the 31st Annual Conference on Neural Information Processing Systems*, pp. 834–843, 2017.

Shirani, A., Dernoncourt, F., Asente, P., Lipka, N., Kim, S., Echevarria, J., and Solorio, T. Learning emphasis selection for written text in visual media from crowd-sourced label distributions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1167–1172, 2019.

Tsoumakas, G., Katakis, I., and Vlahavas, I. Effective and efficient multilabel classification in domains with large number of labels. In *ECML/PKDD 2008 Workshop on Mining Multidimensional Data*, volume 21, pp. 53–59, 2008.

Tsoumakas, G., Katakis, I., and Vlahavas, I. Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1079–1089, 2010.

Wang, J. and Geng, X. Theoretical analysis of label distribution learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5256–5263, 2019.

Wang, J. and Geng, X. Learn the highest label and rest label description degrees. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pp. 3097–3103, 2021a.

Wang, J. and Geng, X. Label distribution learning machine. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 10749–10759. PMLR, 2021b.

Wang, K., Xu, N., Ling, M., and Geng, X. Fast label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(02): 1502–1514, 2023.

Wang, R., Kwong, S., Wang, X., and Jia, Y. Active k-labelsets ensemble for multi-label classification. *Pattern Recognition*, 109:107583, 2021.

Wang, Jing and Geng, Xin. Classification with label distribution learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 3712–3718, 2019.

Wen, C., Zhang, X., Yao, X., and Yang, J. Ordinal label distribution learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23481–23491, 2023.

Wu, X., Wen, N., Liang, J., Lai, Y., She, D., Cheng, M., and Yang, J. Joint acne image grading and counting via label distribution learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10642–10651, 2019.

Xu, M. and Zhou, Z.-H. Incomplete label distribution learning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3175–3181, 2017.

Xu, N., Liu, Y., and Geng, X. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1632–1643, 2019.

Xu, N., Shu, J., Zheng, R., Geng, X., Meng, D., and Zhang, M.-L. Variational label enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (5):6537–6551, 2023.

Yang, J., She, D., and Sun, M. Joint image emotion classification and distribution learning via deep convolutional neural network. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3266–3272, 2017a.

Yang, J., Sun, M., and Sun, X. Learning visual sentiment distributions via augmented conditional probability neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 224–230, 2017b.

Zhai, Y., Dai, J., and Shi, H. Label distribution learning based on ensemble neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing*, pp. 593–602, 2018.

Zhang, M.-L. and Zhou, Z.-H. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2013.

Zhao, X., An, Y., Xu, N., Wang, J., and Geng, X. Imbalanced label distribution learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 11336–11344, 2023.

Zheng, Q., Zhu, J., and Tang, H. Label information bottleneck for label enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7497–7506, 2023.

A. Appendix: details of experiments

Here we try our best to provide as much information as possible for reproducible research.

Table 7. Summary of the metrics

Name	Formula	Name	Formula
`Cheby.` $\downarrow$	$\mathrm{Dis}_1(\boldsymbol{u},\,\boldsymbol{v}) = \max_j \lvert u_j - v_j \rvert$	`Cosine` $\uparrow$	$\mathrm{Sim}_1(\boldsymbol{u},\,\boldsymbol{v}) = \frac{\sum_{j=1}^{L} u_j v_j}{\sqrt{\sum_{j=1}^{L} u_j^2}\sqrt{\sum_{j=1}^{L} v_j^2}}$
`Clark` $\downarrow$	$\mathrm{Dis}_2(\boldsymbol{u},\,\boldsymbol{v}) = \sqrt{\sum_{j=1}^{L} \frac{(u_j - v_j)^2}{(u_j + v_j)^2}}$	`Int.` $\uparrow$	$\mathrm{Sim}_2(\boldsymbol{u},\,\boldsymbol{v}) = \sum_{j=1}^{L} \min(u_j,\,v_j)$
`Can.` $\downarrow$	$\mathrm{Dis}_3(\boldsymbol{u},\,\boldsymbol{v}) = \sum_{j=1}^{L} \frac{\lvert u_j - v_j \rvert}{u_j + v_j}$	`Spear.` $\uparrow$	$\mathrm{Rnk}_1(\boldsymbol{u},\,\boldsymbol{v}) = 1 - \frac{6\sum_{j=1}^{L}(\rho(u_j)-\rho(v_j))^2}{L(L^2-1)}$
`KLD` $\downarrow$	$\mathrm{Dis}_4(\boldsymbol{u},\,\boldsymbol{v}) = \sum_{j=1}^{L} u_j \ln \frac{u_j}{v_j}$	`Ken.` $\uparrow$	$\mathrm{Rnk}_2(\boldsymbol{u},\,\boldsymbol{v}) = \frac{2\sum_{j<k}\mathrm{sgn}(u_j-u_k)\mathrm{sgn}(v_j-v_k)}{L(L-1)}$
`0/1 loss` $\downarrow$	$\mathrm{C}_1(\boldsymbol{u},\,\boldsymbol{v}) = \delta(\arg\max(\boldsymbol{u}),\ \arg\max(\boldsymbol{v}))$	`Err. prob.` $\downarrow$	$\mathrm{C}_2(\boldsymbol{u},\,\boldsymbol{v}) = 1 - u_{\arg\max(\boldsymbol{v})}$

A.1. Metrics

For LDL, IncomLDL, and LE, we use the same metrics suggested by Geng (2016), which are `Cheby.` $\downarrow$ (Chebyshev distance), `Clark` $\downarrow$ (Clark distance), `Can.` $\downarrow$ (Canberra distance), `KLD` $\downarrow$ (Kullback-Leibler divergence), `Cosine` $\uparrow$ (cosine similarity), and `Int.` $\uparrow$ (intersection similarity), respectively. Here $\downarrow$ ($\uparrow$) indicates "the lower (higher) the better". For LDL and IncomLDL, we additionally use two ranking metrics: `Spear.` $\uparrow$ (Spearman's coefficient) and `Ken.` $\uparrow$ (Kendall's coefficient) (Jia et al., 2023). Note that these metrics are *not* as intuitive as accuracy or error rate, i.e., *small changes can mean large performance differences.* For LDL4C, objective of which is different from LDL, we use `0/1 loss` $\downarrow$ (zero one loss) and `Err. prob.` $\downarrow$ (error probability) as metrics (Wang, Jing and Geng, Xin, 2019). Let the real distribution be denoted by $\boldsymbol{u} = \{u_j\}_{j=1}^{L}$, and the predicted distribution be denoted by $\boldsymbol{v} = \{v_j\}_{j=1}^{L}$, then the above metrics can be summarized in Table 7, where $\rho(\cdot)$ and $\delta(\cdot,\,\cdot)$ are the ranking function and the Kronecker delta function, respectively.

Table 8. Summary of datasets

Dataset	# Instances N	# Features P	# Labels L
`JAFFE`	213	243	6
`sBU_3DFE`	2500	243	6
`Movie`	7755	1869	5
`Nature_Scene`	2000	294	9
`fbp5500`	5500	512	5
`Yeast_heat`	2465	24	6
`Yeast_diau`	2465	24	7
`Yeast_cold`	2465	24	4
`Yeast_dtt`	2465	24	4
`emotion6`	1980	168	7
`Twitter`	10045	168	8
`Flickr`	11150	168	8

A.2. Datasets

We adopt several widely used label distribution datasets, including: `JAFFE` (Lyons et al., 1998);[5] `fbp5500` (Liang et al., 2018);[6] `sBU_3DFE`, `Movie`, `Natural_Scene`, `Yeast_heat`, `Yeast_diau`, `Yeast_cold`, and `Yeast_dtt` provided by Geng (2016);[7] `emotion6`, `Twitter`, and `Flickr` provided by Yang et al. (2017b).[8] The information of these datasets are summarized in Table 8.

[5] https://zenodo.org/records/3451524

[6] https://github.com/HCIILAB/SCUT-FBP5500-Database-Release

[7] https://palm.seu.edu.cn/xgeng/LDL/download.htm

[8] https://cv.nankai.edu.cn/projects

A.3. Comparison methods

On the one hand, we apply our proposed $\mathcal{S}$-LDL to existing methods to demonstrate performance improvements in the LDL task (denoted by the "$\mathcal{S}$-" prefix). These methods are BFGS-LLD (KLD) (Geng, 2016), SCL (Jia et al., 2019), LRR (Jia et al., 2023), QFD^2 (Wen et al., 2023), and CJS (Wen et al., 2023) (the losses of these methods constitute the set $\mathcal{L}_{\text{LDL}}$). On the other hand, we compare $\mathcal{S}$-LDL with methods that have specialized structure, which our proposed cannot directly adapt to. These methods are CPNN (Geng et al., 2013), LDSVR (Geng & Hou, 2015), AA-kNN (Geng, 2016), LDLFs (Shen et al., 2017), and DF-LDL (denoted by DF-BFGS since we use BFGS-LLDs as base estimators) (González et al., 2021a). Moreover, we apply our proposed to derivative tasks of LDL (i.e., LDL4C, IncomLDL, and LE) and the comparison methods involved are LDL4C (Wang, Jing and Geng, Xin, 2019), HR (Wang & Geng, 2021a), LDLM (Wang & Geng, 2021b), IncomLDL (Xu & Zhou, 2017), LP (Xu et al., 2019), GLLE (Xu et al., 2019), LEVI (Xu et al., 2023), and LIBLE (Zheng et al., 2023).

Table 9. Summary of algorithms and parameter settings

Algorithms	Parameter	Value (Range)
AA-kNN	k: # Neighbors	5
LDLFs	# Estimators (trees)	5
	Depth	6
	Latent units (leaves)	64
BFGS-LLD	ε: Convergence criterion	10^{-6}
	Max iteration	600
SCL	m: # Clusters	5
	λ_1, λ_2, λ_3: Trade-off	10^{-3}, 10^{-3}, 0.1
LRR	λ: Trade-off (ranking loss)	$10^{\{-5,\,-4,\,-3,\,-2,\,-1\}}$
	β: Trade-off (regularization)	$10^{\{-3,\,-2,\,-1,\,0,\,1,\,2\}}$
LDL4C	C_1, C_2: Balance coefficients	10^{-2}, 10^{-6}
	ρ: Margin	10^{-2}
HR	λ_1, λ_2, λ_3: Trade-off	10^{-2}, 10^{-6}
	ρ: Margin	10^{-2}
LDLM	λ_1, λ_2, λ_3: Trade-off	10^{-6}, $10^{\{-3,\,-2,\,-1\}}$, $10^{\{-3,\,-2,\,-1\}}$
	ρ: Margin	10^{-2}
IncomLDL	ε: Convergence criterion	10^{-6}
	γ: Factor of Lipschitz constant	2
	λ: Trade-off	1
LP	α: Balance coefficient	0.5
GLLE	λ_1, λ_2: Trade-off	10^{-2}, 10^{-4}
	σ: Width parameter for similarity calculation	10
LEVI	λ: Trade-off	1
LIBLE	α, β: Trade-off	$10^{\{-3,\,-2,\,-1,\,0,\,1,\,2\}}$
$\mathcal{S}$-LDL	μ, λ, T	0.1, 0.2, 10

A.4. Parameter settings and experimental environment

The parameter settings of the proposed $\mathcal{S}$-LDL and comparison algorithms are summarized in Table 9. Note that DF-LDL is parameter-free, and we use BFGS-LLDs as its base estimators, parameter settings of which are the same as BFGS-LLD as the comparison algorithm. We use Adam (Kingma & Ba, 2015) for the optimization of $\mathcal{S}$-LDL. For all methods of the deep regime, the learning rate is chosen among $\{1,\ 2,\ 5\} \times 10^{\{-4,\,-3,\,-2\}}$, and the selection of the number of epochs is nested into a ten-fold cross validation. All the results are obtained on a Linux workstation with Intel Core i9 (3.70GHz), NVIDIA GeForce RTX 3090 (24GB), and 32GB memory.

A.5. Full analytical/experimental results

Here we provide complete results of all conducted analysis/experiments. Figures 5 to 9 are full results of validity analysis in Section 4.1. Tables 10 to 21 are results on the LDL task with different datasets. For IncomLDL, we follow *the incomplete* *settings* (Xu & Zhou, 2017) and vary the observed rate $\omega\%$ from 20% to 40%. Tables 22 to 23 are results on the IncomLDL task. Tables 24 to 25 are on the LDL4C task. For LE, we follow *the settings of the recovery experiment* (Xu et al., 2019). Tables 26 to 27 show results on the LE task.

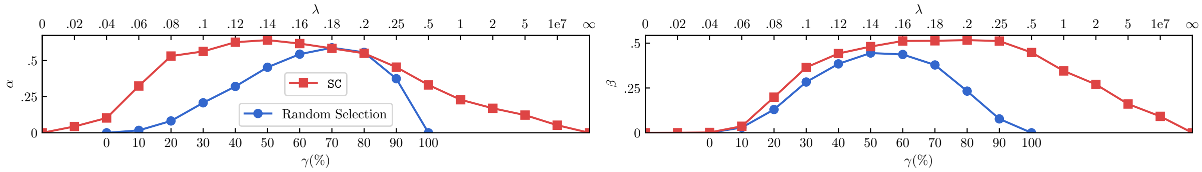


Figure 5. Information/diversity index w.r.t. λ on `emotion6`.

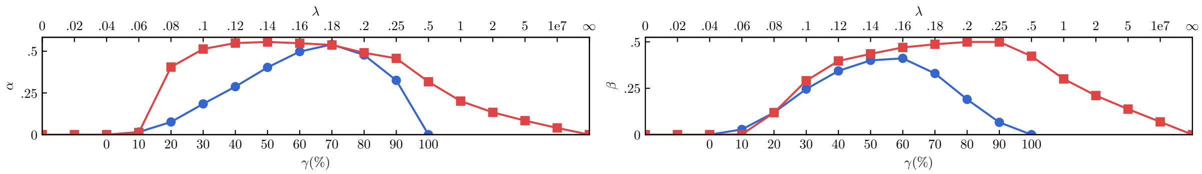


Figure 6. Information/diversity index w.r.t. λ on `JAFFE`.

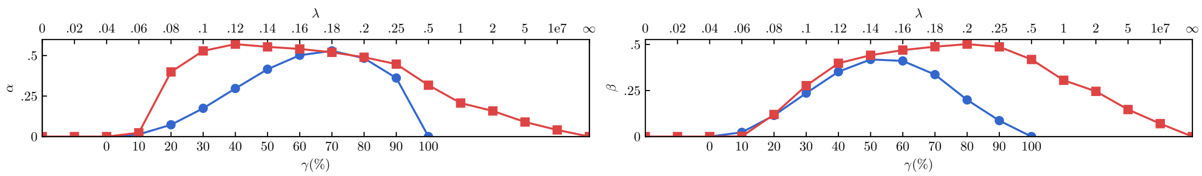


Figure 7. Information/diversity index w.r.t. λ on `SBU_3DFE`.

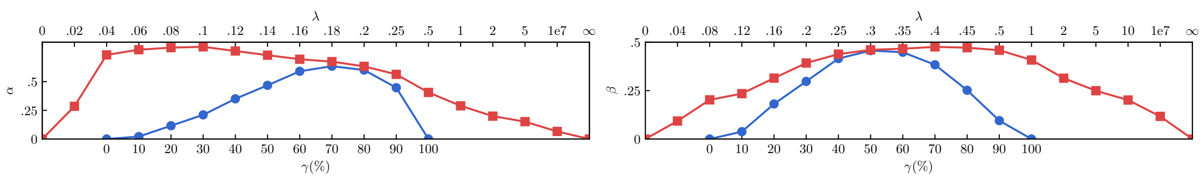


Figure 8. Information/diversity index w.r.t. λ on `Twitter`.

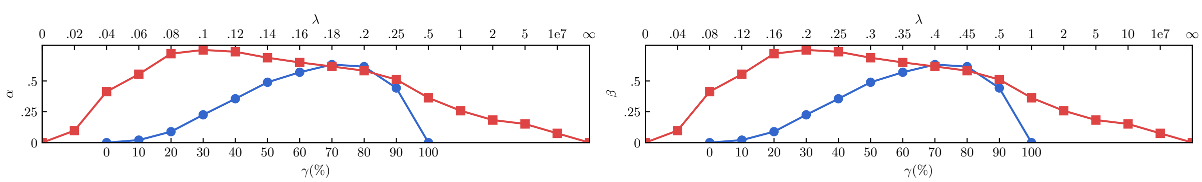


Figure 9. Information/diversity index w.r.t. λ on `Flickr`.

Table 10. Experimental results of LDL on the `JAFFE` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR •	$.0959_{\pm.013}$	$.3280_{\pm.027}$	$.6778_{\pm.058}$	$.0476_{\pm.011}$	$.9549_{\pm.010}$	$.8838_{\pm.012}$	$.5175_{\pm.102}$	$.4508_{\pm.086}$
$\mathcal{S}$-LDSVR	$.0859_{\pm.012}$	$.3024_{\pm.033}$	$.6143_{\pm.066}$	$.0449_{\pm.012}$	$.9580_{\pm.011}$	$.8941_{\pm.012}$	$.5237_{\pm.091}$	$.4580_{\pm.079}$
AA-kNN	$.0978_{\pm.012}$	$.3483_{\pm.032}$	$.7164_{\pm.066}$	$.0527_{\pm.011}$	$.9497_{\pm.010}$	$.8766_{\pm.012}$	$.4111_{\pm.083}$	$.3514_{\pm.070}$
LDLFs	$.0940_{\pm.010}$	$.3637_{\pm.032}$	$.7355_{\pm.066}$	$.0550_{\pm.009}$	$.9494_{\pm.009}$	$.8766_{\pm.011}$	$.4364_{\pm.108}$	$.3749_{\pm.093}$
DF-BFGS	$.0827_{\pm.009}$	$.3062_{\pm.025}$	$.6239_{\pm.052}$	$.0388_{\pm.007}$	$.9633_{\pm.007}$	$.8944_{\pm.010}$	$.5244_{\pm.087}$	$.4493_{\pm.077}$
KLD •	$.0925_{\pm.010}$	$.3608_{\pm.031}$	$.7363_{\pm.064}$	$.0508_{\pm.009}$	$.9538_{\pm.008}$	$.8777_{\pm.011}$	$.4572_{\pm.097}$	$.3873_{\pm.084}$
$\mathcal{S}$-KLD	$.0818_{\pm.011}$	$.3007_{\pm.032}$	$.6132_{\pm.067}$	$.0395_{\pm.010}$	$.9625_{\pm.009}$	$.8960_{\pm.012}$	$\mathbf{.5461}_{\pm.105}$	$.4769_{\pm.096}$
SCL •	$.0873_{\pm.008}$	$.3358_{\pm.024}$	$.6874_{\pm.051}$	$.0439_{\pm.006}$	$.9592_{\pm.006}$	$.8851_{\pm.009}$	$.4744_{\pm.092}$	$.4020_{\pm.080}$
$\mathcal{S}$-SCL	$.0854_{\pm.010}$	$.3184_{\pm.025}$	$.6526_{\pm.053}$	$.0420_{\pm.008}$	$.9604_{\pm.008}$	$.8896_{\pm.010}$	$.5110_{\pm.095}$	$.4388_{\pm.084}$
LRR •	$.0853_{\pm.010}$	$.3230_{\pm.027}$	$.6560_{\pm.055}$	$.0412_{\pm.008}$	$.9616_{\pm.008}$	$.8906_{\pm.010}$	$.5117_{\pm.094}$	$.4420_{\pm.084}$
$\mathcal{S}$-LRR	$\mathbf{.0804}_{\pm.009}$	$\mathbf{.2934}_{\pm.028}$	$\mathbf{.5989}_{\pm.059}$	$\mathbf{.0383}_{\pm.009}$	$\mathbf{.9635}_{\pm.008}$	$\mathbf{.8981}_{\pm.011}$	$.5448_{\pm.092}$	$\mathbf{.4819}_{\pm.084}$

Table 11. Experimental results of LDL on the `sBU_3DFE` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR •	$.1250_{\pm.005}$	$.3710_{\pm.010}$	$.8009_{\pm.021}$	$.0720_{\pm.004}$	$.9298_{\pm.004}$	$.8559_{\pm.004}$	$.3524_{\pm.031}$	$.3011_{\pm.026}$
$\mathcal{S}$-LDSVR	$.1193_{\pm.004}$	$.3573_{\pm.010}$	$.7600_{\pm.022}$	$.0672_{\pm.004}$	$.9338_{\pm.004}$	$.8625_{\pm.004}$	$.3759_{\pm.030}$	$.3236_{\pm.026}$
AA-kNN	$.1272_{\pm.004}$	$.4001_{\pm.009}$	$.8281_{\pm.020}$	$.0801_{\pm.004}$	$.9217_{\pm.004}$	$.8488_{\pm.004}$	$.2053_{\pm.030}$	$.1767_{\pm.026}$
LDLFs	$.1016_{\pm.003}$	$.3262_{\pm.008}$	$.6841_{\pm.017}$	$.0504_{\pm.003}$	$.9499_{\pm.003}$	$.8776_{\pm.003}$	$.4212_{\pm.023}$	$.3620_{\pm.019}$
DF-BFGS	$.1146_{\pm.004}$	$.3616_{\pm.008}$	$.7627_{\pm.019}$	$.0618_{\pm.003}$	$.9388_{\pm.003}$	$.8626_{\pm.004}$	$.3026_{\pm.031}$	$.2621_{\pm.026}$
KLD •	$.1147_{\pm.004}$	$.3697_{\pm.008}$	$.7804_{\pm.019}$	$.0624_{\pm.003}$	$.9387_{\pm.003}$	$.8604_{\pm.003}$	$.3021_{\pm.026}$	$.2643_{\pm.022}$
$\mathcal{S}$-KLD	$.1014_{\pm.004}$	$.3203_{\pm.009}$	$.6736_{\pm.018}$	$.0514_{\pm.003}$	$.9487_{\pm.003}$	$.8789_{\pm.004}$	$.4334_{\pm.025}$	$.3729_{\pm.022}$
SCL •	$.1145_{\pm.004}$	$.3648_{\pm.008}$	$.7748_{\pm.018}$	$.0605_{\pm.003}$	$.9404_{\pm.003}$	$.8614_{\pm.003}$	$.3091_{\pm.026}$	$.2701_{\pm.021}$
$\mathcal{S}$-SCL	$.1041_{\pm.004}$	$.3301_{\pm.009}$	$.6936_{\pm.019}$	$.0535_{\pm.003}$	$.9468_{\pm.003}$	$.8754_{\pm.004}$	$.3956_{\pm.030}$	$.3381_{\pm.027}$
LRR •	$.1067_{\pm.003}$	$.3476_{\pm.008}$	$.7320_{\pm.017}$	$.0543_{\pm.003}$	$.9465_{\pm.003}$	$.8695_{\pm.003}$	$.3626_{\pm.026}$	$.3123_{\pm.022}$
$\mathcal{S}$-LRR	$\mathbf{.0996}_{\pm.004}$	$\mathbf{.3157}_{\pm.008}$	$\mathbf{.6610}_{\pm.017}$	$\mathbf{.0499}_{\pm.003}$	$\mathbf{.9502}_{\pm.003}$	$\mathbf{.8812}_{\pm.003}$	$\mathbf{.4455}_{\pm.026}$	$\mathbf{.3837}_{\pm.023}$

Table 12. Experimental results of LDL on the `Yeast_heat` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
CPNN	$.0419_{\pm.001}$	$.1818_{\pm.005}$	$.3633_{\pm.009}$	$.0125_{\pm.001}$	$.9881_{\pm.001}$	$.9404_{\pm.001}$	$.1507_{\pm.034}$	$.1221_{\pm.028}$
AA-kNN	$.0441_{\pm.001}$	$.1913_{\pm.005}$	$.3840_{\pm.010}$	$.0140_{\pm.001}$	$.9867_{\pm.001}$	$.9370_{\pm.002}$	$.1678_{\pm.031}$	$.1384_{\pm.026}$
LDLFs	$.0420_{\pm.001}$	$.1818_{\pm.005}$	$.3627_{\pm.009}$	$.0125_{\pm.001}$	$.9881_{\pm.001}$	$.9405_{\pm.001}$	$.1731_{\pm.032}$	$.1409_{\pm.026}$
DF-BFGS	$.0420_{\pm.001}$	$.1816_{\pm.005}$	$.3624_{\pm.009}$	$.0125_{\pm.001}$	$.9881_{\pm.001}$	$.9405_{\pm.001}$	$\mathbf{.1964}_{\pm.034}$	$\mathbf{.1624}_{\pm.028}$
LRR •	$.0423_{\pm.001}$	$.1828_{\pm.005}$	$.3644_{\pm.009}$	$.0126_{\pm.001}$	$.9880_{\pm.001}$	$.9402_{\pm.001}$	$.1655_{\pm.033}$	$.1351_{\pm.028}$
$\mathcal{S}$-LRR	$\mathbf{.0417}_{\pm.001}$	$.1806_{\pm.005}$	$.3609_{\pm.009}$	$\mathbf{.0124}_{\pm.001}$	$\mathbf{.9882}_{\pm.001}$	$.9408_{\pm.001}$	$.1882_{\pm.034}$	$.1548_{\pm.028}$
QFD^2 •	$.0423_{\pm.001}$	$.1827_{\pm.005}$	$.3644_{\pm.009}$	$.0126_{\pm.001}$	$.9880_{\pm.001}$	$.9402_{\pm.001}$	$.1677_{\pm.032}$	$.1351_{\pm.027}$
$\mathcal{S}$-QFD^2	$\mathbf{.0417}_{\pm.001}$	$.1808_{\pm.005}$	$.3611_{\pm.009}$	$\mathbf{.0124}_{\pm.001}$	$\mathbf{.9882}_{\pm.001}$	$.9408_{\pm.001}$	$.1880_{\pm.032}$	$.1544_{\pm.027}$
CJS •	$.0423_{\pm.001}$	$.1827_{\pm.005}$	$.3643_{\pm.009}$	$.0126_{\pm.001}$	$.9880_{\pm.001}$	$.9402_{\pm.001}$	$.1632_{\pm.032}$	$.1329_{\pm.027}$
$\mathcal{S}$-CJS	$\mathbf{.0417}_{\pm.001}$	$\mathbf{.1804}_{\pm.005}$	$\mathbf{.3603}_{\pm.009}$	$\mathbf{.0124}_{\pm.001}$	$\mathbf{.9882}_{\pm.001}$	$\mathbf{.9409}_{\pm.001}$	$.1940_{\pm.030}$	$.1589_{\pm.025}$

Table 13. Experimental results of LDL on the `Yeast_diau` dataset formatted as (`mean` $\pm$ `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
CPNN	.0385$_{\pm.001}$	.2069$_{\pm.006}$	.4439$_{\pm.012}$	.0138$_{\pm.001}$	.9872$_{\pm.001}$	.9383$_{\pm.002}$	.2962$_{\pm.034}$	.2427$_{\pm.027}$
AA-kNN	.0385$_{\pm.001}$	.2085$_{\pm.006}$	.4487$_{\pm.014}$	.0145$_{\pm.001}$	.9867$_{\pm.001}$	.9377$_{\pm.002}$	.3674$_{\pm.029}$	.2976$_{\pm.024}$
LDLFs	.0371$_{\pm.001}$	.2014$_{\pm.006}$	.4324$_{\pm.012}$	.0132$_{\pm.001}$	.9879$_{\pm.001}$	.9401$_{\pm.002}$	.4088$_{\pm.021}$	.3254$_{\pm.018}$
DF-BFGS	.0368$_{\pm.001}$	.1999$_{\pm.006}$	.4294$_{\pm.013}$	.0131$_{\pm.001}$	.9879$_{\pm.001}$	.9405$_{\pm.002}$	.4161$_{\pm.027}$	**.3404**$_{\pm.022}$
LRR •	.0370$_{\pm.001}$	.2007$_{\pm.006}$	.4307$_{\pm.012}$	.0131$_{\pm.001}$	.9879$_{\pm.001}$	.9403$_{\pm.002}$	.4154$_{\pm.023}$	.3343$_{\pm.020}$
$\mathcal{S}$-LRR	**.0366**$_{\pm.001}$	**.1983**$_{\pm.006}$	**.4257**$_{\pm.012}$	**.0129**$_{\pm.001}$	**.9881**$_{\pm.001}$	**.9410**$_{\pm.002}$	.4198$_{\pm.023}$	.3389$_{\pm.019}$
QFD^2 •	.0369$_{\pm.001}$	.2000$_{\pm.006}$	.4296$_{\pm.012}$	.0131$_{\pm.001}$	.9879$_{\pm.001}$	.9404$_{\pm.002}$	.4118$_{\pm.025}$	.3326$_{\pm.021}$
$\mathcal{S}$-QFD^2	**.0366**$_{\pm.001}$	.1985$_{\pm.006}$	.4261$_{\pm.012}$	**.0129**$_{\pm.001}$	**.9881**$_{\pm.001}$	.9409$_{\pm.002}$	**.4203**$_{\pm.021}$	.3387$_{\pm.018}$
CJS	.0367$_{\pm.001}$	.1989$_{\pm.006}$	.4272$_{\pm.012}$	.0130$_{\pm.001}$	.9880$_{\pm.001}$	.9408$_{\pm.002}$	.4164$_{\pm.025}$	.3366$_{\pm.021}$
$\mathcal{S}$-CJS	**.0366**$_{\pm.001}$	.1984$_{\pm.006}$	.4260$_{\pm.012}$	.0130$_{\pm.001}$	**.9881**$_{\pm.001}$	.9409$_{\pm.002}$	.4198$_{\pm.024}$	.3392$_{\pm.019}$

Table 14. Experimental results of LDL on the `Yeast_cold` dataset formatted as (`mean` $\pm$ `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
CPNN	**.0510**$_{\pm.002}$	.1392$_{\pm.005}$	.2396$_{\pm.008}$	**.0121**$_{\pm.001}$	**.9886**$_{\pm.001}$	**.9410**$_{\pm.002}$	**.2651**$_{\pm.036}$	**.2263**$_{\pm.032}$
AA-kNN	.0542$_{\pm.002}$	.1476$_{\pm.005}$	.2549$_{\pm.008}$	.0135$_{\pm.001}$	.9872$_{\pm.001}$	.9371$_{\pm.002}$	.2189$_{\pm.035}$	.1866$_{\pm.031}$
LDLFs	.0511$_{\pm.002}$	.1396$_{\pm.005}$	.2404$_{\pm.009}$	.0122$_{\pm.001}$	.9885$_{\pm.001}$	.9408$_{\pm.002}$	.2482$_{\pm.038}$	.2112$_{\pm.033}$
DF-BFGS	.0514$_{\pm.002}$	.1404$_{\pm.005}$	.2424$_{\pm.008}$	.0123$_{\pm.001}$	.9885$_{\pm.001}$	.9403$_{\pm.002}$	.2581$_{\pm.036}$	.2190$_{\pm.030}$
LRR	.0511$_{\pm.002}$	.1395$_{\pm.005}$	.2402$_{\pm.009}$	.0122$_{\pm.001}$	**.9886**$_{\pm.001}$	.9408$_{\pm.002}$	.2490$_{\pm.035}$	.2111$_{\pm.030}$
$\mathcal{S}$-LRR	**.0510**$_{\pm.002}$	**.1391**$_{\pm.005}$	**.2395**$_{\pm.009}$	**.0121**$_{\pm.001}$	**.9886**$_{\pm.001}$	**.9410**$_{\pm.002}$	.2618$_{\pm.037}$	.2238$_{\pm.032}$
QFD^2	.0513$_{\pm.002}$	.1401$_{\pm.005}$	.2413$_{\pm.009}$	.0123$_{\pm.001}$	.9885$_{\pm.001}$	.9405$_{\pm.002}$	.2534$_{\pm.037}$	.2158$_{\pm.032}$
$\mathcal{S}$-QFD^2	**.0510**$_{\pm.002}$	**.1391**$_{\pm.005}$	.2396$_{\pm.008}$	**.0121**$_{\pm.001}$	**.9886**$_{\pm.001}$	**.9410**$_{\pm.002}$	.2571$_{\pm.039}$	.2197$_{\pm.033}$
CJS	.0513$_{\pm.002}$	.1401$_{\pm.005}$	.2412$_{\pm.008}$	.0123$_{\pm.001}$	.9884$_{\pm.001}$	.9406$_{\pm.002}$	.2535$_{\pm.038}$	.2152$_{\pm.032}$
$\mathcal{S}$-CJS	**.0510**$_{\pm.002}$	.1392$_{\pm.005}$	.2396$_{\pm.009}$	**.0121**$_{\pm.001}$	**.9886**$_{\pm.001}$	**.9410**$_{\pm.002}$	.2621$_{\pm.037}$	.2241$_{\pm.031}$

Table 15. Experimental results of LDL on the `Yeast_dtt` dataset formatted as (`mean` $\pm$ `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
CPNN	.0361$_{\pm.001}$	.0984$_{\pm.004}$	.1690$_{\pm.006}$	.0063$_{\pm.001}$	.9941$_{\pm.000}$	.9583$_{\pm.001}$	.1735$_{\pm.035}$	.1494$_{\pm.030}$
AA-kNN	.0386$_{\pm.001}$	.1047$_{\pm.004}$	.1797$_{\pm.006}$	.0071$_{\pm.001}$	.9933$_{\pm.000}$	.9556$_{\pm.001}$	.1591$_{\pm.033}$	.1399$_{\pm.030}$
LDLFs	.0360$_{\pm.001}$	.0981$_{\pm.004}$	.1689$_{\pm.006}$	.0063$_{\pm.001}$	**.9941**$_{\pm.000}$	.9583$_{\pm.001}$	.1986$_{\pm.038}$	.1727$_{\pm.034}$
DF-BFGS	.0365$_{\pm.001}$	.0995$_{\pm.004}$	.1712$_{\pm.006}$	.0064$_{\pm.001}$	.9939$_{\pm.000}$	.9578$_{\pm.001}$	.1804$_{\pm.033}$	.1592$_{\pm.030}$
LRR	.0360$_{\pm.001}$	.0982$_{\pm.004}$	.1690$_{\pm.006}$	.0063$_{\pm.001}$	**.9941**$_{\pm.000}$	.9583$_{\pm.001}$	.2016$_{\pm.037}$	.1738$_{\pm.032}$
$\mathcal{S}$-LRR	**.0359**$_{\pm.001}$	**.0977**$_{\pm.004}$	**.1680**$_{\pm.006}$	**.0062**$_{\pm.001}$	**.9941**$_{\pm.000}$	**.9585**$_{\pm.001}$	.2068$_{\pm.035}$	.1811$_{\pm.031}$
QFD^2	.0362$_{\pm.001}$	.0986$_{\pm.004}$	.1696$_{\pm.006}$	.0063$_{\pm.001}$	.9940$_{\pm.000}$	.9582$_{\pm.001}$	.1917$_{\pm.035}$	.1665$_{\pm.031}$
$\mathcal{S}$-QFD^2	**.0359**$_{\pm.001}$	**.0977**$_{\pm.004}$	.1681$_{\pm.006}$	**.0062**$_{\pm.001}$	**.9941**$_{\pm.000}$	**.9585**$_{\pm.001}$	**.2086**$_{\pm.036}$	**.1822**$_{\pm.032}$
CJS	.0361$_{\pm.001}$	.0984$_{\pm.004}$	.1692$_{\pm.006}$	.0063$_{\pm.001}$	**.9941**$_{\pm.000}$	.9582$_{\pm.001}$	.1975$_{\pm.040}$	.1722$_{\pm.035}$
$\mathcal{S}$-CJS	**.0359**$_{\pm.001}$	.0978$_{\pm.004}$	.1682$_{\pm.006}$	**.0062**$_{\pm.001}$	**.9941**$_{\pm.000}$	**.9585**$_{\pm.001}$	.2080$_{\pm.035}$	.1804$_{\pm.031}$

Table 16. Experimental results of LDL on the `emotion6` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR	$.3152_{\pm.010}$	$1.8217_{\pm.020}$	$4.1452_{\pm.064}$	$1.0744_{\pm.081}$	$.6906_{\pm.015}$	$.5773_{\pm.012}$	$.3915_{\pm.030}$	$.3235_{\pm.025}$
$\mathcal{S}$-LDSVR	$.3132_{\pm.010}$	$1.8236_{\pm.016}$	$4.1484_{\pm.055}$	$1.0672_{\pm.072}$	$.6922_{\pm.014}$	$.5792_{\pm.011}$	$.3952_{\pm.028}$	$.3280_{\pm.023}$
AA-kNN	$.3288_{\pm.011}$	$1.7116_{\pm.026}$	$3.8757_{\pm.076}$	$.9512_{\pm.115}$	$.6632_{\pm.013}$	$.5564_{\pm.010}$	$.2920_{\pm.027}$	$.2401_{\pm.022}$
LDLFs	$.3120_{\pm.010}$	$1.6625_{\pm.026}$	$3.7330_{\pm.075}$	$.5871_{\pm.024}$	$.7143_{\pm.011}$	$.5802_{\pm.010}$	$.3631_{\pm.029}$	$.3025_{\pm.024}$
DF-BFGS	$.3026_{\pm.010}$	$1.6765_{\pm.025}$	$3.7675_{\pm.071}$	$.5805_{\pm.026}$	$.7206_{\pm.013}$	$.5909_{\pm.010}$	$.3940_{\pm.027}$	$.3256_{\pm.022}$
KLD •	$.3037_{\pm.010}$	$1.6774_{\pm.025}$	$3.7729_{\pm.074}$	$.5863_{\pm.027}$	$.7191_{\pm.013}$	$.5897_{\pm.011}$	$.3959_{\pm.028}$	$.3259_{\pm.023}$
$\mathcal{S}$-KLD	$.3024_{\pm.010}$	$1.6548_{\pm.026}$	$3.6984_{\pm.075}$	$.5631_{\pm.024}$	$.7282_{\pm.012}$	$.5926_{\pm.010}$	$.4063_{\pm.027}$	$.3361_{\pm.023}$
SCL •	$.3020_{\pm.010}$	$1.6750_{\pm.025}$	$3.7642_{\pm.073}$	$.5803_{\pm.027}$	$.7219_{\pm.013}$	$.5917_{\pm.011}$	$.4003_{\pm.028}$	$.3299_{\pm.023}$
$\mathcal{S}$-SCL	$\mathbf{.3018}_{\pm.010}$	$1.6554_{\pm.026}$	$3.6993_{\pm.076}$	$.5631_{\pm.025}$	$.7281_{\pm.012}$	$\mathbf{.5936}_{\pm.010}$	$\mathbf{.4089}_{\pm.027}$	$\mathbf{.3383}_{\pm.023}$
LRR •	$.3030_{\pm.010}$	$1.6736_{\pm.025}$	$3.7601_{\pm.073}$	$.5804_{\pm.026}$	$.7212_{\pm.013}$	$.5899_{\pm.010}$	$.3941_{\pm.027}$	$.3243_{\pm.023}$
$\mathcal{S}$-LRR	$.3028_{\pm.009}$	$\mathbf{1.6524}_{\pm.026}$	$\mathbf{3.6923}_{\pm.074}$	$\mathbf{.5607}_{\pm.023}$	$\mathbf{.7299}_{\pm.011}$	$.5923_{\pm.010}$	$.4078_{\pm.027}$	$.3373_{\pm.022}$

Table 17. Experimental results of LDL on the `Twitter` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR ○	$.4236_{\pm.008}$	$2.6722_{\pm.002}$	$7.3015_{\pm.009}$	$5.0018_{\pm.115}$	$.7627_{\pm.008}$	$.5761_{\pm.008}$	$.5237_{\pm.008}$	$.4246_{\pm.007}$
$\mathcal{S}$-LDSVR	$.4238_{\pm.009}$	$2.6723_{\pm.002}$	$7.3018_{\pm.010}$	$5.0051_{\pm.124}$	$.7623_{\pm.009}$	$.5229_{\pm.007}$	$.5229_{\pm.007}$	$.4238_{\pm.006}$
AA-kNN	$.3172_{\pm.004}$	$\mathbf{2.0142}_{\pm.012}$	$\mathbf{4.5597}_{\pm.043}$	$3.1429_{\pm.148}$	$.7926_{\pm.006}$	$.6024_{\pm.005}$	$.5014_{\pm.009}$	$.4432_{\pm.008}$
LDLFs	$.4035_{\pm.014}$	$2.5461_{\pm.010}$	$6.8269_{\pm.040}$	$1.6884_{\pm.115}$	$.6756_{\pm.018}$	$.5318_{\pm.013}$	$.4164_{\pm.013}$	$.3349_{\pm.010}$
DF-BFGS	$.2982_{\pm.004}$	$2.4025_{\pm.005}$	$6.2416_{\pm.020}$	$.6304_{\pm.012}$	$.8250_{\pm.006}$	$.6220_{\pm.004}$	$.5467_{\pm.008}$	$.4454_{\pm.007}$
KLD ○	$.2966_{\pm.004}$	$2.4059_{\pm.005}$	$6.2558_{\pm.020}$	$.6307_{\pm.013}$	$.8243_{\pm.006}$	$.6249_{\pm.005}$	$.5470_{\pm.008}$	$.4456_{\pm.007}$
$\mathcal{S}$-KLD	$.2995_{\pm.005}$	$2.4112_{\pm.005}$	$6.2883_{\pm.020}$	$.6491_{\pm.013}$	$.8203_{\pm.006}$	$.6205_{\pm.005}$	$.5384_{\pm.009}$	$.4385_{\pm.008}$
SCL •	$.2977_{\pm.004}$	$2.4028_{\pm.005}$	$6.2435_{\pm.021}$	$.6262_{\pm.013}$	$.8256_{\pm.006}$	$.6233_{\pm.005}$	$.5488_{\pm.008}$	$.4471_{\pm.007}$
$\mathcal{S}$-SCL	$.2940_{\pm.005}$	$2.4059_{\pm.006}$	$6.2589_{\pm.023}$	$.6203_{\pm.013}$	$.8268_{\pm.006}$	$.6281_{\pm.006}$	$.5518_{\pm.008}$	$.4497_{\pm.007}$
LRR •	$.2984_{\pm.004}$	$2.4046_{\pm.005}$	$6.2525_{\pm.019}$	$.6351_{\pm.012}$	$.8232_{\pm.006}$	$.6220_{\pm.004}$	$.5443_{\pm.008}$	$\mathbf{.4636}_{\pm.007}$
$\mathcal{S}$-LRR	$\mathbf{.2937}_{\pm.004}$	$2.4056_{\pm.005}$	$6.2589_{\pm.019}$	$\mathbf{.6189}_{\pm.013}$	$\mathbf{.8271}_{\pm.006}$	$\mathbf{.6283}_{\pm.005}$	$\mathbf{.5519}_{\pm.008}$	$.4498_{\pm.007}$

Table 18. Experimental results of LDL on the `Flickr` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR •	$.5174_{\pm.006}$	$2.6364_{\pm.002}$	$7.2094_{\pm.011}$	$5.0366_{\pm.086}$	$.6636_{\pm.008}$	$.4683_{\pm.006}$	$.4622_{\pm.009}$	$.3811_{\pm.008}$
$\mathcal{S}$-LDSVR	$.4991_{\pm.006}$	$2.6201_{\pm.003}$	$7.1402_{\pm.012}$	$4.9196_{\pm.085}$	$.6721_{\pm.007}$	$.4814_{\pm.006}$	$.4667_{\pm.009}$	$.3850_{\pm.008}$
AA-kNN	$.3286_{\pm.005}$	$\mathbf{2.0685}_{\pm.009}$	$\mathbf{4.9363}_{\pm.033}$	$2.2172_{\pm.107}$	$.7200_{\pm.006}$	$.5582_{\pm.005}$	$.4265_{\pm.009}$	$.3465_{\pm.007}$
LDLFs	$.4051_{\pm.011}$	$2.4012_{\pm.012}$	$6.3262_{\pm.050}$	$1.4274_{\pm.077}$	$.6073_{\pm.015}$	$.4822_{\pm.011}$	$.3478_{\pm.014}$	$.2847_{\pm.012}$
DF-BFGS	$.3007_{\pm.005}$	$2.1995_{\pm.007}$	$5.4900_{\pm.025}$	$.6309_{\pm.011}$	$.7801_{\pm.005}$	$.5979_{\pm.004}$	$.5102_{\pm.009}$	$.4226_{\pm.008}$
KLD ○	$.3015_{\pm.005}$	$2.2008_{\pm.007}$	$5.4969_{\pm.025}$	$.6348_{\pm.012}$	$.7787_{\pm.005}$	$.5973_{\pm.004}$	$.5113_{\pm.009}$	$.4234_{\pm.008}$
$\mathcal{S}$-KLD	$.3052_{\pm.005}$	$2.2044_{\pm.007}$	$5.5222_{\pm.026}$	$.6485_{\pm.012}$	$.7720_{\pm.005}$	$.5926_{\pm.004}$	$.5030_{\pm.009}$	$.4166_{\pm.008}$
SCL •	$.3280_{\pm.013}$	$2.2986_{\pm.024}$	$5.9247_{\pm.099}$	$.8301_{\pm.055}$	$.7268_{\pm.018}$	$.5713_{\pm.015}$	$.4566_{\pm.024}$	$.3748_{\pm.022}$
$\mathcal{S}$-SCL	$\mathbf{.2929}_{\pm.005}$	$2.2045_{\pm.007}$	$5.5289_{\pm.028}$	$.6113_{\pm.012}$	$.7862_{\pm.005}$	$\mathbf{.6070}_{\pm.005}$	$\mathbf{.5265}_{\pm.008}$	$\mathbf{.4373}_{\pm.007}$
LRR •	$.3057_{\pm.005}$	$2.1969_{\pm.007}$	$5.4763_{\pm.025}$	$.6431_{\pm.012}$	$.7752_{\pm.006}$	$.5929_{\pm.004}$	$.5047_{\pm.009}$	$.4229_{\pm.008}$
$\mathcal{S}$-LRR	$.2938_{\pm.005}$	$2.2013_{\pm.006}$	$5.5138_{\pm.023}$	$\mathbf{.6105}_{\pm.012}$	$\mathbf{.7864}_{\pm.005}$	$.6058_{\pm.005}$	$.5261_{\pm.009}$	$.4369_{\pm.008}$

Table 19. Experimental results of LDL on the `Natural_Scene` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
LDSVR ●	.4899 $\pm .016$	2.0831 $\pm .025$	5.7724 $\pm .092$	2.0862 $\pm .085$	.5740 $\pm .017$	.4430 $\pm .015$	.4997 $\pm .015$	.3695 $\pm .012$
$\mathcal{S}$-LDSVR	.4692 $\pm .017$	1.9231 $\pm .034$	5.2307 $\pm .113$	2.2261 $\pm .121$	.5893 $\pm .018$	.4602 $\pm .018$	.5236 $\pm .017$	.3888 $\pm .013$
AA-kNN	.3113 $\pm .014$	**1.9066** $\pm .034$	**4.5413** $\pm .110$	1.0874 $\pm .082$	.7113 $\pm .015$	.5636 $\pm .013$	.4921 $\pm .021$	.3518 $\pm .016$
LDLFs	.2808 $\pm .034$	2.4329 $\pm .024$	6.6027 $\pm .108$	**.6464** $\pm .118$	.7679 $\pm .046$	.5839 $\pm .043$	.5406 $\pm .058$	.4072 $\pm .045$
DF-BFGS	.3074 $\pm .013$	2.4126 $\pm .017$	6.5896 $\pm .072$	.7603 $\pm .033$	.7381 $\pm .013$	.5568 $\pm .011$	.5110 $\pm .017$	.3837 $\pm .013$
KLD ●	.3201 $\pm .013$	2.4242 $\pm .017$	6.6560 $\pm .070$	.8285 $\pm .044$	.7172 $\pm .015$	.5485 $\pm .011$	.4958 $\pm .016$	.3715 $\pm .012$
$\mathcal{S}$-KLD	.2743 $\pm .013$	2.3866 $\pm .020$	6.4733 $\pm .077$	.6608 $\pm .039$	**.7751** $\pm .014$	.6133 $\pm .012$	.5592 $\pm .017$	.4221 $\pm .014$
SCL ●	.3379 $\pm .014$	2.4800 $\pm .018$	6.8659 $\pm .076$	.8867 $\pm .035$	.7014 $\pm .014$	.4801 $\pm .014$	.4109 $\pm .018$	.3025 $\pm .013$
$\mathcal{S}$-SCL	**.2733** $\pm .013$	2.3734 $\pm .018$	6.4376 $\pm .072$	.6703 $\pm .043$	.7744 $\pm .015$	.6156 $\pm .013$	.5573 $\pm .018$	.4207 $\pm .014$
LRR ●	.3138 $\pm .013$	2.4469 $\pm .018$	6.7118 $\pm .074$	.7703 $\pm .032$	.7363 $\pm .013$	.5456 $\pm .011$	.5056 $\pm .016$	.3782 $\pm .012$
$\mathcal{S}$-LRR	.2740 $\pm .018$	2.3461 $\pm .023$	6.3467 $\pm .087$	.6867 $\pm .070$	.7715 $\pm .021$	**.6199** $\pm .017$	**.5595** $\pm .023$	**.4228** $\pm .018$

Table 20. Experimental results of LDL on the `Movie` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
CPNN	.1337 $\pm .003$	.5639 $\pm .010$	1.0746 $\pm .020$	.1191 $\pm .005$	.9194 $\pm .003$	.8164 $\pm .004$	.6610 $\pm .013$	**.7080** $\pm .002$
AA-kNN	.1223 $\pm .002$	.5451 $\pm .009$	1.0445 $\pm .018$	.1129 $\pm .004$	.9254 $\pm .003$	.8250 $\pm .003$	.6557 $\pm .011$	.5710 $\pm .010$
LDLFs	.1172 $\pm .003$	.5233 $\pm .013$	1.0134 $\pm .026$	.1086 $\pm .006$	.9305 $\pm .003$	.8324 $\pm .004$	.6929 $\pm .013$	.6051 $\pm .012$
DF-BFGS	.1210 $\pm .002$	.5282 $\pm .009$	1.0158 $\pm .019$	.1084 $\pm .005$	.9289 $\pm .003$	.8301 $\pm .003$	.6848 $\pm .012$	.5963 $\pm .012$
LRR ●	.1135 $\pm .002$	.5101 $\pm .009$	.9770 $\pm .018$	.0957 $\pm .004$	.9369 $\pm .002$	.8385 $\pm .003$	.7119 $\pm .011$	.6203 $\pm .011$
$\mathcal{S}$-LRR	.1125 $\pm .002$	.5086 $\pm .009$	.9717 $\pm .018$	**.0945** $\pm .004$	**.9376** $\pm .002$	.8398 $\pm .003$	**.7126** $\pm .011$	.6227 $\pm .011$
QFD^2 ●	.1159 $\pm .002$	.5200 $\pm .009$	.9920 $\pm .018$	.0975 $\pm .004$	.9355 $\pm .002$	.8357 $\pm .003$	.7075 $\pm .011$	.6158 $\pm .011$
$\mathcal{S}$-QFD^2	**.1123** $\pm .002$	.5073 $\pm .009$	.9700 $\pm .018$	**.0945** $\pm .004$	**.9376** $\pm .002$	**.8401** $\pm .003$	.7125 $\pm .011$	.6224 $\pm .011$
CJS ●	.1153 $\pm .002$	.5127 $\pm .009$	.9845 $\pm .019$	.0984 $\pm .004$	.9352 $\pm .002$	.8368 $\pm .003$	.7103 $\pm .012$	.6178 $\pm .011$
$\mathcal{S}$-CJS	**.1123** $\pm .002$	**.5072** $\pm .009$	**.9699** $\pm .018$	**.0945** $\pm .004$	**.9376** $\pm .002$	**.8401** $\pm .003$	.7125 $\pm .011$	.6223 $\pm .011$

Table 21. Experimental results of LDL on the `fbp5500` dataset formatted as (`mean` ± `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
CPNN	.1864 $\pm .005$	1.3367 $\pm .009$	2.3604 $\pm .020$	.1664 $\pm .005$	.9281 $\pm .004$	.7958 $\pm .005$	.8688 $\pm .005$	.7831 $\pm .007$
AA-kNN	.1515 $\pm .004$	**1.0443** $\pm .015$	**1.7295** $\pm .031$	.1846 $\pm .016$	.9419 $\pm .004$	.8317 $\pm .005$	.8865 $\pm .006$	.8123 $\pm .008$
LDLFs	.1307 $\pm .003$	1.2787 $\pm .010$	2.1703 $\pm .024$	.1002 $\pm .005$	.9575 $\pm .003$	.8552 $\pm .004$	.9060 $\pm .005$	.8352 $\pm .007$
DF-BFGS	.1341 $\pm .003$	1.2889 $\pm .010$	2.1982 $\pm .023$	.1050 $\pm .005$	.9551 $\pm .003$	.8523 $\pm .004$	.9047 $\pm .005$	.8337 $\pm .007$
LRR	.1312 $\pm .003$	1.2767 $\pm .010$	2.1655 $\pm .024$	.1004 $\pm .004$	.9575 $\pm .002$	.8547 $\pm .003$	.9059 $\pm .004$	.8350 $\pm .006$
$\mathcal{S}$-LRR	**.1302** $\pm .003$	1.2796 $\pm .010$	2.1717 $\pm .024$	**.0997** $\pm .005$	**.9576** $\pm .002$	.8558 $\pm .003$	.9063 $\pm .004$	.8425 $\pm .006$
QFD^2 ●	.1380 $\pm .003$	1.2803 $\pm .010$	2.1858 $\pm .024$	.1084 $\pm .005$	.9535 $\pm .003$	.8476 $\pm .004$	.9021 $\pm .004$	.8297 $\pm .006$
$\mathcal{S}$-QFD^2	.1321 $\pm .004$	1.2811 $\pm .010$	2.1779 $\pm .024$	.1027 $\pm .006$	.9561 $\pm .003$	.8537 $\pm .004$	.9044 $\pm .005$	.8330 $\pm .007$
CJS ●	.1343 $\pm .003$	1.3057 $\pm .010$	2.2374 $\pm .024$	.1084 $\pm .005$	.9544 $\pm .003$	.8527 $\pm .004$	.9044 $\pm .004$	.8334 $\pm .006$
$\mathcal{S}$-CJS	**.1302** $\pm .003$	1.2802 $\pm .010$	2.1731 $\pm .024$	**.0997** $\pm .005$	.9575 $\pm .002$	**.8559** $\pm .003$	**.9066** $\pm .004$	**.8429** $\pm .007$

Table 22. Experimental results of IncomLDL on the `JAFFE` dataset formatted as (`mean` ± `std`)

Algorithms	$\omega = 20\%$							
	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
IncomLDL ●	.0898 $\pm .010$	.3304 $\pm .024$	.6742 $\pm .049$	**.0425** $\pm .007$	**.9598** $\pm .007$	.8861 $\pm .009$	.4742 $\pm .094$	.4017 $\pm .082$
$\mathcal{S}$-IncomLDL	**.0863** $\pm .013$	**.3179** $\pm .036$	**.6525** $\pm .077$	.0433 $\pm .012$	.9590 $\pm .011$	**.8893** $\pm .014$	**.5034** $\pm .114$	**.4401** $\pm .100$
Algorithms	$\omega = 40\%$							
	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
IncomLDL ●	.0946 $\pm .010$	.3454 $\pm .026$	.7073 $\pm .053$	.0465 $\pm .007$	.9558 $\pm .007$	.8801 $\pm .010$	.4231 $\pm .086$	.3534 $\pm .074$
$\mathcal{S}$-IncomLDL	**.0868** $\pm .013$	**.3211** $\pm .038$	**.6568** $\pm .081$	**.0439** $\pm .014$	**.9585** $\pm .012$	**.8886** $\pm .015$	**.5075** $\pm .115$	**.4434** $\pm .098$

Table 23. Experimental results of IncomLDL on the `SBU_3DFE` dataset formatted as (`mean` $\pm$ `std`)

Algorithms	$\omega = 20\%$							
	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
IncomLDL ●	.1088 $\pm$.003	.3586 $\pm$.008	.7586 $\pm$.017	.0574 $\pm$.003	.9439 $\pm$.003	.8655 $\pm$.003	.3171 $\pm$.027	.2746 $\pm$.023
$\mathcal{S}$-IncomLDL	**.1014** $\pm$.004	**.3208** $\pm$.009	**.6727** $\pm$.019	**.0516** $\pm$.003	**.9485** $\pm$.003	**.8790** $\pm$.004	**.4255** $\pm$.026	**.3679** $\pm$.023
Algorithms	$\omega = 40\%$							
	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑	`Spear.` ↑	`Ken.` ↑
IncomLDL ●	.1104 $\pm$.003	.3621 $\pm$.008	.7673 $\pm$.017	.0586 $\pm$.003	.9426 $\pm$.003	.8638 $\pm$.003	.3003 $\pm$.024	.2595 $\pm$.020
$\mathcal{S}$-IncomLDL	**.1016** $\pm$.004	**.3213** $\pm$.009	**.6748** $\pm$.019	**.0516** $\pm$.003	**.9485** $\pm$.003	**.8786** $\pm$.004	**.4232** $\pm$.025	**.3659** $\pm$.022

Table 24. Experimental results of LDL4C on `JAFFE` and `Twitter` formatted as (`mean` $\pm$ `std`)

Algorithms	`JAFFE`		Algorithms	`Twitter`	
	`0/1 loss` ↓	`Err. prob.` ↓		`0/1 loss` ↓	`Err. prob.` ↓
LDL4C ●	.4973 $\pm$.108	.7665 $\pm$.020	LDL4C	.9081 $\pm$.009	.8846 $\pm$.005
$\mathcal{S}$-LDL4C	**.4453** $\pm$.102	**.7600** $\pm$.019	$\mathcal{S}$-LDL4C	.8714 $\pm$.207	.8729 $\pm$.156
LDL-HR	.4786 $\pm$.097	.7676 $\pm$.020	LDL-HR ●	.3656 $\pm$.017	.4928 $\pm$.011
$\mathcal{S}$-HR	.4653 $\pm$.105	.7655 $\pm$.019	$\mathcal{S}$-HR	**.2753** $\pm$.013	**.4250** $\pm$.008
LDLM	.4787 $\pm$.109	.7687 $\pm$.021	LDLM ●	.2814 $\pm$.013	.4291 $\pm$.008
$\mathcal{S}$-LDLM	.4737 $\pm$.097	.7689 $\pm$.019	$\mathcal{S}$-LDLM	**.2753** $\pm$.014	**.4250** $\pm$.008

Table 25. Experimental results of LDL4C on `sBU_3DFE` and `Flickr` formatted as (`mean` $\pm$ `std`)

Algorithms	`sBU_3DFE`		Algorithms	`Flickr`	
	`0/1 loss` ↓	`Err. prob.` ↓		`0/1 loss` ↓	`Err. prob.` ↓
LDL4C	.5578 $\pm$.028	.7671 $\pm$.007	LDL4C	.8971 $\pm$.008	.8884 $\pm$.004
$\mathcal{S}$-LDL4C	.5526 $\pm$.025	.7686 $\pm$.006	$\mathcal{S}$-LDL4C	.8705 $\pm$.138	.8702 $\pm$.100
LDL-HR ●	.5167 $\pm$.027	.7596 $\pm$.006	LDL-HR ●	.4513 $\pm$.015	.5823 $\pm$.007
$\mathcal{S}$-HR	.5069 $\pm$.025	.7598 $\pm$.006	$\mathcal{S}$-HR	**.4219** $\pm$.015	**.5639** $\pm$.007
LDLM ●	.5258 $\pm$.034	.7619 $\pm$.009	LDLM ●	.4384 $\pm$.014	.5740 $\pm$.007
$\mathcal{S}$-LDLM	**.4809** $\pm$.024	**.7524** $\pm$.005	$\mathcal{S}$-LDLM	.4321 $\pm$.016	.5667 $\pm$.007

Table 26. Experimental results of LE on the `JAFFE` dataset formatted as (`mean` $\pm$ `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑
LP	.0812 $\pm$.001	.3446 $\pm$.002	.7125 $\pm$.005	.0424 $\pm$.001	.9618 $\pm$.001	.8808 $\pm$.001
GLLE	.0821 $\pm$.002	.3196 $\pm$.013	.6518 $\pm$.028	.0386 $\pm$.003	.9638 $\pm$.002	.8901 $\pm$.004
LEVI	.0787 $\pm$.003	.3316 $\pm$.013	.6864 $\pm$.028	.0391 $\pm$.003	.9649 $\pm$.002	.8860 $\pm$.004
LIBLE ●	.0813 $\pm$.006	.3106 $\pm$.020	.6358 $\pm$.044	.0370 $\pm$.005	.9652 $\pm$.005	.8929 $\pm$.008
$\mathcal{S}$-LIBLE	**.0770** $\pm$.003	**.2942** $\pm$.007	**.5997** $\pm$.016	**.0332** $\pm$.002	**.9685** $\pm$.002	**.8987** $\pm$.003

Table 27. Experimental results of LE on the `Yeast_heat` dataset formatted as (`mean` $\pm$ `std`)

Algorithms	`Cheby.` ↓	`Clark` ↓	`Can.` ↓	`KLD` ↓	`Cosine` ↑	`Int.` ↑
LP	**.0421** $\pm$.000	.2148 $\pm$.000	.4711 $\pm$.001	.0153 $\pm$.000	.9860 $\pm$.000	.9235 $\pm$.000
GLLE	.0481 $\pm$.001	.2114 $\pm$.005	.4282 $\pm$.011	.0168 $\pm$.001	.9842 $\pm$.001	.9298 $\pm$.002
LEVI	.0494 $\pm$.007	.2125 $\pm$.027	.4307 $\pm$.056	.0169 $\pm$.004	.9838 $\pm$.004	.9289 $\pm$.009
LIBLE ●	.0453 $\pm$.000	.1973 $\pm$.001	.3982 $\pm$.003	.0148 $\pm$.000	.9859 $\pm$.000	.9346 $\pm$.000
$\mathcal{S}$-LIBLE	.0445 $\pm$.000	**.1901** $\pm$.002	**.3790** $\pm$.005	**.0137** $\pm$.000	**.9869** $\pm$.000	**.9376** $\pm$.001