Abstract—Deep learning on large-scale data is currently dominant nowadays. The unprecedented scale of data has been arguably one of the most important driving forces behind its success. However, there still exist scenarios where collecting data or labels could be extremely expensive, e.g., medical imaging and robotics. To fill up this gap, this paper considers the problem of data-efficient learning from scratch using a small amount of representative data. First, we characterize this problem by active learning on homeomorphic tubes of spherical manifolds. This naturally generates feasible hypothesis class. With homologous topological properties, we identify an important connection—finding tube manifolds is equivalent to minimizing hyperspherical energy (MHE) in physical geometry. Inspired by this connection, we propose a MHE-based active learning (MHEAL) algorithm, and provide comprehensive theoretical guarantees for MHEAL, covering convergence and generalization analysis. Finally, we demonstrate the empirical performance of MHEAL in a wide range of applications for data-efficient learning, including deep clustering, distribution matching, version space sampling, and deep active learning.

Index Terms—Deep learning, representative data, active learning, homeomorphic tubes, hyperspherical energy, convergence analysis.

I. INTRODUCTION

RECENT years have witnessed the success of deep learning [1] in a wide range of applications in computer vision [2], natural language processing [3], and speech processing [4]. The unprecedented scale of available datasets [5], [6] has been the driving force behind the progress made by deep learning. By leveraging vast amounts of data and annotations, powerful artificial intelligence (AI) systems have been developed, such as AlphaFold [7] and AlphaGo [8]. Despite their superior performance, the process of training and annotating large-scale data can be prohibitively expensive. In contrast, humans can easily distinguish between a dog and a wolf by simply glancing at a single picture. This has motivated researchers to explore scenarios where only a small amount of data is available for AI systems.

There are several related techniques that study how to learn efficiently from a small amount of data or annotations, such as few-shot learning [9], active learning [10], and unsupervised representation learning [11]. Besides, data augmentation [12] and generative models [13] can also be used to help data-efficient learning. This leads to a broad interest for our study. To narrow down our scope, we focus on the generalization performance of a small amount of data distributed over the version space. Given a finite hypothesis class, data distributed around the decision boundary of version space usually generate effective hypotheses, and also produce significant updates for the model. We thus adopt a novel perspective for data-efficient learning—characterizing data representation via the topology of decision boundary.

Active learning (AL), which can be initialized with any hypothesis (even a null hypothesis), is a promising tool to learn the topology of decision boundary via iteratively shrinking the version space (see Fig. 1). Specifically, the highly informative data which derive the hypothesis updates are distributed in version space regions (called in-version-space), which maintain homeomorphic manifolds [14] over the topological properties of decision boundaries. Hereafter, an AL algorithm that uses in-version-space sampling is termed version space-based AL (VSAL) [15]. Usually, a typical sampling criterion of VSAL

![Fig. 1. Active learning shrinks the version space \( \mathcal{V} \) into its optimal expression \( \mathcal{V}^* \) which tightly covers the decision boundaries of the two spherical classes, deriving effective pruning for candidate classification hypotheses. Geometrically, there are a small amount of data distributed around the decision boundaries, which characterize their topological structures.](image-url)
Solution: We propose the MHEAL algorithm with theoretical guarantees on generalization error and label complexity. Our major contributions are summarized as follows:

1. We study data-efficient learning as an important step toward training AI systems with a small amount of data. In a nutshell, we propose to characterize the decision boundaries of version space by minimizing the hyperspherical energy of a small amount of representative data.
2. We derive a lower bound maximization scheme to approximate the greedy sequential solution of $\ell_0$ expression of MHE, yielding a much lower computation complexity.
3. We give theoretical analyses for our approximation optimization scheme which guarantees both geometric preservation and approximation accuracy.
4. We propose a novel active learning algorithm called MHEAL, which performs MHE over each pre-estimated spherical cluster in an unsupervised manner and effectively characterizes their decision boundaries.
5. We prove the theoretical convergence of MHEAL by deriving its generalization error and label complexity bounds. Moreover, we conduct an extensive empirical study on the generalization performance to demonstrate the effectiveness of our bounds.

II. Related Work

VSAL presents a theoretical interpretation for learning from representative data via AL, and MHE implements an advanced VSAL algorithm from the perspective of physical geometry. We briefly introduce related work from these two aspects.

VSAL: The VSAL algorithm [15, 17] prunes the hypothesis class by annotating the unlabeled data via the hypothesis disagreement maximization, which requires the optimal hypothesis always to be included. Specifically, the hypothesis-pruning [20] always needs to maintain a correct update, where error disagreement [21] is an important indicator to control these updates. With the estimation on errors, the error disagreement indicator can be specified as the best-in-class error [22], average-in-class error [23], entropy of error [24], etc. A series of confidence label complexity bounds were derived under noise-free settings, such as the agnostic PAC bound [25, 26, 27, 28]. With a desired error threshold, the annotating budget is also bounded. By employing importance weighting, importance weighted AL (IWAL) [29] utilizes the on-line sampling theory to tighten the label complexity bounds, before converging into the optimal hypothesis. To further improve it, [32] gives more refined analysis on the current hypothesis class to reduce the best-in-class error, resulting in a tighter bound on label complexity [16].

Guarantees under different noise settings are also studied, e.g., adversarial noise [30], malicious noise [31], random classification noise [32], and bounded noise [33].

Few-shot learning [9] also studies how to adapt to a new task with a very small amount of data, but it typically requires extra information about the base tasks. Moreover, few-shot learning mostly considers the passive setting where the label distribution is explicitly controlled by one specific sampling scenario in the pre-defined training set. In contrast, AL that learns from scratch...
using representative data is not as limited as few-shot learning, since its algorithms can stop their iterative sampling either when they achieve desired accuracy or when the annotation budgets are exhausted.

**MHE:** Thomson problem [34] in physics describes the ground state energy and configuration of a set of electrons on a unit hypersphere. Given $N$ electrons, Liu et al. [18] seek to achieve their minimum potential energy by considering their interactions on a unit hypersphere. This also can be viewed as maximizing hyperspherical uniformity [35]. Recent studies [36], [37], [38], [39] have shown that hyperspherical similarity preserves the most abundant and discriminant information. Modeling with DNNs, geodesic distances [40] between neurons that discriminate the features, are projected and optimized on the hypersphere. However, naively minimizing hyperspherical energy from geometry suffers from some difficulties due to the underlying non-linearity and non-convexity. To alleviate these difficulties, Lin et al. [41] propose the compressive MHE (CoMHE) as a more effective regularization to minimize hyperspherical energy for neural networks. Following [18], [41], Perez-Lapillo et al. [42] and Shah et al. [43] improve voice separation by applying MHE to Wave-U-Net and time-frequency domain networks, respectively. MHE has wide applications in image recognition [39], [44], [45], face recognition [18], [36], [46], speaker verification [47], adversarial robustness [48], few-shot learning [49], [50], etc.

### III. MHE and Decision Boundary

Section III-A characterizes the homologous topological properties of decision boundaries for tube manifolds. Section III-B presents the MHE problem of physical geometry. Section III-C then proposes a sequential solution for $\ell_0$ expression of MHE. To find an alternative scheme for such a greedy solution, Section III-D maximizes the lower bound of the optimal $\ell_0$ expression of MHE as a feasible approximation.

#### A. Decision Boundary

We start by characterizing the decision boundaries. Given a feature space $\mathcal{X}$ associating with a label space $\mathcal{Y}$, let $\mathcal{Y} = \{1, 2, \ldots, K\}$, we define $P_{\mathcal{X}Y}$ as the joint distribution over the mapping from $\mathcal{X}$ to $\mathcal{Y}$. According to the Bayesian theorem, decision boundary of a class (cluster) can be constructed on a tube manifold $\mathcal{M}$ [14]. We next give a conceptual description for decision boundaries in the multi-class setting.

**Definition 1 (Decision Boundary):** Given a class i.i.d. drawn from $P_{\mathcal{X}Y}$ with label $k$ embedded in a label space $\mathcal{Y} = \{1, 2, 3, \ldots, K\}$, $k' \in \mathcal{Y}$ and $k' \neq k$, the decision boundary of this class is distributed over a tube manifold $\mathcal{M}$ with uncertain label predictions, which satisfy $\mathcal{M} := \{x \in \mathcal{X} | P_{y|\mathcal{X}}(k'|x) = P_{y|\mathcal{X}}(k|x)\}$, where the optimal Bayesian classifier is $f(x) = k$ if $P_{y|\mathcal{X}}(k|x) \geq 0.5$, otherwise $f(x) = k' \in \{1, 2, \ldots, k-1, k+1, \ldots, K\}$.

**Remark 1:** Definition 1 characterizes the decision boundary of a tube manifold over a label space $\mathcal{Y}$, where those samples around the tube are with uncertain class labels. For any high-dimensional sphere, the tube manifold still covers those samples with uncertain labels distributed near the boundary of the class.

If $\mathcal{Y}$ is a binary label space, that is, $\mathcal{Y} = \{+1, -1\}$, we then adapt Definition 1 into the following specific case.

**Example 1 (Binary Case of Decision Boundary):** Given a class embedded in a label space $\mathcal{Y} = \{+1, -1\}$, its decision boundary is distributed over a tube manifold $\mathcal{M}$ with uncertain label predictions, which satisfy $\mathcal{M} := \{x \in \mathcal{X} | P_{y|\mathcal{X}}(+1|x) = P_{y|\mathcal{X}}(-1|x)\}$, where the optimal Bayesian classifier is $f(x) = +1$ if $P_{y|\mathcal{X}}(+1|x) \geq 0.5$, otherwise $f(x) = -1$.

From a geometric perspective [19], the manifold of decision boundary can be characterized by a tube with a width $\gamma$, which keeps homologous properties with a ball $B(c, r)$, where $c$ denotes the center and $r$ denotes the radius (see Fig. 4). We give the following assumption for the homeomorphic tube manifold, where $B(c, r - \gamma)$ denotes the concentric ball of $B(c, r)$. Specifically, $r - \gamma$ is the radius, $c$ is the center, and $\gamma$ is a variable to denote the width of the tube manifold.

**Assumption 1:** Assume that the label complexity of a version space-based hypothesis is characterized by a set $S$ over the tube manifold $\mathcal{M} := \{B(c, r) \setminus B(c, r - \gamma)\}$, if $S \subseteq D^1 = \{x \in \mathcal{X} : f(x) = +1\}$, we have: 1) $\bigcap_{(x_i, \gamma)} M(x_i, \gamma) \neq \emptyset$, where $M(x_i, \gamma)$ denotes the tube manifold covering $x_i$ with...
a width $\gamma$; 2) let $D^{-1} = \{ x \in \mathcal{X} : f(x) = -1 \}$, if $x_i \subseteq D^{-1}$ and $x_j \subseteq D^{-1}$, there exists $\|x_i - x_j\| \leq \gamma$, where $\| \cdot \|$ denotes the $\ell_2$-norm.

In Assumption 1, the first condition requires that there is a homeomorphic tube of $M$ with a width $\gamma$ which covers $x_i \in S$ to contain shared elements from both classes. This means that $M$ is a tight covering for $S$. The second condition describes the width of $M$, i.e., $\gamma$. With these settings, we have defined decision boundaries to characterize those feasible hypothesis class as a consistent concept, and then generalize it with MHE.

### B. MHE

MHE [18] seeks to find the configuration of $N$ mutually-repelling electrons that minimizes the potential energy. These electrons are distributed on a unit hypersphere. In physics, such a potential energy is used to describe a balanced state of the distribution of electrons. More generally, MHE corresponds to the uniform distribution on the hypersphere [35]. From a geometric perspective, it can be used to characterize the decision boundary distributed in a homeomorphic tube manifold of a hypersphere.

Given $N$ samples, i.e., $W = \{ w_1, w_2, \ldots, w_N \} \subseteq \mathbb{R}^d$, let $\hat{w}_i$ denote the $\ell_2$-norm projection of $w_i$ on the unit hypersphere. Then their hyperspherical energy can be defined as

$$
\mathcal{E}_{s,d}(\hat{w}_i) = \sum_{i=1}^{N} \sum_{j=1}^{N} f_s(\|\hat{w}_i - \hat{w}_j\|) = \left\{ \begin{array}{ll} 
\sum_{i>j} \|\hat{w}_i - \hat{w}_j\|^{-s}, & s > 0 \\
\sum_{i>j} \log(\|\hat{w}_i - \hat{w}_j\|^{-1}), & s = 0 
\end{array} \right.,
$$

where $f_s$ denotes an energy function, and $i, j \geq 1$. Specifically, $\mathcal{E}_{s,d}$ denotes the logarithmic potential energy, that is, the $\ell_0$ expression of MHE (also termed as $\ell_0$ MHE), i.e., $\mathcal{E}_{0,d} = \sum_{i>j} \log(\|\hat{w}_i - \hat{w}_j\|^{-1})$. Then, minimizing the $\ell_0$ hyperspherical energy can be viewed as an approximation to minimizing $\mathcal{E}_{s,d}$ [18]. Specifically, $\ell_0$ MHE is formally defined as

$$
\arg\min_{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N} \mathcal{E}_{0,d} = \sum_{i>j} \log(\|\hat{w}_i - \hat{w}_j\|^{-1}) = \arg\max_{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N} \prod_{i>j} \|\hat{w}_i - \hat{w}_j\|. 
$$

Equation (2) is essentially a limiting case of the MHE objective $\mathcal{E}_{s,d}$ [18]. Besides this, $\mathcal{E}_{1,d}$ and $\mathcal{E}_{2,d}$ are also feasible alternatives, namely $\ell_1$ and $\ell_2$ expressions of MHE, respectively (also term $\ell_1$ MHE and $\ell_2$ MHE). However, minimizing $\mathcal{E}_{1,d}$ is a typical NP hard problem, which usually requires a sequential search. $\mathcal{E}_{2,d}$ can be solved by gradient descent with the following gradient [41]:

$$
\nabla \hat{w}_i, \mathcal{E}_{2,d} = -\frac{2(\hat{w}_i - \hat{w}_j)}{\|\hat{w}_i - \hat{w}_j\|^4},
$$

whose solution yields $\hat{w}_i = \frac{\sum_{j=1}^{N} \|\hat{w}_i - \hat{w}_j\|^4}{\sum_{j=1}^{N} \|\hat{w}_i - \hat{w}_j\|^4}$, s.t. $j \neq i$.

### C. Sequential Optimization of MHE

We here present the optimal solution of $\ell_0$ MHE by employing a greedy strategy. Given $P$ containing $M$ feasible data points in $\mathbb{R}^d$, MHE is performed to find $N$ data points for $W$. Let $\hat{w}_1 \in P$ and $W$ be initialized by $W = \{ \hat{w}_1 \}$. At the $t$-th step, a sequential optimization is adopted to find $\hat{w}_t$ from $P$

$$
\hat{w}_t = \arg\max_{\hat{w}_t \in P} \prod_{t-1 \geq i \geq 1} \|\hat{w}_i - \hat{p}\|,
$$

after which $W$ is updated by adding a new data point, i.e., $W \leftarrow W \cup \{ \hat{w}_t \}$ at the $t$-th step. In this setting, computing one sample costs $O(MN)$ complexity, where the argmax operation costs $O(M)$, and the computation of $\prod_{t-1 \geq i \geq 1}$ costs $O(N)$. To obtain $N$ feasible samples, (4) is computed until $t$ arrives at $N$, thereby costing $O(MN^2)$.

Let $\mathcal{E}_{\ell_0}(\hat{w}_1)$ denote the $\ell_0$ hyperspherical energy of $N$ data points with an initial sample $\hat{w}_1$, the optimal initialization on $\hat{w}_1$, i.e., $\hat{w}_1^*$, is obtained by

$$
\hat{w}_1^* = \arg\max_{\hat{w}_1 \in P} \mathcal{E}_{\ell_0}(\hat{w}_1).
$$

For $M$ times of the sequential optimization in (5), it will cost $O(M^2 N^2)$ to obtain the optimal $\hat{w}_1$. A detailed process is then presented in Algorithm 1.
D. Approximation by Lower Bound Maximization

The sequential optimization comes with a computation complexity of \(O(M^2 N^2)\), which is not efficient to solve. We thus aim to find a feasible approximation.

**Proposition 1:** Assume that \(\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{N-1}\) are known samples for \(W\), the following step is to optimize \(\hat{w}_N\), that is, \(\hat{w}_N = \arg\max_{\hat{w}_N} P_{\hat{w}_N} \mathcal{L}_{\hat{w}_N}(\hat{p})\), s.t. \(W = \{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{N-1}\}\), where this argmax operation invokes (4) one time since \(W\) has already included \(N - 1\) samples. With such setting, assume that \(\min_{\hat{w}_j \in W} \|\hat{w}_N - \hat{w}_j\| \leq \min_{\hat{w}_j \in W} \|\hat{w}_N - \hat{w}_j\|\), recalling the energy definition of (2), there exists an inequality

\[
\left( \min_{\hat{w}_j \in W} \|\hat{w}_N - \hat{w}_j\| \right)^{\frac{N^2-N}{2}} \leq \left( \prod_{i=1}^{N-1} \|\hat{w}_i - \hat{w}_j\| \right)
\]

where \(\min_{\hat{w}_j \in W} \|\hat{w}_N - \hat{w}_j\|\) returns the minimal geodesic distance of \(\hat{w}_N\) to the data point of \(W\).

To tightly approximate the optimal energy of \(\mathcal{L}_{\hat{w}_N}(\hat{w}_N)\), an effective way is to maximize its lower bound \(\min_{\hat{w}_j \in W} \|\hat{w}_N - \hat{w}_j\|\). We thus present Corollary 1 to state our alternative scheme.

**Corollary 1 (Lower Bound Maximization):** To obtain an approximated optimal \(\hat{w}_N\), one feasible method is to maximize the lower bound of \(\mathcal{L}_{\hat{w}_N}(\hat{w}_N)\), that is,

\[
\max_{\hat{w}_N} \left( \min_{\hat{w}_j \in W} \|\hat{p} - \hat{w}_j\| \right)^{\frac{N^2-N}{2}}.
\]

Equivalently, this max-min solution is simplified as

\[
\max_{\hat{w}_N} \min_{\hat{w}_j \in W} \|\hat{p} - \hat{w}_j\|. \tag{6}
\]

**Sequential max-min:** With Corollary 1, at \(t\)-time, acquiring \(\hat{w}_t\) is sequentially obtained from

\[
\arg\max_{\hat{w}_N} \min_{\hat{w}_j \in W} \|\hat{p} - \hat{w}_j\|. \tag{8}
\]

where \(W\) is updated by \(\hat{w}_{t-1}\). In this max-min optimization, the argmax needs to perform \(M\) times of min operation. On such setting, acquiring a sample \(\hat{w}_t\) at \(t\)-time costs about \(O(MN)\).

**Computation complexity:** To obtain \(N\) feasible data from \(P\), (8) is iteratively invoked \(N\) times, which costs about \(O(MN^2)\). To find the optimal \(\hat{w}_1\), (5) needs to be performed \(M\) times. Therefore, the total computation complexity of our lower bound approximation scheme costs about \(O(M^2 N^2)\).

IV. THEORETICAL INSIGHTS

In physics, MHE projects the vectors on a unit sphere to maximize their potential energies. Ata Kaban et al. [51] proposed Gaussian random projection to redefine it from a vector view. Following their works, we present theoretical insights for our proposed optimization scheme.

A. Geometric Projection

We first review the Gaussian random projection that maps vectors from \(\mathbb{R}^d\) to \(\mathbb{R}^k\), where \(k < d\).

**Lemma 1:** Given \(\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N \in \mathbb{R}^d, P \in \mathbb{R}^{k \times d}\) denotes a Gaussian random projection matrix \([51]\) where \(P_{ij} = \frac{1}{\sqrt{d}} r_{ij}\), \(r_{ij}\) is i.i.d. drawn from \(N(0, \sigma^2)\), and \(P\hat{w}_1, P\hat{w}_2 \in \mathbb{R}^k\) are the random projection of \(\hat{w}_1, \hat{w}_2\) under \(P\). Then, for any variable \(\varepsilon \in [0, 1]\), there exists

\[
\Pr\left\{\left(1 - \varepsilon\right)\|\hat{w}_i - \hat{w}_j\|^2 \geq \|P\hat{w}_i - P\hat{w}_j\|^2 \right\} \\
< 1 - \exp\left(-\frac{k\varepsilon^2}{8}\right),
\]

which satisfies \(k\varepsilon^2 > 5.54517744\).

Note that the inequality of \(k\varepsilon^2\) is limited by \(2\exp(\frac{\varepsilon^2}{8}) \leq 1\) and the proof of Theorem 1.1 of [51] can be adopted here. With Lemma 1, \(\arg\max_{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N} \prod_{j > i} \|\hat{w}_i - \hat{w}_j\|\) approximated by \(\arg\max_{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N} \prod_{j > i} \|\hat{w}_i - \hat{w}_j\|\). Liu et al. [18] used this idea to regularize the network generalization. In our scheme, to project \(\{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N\}\) onto a sphere is a necessary process. However, most of the input distributions are aspherical [52], and we usually project the data into a spherical-likeness geometry. Typical data normalization methods are adopted to relieve this problem. For example, we project \(\arg\max_{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N}\) with a Gaussian projection to obtain a \(0\)-mean and \(\sigma^2\)-variance distribution. With such idea, we present our projection proposition.

**Proposition 2:** Given \(\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N \in \mathbb{R}^d, P \in \mathbb{R}^{k \times d}\) denotes a Gaussian random projection matrix where \(P_{ij} = \frac{1}{\sqrt{d}} r_{ij}\), \(r_{ij}\) is i.i.d. drawn from \(N(0, \sigma^2)\), and \(P\hat{w}_1, P\hat{w}_2 \in \mathbb{R}^k\) are the random projection of \(\hat{w}_1, \hat{w}_2\) under \(P\). To satisfy the condition of \(k < d\) in Lemma 1, we extend a vector with zero-element to the \((d + 1)\) dimension of \(\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N \in \mathbb{R}^d\), and obtain \(\hat{w}_1', \hat{w}_2', \ldots, \hat{w}_N' \in \mathbb{R}^{d+1}\), which require \(\hat{w}_{id} = 0\) for any \(1 \leq i \leq N\), where \(d' = d + 1\). Considering that \(P\) still is a \(d\)-dimensional projection matrix, for any variable \(\varepsilon \in [0, 1]\), the inequality of Lemma 1 still holds.

In short, Lemma 1 still holds if the Gaussian projection limits \(k = d\). Then, the projection is a more specific normalization guaranteed by Lemma 1.

B. Geometric Preservation

With Proposition 2, under the limitation of \(k = d\), we observe the geometric preservation for our projection idea.

**Lemma 2 (Preservation of Geodesic Inequality):** Given \(\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N \in \mathbb{R}^d, P \in \mathbb{R}^{k \times d}\) denotes a Gaussian random projection matrix \(P_{ij} = \frac{1}{\sqrt{d}} r_{ij}\), \(r_{ij}\) is i.i.d. drawn from \(N(0, \sigma^2)\), and \(P\hat{w}_1, P\hat{w}_2 \in \mathbb{R}^k\) are the random projections of \(\hat{w}_1, \hat{w}_2\) under \(P\), respectively. Let \(d_M\) be the geodesic metric over a curve \(g\) of \(\mathbb{R}^d\) which satisfies \(d_M(g(w_1), g(w_2)) := \nu\|w_1 - w_2\|\), where \(\nu \leq 0\), \(d_M\) be the geodesic metric over a curve \(g'\) of \(\mathbb{R}^k\) which satisfies \(d_M'(g(w_1), g(w_2)) := \nu'\|w_1 - w_2\|\), where \(\nu' \leq 0\), if \(d_M(\hat{w}_1, \hat{w}_2) \leq d_M(\hat{w}_2, \hat{w}_3)\) and \(\nu = \nu'\), the
inequality still holds for $d'_{M}$, i.e., there exists $d'_{M}(P\hat{w}_1, P\hat{w}_2) \leq d'_{M}(P\tilde{w}_1, P\tilde{w}_3)$.

**Lemma 3 (Preservation of Angle Inequality):** Given $\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_N \in \mathbb{R}^d$, $P \in \mathbb{R}^{n \times d}$ denotes a Gaussian random projection matrix where $P_{ij} = \frac{1}{\sqrt{n}}r_{ij}$, $r_{ij}$ is i.i.d. drawn from $N(0, \sigma^2)$, and $P\hat{w}_1, P\hat{w}_2 \in \mathbb{R}^n$ are the random projections of $\hat{w}_1, \hat{w}_2$ under $P$. Let $\theta_M(\cdot, \cdot)$ be the angle metric over $\mathbb{R}^d$, if $\theta_M(\hat{w}_1, \hat{w}_2) \leq \theta_M(\hat{w}_2, \hat{w}_3)$, then exists $\theta_M(P\hat{w}_1, P\hat{w}_2) \leq \theta_M(P\hat{w}_2, P\hat{w}_3)$, otherwise, given $\theta_M(\hat{w}_1, \hat{w}_2) = \frac{1}{\varepsilon}||P\hat{w}_1||^2(||P\hat{w}_2||^2)^{1/2}$, following [41], there then exists

$$\theta_M(P\hat{w}_1, P\hat{w}_2) - e \varepsilon < \theta_M(P\tilde{w}_2, P\tilde{w}_3),$$

where $\varepsilon \in [0, 1]$, $e > 0$, and $\kappa = d$.

C. Convergence of Approximation

We present the bound of the approximation loss of Proposition 1, and then extend it into a more general bound of acquiring $N$ samples.

**Proposition 3 (Controlled Approximation Loss):** Let $\ell_{LBM}^{\text{upper}}(\hat{w}_1; N)$ be the upper bound of the approximation loss of acquiring $N$ samples, given $\ell_{LBM}^{\text{upper}}(\hat{w}_1; N) = \frac{N^2}{2\sqrt{\ell_{LBM}(\hat{w}_1; N)}}$, there exists

$$\min_{p \in P} \left(\max_{\hat{w}_j \in W} ||\hat{p} - \hat{w}_j|| - \min_{\hat{w}_j \in W} ||\hat{p} - \hat{w}_j||\right) \leq \ell_{LBM}^{\text{upper}}(\hat{w}_1; N)$$

Specifically, for any approximation, there exists

$$O\left(\min_{\hat{w}_1, \hat{w}_j \in W} ||\hat{w}_1 - \hat{w}_j||\right) \leq \ell_{LBM}^{\text{upper}}(\hat{w}_1; N)$$

that is, any upper bound of the MHE approximation to the optimal solution is proportional to the maximal length of the sphere chord, and any of its lower bound is proportional to the minimal length of the sphere chord.

Proposition 3 presents a natural derivation process for the approximation loss of Proposition 1. We can see that our proposed lower bound maximization can properly converge.

V. MHEAL: MHE-BASED ACTIVE LEARNING

Based on the theoretical results of Sections III and IV, MHEAL adopts a max-min solution of (8) in a set of hyperspheres as shown in Fig. 3. This section thus interprets this process beginning from the hyperspherical clustering.

A. Hyperspherical Energies

**MHE with Clustering:** Our goal is to characterize the decision boundaries over each cluster. Therefore, it is necessary to perform clustering in input distribution. Moreover, adopting clustering to reduce $N, M$ into $N/k, M/k$, respectively, can effectively reduce the calculation complexity of sequential solution of $\ell_0$ MHE. In this way, the calculation complexity of our optimization scheme with $k$ clusters reduces to $O(\frac{M^2N^2}{k^3})$, which is lower than previous $O(M^2N^2)$.

**Hyperspherical Clustering:** To characterize the hyperspherical energies, we use hyperspherical $k$-means (termed SphericalKmeans) to obtain $k$ hyperspheres. Following [53], the standard hyperspherical $k$-means is to optimize a set of sphere centers $C = \{c_1, c_2, \ldots, c_k\}$:

$$\arg\min_{c_k} \sum_{x_i} (1 - \cos(c_k, x_i)), \quad (9)$$

where $c_k = \sum_{x_i} \frac{1_{x_i = c_k}}{\sqrt{x_i}}$, and $1$ denotes the indication function. With (9), minimizing hyperspherical energies over each hypersphere is transferred into learning tube energies in decision boundaries around those hyperspheres. Note that applying SphericalKmeans needs to normalize the input features with Gaussian projection following the theoretical results of Section IV-C, where we specify $\mu = 0$ and $\sigma = 1$ for the Gaussian projection.

**MHE over Each Pre-estimated Cluster:** Let $\{B_1, B_2, \ldots, B_k\}$ be the clustered $k$ hyperspheres where $B_i$ is with a center $c_i$ and radius $r_i$, $\forall i, 1, 2, 3, \ldots, k$. We employ (8) to perform MHE over each pre-estimated cluster. The selection of $\hat{w}_1$ in $B_i$ follows the sequential optimization of (5), that is,

$$\arg\max_{\hat{w}_1 \in B_i} L_{\hat{w}_1}(\hat{w}_1). \quad (10)$$

Assume that $B_i$ has $N_i$ data, $\hat{w}_1 \in B_i$ thus needs to perform $N_i$ times of argmax operation, that is, (10) needs to initialize $\hat{w}_1$ for $N_i$ times. To reduce the calculation complexity, we degenerate the sequential selection into vector rotation maximization, which maximizes the vector rotation of $\hat{w}_1^{-1}$ and its next initialization $\hat{w}_1$, that is,

$$\hat{w}_1 = \arg\max_{x \in B_i} \left(1 - \cos(\gamma_u^{\hat{w}_1^{-1}}, \gamma_u^{\hat{w}_1})\right), \quad (11)$$

where $\gamma_u$ denotes the geodesic from $u$ to $v$. On such setting, we perform (11) $m$ times and obtain an initialization set for $\hat{w}_1$, that is, $W = \{\hat{w}_1, \hat{w}_1^2, \ldots, \hat{w}_1^m\}$ with $m$ candidates, covering the original $B_i$. See Fig. 5. With this degeneration, the sequential

![Fig. 5. Vector rotation maximization. MHEAL selects the optimal $\hat{w}_1$ by observing a set of candidate $\hat{w}_1, t = 1, 2, 3, \ldots, m$, where the vector over the geodesic $\gamma_u^{\hat{w}_1}$ holds the maximal disagreement with $\gamma_u^{\hat{w}_1^{-1}}$.](https://scikit-learn.org/stable/modules/preprocessing.html)
Algorithm 2: Minimizing Hyperspherical Energy-Based Active Learning (MHEAL).

1 Input: Data set $\mathcal{X}$, number of clusters $k$, number of representative data $\tau$.
2 Initialize: Randomly initialize $c_1, c_2, c_3, \ldots, c_k$ from $\mathcal{X}$, and then set $S, S^* = \emptyset$.
3 Solve arg min $\sum_{c_k} \left(1 - \cos(c_k, x_i)\right)$.
4 for $i = 1, 2, 3, \ldots, k$ do
5   $\hat{w}_i = \max_{\hat{w}_i \in \mathcal{W}} \mathcal{L}_{E_{\theta, d}}(\hat{w}_i)$ and $S = \hat{w}_i^*$.
6   for $t = 2, 3, \ldots, \tau$ do
7     $x_t = \max_{x_t \in \mathcal{B}_t} \min_{w, s \in S} \|w_i - x_j\|$, $S = S \cup x_{t-1}$.
8 end
9 $S^* = S^* \cup S$.
10 end
11 Output: Representative set $S^*$.

The optimization of (10) is redefined as

$$\arg \min_{\hat{w}_1 \in \mathcal{W}} \mathcal{L}_{E_{\theta, d}}(\hat{w}_1).$$

**Remark 2:** With the vector rotation maximization of (11), then $\hat{w}_1 \in \mathcal{B}_1$ of (10) is updated into $\hat{w}_1 \in \mathcal{W}$ of (11). In this way, the calculation complexity of (10) degenerates into $O(\frac{m^2}{k}N^2)$, that is, (12) results in a much lower calculation complexity after restricting $\hat{w}_1 \in \mathcal{W}$.

**B. Algorithm**

We present MHEAL algorithm to learn representative data, which characterizes decision boundaries in a balanced manner over each pre-estimated hyperspherical cluster.

The detailed steps of the MHEAL algorithm are as follows. Step 3 optimizes a set of spherical clusters adopting hyperspherical clustering w.r.t. (9), where we randomly initialize $c_1, c_2, c_3, \ldots, c_k$ from $\mathcal{X}$. For each hypersphere $\mathcal{B}_t$, Step 5 calculates the optimal initialization on $\hat{w}_i$ following (12), where the initialization set $W$ is obtained from (11); Steps 6 to 8 select $\tau$ data to match $\mathcal{B}_t$. Step 9 merges $\tau$ representative data from $\mathcal{B}_t$ into the representation set $S$, for any $i \leq k$. In this way, the final outputs on $S$ (i.e., $S^*$) has $\tau k$ representative data, which approximately represent the $k$ spherical clusters.

**VI. IMPROVING GENERALIZATION FOR MHEAL**

Section VI-A presents the generalization analysis of MHEAL on error and label complexity bounds. Section VI-B then presents the empirical study to explain the theoretical results.

A. **Error and Label Complexity Bounds**

MHEAL adopts a version space view that yields an unsupervised manner to extract representative data. This reduces the typical label complexity bound of VSAL into a set of local polynomial label complexities [54] over clusters. Here, we follow the IWAL [29] scenario to present the convergence analysis of MHEAL on the generalization error and label complexity bounds.

**IWAL scenario:** Given a finite hypothesis class $\mathcal{H}$, IWAL tries to update the current hypothesis $h_t \in \mathcal{H}$ at $t$-time into the optimal hypothesis $h^* \in \mathcal{H}$. Let IWAL perform $T$ rounds of querying from $\mathcal{X}$, assume that $\ell(\cdot, \cdot)$ denotes the loss of mapping $\mathcal{X}$ into $\mathcal{Y}$ with multi-class setting, we define the total loss of the $T$ rounds of querying as $R(h_T) = \sum_{t=1}^{T} Q_t \ell(h(x_t), y_t)$, where $y_t$ denotes the label of $x_t$, $Q_t$ satisfies the Bernoulli distribution of $Q_t \in [0, 1]$, and $\frac{1}{\delta}$ denotes the weight of sampling $x_t$. The sampling process adopts an error disagreement to control the hypothesis updates:

$$\theta_{\text{IWAL}} = \mathbb{E}_{x_t \in \mathcal{X}} \sup_{h \in \mathcal{B}(h^*, r)} \frac{\ell(h(x_t), y_t) - \ell(h^*(x_t), y_t)}{r},$$

where $B(h^*, r)$ denotes the ball with a center $h^*$ and radius $r$. In hypothesis class, $\ell(h(x_t), y_t) - \ell(h^*(x_t), y_t)$ denotes the maximal hypothesis disagreement over the loss $\ell$. On this setting, $\theta_{\text{IWAL}}$ denotes the minimal step on the significant update of hypothesis. Once the hypothesis update w.r.t. error after adding $x_t$ is larger than $\theta_{\text{IWAL}}$, IWAL solicits $x_t$ as a significant update. Otherwise, $x_t$ will not be an ideal update.

**MHEAL scenario:** Given the input dataset $\mathcal{X}$ with $n$ samples, it is divided into $k$ clusters: $\{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_k\}$, where $\mathcal{B}_i$ has $N_i$ samples. MHEAL performs IWAL for any $\mathcal{B}_i$. Specifically, MHEAL uses a new error disagreement $\theta_{\text{MHEAL}}$ to control the hypothesis updates:

$$\theta_{\text{MHEAL}} = \mathbb{E}_{x_t \in \mathcal{B}_i} \sup_{h \in \mathcal{B}(h^*, r)} \frac{\ell(h(x_t), y_t) - \ell(h^*(x_t), y_t)}{r}.$$  \hspace{1cm} (14)

**Theorem 1:** Given $T$ rounds of querying by employing IWAL, let $Q$ be the number of ground-truth queries, i.e., label complexity. If MHEAL performs IWAL for any $\mathcal{B}_i$, each cluster will have $\tau = T/k$ rounds of querying. Then, with a probability at least $1 - \delta$, for all $\delta > 0$, for any $T > 0$, the error disagreement of $R(h_T)$ and $R(h^*)$ of MHEAL is bounded by $k$ times of polynomial label complexities over each cluster

$$R(h_T) - R(h^*) \leq k \times \max_{H_i, i=1, \ldots, k} \left\{ \sum_{t=1}^{\tau} p_t + 6 \sqrt{\frac{2(3 + \tau)p_T^2}{\delta}} \right\} \times \sqrt{\log \left( \frac{16\tau^2|H_i|^2\log \tau}{\delta} \right)}.$$

Then, with a probability at least $1 - 2\delta$, for all $\delta > 0$, the label complexity of MHEAL can be bounded by

$$Q \leq 8k \times \max_{H_i, i=1, \ldots, k} K_i \left\{ \sum_{j=1}^{N_i} \theta_{\text{MHEAL}} R_{ij}^2 \tau p_j \right\} + \sum_{j=1}^{N_i} O \left( R_{ij}^2 \tau p_j \log \frac{\tau|H_i|N_i}{\delta} \right) + O \left( N_i \log \frac{\tau|H_i|N_i}{\delta} \right).
where $K_\ell$ is the slope asymmetry over the limited loss $\ell$ on $B_j$, i.e., $K_\ell = \sup_{x_i, x_j \in B_j} \left( \max_{h \in H} \left( \ell(h(x_i), y) - \ell(h(x_j), y) \right) \right) - \left( \min_{h \in H} \left( \ell(h(x_i), y) - \ell(h(x_j), y) \right) \right)$, $R_j$ denotes the best-in-class risk at $j$-time querying, and $|H|$ denotes the element number of $H$ ($\ell_0$-norm).

### B. Empirical Study

Datasets selection: Following Cortes’s work in [54], we select six UCI datasets for our empirical study, including skin, shuttle, magic04, jm1, covtype, and nomao.

Hypotheses generation: We take logistic loss defined by $\log(1 + \exp(-y h(x)))$ as the hypothesis prototype. For $d$-dimensional data space, we randomly generate 10,000 $d$-dimensional hyperplanes with a prototype of $\sum_{i=1}^{d} w_i x_i + b = 0$, where any $w_i \in N(0, 1)$, $b$ also follows, and $h^*$ is defined as that hyperplane which generates the minimum loss. Note that $N(0, 1)$ denotes a standard Gaussian distribution.

Specification on $\theta_{\text{IWAL}}$ and $\theta_{\text{MHEAL}}$: With Theorem 1, we know the error disagreement of MHEAL is a tighter polynomial expression of IWAL. We thus need to specify $\theta_{\text{IWAL}}$ and $\theta_{\text{MHEAL}}$ which satisfy $\theta_{\text{MHEAL}} \leq \theta_{\text{IWAL}}$. Recalling (11) and (12), we know 1) $\ell(h(x_i), y) - \ell(h^*(x_i), y) \leq 1$, and 2) $\ell(h(x_i), y) - \ell(h^*(x_i), y)$ of learning from $X$ is tighter than that of learning from $B$, since $B \subset X$. Let $\text{Vol}()$ denote the volume function, we know $\text{Vol}(B) = \text{Vol}(X)$. Let $\epsilon_{\text{IWAL}} = \ell(h(x_i), y) - \ell(h^*(x_i), y)$ and $\epsilon_{\text{MHEAL}} = \ell(h(x_i), y) - \ell(h^*(x_i), y)$, we know $\epsilon_{\text{MHEAL}} = \frac{1}{2} \epsilon_{\text{IWAL}}$. To enlarge the disagreement of $\theta_{\text{IWAL}}$ and $\theta_{\text{MHEAL}}$, we force $\epsilon_{\text{IWAL}}$ to arrive at its upper bound, that is, $\epsilon_{\text{IWAL}} = 1$. Given a radius $r = 0.1$, we then specify $\theta_{\text{IWAL}} = 10$. Given that there are three clusters in $X$, i.e., $k = 3$, with the condition of $\epsilon_{\text{MHEAL}} = \frac{1}{k} \epsilon_{\text{IWAL}}$, we then know $\epsilon_{\text{MHEAL}} = \frac{1}{3} \times 1 = 0.333$. With a same setting on $r = 0.1$, we then know $\theta_{\text{MHEAL}} = 3.333$.

Hypothesis-pruning: We start the hypothesis-pruning using $\theta_{\text{IWAL}}$ and $\theta_{\text{MHEAL}}$, where the pruning condition is that any hypothesis update needs to satisfy a minimal disagreement.

Following [54], we invoke a feasible pruning manner to prune those 10,000 hyperplanes, which can guarantee a near-optimal convergence. The specified way is to iteratively update $\theta_{\text{IWAL}}$ and $\theta_{\text{MHEAL}}$ using $\sqrt{\frac{1}{10}}/t$ for IWAL and $\sqrt{\frac{3}{10}}/t$ for MHEAL to eliminate more loose hypotheses. The specified pruning operation is to eliminate the hyperplane which holds a logistic loss looser than $\sqrt{\frac{1}{10}}/t$ for IWAL or $\sqrt{\frac{3}{10}}/t$ for MHEAL will be eliminated. Fig. 6 presents the comparison of MHEAL and IWAL on their hypothesis-pruning speeds. The results show that MHEAL can prune the hypothesis class faster than typical IWAL.

Generalization on error: Error disagreement of MHEAL is a tighter polynomial expression of IWAL, which prunes the hypothesis faster. We thus study its hypothesis-pruning speed, that is, whether MHEAL can eliminate more insignificant hypotheses using a given coefficient. We thus follow the specification on $\theta_{\text{IWAL}}$ and $\theta_{\text{MHEAL}}$, and present the generalization error of pruning those 10,000 hypotheses into their optimal hypothesis. The error curves are presented in Fig. 7. The results show that MHEAL can obtain a tighter error than IWAL.

Generalization on label complexity: Label complexity bound of MHEAL is also a tighter polynomial expression of IWAL. We thus try to observe their label costs of converging into a desired hypothesis. Fig. 8 presents the label complexities of IWAL and MHEAL. The results show that MHEAL can spend fewer labels to converge than IWAL.

In conclusion, MHEAL employs a tighter error disagreement coefficient than IWAL to pure the hypothesis class, but results in a faster pruning speed. The generalization results on error and label complexity then show tighter bounds than that of IWAL, which verifies our theoretical results in Theorem 1.

VII. EXPERIMENTS

According to the statement of Section V-A, MHEAL algorithm adopts hyperspherical clustering to learn the representative
data in each pre-estimated cluster. The experiments discuss the following five questions.

- Why SphericalKmeans clustering is applied? How about by adopting a gradient solver of expressions of MHE. Specifically, \(-X^2\) Why do we follow \(\ell_0\) expression of MHE? How about \(\ell_1\) and \(\ell_2\) expressions of MHE?
- Why decision boundaries can characterize better representative data? What is the difference of sampling in out-version-space and in-version-space?
- Can MHEAL derive more expressive representative data than the state-of-the-art deep AL baselines?
- Can MHEAL keep solid performance against special data-efficiency learning settings, such as repeated and noisy scenario?

### A. Data-Efficient Spherical Clustering

Clustering model on representative data can be transferred into large-scale learning [55]. In this section, we focus on spherical clustering on representative data, and the related baselines are K-means, GMM, and SphericalKmeans. Fig. 9 presents the performance of clustering on representative data of the three baselines on MNIST and Fashion-MNIST, where the representative data are randomly sampled from the original training sets of the datasets, with a varying number from 1,000 to 6,000. The maximum iterations of these baselines are set as 60.

To improve the non-deep clustering algorithms, we adopt the AutoEncoder and Kullback-Leibler (KL) divergence loss of deep embedding clustering (DEC) [11] to enhance their unsupervised learning results. The dimension of the input layer is 784, the encoding layers follow the (input,output) dimension settings of (784, 500), (500, 500), (500, 2000), (2000, 10), then decoding layers follow the (input,output) dimension settings of (10, 2000), (2000, 500), (500, 500), (500, 784). We also build one clustering layer, which converts the input features to label probability of clusters calculated by student’s t-distribution, measuring the similarity between embedded data and centroids.

The deep clustering results also are presented in Fig. 9. It is clear that the typical clustering baselines have significant accuracy improvements by adopting AutoEncoder and DEC, where SphericalKmeans performs best most of the time whether in non-deep or deep clustering. To analyze their perturbations to the number of representative data, Table I presents the unsupervised accuracy statistics for Fig. 9, where optimal accuracy denotes the best accuracy with a given number of representative data, varying from \{10,000, 20,000, \ldots, 60,000\}, and the best ‘optimal accuracy’ and the lowest ‘standard deviation’ are marked in bold. From those metrics, we find that SphericalKmeans can achieve better clustering performance on representative data than other spherical clustering approaches, characterizing better geometric clustering features.

### B. Data-Efficient Distribution Matching

Distribution matching [56] on representative data presents expressive modeling for input features. We compare the performance disagreements of distribution matching using \(\ell_0\), \(\ell_1\), and \(\ell_2\) expressions of MHE. Specifically, \(\ell_0\) MHE minimizes \(\mathbb{E}_{0,d}\) by adopting our max-min solution of (8), \(\ell_1\) MHE minimizes \(\mathbb{E}_{1,d}\) by adopting a heuristic search with a random beginning, and \(\ell_2\) MHE minimizes \(\mathbb{E}_{2,d}\) by adopting a gradient solver of (3) with a learning rate 0.001. We select KL divergence and Maximum Mean Discrepancy (MMD) [57] as the loss metrics of learning by representative data via MHE.

Given \(\mathcal{X}'\) be the representative data of \(\mathcal{X}\) with a size \(m\), we define \(\ell_{KL} := \sum_i \mathcal{X}'(i) \log \frac{\mathcal{X}(i)}{\mathcal{X}'(i)} + \beta \sum_{i,j} \mathcal{X}(i) - \mathcal{X}'(j)\| + \frac{1}{m^2} \sum_{i,j} \mathcal{X}'(i) - \mathcal{X}'(j)\|_2^2\), where \(\beta\) avoids the calculation infeasible of \(\mathcal{X}'(i) = 0\), and the \(\ell_2\) operation of \(\ell_{MMD}\) avoids the negative values of the \(\ell_2\) metrics on MMD.

Note \(\|\cdot\|_1\) denotes the \(\ell^1\)-norm. Since kernel MMD may result in high computational cost and has parameter perturbations, we select the unbiased \(\ell_2\) operator to define MMD from a
Table I

| Algorithms                  | MNIST       | Fashion-MNIST |
|-----------------------------|-------------|---------------|
|                             | Optimal Acc | Mean Acc      | Std   | Optimal Acc | Mean Acc      | Std   |
| K-means                     | (40,000, 0,53961) | 0.5298        | 0.0095 | (40,000, 0.55031) | 0.5083        | 0.0372 |
| GMM                         | (30000, 0.41239) | 0.3695        | 0.0420 | (50,000, 0.52611) | 0.4566        | 0.0391 |
| Spherical K-means           | (60,000, 0.55188) | 0.5490        | **0.0025** | (10,000, 0.53125) | 0.5262        | **0.0050** |
| K-means + Auto + DEC        | (60000, 0.96221) | 0.9257        | 0.0490 | (60,000, 0.62419) | 0.5821        | 0.0309 |
| GMM + Auto + DEC            | (60000, 0.96256) | 0.8615        | 0.0536 | (60,000, 0.61319) | 0.5743        | 0.0307 |
| Spherical K-means + Auto + DEC | (60,000, 0.96864) | **0.9362**    | 0.0172 | (60,000, 0.63448) | **0.6133**    | 0.0230 |

Fig. 10. \( \ell_{KL} \), \( \ell_{MMD} \), and \( MMD_\mu \) loss curves of distribution matching on representative data via MHE. The curves show that distribution matching by \( \ell_0 \) MHE approximates the input distribution more tightly than \( \ell_1 \) and \( \ell_2 \) MHE. (F-MNIST denotes Fashion-MNIST).

Table II

| Loss Metrics | Algorithms | MNIST       | Fashion-MNIST |
|--------------|------------|-------------|---------------|
|              | Initial Loss | Mean Loss | Std | Initial Loss | Mean Loss | Std |
| \( \ell_{KL} \) | \( \ell_2 \) MHE | 0.1723 | 0.2751 | 0.1396 | 1.5013 | 0.4603 | 0.5325 |
|              | \( \ell_1 \) MHE | 0.1456 | **0.2103** | 0.1331 | **0.5194** | 0.3698 | 0.3409 |
|              | \( \ell_0 \) MHE | **0.0912** | 0.2183 | **0.0788** | 0.7869 | **0.2451** | **0.2788** |
| \( \ell_{MMD} \) | \( \ell_2 \) MHE | 0.0985 | 0.0402 | 0.0354 | 0.1225 | 0.0527 | 0.0418 |
|              | \( \ell_1 \) MHE | 0.0945 | **0.0338** | 0.0385 | 0.1325 | 0.0533 | 0.0455 |
|              | \( \ell_0 \) MHE | **0.0915** | 0.0385 | **0.0330** | **0.1180** | **0.0494** | **0.0406** |
| MMD_\mu      | \( \ell_2 \) MHE | 0.0634 | 0.0284 | 0.0222 | 0.1148 | 0.0410 | **0.0348** |
|              | \( \ell_1 \) MHE | 0.0690 | 0.0290 | 0.0238 | 0.1039 | 0.0395 | 0.0351 |
|              | \( \ell_0 \) MHE | **0.0618** | **0.0249** | **0.0212** | **0.1010** | **0.0364** | 0.0359 |

C. Data-Efficient Version Space Representation

Decision boundaries [15] are distributed in the version space over the tube manifold of hyperspherical distribution. Effective representative data are derived from decision boundaries. This section thus reveals the effectiveness of MHE in characterizing the topology of decision boundaries. Following the cluster boundary detection [58] and out-of-distribution detection [59], we also use a hyper-parameterized threshold to divide one cluster into two parts: 1) in-version-space i.e., tube manifold \( \mathcal{M} \), and 2) out-version-space i.e., inner regions of \( B(c, R - \gamma) \). To specify \( \mathcal{M} \) or \( B(c, R - \gamma) \), we use hyperspherical energy as the parameterized variable.

**Specification of Definition 1:** Given \( K \) samples around \( x_i \), \( \mathbb{E}_{0,d}(x_i) \) denotes the hyperspherical energy of \( B(c, r) \). Assume that there is a threshold \( g' \) which divides \( B(c, r) \) into \( \mathcal{M} \) and \( B(c, R - \gamma) \); if \( \mathbb{E}_{0,d}(x_i) > g', x_i \in \mathcal{M}, \) else \( x_i \in B(c, R - \gamma) \). Specifically, we use PCA to project 2 dimensions of MNIST and Fashion-MNIST as its extracted features. To free \( g' \), we set \( K = 5 \) and collect the top 30% training data with large \( \mathbb{E}_{0,d} \) to specify \( \mathcal{M} \).
We next randomly sample representative data from $\mathcal{M}$ and $B(c, R - \gamma)$ to compare their learning disagreements. Fig. 11 presents the average classification accuracy results with 10 times of random sampling, where the used prediction model is a CNN following [10]. The learning curves clearly show that sampling representative data in in-version-space has significant superiority than that of sampling in out-of-version-space. This explains our motivation of sampling representative data from in-version-space regions, covering decision boundaries of $\gamma$-tube.

D. Data-Efficient Classification

The effectiveness of learning on representative data is evaluated by active sampling from input distribution. The selected sampling baselines are Margin [65], TED [66], GEN [67], SPAL [68], ALDR+ [69], and our MHEAL.

Details of the above baselines are as follows. 1) Margin selects those unlabeled data which derive the minimum margin distances. 2) Transductive experimental design (TED) selects data that represent the unlabeled data using kernel incremental tricks. 3) General AL framework (GEN) queries the samples with both representativeness and diversity. 4) Self-paced AL (SPAL) takes easiness, informativeness, and representativeness into an uniform optimization. 5) ALDR+ selects those discriminative and representative samples.

The selected experimental data are typical UCI datasets, including Phishing, Adult, Satimage, and MNIST. The used classifier tool is LIBSVM following [69]. To present fair comparison, we use SphericalK-maps without AutoEncoder and DEC to perform MHEAL, and use a single query version of ALDR+. Note that Gu et al. [69] presented a bath query strategy for AL. Learning curves of these baselines with average accuracies over 100 times of running are presented in Fig. 12, where the initial number of labeled data are all the class numbers of datasets. The results show that the unsupervised SphericalK-means of MHEAL has natural advantages for few labeled data. The other supervised baselines which need sufficient labels to estimate subsequent sampling, show poor performance against sufficient numbers of representative data. With the increase of the number of representative data, the performance disagreements of those baselines are reduced.

As we can observe from the above analysis, MHEAL adopts an unsupervised manner to learn on representative data, which shows expressive representation performance than those supervised methods. This helps MHEAL to actively spend tighter label complexity to converge into a desired generalization error, verifying the theoretical results of Theorem 1.

E. Data-Efficient Deep Classification

As we stated in the introduction, the typical classification learning by representative data can be implemented by AL. We thus study the classification ability with regard on representative data derived by our MHEAL and the state-of-the-art deep AL baselines. Table III presents the accuracies of classification on representative data of ten baselines on the datasets of MNIST, CIFAR-10, and CIFAR-100, respectively. For the typical AL baselines, e.g., the first seven baselines and MHEAL, training MNIST involves the Bayesian CNN of [10], and training CIFAR-10 and CIFAR-10 involve the ResNet20 of [70]. (Our experience shows that CNN has expressive modeling using few labels on MNIST.) Besides them, CDAL+CoreSet [63], CDAL+Reinforcement [63], and VAAL+VAE+Adversial [64] are set as the network architectures-based deep AL algorithms. Their details are presented as follows.

Details of network architectures-based deep AL baselines.

(1) VAAL+VAE+Adversarial (VAAL: Variational adversarial active learning) [64]. It is a deep AL framework, which adopts a variational autoencoder (VAE) and an adversarial network to learn the latent space.\(^3\) (2) CDAL+CoreSet (CDAL: Contextual Diversity for Active Learning) [63]. This deep AL approach introduces the contextual diversity hinges to observe the probability vectors predicted by a CNN, where the vector regions of interest typically contain diverse label information. This approach cooperates with a core-set to evaluate the contextual diversity.\(^4\) (3) CDAL+Reinforcement (CDAL: Contextual Diversity for Active Learning) [63]. Different from CDAL+CoreSet, this deep AL approach cooperates with a reinforcement learning policy.\(^5\)

We follow [20] to select 20 random samples to initialize the hypothesis of AL. As shown in Table III, 1) the unsupervised baselines such as K-means, K-medoids, Hierarchical Tree, and Core-set perform stably, varying the number of desired representative data, 2) the typical supervised baselines such as Max Entropy, Variation Ratios, and BALD perform not well if annotating few data; only sufficient labeled data can rapidly improve their accuracies, 3) the network architectures-based deep AL, e.g., CDAL+CoreSet and CDAL+Reinforcement, and VAAL+VAE+Adversial need large number of data and labels can show expressive modeling, while show very general accuracies among insufficient annotation budgets, 4) MHEAL employs deep clustering to extract structural clusters from the inputs, showing the state-of-the-art accuracies than other baselines on MNIST. Technically, our locally MHE optimization in each

\(^3\)Code: https://github.com/sinhasam/vaal
\(^4\)Code: https://github.com/sharat29ag/CDAL
\(^5\)Code: https://github.com/sharat29ag/CDAL
cluster yields expressive distribution matching, which derives comprehensive covering on the input features. Better accuracies then are naturally presented.

Fig. 13, 14, and 15 present the learning curves for Table III. This visualizes the performance disagreement of those baselines in a different view. Compared to the experimental results of Section VII-D, those deep learning baselines need more labeled data to strength the advantages of their network configurations. However, unsupervised deep clustering does not need labels, and can obtain more useful information for subsequent MHE optimization, presenting positive guarantees.

F. Data-Efficient Deep Classification With Repeated Data

Sections VII-D and VII-E have presented the classification results of learning representative data by employing several state-of-the-art baselines. However, some of them depend on incremental updates of classification hypothesis, in which those
updates may prefer similar or repeated samples due to the unbalanced class labels in initial data [71]. We thus consider a special scenario: data-efficient deep classification with repeated data following [71]. The goal is to test whether the baseline will be mislead by repeated data.

Following the settings of Section VII-E, we randomly select 10,000 training data and add them into the original training sets of MNIST, CIFAR10, and CIFAR100, respectively. We restart the experiments of Table III and report the results in Table IV. The learning curves are then represented in Figs. 16 to 18 (see supplementary material, available online). The results show that those baselines who invoke the incremental updates of hypothesis may continuously select those scarce repeated samples. However, the unsupervised baselines aim to find the representation samples, which provide fair or sufficient data for each class. It is thus that their biases are lower than those baselines using incremental updates of hypothesis.

### Table IV

| Algorithms                  | MNIST (CNN) | CIFAR-10 (ResNet20) | CIFAR-100 (ResNet20) |
|----------------------------|-------------|---------------------|----------------------|
|                            | 100 200 300 500 | 1K 2K 5K 10K       | 2K 5K 10K 20K       |
| K-means [20]               | 0.7102 0.8687 0.9312 0.9309 | 0.3198 0.3920 0.4989 0.7210 | 0.2201 0.2603 0.3001 0.4401 |
| K-medoids [20]             | 0.8345 0.9217 0.9330 0.9401 | 0.4702 0.5602 0.7102 0.7670 | 0.2526 0.3021 0.4405 0.5201 |
| Hierarchical Tree [60]     | 0.7925 0.8698 0.8875 0.9103 | 0.5012 0.5623 0.7109 0.7587 | 0.2454 0.2901 0.4058 0.4623 |
| Max Entropy [10]           | 0.6846 0.8952 0.9105 0.9567 | 0.4598 0.5412 0.7201 0.7712 | 0.2201 0.2989 0.4108 0.5058 |
| Variation Ratios [10]      | 0.6608 0.8436 0.9106 0.9219 | 0.4802 0.5875 0.7398 0.7874 | 0.2048 0.3130 0.4257 0.5147 |
| Core-set [61]              | 0.6293 0.8386 0.8956 0.9103 | 0.4501 0.6315 0.7309 0.7717 | 0.2587 0.3212 0.4536 0.5287 |
| BALD [62]                  | 0.7542 0.8798 0.9288 0.9541 | 0.5102 0.6426 0.6923 0.7654 | 0.1987 0.3026 0.4156 0.5185 |
| CDAL+CoreSet [63]          | 0.8502 0.8913 0.92986 0.9510 | 0.5065 0.5687 0.6823 0.7689 | 0.2103 0.2987 0.4256 0.4745 |
| CDAL+Reinforcement [63]    | 0.8298 0.8768 0.9201 0.9398 | 0.4687 0.4770 0.6103 0.7792 | 0.2365 0.3108 0.4243 0.4158 |
| VAAL+VAE+Adversial [64]    | 0.8023 0.8409 0.8956 0.9269 | 0.3128 0.3956 0.6012 0.7226 | 0.1948 0.2745 0.3657 0.4635 |
| MHEAL                      | 0.8600 0.9302 0.9369 0.9701 | 0.5378 0.6156 0.7800 0.8042 | 0.3187 0.3745 0.4989 0.5459 |
G. Data-Efficient Deep Classification With Noisy Data

Learning with repeated data is inevitable. However, the tested datasets are clean data; hypothesis updates usually will be misguided by noisy labels due to their larger incremental estimations. We thus consider a more practical scenario: learning from noisy representative data. The goal is to observe the responses of those compared baselines.

Following the settings of Section VII-E, we randomly select 10,000 noisy data and add them into the original training sets of MNIST, CIFAR10, and CIFAR100, respectively. We restart the experiments of Table III and report the results in Table V (see supplementary material, available online). The learning curves are then represented in Figs. 19 to 21 (see supplementary material, available online). The results show that those baselines which invoke the hypothesis incremental updates may be more easily mislead by noises than clean samples. However, the unsupervised baselines use a global manner to find those representation samples, which fairly glances the samples of each class. Therefore, noisy biases of unsupervised methods are lower than supervised baselines since noises may return large perturbations to incremental updates of hypothesis.

VIII. DISCUSSION

Data-efficient learning using representative data is the future trend for AI, and deep AL is one of the effective ways to implement this goal. In this work, MHE is utilized to extract representative data from the topology of decision boundaries. Deep clustering first provides a rough characterization for the clusters before the extraction, yielding superior performance than learning without boundary interactions.

However, such improvements depend on the clustering structures. Once the input distribution has no inherent clustering prototypes, our idea will degenerate into pure MHE optimizations in splitting regions. For real-world tasks, MHEAL will benefit the weakly-supervised [72] and semi-supervised [73] sampling issues. For noisy supervision [74], adopting our idea may reduce the wrong estimation rate on noises. Moreover, MHE optimization may bring geometric benefits for the outlier recognition and out-of-distribution detection [75]. Learning the representative data from the spherical topology, adversarial attacks [76], and malicious intrusion around the decision boundaries may be easier. Capturing the characterization of decision boundaries will fool the network filtering and mailbox defense systems. (Note that [77], [78], [79] are cited in the supplementary material, available online.)

IX. CONCLUSION

Learning from scratch using representative data is helpful for data-efficient AI, and deep AL is one of the effective ways to implement this goal. This paper presents a novel idea of hyperspherical $\ell_0$ MHE from the physical geometry, to actively learn the representative data from the homeomorphic tubes of spherical manifolds. The proposed MHEAL algorithm employs deep spherical clustering, which provides a pre-estimation for the input distribution, characterizing the topology of version space. Then the max-min optimization for MHE derives effective decision boundaries to match each cluster, resulting in a lower distribution loss than $\ell_1$ and $\ell_2$ MHE. In-version-space sampling also showed more expressive modeling for classification than sampling in out-version-space, which manifests the significance of our MHEAL. Generalization error and label complexity bounds prove that MHEAL can safely converge. A series of experiment support the claimed theoretical results. In future, semi-supervised theory of MHEAL may have potential interests.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.
[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2012, pp. 1106–1114.
[3] A. Vaswani et al., “Attention is all you need,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2017, pp. 5998–6008.
[4] D. Amodei et al., “Deep speech 2: End-to-end speech recognition in English and Mandarin,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 173–182.
[5] D. E. O’Leary, “Artificial intelligence and big data,” IEEE Intell. Syst., vol. 28, no. 2, pp. 96–99, Mar./Apr. 2013.
[6] A. Labrinidis and H. V. Jagadish, “Challenges and opportunities with big data,” in Proc. VLDB Endowment, vol. 5, no. 12, pp. 2032–2033, 2012.
[7] J. Jumper et al., “Highly accurate protein structure prediction with AlphaFold,” Nature, vol. 596, no. 7873, pp. 583–589, 2021.
[8] F.-Y. Wang et al., “Where does AlphaGo go: From church-turing thesis to AlphaGo thesis and beyond,” IEEE/CAA J. Automatica Sinica, vol. 3, no. 2, pp. 113–120, Apr. 2016.
[9] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, “Generalizing from a few examples: A survey on few-shot learning,” ACM Comput. Surveys, vol. 53, no. 3, pp. 1–34, 2020.
[10] Y. Gal, R. Islam, and Z. Ghahramani, “Deep Bayesian active learning with image data,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 1183–1192.
[11] J. Xie, R. Girshick, and A. Farhadi, “Unsupervised deep embedding for clustering analysis,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 478–487.
[12] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, “Random erasing data augmentation,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 13 001–13 008.
[13] L. Liu, M. Muely, J. Deng, T. Pifer, and L.-J. Li, “Generative modeling for small-data object detection,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 6073–6081.
[14] W. Li, G. Dasarathy, K. N. Ramamurthy, and V. Berisha, “Finding the homology of decision boundaries with active learning,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2020, Art. no. 700.
[15] A. Beygelzimer, D. Hsu, J. Langford, and T. Zhang, “Agnostic active learning without constraints,” in Proc. 23rd Int. Conf. Neural Inf. Process. Syst., 2010, pp. 199–207.
[16] S. Hanneke, “A bound on the label complexity of agnostic active learning,” in Proc. 24th Int. Conf. Mach. Learn., 2007, pp. 353–360.
[17] A. Beygelzimer, D. Hsu, N. Karampatziakis, J. Langford, and T. Zhang, “Efficient active learning,” in Proc. ICML Workshop On-line Trading Explanation Exploitation, 2011.
[18] W. Liu et al., “Learning towards minimum hyperspherical energy,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2018, pp. 6222–6233.
[19] S. Ben-David and U. Von Luxburg, “Relating clustering stability to properties of cluster boundaries,” in Proc. 21st Annu. Conf. Learn. Theory, Omnipress, 2008, pp. 379–390.
[20] Y. Chen and R. Shibuya, “Shattering distribution for active learning,” IEEE Trans. Neural Netw. Learn. Syst., vol. 33, no. 1, pp. 215–228, Jan. 2022.
[21] S. Hanneke et al., “Theory of disagreement-based active learning,” Found. Trends Mach. Learn., vol. 7, no. 2/3, pp. 131–309, 2014.
[22] C. Cortes, G. DeSalvo, C. Gentile, M. Mohri, and N. Zhang, “Adaptive region-based active learning,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 2144–2153.
[23] C. Cortes, G. DeSalvo, M. Mohri, N. Zhang, and C. Gentile, “Active learning with disagreement graphs,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 1379–1387.
[24] N. Roy and A. McCallum, “Toward optimal active learning through Monte Carlo estimation of error reduction,” in Proc. Int. Conf. Mach. Learn., 2001, pp. 441–448.
Xiaofeng Cao received the PhD degree from the Australian Artificial Intelligence Institute, University of Technology Sydney. He is currently an associate professor with the School of Artificial Intelligence, Jilin University, China and leading a Machine Perceptron Research Group. His research interests include the PAC learning theory, agnostic learning algorithm, generalization analysis, and hyperbolic geometry.

Weiyang Liu is currently conducting research with the University of Cambridge, U.K. and the Max Planck Institute for Intelligent Systems, Tübingen, Germany under the Cambridge-Tübingen Fellowship Program. Prior to joining this program, he has been with the College of Computing, Georgia Institute of Technology, Atlanta, GA, USA. His research interests broadly lie in deep learning, representation learning, interactive machine learning and causality.

Ivor W. Tsang (Fellow IEEE) is the director of A*STAR Centre for Frontier AI Research (CFAR), and the research director of the Australian Artificial Intelligence Institute (AAII). His research focuses on transfer learning, deep generative models, learning with weakly supervision, Big Data analytics for data with extremely high dimensions in features, samples and labels. In 2013, he received his ARC Future Fellowship for his outstanding research on Big Data analytics and large-scale machine learning. In 2019, he received the International Consortium of Chinese Mathematicians Best Paper Award. In 2020, he was recognized as the AI 2000 AAAI/IJCAI Most Influential Scholar in Australia. In addition, he was awarded the IEEE TNN Outstanding 2004 Paper Award, the Best Student Paper Award at CVPR 2010, and the 2014 IEEE TMM Prize Paper Award. In 2021, he was conferred the IEEE fellow for his outstanding contributions to large-scale machine learning and transfer learning. He serves as the editorial board for the Journal of Machine Learning Research, Machine Learning, Journal of Artificial Intelligence Research, IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Artificial Intelligence, IEEE Transactions on Big Data, and IEEE Transactions on Emerging Topics in Computational Intelligence. He serves as a senior area chair/area chair for NeurIPS, ICML, AAAI, and IJCAI, and the steering committee of ACML.