LEAST DISAGREE METRIC-BASED ACTIVE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

The most popular class of active learners today queries for the labels of the samples for which the prediction is most uncertain and uses the labeled samples to update its prediction. Unfortunately, quantifying uncertainty is an open question. This paper mathematically defines uncertainty in terms of the *least disagree metric* (LDM), which is the smallest perturbation required to alter the sample prediction. Based on this metric, the predictor is updated by querying the label of the most uncertain samples. Given a finite-sized training set, empirical LDM is incorporated into an active learning algorithm and used to approximate the theoretical LDM of each sample. Theoretical convergence properties between the empirical and the mathematical definition of LDM are provided. Experimental results show that our algorithm mostly outperforms other high-performing active learning algorithms and leads to state-of-the-art performance on various datasets and deep networks.

1 INTRODUCTION

Active learning (Cohn et al., 1996) is a sub-field in machine learning for attaining sample efficiency by sequentially selecting unlabeled samples for their labels. When selection is performed from a large collection of unlabeled samples, this type of active learning is coined pool-based active learning. Of the various active learning strategies, uncertainty-based sampling (Lewis & Gale, 1994; Sharma & Bilgic, 2017; Nguyen et al., 2022) for its simplicity and relatively low computational load is the most popular. Here, the focus is on determining the uncertainty of each unlabeled sample for a given predictor. However, there is no consensus on quantifying the uncertainty (Balcan et al., 2007; Settles, 2009; Houlsby et al., 2011; Yang et al., 2015; Sharma & Bilgic, 2017; Beluch et al., 2018; Ash et al., 2020).

In this paper, we measure the uncertainty of a sample by perturbing the predictor to see how the predicted label is altered w.r.t. that perturbation. To the best of our knowledge, this approach has rarely been considered in active learning literature. In active learning, the most similar works to ours are the query-by-committee (QBC; Seung et al., 1992) and disagreement region-based approach (Hanneke, 2014), in which a committee of *diverse* experts is formed to identify the most disagreed samples in that the experts give differing predictions. Despite its strong theoretical guarantee of achieving exponential sample complexity (Hanneke, 2014), the computational cost of creating numerous experts (e.g. by training neural networks several times) is very heavy and often in modern deep learning frameworks, infeasible.

As we will see later, our approach suffers from none of the aforementioned problems, largely because ours is based on a completely different framework. The main contributions of this paper are as follows:

1. This paper proposes the *least disagree metric* (LDM) as the measure of the sample's uncertainty for the predictor, defined as the least probability of disagreement of the predicted label with a perturbed predictor.
2. This paper then introduces a finite sample approximation to the true LDM, called *empirical LDM*, along with some theoretical guarantees. We then provide a brute-force algorithm that evaluates the empirical LDM.
3. This paper proposes an active learning algorithm querying unlabeled samples based on (empirical) LDM. Experiments show the algorithm leads to state-of-the-art performance on various datasets and deep networks.

2 LEAST DISAGREE METRIC (LDM)

This section mathematically defines the *least disagree metric* (LDM) of a sample for a given predictor in terms of the sample distribution. Henceforth, it will be assumed that the predictor belongs to a certain hypothesis space. For a given finite sample set, an *empirical LDM* is incorporated into a brute-force algorithm for estimating the theoretical LDM of each sample. This paper will focus on the multi-class classification task.

2.1 DEFINITION OF LDM

Let $\mathcal{X}$ and $\mathcal{Y}$ be the instance and label space with $\mathcal{Y}$, and $\mathcal{H}$ be the hypothesis space of $h : \mathcal{X} \to \mathcal{Y}$. Let $\mathcal{D}$ be the joint distribution over $\mathcal{X} \times \mathcal{Y}$, and $\mathcal{D}_{\mathcal{X}}$ be the instance distribution. Recall the *disagree metric* between two hypotheses (Hanneke, 2014) defined as

$$\rho(h_1, h_2) := \mathbb{P}_{X \sim \mathcal{D}_{\mathcal{X}}}[h_1(X) \neq h_2(X)]$$

where $\mathbb{P}_{X \sim \mathcal{D}_{\mathcal{X}}}$ is the probability measure on $\mathcal{X}$, induced by $\mathcal{D}_{\mathcal{X}}$. For a reference hypothesis $\hat{h} \in \mathcal{H}$ and $\boldsymbol{x}_0 \in \mathcal{X}$, let $\mathcal{H}^{\hat{h}, \boldsymbol{x}_0}$ be the set of hypotheses disagreeing with $\hat{h}$ in their prediction of $\boldsymbol{x}_0$ i.e.

$$\mathcal{H}^{\hat{h}, \boldsymbol{x}_0} := \{h \in \mathcal{H} \mid h(\boldsymbol{x}_0) \neq \hat{h}(\boldsymbol{x}_0)\}.$$

Based on the above set and disagree metric, the uncertainty of a sample to a reference hypothesis is defined as follows:

Definition 1. For given $\hat{h} \in \mathcal{H}$ and $\boldsymbol{x}_0 \in \mathcal{X}$, the **least disagree metric (LDM)** is defined as

$$L(\hat{h}, \boldsymbol{x}_0) := \inf_{h \in \mathcal{H}^{\hat{h}, \boldsymbol{x}_0}} \rho(h, \hat{h}). \tag{1}$$

Conceptually, a sample with a small LDM indicates that even a small perturbation in the predictor can alter prediction and vice versa. Precisely, assuming the the hypothesis h and $\hat{h}$ is parameterized by $\boldsymbol{w}$ and $\hat{\boldsymbol{w}}$,

$$L(\hat{h}, \boldsymbol{x}_1) < L(\hat{h}, \boldsymbol{x}_2) \iff \mathbb{P}_{\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma^2)}[h(\boldsymbol{x}_1) \neq \hat{h}(\boldsymbol{x}_1)] > \mathbb{P}_{\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma^2)}[h(\boldsymbol{x}_2) \neq \hat{h}(\boldsymbol{x}_2)].$$

That is, the sample with the smallest LDM is the most uncertain. This intuition is verified in Appendix B.1.

To illustrate this concept more clearly, we provide a simple example. Consider a two-dimensional binary classification with a set of linear classifiers,

$$\mathcal{H} = \{h : h(\boldsymbol{x}) = \operatorname{sgn}(\boldsymbol{x}^{\mathsf{T}}\boldsymbol{w}), \boldsymbol{w} \in \mathbb{R}^2\}$$

where $\boldsymbol{x}$ is uniformly distributed on $\mathcal{X} = \{\boldsymbol{x} : \|\boldsymbol{x}\| \leq 1\} \subset \mathbb{R}^2$. The decision boundary of $\boldsymbol{w}$ is denoted as $l_{\boldsymbol{w}} = \{\boldsymbol{x} : \boldsymbol{x}^{\mathsf{T}}\boldsymbol{w} = 0\}$. For given $\hat{h}$ and $\boldsymbol{x}_0$, let $\theta \in (-\pi, \pi)$ be the (radian) angle between $l_{\boldsymbol{w}}$ and $l_{\hat{\boldsymbol{w}}}$. Then we have that

$$\rho(h, \hat{h}) = |\theta|/\pi,$$

since $\rho(h, \hat{h})$ is the probability of $X \in \{\boldsymbol{x} \in \mathcal{X} \mid h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})\}$ and $h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})$ for all $\boldsymbol{x}$ in the region between $l_{\boldsymbol{w}}$ and $l_{\hat{\boldsymbol{w}}}$. This is shown in Figure 1; for instance,

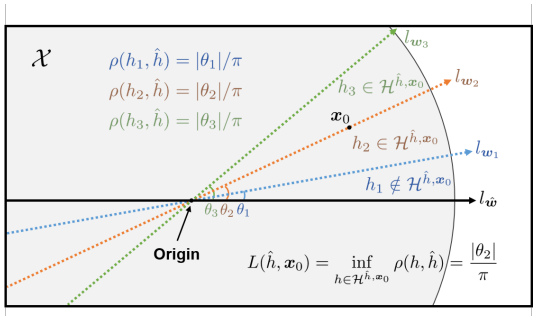


Figure 1: An example of LDM of $\boldsymbol{x}$ for given $\hat{h}$ in binary classification with the linear classifier. Here $\boldsymbol{x}$ is uniformly distributed on $\mathcal{X} \subset \mathbb{R}^2$. The h_2 and h_3 disagree with $\hat{h}$ in prediction for $\boldsymbol{x}_0$, and $L(\hat{h}, \boldsymbol{x}_0) = \inf_{h \in \mathcal{H}^{\hat{h}, \boldsymbol{x}_0}} \rho(h, \hat{h}) = |\theta_2|/\pi$.

$$h_2, h_3 \in \mathcal{H}^{\hat{h}, \boldsymbol{x}_0} = \{h \in \mathcal{H} \mid h(\boldsymbol{x}_0) \neq \hat{h}(\boldsymbol{x}_0)\},$$

and thus,

$$L(\hat{h}, \boldsymbol{x}_0) = \inf_{h \in \mathcal{H}^{\hat{h}, \boldsymbol{x}_0}} \rho(h, \hat{h}) = |\theta_2|/\pi.$$

2.2 EMPIRICAL LDM

In most cases, LDM is not computable for the following two reasons: 1) ρ is generally intractable, especially when $\mathcal{D}_{\mathcal{X}}$ and the form of h are both complicated, e.g., neural networks over real-world image datasets, and 2) one needs to take an infimum over $\mathcal{H}$, which is usually an infinite set. We can consider two approximations: 1) $\mathbb{P}$ in the definition of ρ is replaced by an empirical probability based on S samples, and 2) $\mathcal{H}$ is replaced by a finite hypothesis set $\mathcal{H}_N$ of cardinality N. More precisely, we consider

$$L_{N,S}(\hat{h}, \boldsymbol{x}_0) := \inf_{h \in \mathcal{H}_N^{\hat{h}, \boldsymbol{x}_0}} \left\{ \rho_S(h, \hat{h}) \triangleq \frac{1}{S} \sum_{i=1}^{S} \mathbb{I}\big[h(X_i) \neq \hat{h}(X_i)\big] \right\}, \tag{2}$$

where $\mathcal{H}_N^{\hat{h}, \boldsymbol{x}_0} := \left\{ h \in \mathcal{H}_N \mid h \in \mathcal{H}^{\hat{h}, \boldsymbol{x}_0} \right\}$, $\mathbb{I}[\cdot]$ is an indicator function, and $X_1, \ldots, X_S \overset{i.i.d.}{\sim} \mathcal{D}_{\mathcal{X}}$. Here, S is the number of instances (sampled i.i.d.) for approximating ρ, and N is the number of sampled hypotheses for approximating L.

Assumption 1. Our hypothesis space $\mathcal{H}$ is a complete, separable metric space with metric $d_{\mathcal{H}}(\cdot, \cdot)$.

Assumption 2. $\rho(\cdot, \cdot)$ is B-Lipschitz for some $B > 0$ i.e. $\rho(h, g) \leq B d_{\mathcal{H}}(h, g),\ \forall h, g \in \mathcal{H}$.

In the following theorem- its proof is deferred to Appendix A, we show that our proposed estimator is asymptotically consistent.

Theorem 1. *Let $\hat{h} \in \mathcal{H}$ and $\boldsymbol{x}_0 \in \mathcal{X}$ be arbitrary. Suppose that $\mathcal{H}_1 \subseteq \mathcal{H}_2 \subseteq \cdots \subset \mathcal{H}$ is arbitrary increasing sequence of sets satisfying the following: there exists $\varepsilon > 0$ s.t.*

$$\inf_{\substack{h^* \in \mathcal{H}^{\hat{h}, \boldsymbol{x}_0} \\ \rho(h^*, \hat{h}) - L(\hat{h}, \boldsymbol{x}_0) < \varepsilon}} \min_{h \in \mathcal{H}_N^{\hat{h}, \boldsymbol{x}_0}} d_{\mathcal{H}}(h^*, h) \xrightarrow{\mathbb{P}} 0 \text{ as } N \to \infty. \tag{3}$$

Then, as[1] $\min(S, N) \to \infty$ *satisfying* $S = \omega(\log(CN))$, $\left| L_{N,S}(\hat{h}, \boldsymbol{x}_0) - L(\hat{h}, \boldsymbol{x}_0) \right| \leq \varepsilon$ *in probability.*

Corollary 2. *If Eqn. (3) holds for any $\varepsilon > 0$, then under the same assumptions as Theorem 1, $L_{N,S}(\hat{h}, \boldsymbol{x}_0) \xrightarrow{\mathbb{P}} L(\hat{h}, \boldsymbol{x}_0)$.*

One important consequence is that the ordering of the empirical LDM is preserved in probability:

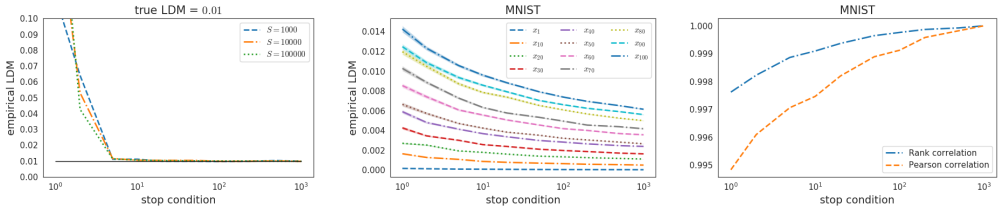
Corollary 3. *Assume that $L(\hat{h}, \boldsymbol{x}_i) < L(\hat{h}, \boldsymbol{x}_j)$. Under the same assumptions as Corollary 2, we have that $L_{N,S}(\hat{h}, \boldsymbol{x}_i) < L_{N,S}(\hat{h}, \boldsymbol{x}_j)$ in probability.*

2.3 BRUTE-FORCE SEARCH FOR EMPIRICAL LDM

Assuming that the hypothesis set is parameterized i.e. $h \in \mathcal{H}$ is of the form $h(\cdot; \boldsymbol{w})$ with $\boldsymbol{w} \in \mathbb{R}^p$, Algorithm 1 describes the algorithm for evaluating empirical LDM of $\boldsymbol{x}$ for $\hat{h}$. Prepare a set of variances $\{\sigma_k^2\}_{k=1}^K$ such that $\sigma_k < \sigma_{k+1}$ and stop condition s. Initialize $L_{\boldsymbol{x}} = 1$. For each k, h is sampled with $\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma_k^2)$, and if $h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})$ then update $L_{\boldsymbol{x}}$ as $\min\{L_{\boldsymbol{x}}, \rho_S(h, \hat{h})\}$. When $L_{\boldsymbol{x}}$ does not change s times consecutively, move on to $k + 1$. The final output $L_{\boldsymbol{x}}$ is the desired empirical LDM.

When sampling the weights, the reason for using several σ^2 is two-fold. We would want the sampled hypothesis h to satisfy $h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})$. Importantly, such probability (over the randomness of $\boldsymbol{w}$ and thus h) is expected to be monotone increasing in σ^2, as larger σ^2 indicates that the sampled hypothesis h is further away from $\hat{h}$. For the same reason, it is expected that the value $\mathbb{E}_{\boldsymbol{w}}[\rho_S(h, \hat{h})]$ is monotone increasing in σ^2. In other words, with too large σ^2, the minimum of $\rho_S(h, \hat{h})$ over the sampled h's may be too far away from the true LDM, especially when the true LDM is close to 0. This intuition is verified in Appendix B.2.

[1] For the asymptotic analyses, we write $f(n) = \omega(g(n))$ if $\lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$.



(a) (b) (c)

Figure 2: Empirical LDMs evaluated by Algorithm 1 by varying the stop condition s. (a) Here, we consider the two-dimensional binary classification with the linear classifier (see Figure 1). The empirical LDM is very close to the true LDM even when $s = 10$, and it reaches the true LDM when $s \geq 20$. (b) Empirical LDMs of MNIST samples with a four-layered CNN. Observe that the empirical LDM monotonically decreases as s increases, and the rank order is well maintained. (c) In the same setting, the rank and Pearson's correlations between the empirical LDMs when $s = 10$ and $s = 1000$ are 0.999 and 0.997, respectively, suggesting that a moderate value of s suffices.

Figure 2a shows the empirical LDM, computed by our Algorithm 1, of a sample in binary classification with the linear classifier as described in Figure 1. The true LDM of the sample is 0.01, and the empirical LDM is evaluated 100 times for $S \in \{1000, 10000, 100000\}$ and $s \in \{1, 2, 5, 10, 20, 50, 100, 200, 500, 1000\}$. The empirical LDM is very close to the true LDM when $s = 10$, and it reaches the true LDM when $s \geq 20$ with the gap being roughly 10^{-4}. This suggests that even with a moderate value of s, our approach can output empirical LDM that is sufficiently close to the true LDM.

Figure 2b shows the empirical LDMs of MNIST samples for a four-layered CNN. The empirical LDMs are evaluated 100 times at $S = 60000$, which is the total number of samples of MNIST. We denote $\boldsymbol{x}_i$ as the i^{th} sample ordered by the computed empirical LDM. Observe that the empirical LDMs are monotonically decreasing as s increases, but even with $s = 1000$, the empirical LDMs do not seem to converge. Furthermore, when $s = 1000$, as our algorithm samples 1M hypotheses, it takes roughly 20 min to evaluate the empirical LDM of a single sample, suggesting that it is computationally infeasible to obtain empirical LDM that is very close to the true LDM; this is further discussed in Appendix D.1. Even though this is the case, one important observation is that the empirical LDM maintains the relative ranking between the samples. Figure 2c shows the rank and Pearson's correlations of the empirical LDMs to that at $s = 1000$. In MNIST, when $s = 10$ and $s = 1000$, the rank and Pearson's correlations reach 0.999 and 0.997, respectively. We set $s = 10$ in Algorithm 1 in our experiments, as the runtime of our algorithm is of few seconds. As we will discuss later, the ranking of the samples via their estimated empirical LDM is what is important.

3 LDM-BASED ACTIVE LEARNING

This section introduces LDM-S, the LDM-based batch sampling algorithm for pool-based active learning. This is the setting in which we have a set of unlabeled samples, $\mathcal{U}$, and we simultaneously query q samples from randomly sampled pool data $\mathcal{P} \subset \mathcal{U}$ of size m.

3.1 LDM-BASED BATCH SAMPLING

One LDM-based approach is to choose q samples with the lowest LDMs, which can be thought of as choosing the most uncertain samples. However, as shown in Appendix D.2, this strategy often does *not* lead to good performance; upon further inspection, we observed that there's a significant overlap of information in the selected batch. This problem is prevalent in batch active learning, and one popular approach to mitigate this is to consider diversity to minimize redundancy among the selected samples (Kirsch et al., 2019; Ash et al., 2020; Citovsky et al., 2021; Yang et al., 2021).

We incorporate diversity via a modification of the k-means++ seeding algorithm (Arthur & Vassilvitskii, 2007), which was reported to be the best algorithm to increase batch diversity without introducing additional hyperparameters (Ash et al., 2020). Intuitively, the k-means++ seeding selects centroids by iteratively sampling points proportional to their squared distance from the nearest

Algorithm 1 Evaluation of Empirical LDM

Input:
$\hat{h}, \boldsymbol{x}$: Given hypothesis and sample
$\{\sigma_k^2\}_{k=1}^K$: Set of Variances
s : Stop condition for parameter sampling

Function:
$L_{\boldsymbol{x}} = 1$
for $k = 1$ **to** K **do**
  $c = 0$
  while $c < s$ **do**
    $\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma_k^2)$
    $c = c + 1$
    if $h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})$ **and** $L_{\boldsymbol{x}} > \rho_S(h, \hat{h})$ **then**
      $L_{\boldsymbol{x}} \leftarrow \rho_S(h, \hat{h})$
      $c = 0$
    end if
  end while
end for
return: $L_{\boldsymbol{x}}$

Algorithm 2 LDM-weighted Seeding (LDM-S)

Input:
$\mathcal{L}_0, \mathcal{U}_0$: Initial labeled and unlabeled samples
m, q : pool and query size

Procedure:
for $t = 0$ **to** $T - 1$ **do**
  Obtain $\hat{\boldsymbol{w}}$ by training on $\mathcal{L}_t$
  Randomly sample $\mathcal{P} \subset \mathcal{U}_t$ with $|\mathcal{P}| = m$
  Evaluate $L_{\boldsymbol{x}}$ for $\boldsymbol{x} \in \mathcal{P}$ by Algorithm 1
  Compute $\gamma(\boldsymbol{x})$ using Eq. 5
  $\mathcal{Q}_1 \leftarrow \{\boldsymbol{x}_1\}$ where $\boldsymbol{x}_1 = \arg\min_{\boldsymbol{x} \in \mathcal{P}} L_{\boldsymbol{x}}$
  for $n = 2$ **to** q **do**
    $p(\boldsymbol{x}) = \gamma(\boldsymbol{x}) * \min_{\boldsymbol{x}' \in \mathcal{Q}_{n-1}} d_{\cos}(\boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{z}_{\boldsymbol{x}'})$
    Sample $\boldsymbol{x}_n \in \mathcal{P}$ w.p. $\mathbb{P}(\boldsymbol{x}_n) = \frac{p(\boldsymbol{x}_n)^2}{\sum_{\boldsymbol{x}_j \in \mathcal{P}} p(\boldsymbol{x}_j)^2}$
    $\mathcal{Q}_n \leftarrow \mathcal{Q}_{n-1} \cup \{\boldsymbol{x}_n\}$
  end for
  $\mathcal{L}_{t+1} \leftarrow \mathcal{L}_t \cup \{(\boldsymbol{x}_i, y_i)\}_{\boldsymbol{x}_i \in \mathcal{Q}_q}$, $\mathcal{U}_{t+1} \leftarrow \mathcal{U}_t \setminus \mathcal{Q}_q$
end for

centroid that has already been chosen, which tends to select a diverse batch. Our proposed modification is to use the cosine distance between features of samples instead of the usual ℓ_2-distance. The reason for considering cosine distance is that for the final prediction, the scales of the features do not matter, and the reason for considering features is that the perturbation is applied to the weights of the last layer that takes the features as input. We introduce two seeding methods based on the principle of querying samples with the least LDMs while pursuing diversity:

Seeding on the subset with small LDM. Let $\mathcal{P}_l \subset \mathcal{P}$ be the set of top l samples with the smallest LDM, where l is about 2-5 times the query size q. The LDM-based seeding algorithm starts by selecting unlabeled samples with the smallest LDM in $\mathcal{P}_l$. The next distinct unlabeled sample is sampled from $\mathcal{P}_l$ by the following probability:

$$\mathbb{P}(\boldsymbol{x}) = \frac{p(\boldsymbol{x})^2}{\sum_{\boldsymbol{x}_j \in \mathcal{P}_l} p(\boldsymbol{x}_j)^2}, \quad p(\boldsymbol{x}) = \min_{\boldsymbol{x}' \in \mathcal{Q}} d_{\cos}(\boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{z}_{\boldsymbol{x}'}) \tag{4}$$

where $\mathcal{Q}$ is the set of selected unlabeled samples for querying, $d_{\cos}(\cdot, \cdot)$ is the cosine distance, and $\boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{z}_{\boldsymbol{x}'}$ are the feature vectors of $\boldsymbol{x} \in \mathcal{P}_l$ and $\boldsymbol{x}' \in \mathcal{Q}$, respectively. Repeating until $|\mathcal{Q}| = q$ completes the selection of unlabeled samples for querying. However, this method requires setting an additional hyperparameter l, and in the chosen subset $\mathcal{P}_l$, the selection probability does not explicitly take LDM into account. Thus the LDMs of the resulting query set are distributed evenly, which is undesirable as we expect for the samples with low LDMs to be chosen more often.

Seeding by LDM-weighted distribution. For querying samples with the smallest LDMs, this paper considers the exponential decay w.r.t LDM, which is a common choice as a decay function in machine learning, e.g., EXP3 in bandits (Auer et al., 2002). In addition, the pool set $\mathcal{P}$ is partitioned as $\mathcal{P}_q$ and $\mathcal{P}_c$ to balance the effect of choosing samples with the least LDMs and diversity, where $\mathcal{P}_q$ is the set of samples with smallest LDM of size q (=query size) and $\mathcal{P}_c = \mathcal{P} \setminus \mathcal{P}_q$. To avoid being biased towards either LDM or diversity regardless of the choice of m and q, the total weights of $\mathcal{P}_q$ and $\mathcal{P}_c$ are balanced to be equal. Precisely, the weights of $\boldsymbol{x} \in \mathcal{P}$ are defined as follows:

$$\gamma(\boldsymbol{x}) = \begin{cases} \tilde{\gamma}(\boldsymbol{x}) / \sum_{\boldsymbol{x}_j \in \mathcal{P}_q} \tilde{\gamma}(\boldsymbol{x}_j) = \frac{1}{q}, & \boldsymbol{x} \in \mathcal{P}_q \\ \tilde{\gamma}(\boldsymbol{x}) / \sum_{\boldsymbol{x}_j \in \mathcal{P}_c} \tilde{\gamma}(\boldsymbol{x}_j), & \boldsymbol{x} \in \mathcal{P}_c \end{cases} \tag{5}$$

where

$$\tilde{\gamma}(\boldsymbol{x}) = \exp\left(-\frac{(L_{\boldsymbol{x}} - L_q)_+}{L_q}\right),$$

$(\cdot)_+ = \max\{0, \cdot\}$, and $L_q = \max_{\boldsymbol{x} \in \mathcal{P}_q} L_{\boldsymbol{x}}$. Then, the Eq. 4 is replaced with the following:

$$\mathbb{P}(\boldsymbol{x}) = \frac{p(\boldsymbol{x})^2}{\sum_{\boldsymbol{x}_j \in \mathcal{P}} p(\boldsymbol{x}_j)^2}, \quad p(\boldsymbol{x}) = \gamma(\boldsymbol{x}) * \min_{\boldsymbol{x}' \in \mathcal{Q}} d_{\cos}(\boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{z}_{\boldsymbol{x}'}).$$

Table 1: Settings for data and acquisition size. Acquisition size denotes the number of initial labeled samples + query size for each step (the size of pool data) $\rightarrow$ the number of final labeled samples.

Dataset	Model	# of parameters sampled / total	Data size train / validation / test	Acquisition size		
MNIST	S-CNN	1.3K/1.2M	55,000 / 5,000 / 10,000	20	+20 (2,000)	$\rightarrow$ 1,020
CIFAR10	K-CNN	5.1K/2.2M	45,000 / 5,000 / 10,000	200	+400 (4,000)	$\rightarrow$ 9,800
SVHN	K-CNN	5.1K/2.2M	68,257 / 5,000 / 26,032	200	+400 (4,000)	$\rightarrow$ 10,200
CIFAR100	WRN-16-8	51.3K/11.0M	45,000 / 5,000 / 10,000	5,000	+2,000 (10,000)	$\rightarrow$ 25,000
Tiny ImageNet	WRN-16-8	409.8K/11.4M	90,000 / 10,000 / 10,000	10,000	+5,000 (20,000)	$\rightarrow$ 50,000
FOOD101	WRN-16-8	206.9K/11.2M	60,600 / 15,150 / 25,250	6,000	+3,000 (15,000)	$\rightarrow$ 30,000

Note that this resolves the disadvantages of the first approach; there's no hyperparameter, and the selection probability is explicitly impacted by LDMs while having diversity as well. This difference is shown in Figure 3.

3.2 Algorithm for LDM-weighted Seeding

We now introduce LDM-S in Algorithm 2, the LDM-weighted seeding algorithm for active learning. Let $\mathcal{L}_t$ and $\mathcal{U}_t$ be the set of labeled and unlabeled samples at iteration t. At each step t, the $\hat{\boldsymbol{w}}$ is obtained by training on $\mathcal{L}_t$, and the set of pool samples $\mathcal{P} \subset \mathcal{U}_t$ with $|\mathcal{P}| = m$ is drawn uniformly at random. Then for each $\boldsymbol{x} \in \mathcal{P}$, $L_{\boldsymbol{x}}$ and $\gamma(\boldsymbol{x})$ are evaluated by Algorithm 1 and Eq. 5, respectively. The set of selected unlabeled samples, $\mathcal{Q}_1$, is initialized as $\{\boldsymbol{x}_1\}$ where $\boldsymbol{x}_1 = \arg\min_{\boldsymbol{x} \in \mathcal{P}} L_{\boldsymbol{x}}$. For $n = 2, \ldots, q$, the algorithm samples $\boldsymbol{x}_n \in \mathcal{P}$ with probability $\mathbb{P}(\boldsymbol{x}) = p(\boldsymbol{x})^2 / \sum_{\boldsymbol{x}_j \in \mathcal{P}} p(\boldsymbol{x}_j)^2$ where $p(\boldsymbol{x}) = \gamma(\boldsymbol{x}) * \min_{\boldsymbol{x}' \in \mathcal{Q}_{n-1}} d_{\cos}(\boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{z}_{\boldsymbol{x}'})$ and appends it to $\mathcal{Q}_n$. Lastly, the algorithm queries the label y_i of each $\boldsymbol{x}_i \in \mathcal{Q}_q$, and the algorithm continues until $t = T - 1$.

4 Experiments

This section presents empirical results of the effect of diverse sampling on the LDM-based algorithm, as well as a comprehensive performance comparison with various uncertainty-based active learning algorithms. Six image classification benchmark datasets are considered: MNIST (Lecun et al., 1998), CIFAR10 (Krizhevsky, 2009), SVHN (Netzer et al., 2011), CIFAR100 (Krizhevsky, 2009), Tiny ImageNet (Le & Yang, 2015), and FOOD101 (Bossard et al., 2014) datasets. S-CNN, K-CNN (Chollet et al., 2015) and Wide-ResNet (WRN-16-8; Zagoruyko & Komodakis, 2016) are used to evaluate the performance. All results are averaged over 5 repetitions. The active learning settings regarding data, initial, and query sizes are summarized in Table 1. More details of datasets, networks, and training settings are presented in Appendix C.

4.1 Effect of Diverse Sampling on LDM-based Algorithm

We first examine the effect of diverse sampling on our LDM-based active learning algorithm by considering various combinations of LDM and diverse sampling. Figure 3 shows the test accuracy with respect to the number of labeled samples and the histogram of samples' indices selected from pool data sorted by their LDMs on MNIST, CIFAR10, and CIFAR100 datasets, with the sizes of the seeding subsets being $5q$, $5q$, and $2q$ respectively. LDM-weighted seeding selects more samples with small LDM while pursuing batch diversity, which leads to significant performance improvement compared to when batch diversity is not considered. However, seeding on a subset of samples with small LDMs shows no significant performance improvement when diversity is not considered. This is because, despite increasing diversity, only a few samples with small LDMs are chosen. As a sanity check, we also consider 'seeding on pool data', which is the modified k-means++ seeding on pool data without considering LDM. The result does indeed show that the success of our LDM-S cannot be solely attributed to the effect of diverse sampling.

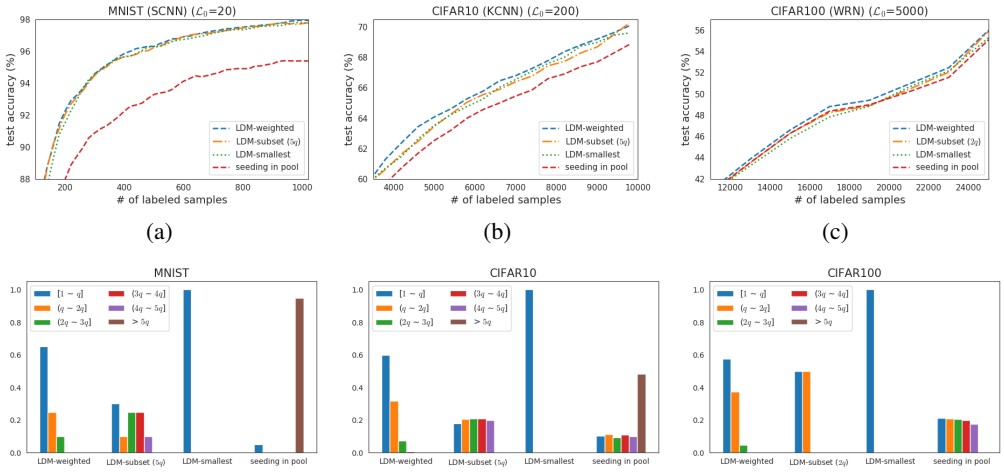


Figure 3: Final performance comparisons (a,b,c) and histograms of samples' indices selected from pool data sorted by LDMs on MNIST, CIFAR10, and CIFAR100 datasets (d,e,f). 'LDM-weighted': seeding by LDM-weighted distribution, 'LDM-subset (kq)': seeding on a subset of size kq with small LDM, 'LDM-smallest': selecting batch with the smallest LDM, 'seeding in pool': modified k-means++ seeding on pool data without considering LDM. LDM-weighted seeding selects more samples with small LDM while pursuing batch diversity, which leads to significant performance improvement compared to those without batch diversity.

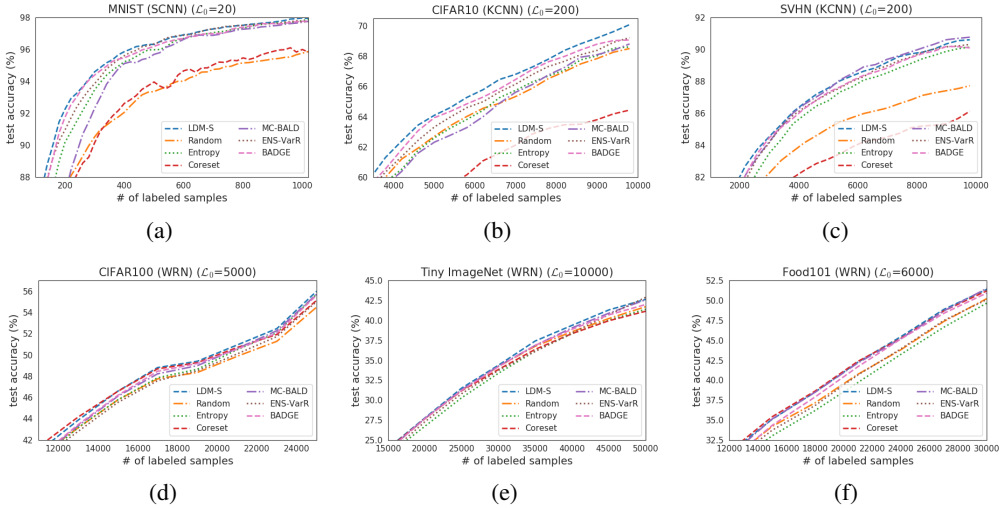


Figure 4: The performance comparison of LDM-S with the other algorithms on MNIST with S-CNN (a), CIFAR10 (b), SVHN (c), CIFAR100 (d), Tiny ImageNet (e), and FOOD101 (f) datasets. Overall, LDM-S consistently performs best or is comparable with all other algorithms.

4.2 Comparing LDM-S to Other Algorithms

We now compare the performance of LDM-S with the baseline active learning algorithms, including state-of-the-art algorithms. Figure 4 shows the test accuracy with respect to the number of labeled samples on MNIST with S-CNN, CIFAR10, and SVHN with K-CNN, and CIFAR100, Tiny ImageNet, and FOOD101, with WRN-16-8[2].

[2]Here, we show plots of test accuracy enlarged appropriately to accentuate the performance difference among various methods. Figures for initially labeled sample sizes are presented in Appendix E.2.

Table 2: The mean ± standard deviation of the averaged performance difference (%) relative to Random overall steps for each repetition. The positive or negative value indicates respectively higher or lower performance compared to Random, and the asterisk (*) indicates that the p-value is less than 0.05 in paired sample t-test between LDM-S and others.

	MNIST	CIFAR10	SVHN	CIFAR100	T. ImageNet	FOOD101
LDM-S[ours]	**3.33±0.43**	**1.34±0.19**	**2.53±0.22**	**0.98±0.44**	**0.55±0.16**	1.27±0.34
Entropy[49]	2.36±0.84*	0.00±0.21*	1.52±0.19*	0.37±0.60	-0.61±0.28*	-0.86±0.20*
Coreset[45]	-0.04±1.23*	-3.71±0.56*	-1.66±0.51*	0.89±0.49	-0.20±0.46*	**1.30±0.16**
MC-BALD[22]	1.68±0.80*	-0.15±0.31*	2.46±0.21	0.55±0.77	0.27±0.19*	1.18±0.35
ENS-VarR[6]	2.98±0.36	0.58±0.28*	2.08±0.22*	0.03±0.41*	-0.15±0.35*	-0.15±0.46*
BADGE[3]	3.01±0.45	0.90±0.21*	2.18±0.23*	0.64±0.48	0.12±0.40*	0.71±0.43*

Each algorithm is denoted as follows 'Entropy': entropy-based uncertainty sampling (Shannon, 1948), 'Coreset': core-set selection (Sener & Savarese, 2018), 'MC-BALD': MC-dropout sampling with BALD (Gal et al., 2017), 'ENS-VarR': ensemble method with variation ratio (Beluch et al., 2018), and 'BADGE': batch active learning by diverse gradient embeddings (Ash et al., 2020). For 'MC-BALD', we use 100 forward passes, and for 'ENS-VarR', we use an ensemble consisting of 5 networks of identical architecture but different random initialization and random batches.

Overall, LDM-S either consistently performs best or is at par with other algorithms for all datasets, while the performance of the algorithms except LDM-S varies depending on datasets. For instance, 'Entropy' and 'Coreset' show poor performance compared to other uncertainty-based algorithms, including ours, on MNIST, CIFAR10, and SVHN, while 'Coreset' performs at par with ours on FOOD101. 'ENS-VarR', although comparable to other algorithms, still underperforms compared to LDM-S on all datasets. A similar trend can also be observed for 'MC-BALD' and 'BADGE'. Furthermore, the runtime of LDM-S is comparable to Entropy, MC-BALD, Coreset, and BADGE; see Appendix E.1 for more details. In Appendix D.3, the performance of LDM-S is compared with the standard uncertainty methods (Entropy, MC-BALD, ENS-VarR) to which weighted seeding is applied, with which these methods could be further benefitted. Even in that case, LDM-S still outperforms the other methods, suggesting that the performance gains of LDM-S are attributed to our newly introduced LDM's superiority over the other uncertainty measures, and are *not* mere artifacts of the seeding.

All in all, Table 2 presents the mean and standard deviation of the averaged performance difference relative to Random overall steps for each repetition. The positive or negative value indicates respectively higher or lower performance compared to Random, and the asterisk (*) indicates the p-value is less than 0.05 in paired sample t-test for the null of no difference versus the alternative that the LDM-S is better than others. We observe that LDM-S either consistently performs best or is comparable with other algorithms for all datasets, while the performance of the algorithms except LDM-S varies depending on the datasets.

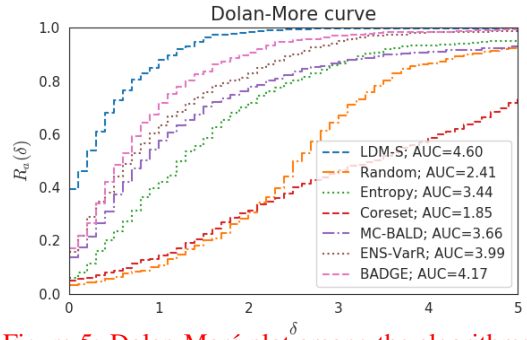


Figure 5: Dolan-Moré plot among the algorithms across all experiments. LDM-S outperforms all other algorithms. Here, AUC is the area under the curve of the plot.

In order to provide a comprehensive comparison across datasets, the performance profile, which is known as Dolan-Moré plot (Dolan & Moré, 2002), is examined. This curve has been widely considered in benchmarking active learning (Tsymbalov et al., 2018; 2019), optimization profiles (Dolan & Moré, 2002), and even general deep learning tasks (Burnaev et al., 2015a;b). To introduce the Dolan-Moré plot, let $\mathrm{acc}_a^{D,r,t}$ be the test accuracy of algorithm a at step t, for dataset D and repetition r, and $\Delta_a^{D,r,t} = \max_{a'}(\mathrm{acc}_{a'}^{D,r,t}) - \mathrm{acc}_a^{D,r,t}$. Then, we define the performance profile as

$$R_a(\delta) := \frac{\sum_{D,r} \sum_t \mathbb{I}(\Delta_a^{D,r,t} \leq \delta\ )}{\sum_{D,r} T_{D,r}} \tag{6}$$

where $T_{D,r}$ is the number of steps for dataset D at repetition r. Intuitively, $R_a(\delta)$ is the fraction of cases where the performance gap between algorithm a and the best competitor is less than δ. Specifically, when $\delta = 0$, $R_a(0)$ is the fraction of cases on which algorithm a performs the best.

Figure 5 shows the performance profile w.r.t. δ, for all algorithms. Overall, it is clear that LDM-S retains the highest $R_a(\delta)$ over all considered δ's. We also observe that $R_{\text{LDM-S}}(0) = 40\%$ while the other algorithms have a value less than 20%. All in all, this clearly shows that our LDM-S outperforms the other considered algorithms.

5 Related Work

There are various active learning strategies such as uncertainty sampling (Lewis & Gale, 1994; Scheffer et al., 2001; Culotta & McCallum, 2005; Wang et al., 2010; Sharma & Bilgic, 2017), expected model change (Settles et al., 2007; Freytag et al., 2014; Ash et al., 2020), expected error reduction (Roy & McCallum, 2001; Yoo & Kweon, 2019; Zhao et al., 2021a), variance reduction (Schein & Ungar, 2007), uncertainty reduction (Zhao et al., 2021b), core-set approach (Sener & Savarese, 2018; Mahmood et al., 2022), clustering (Yang et al., 2021; Citovsky et al., 2021), Bayesian active learning (Pinsler et al., 2019; Shi & Yu, 2019), discriminative sampling (Sinha et al., 2019; Zhang et al., 2020; Gu et al., 2021; Caramalau et al., 2021), Fisher information (Ash et al., 2021), multi-armed bandit (Bouneffouf et al., 2014), bidirectional exploration (Zhang et al., 2015), and data augmentation (Kim et al., 2021)

For the uncertainty-based approach, various forms of uncertainty measures have been studied. *Entropy* (Shannon, 1948) based uncertainty sampling strategies query unlabeled samples yielding the maximum entropy from the predictive distribution, but it does not perform well for multiclass classification tasks as the entropy is heavily influenced by probabilities of less important classes (Joshi et al., 2009). *Margin* based strategies query unlabeled samples closest to the decision boundary, and it is generally understood that unlabeled sample closest to the decision boundary is the most uncertain (Balcan et al., 2007; Kremer et al., 2014; Ducoffe & Precioso, 2018). However, it is difficult to identify samples closest to the decision boundary for multiclass classification with the deep network as the Euclidean distance is often not readily measurable (Ducoffe & Precioso, 2018; Mickisch et al., 2020). *Mutual information* based strategies, such as BALD (Houlsby et al., 2011), DBAL (Gal et al., 2017), and BatchBALD (Kirsch et al., 2019), query unlabeled samples yielding the maximum mutual information between predictions and model parameters. The DBAL approximates the posterior of the model parameters of the deep network by MC-dropout sampling, but each batch selection is independently conducted, and this leads to data inefficiency as correlations between data points in the batch are not taken into account (Kirsch et al., 2019). To address this deficiency, BatchBALD was introduced, but because BatchBALD theoretically computes all possible mutual information between batch-wise predictions and model parameters. For this reason, it is not appropriate for large query sizes. *Variation ratio* (Freeman, 1965) with ensemble method (Beluch et al., 2018) based on query by committee (QBC) strategy (Seung et al., 1992) queries unlabeled samples yielding the maximum variation ratio in labels predicted by the multiple networks, but it requires high computational load: each network belonging to the ensemble must be individually trained. *Gradient* based strategy (Ash et al., 2020) measures uncertainty as the gradient magnitude with respect to parameters in the final layer and queries unlabeled samples where these gradients span a diverse set of directions, but it requires a high computational load when the dimension of parameters is large.

6 Conclusion

This paper defines the least disagree metric (LDM), which measures the uncertainty of samples by perturbing the predictor, and introduces a hypothesis sampling method for evaluating approximated LDM (empirical LDM). In addition, this paper proposes an LDM-based active learning algorithm to select unlabeled samples with small LDM while pursuing batch diversity. The proposed algorithm either consistently performs best or is comparable with other high-performing active learning algorithms, leading to state-of-the-art active learning performance on most of the datasets considered in this paper. Our algorithm is simple and relatively light in terms of computational cost, and thus we expect it to be a meaningful addition to the field of uncertainty-based active learning.

REFERENCES

David Arthur and Sergei Vassilvitskii. K-Means++: The Advantages of Careful Seeding. In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07, pp. 1027–1035, USA, 2007. Society for Industrial and Applied Mathematics. ISBN 9780898716245.

Jordan Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham Kakade. Gone Fishing: Neural Active Learning with Fisher Embeddings. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), Advances in Neural Information Processing Systems, volume 34, pp. 8927–8939. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/4afe044911ed2c247005912512ace23b-Paper.pdf.

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. In International Conference on Learning Representations, 2020. URL https://openreview.net/forum?id=ryghZJBKPS.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multi-armed bandit problem. SIAM journal on computing, 32(1):48–77, 2002.

Maria-Florina Balcan, Andrei Broder, and Tong Zhang. Margin based active learning. In Nader H. Bshouty and Claudio Gentile (eds.), Learning Theory – COLT 2007, pp. 35–50, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-72927-3.

William H. Beluch, Tim Genewein, Andreas Nurnberger, and Jan M. Kohler. The Power of Ensembles for Active Learning in Image Classification. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9368–9377, 2018. doi: 10.1109/CVPR.2018.00976.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), Computer Vision – ECCV 2014, pp. 446–461, Cham, 2014. Springer International Publishing.

Djallel Bouneffouf, Romain Laroche, Tanguy Urvoy, Raphael Feraud, and Robin Allesiardo. Contextual bandit for active learning: Active thompson sampling. In Chu Kiong Loo, Keem Siah Yap, Kok Wai Wong, Andrew Teoh, and Kaizhu Huang (eds.), Neural Information Processing – ICONIP 2014, pp. 405–412, Cham, 2014. Springer International Publishing. ISBN 978-3-319-12637-1.

G. E. P. Box and Mervin E. Muller. A Note on the Generation of Random Normal Deviates. The Annals of Mathematical Statistics, 29(2):610 – 611, 1958. doi: 10.1214/aoms/1177706645. URL https://doi.org/10.1214/aoms/1177706645.

E. Burnaev, P. Erofeev, and A. Papanov. Influence of resampling on accuracy of imbalanced classification. In Antanas Verikas, Petia Radeva, and Dmitry Nikolaev (eds.), Eighth International Conference on Machine Vision (ICMV 2015), volume 9875, pp. 987521. International Society for Optics and Photonics, SPIE, 2015a. doi: 10.1117/12.2228523. URL https://doi.org/10.1117/12.2228523.

E. Burnaev, P. Erofeev, and D. Smolyakov. Model selection for anomaly detection. In Antanas Verikas, Petia Radeva, and Dmitry Nikolaev (eds.), Eighth International Conference on Machine Vision (ICMV 2015), volume 9875, pp. 987525. International Society for Optics and Photonics, SPIE, 2015b. doi: 10.1117/12.2228794. URL https://doi.org/10.1117/12.2228794.

Razvan Caramalau, Binod Bhattarai, and Tae-Kyun Kim. Sequential Graph Convolutional Network for Active Learning. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9578–9587, 2021. doi: 10.1109/CVPR46437.2021.00946.

François Chollet et al. Keras. https://keras.io, 2015.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch Active Learning at Scale. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), Advances in Neural Information Processing Systems, volume 34, pp. 11933–11944. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/64254db8396e404d9223914a0bd355d2-Paper.pdf.

David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active Learning with Statistical Models. Journal of Artificial Intelligence Research, 4(1):129–145, Mar 1996. ISSN 1076-9757.

Aron Culotta and Andrew McCallum. Reducing Labeling Effort for Structured Prediction Tasks. In Proceedings of the 20th National Conference on Artificial Intelligence - Volume 2, AAAI'05, pp. 746–751. AAAI Press, 2005. ISBN 157735236x.

Elizabeth D. Dolan and Jorge J. Moré. Benchmarking optimization software with performance profiles. Mathematical Programming, 91(2):201–213, Jan 2002. ISSN 1436-4646. doi: 10.1007/s101070100263. URL https://doi.org/10.1007/s101070100263.

Melanie Ducoffe and Frederic Precioso. Adversarial active learning for deep networks: a margin based approach. arXiv preprint arXiv:1802.09841, 2018.

R. M. Dudley. Real Analysis and Probability. Cambridge Series in Advanced Mathematics. Cambridge University Press, 2002.

Linton C Freeman. Elementary Applied Statistics: For Students in Behavioral Science. John Wiley & Sons, 1965.

Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: Active learning with expected model output changes. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), Computer Vision – ECCV 2014, pp. 562–577, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10593-2.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian Active Learning with Image Data. In Doina Precup and Yee Whye Teh (eds.), Proceedings of the 34th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research, pp. 1183–1192. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/gal17a.html.

Bin Gu, Zhou Zhai, Cheng Deng, and Heng Huang. Efficient Active Learning by Querying Discriminative and Representative Samples and Fully Exploiting Unlabeled Data. IEEE Transactions on Neural Networks and Learning Systems, 32(9):4111–4122, 2021. doi: 10.1109/TNNLS.2020.3016928.

Steve Hanneke. Theory of Disagreement-Based Active Learning. Foundations and Trends® in Machine Learning, 7(2-3):131–309, 2014. ISSN 1935-8237. doi: 10.1561/2200000037. URL http://dx.doi.org/10.1561/2200000037.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1026–1034, 2015. doi: 10.1109/ICCV.2015.123.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. arXiv preprint arXiv:1112.5745, 2011.

Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2372–2379, 2009. doi: 10.1109/CVPR.2009.5206627.

Yoon-Yeong Kim, Kyungwoo Song, JoonHo Jang, and Il-chul Moon. LADA: Look-Ahead Data Acquisition via Augmentation for Deep Active Learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), Advances in Neural Information Processing Systems, volume 34, pp. 22919–22930. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/c1b70d965ca504aa751ddb62ad69c63f-Paper.pdf.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett (eds.), Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/95323660ed2124450caaac2c46b5ed90-Paper.pdf.

Jan Kremer, Kim Steenstrup Pedersen, and Christian Igel. Active learning with support vector machines. WIREs Data Mining and Knowledge Discovery, 4(4):313–326, 2014. doi: https://doi.org/10.1002/widm.1132. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1132.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. Technical report, CS231N, Stanford University, 2015. URL http://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In Bruce W. Croft and C. J. van Rijsbergen (eds.), SIGIR '94, pp. 3–12, London, 1994. Springer London. ISBN 978-1-4471-2099-5.

Rafid Mahmood, Sanja Fidler, and Marc T Law. Low-Budget Active Learning via Wasserstein Distance: An Integer Programming Approach. In International Conference on Learning Representations, 2022. URL https://openreview.net/forum?id=v8OlxjGn23S.

Pascal Massart. Some applications of concentration inequalities to statistics. Annales de la Faculté des sciences de Toulouse : Mathématiques, Ser. 6, 9(2):245–303, 2000. URL http://archive.numdam.org/item/AFST_2000_6_9_2_245_0/.

David Mickisch, Felix Assion, Florens Greßner, Wiebke Günther, and Mariele Motta. Understanding the Decision Boundary of Deep Neural Networks: An Empirical Study. arXiv preprint arXiv:2002.01810, 2020.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011, 2011. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Vu-Linh Nguyen, Mohammad Hossein Shaker, and Eyke Hüllermeier. How to measure uncertainty in uncertainty sampling for active learning. Machine Learning, 111(1):89–122, Jan 2022. ISSN 1573-0565. doi: 10.1007/s10994-021-06003-9. URL https://doi.org/10.1007/s10994-021-06003-9.

Robert Pinsler, Jonathan Gordon, Eric Nalisnick, and José Miguel Hernández-Lobato. Bayesian Batch Active Learning as Sparse Subset Approximation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett (eds.), Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/84c2d4860a0fc27bcf854c444fb8b400-Paper.pdf.

Nicholas Roy and Andrew McCallum. Toward Optimal Active Learning through Sampling Estimation of Error Reduction. In Carla E. Brodley and Andrea Pohoreckyj Danyluk (eds.), Proceedings of the 18th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research, pp. 441–448. Morgan Kaufmann, 28 Jun –01 Jul 2001. URL http://groups.csail.mit.edu/rrg/papers/icml01.pdf.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3):211–252, Dec 2015. ISSN 1573-1405. doi: 10.1007/s11263-015-0816-y. URL https://doi.org/10.1007/s11263-015-0816-y.

Tobias Scheffer, Christian Decomain, and Stefan Wrobel. Active hidden markov models for information extraction. In Frank Hoffmann, David J. Hand, Niall Adams, Douglas Fisher, and Gabriela Guimaraes (eds.), Advances in Intelligent Data Analysis, pp. 309–318, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg. ISBN 978-3-540-44816-7.

Andrew I. Schein and Lyle H. Ungar. Active learning for logistic regression: an evaluation. Machine Learning, 68(3):235–265, Oct 2007. ISSN 1573-0565. doi: 10.1007/s10994-007-5019-5. URL https://doi.org/10.1007/s10994-007-5019-5.

Ozan Sener and Silvio Savarese. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In International Conference on Learning Representations, 2018.

Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

Burr Settles, Mark Craven, and Soumya Ray. Multiple-Instance Active Learning. In J. Platt, D. Koller, Y. Singer, and S. Roweis (eds.), Advances in Neural Information Processing Systems, volume 20, pp. 1289–1296. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper/2007/file/a1519de5b5d44b31a01de013b9b51a80-Paper.pdf.

H. S. Seung, M. Opper, and H. Sompolinsky. Query by Committee. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT '92, pp. 287–294, New York, NY, USA, 1992. Association for Computing Machinery. ISBN 089791497X. doi: 10.1145/130385.130417. URL https://doi.org/10.1145/130385.130417.

C. E. Shannon. A mathematical theory of communication. The Bell System Technical Journal, 27 (3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x.

Manali Sharma and Mustafa Bilgic. Evidence-based uncertainty sampling for active learning. Data Mining and Knowledge Discovery, 31(1):164–202, Jan 2017. ISSN 1573-756X. doi: 10.1007/s10618-016-0460-3. URL https://doi.org/10.1007/s10618-016-0460-3.

Weishi Shi and Qi Yu. Integrating Bayesian and Discriminative Sparse Kernel Machines for Multi-class Active Learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett (eds.), Advances in Neural Information Processing Systems, volume 32, pp. 2285–2294. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf.

Samrath Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational Adversarial Active Learning. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 5971–5980, 2019. doi: 10.1109/ICCV.2019.00607.

Simon Tong and Edward Chang. Support Vector Machine Active Learning for Image Retrieval. In Proceedings of the Ninth ACM International Conference on Multimedia, MULTIMEDIA '01, pp. 107–118, New York, NY, USA, 2001. Association for Computing Machinery. ISBN 1581133944. doi: 10.1145/500141.500159. URL https://doi.org/10.1145/500141.500159.

Evgenii Tsymbalov, Maxim Panov, and Alexander Shapeev. Dropout-based active learning for regression. In Wil M. P. van der Aalst, Vladimir Batagelj, Goran Glavaš, Dmitry I. Ignatov, Michael Khachay, Sergei O. Kuznetsov, Olessia Koltsova, Irina A. Lomazova, Natalia Loukachevitch, Amedeo Napoli, Alexander Panchenko, Panos M. Pardalos, Marcello Pelillo, and Andrey V. Savchenko (eds.), Analysis of Images, Social Networks and Texts, pp. 247–258, Cham, 2018. Springer International Publishing.

Evgenii Tsymbalov, Sergei Makarychev, Alexander Shapeev, and Maxim Panov. Deeper Connections between Neural Networks and Gaussian Processes Speed-up Active Learning. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, pp. 3599–3605. International Joint Conferences on Artificial Intelligence Organization, 7 2019. doi: 10.24963/ijcai.2019/499. URL https://doi.org/10.24963/ijcai.2019/499.

Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. Journal of Machine Learning Research, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

Martin J. Wainwright. High-Dimensional Statistics: A Non-Asymptotic Viewpoint. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019. doi: 10.1017/9781108627771.

Shuo Wang, Jian-Jian Wang, Xiang-Hui Gao, and Xue-Zheng Wang. Pool-based active learning based on incremental decision tree. In 2010 International Conference on Machine Learning and Cybernetics, volume 1, pp. 274–278, 2010. doi: 10.1109/ICMLC.2010.5581052.

Yazhou Yang, Xiaoqing Yin, Yang Zhao, Jun Lei, Weili Li, and Zhe Shu. Batch Mode Active Learning Based on Multi-Set Clustering. IEEE Access, 9:51452–51463, 2021. doi: 10.1109/ACCESS.2021.3053003.

Yi Yang, Zhigang Ma, Feiping Nie, Xiaojun Chang, and Alexander G. Hauptmann. Multi-Class Active Learning by Uncertainty Sampling with Diversity Maximization. International Journal of Computer Vision, 113(2):113–127, Jun 2015. ISSN 1573-1405. doi: 10.1007/s11263-014-0781-x. URL https://doi.org/10.1007/s11263-014-0781-x.

Donggeun Yoo and In So Kweon. Learning Loss for Active Learning. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 93–102, 2019. doi: 10.1109/CVPR.2019.00018.

Sergey Zagoruyko and Nikos Komodakis. Wide Residual Networks. In Edwin R. Hancock Richard C. Wilson and William A. P. Smith (eds.), Proceedings of the British Machine Vision Conference (BMVC), pp. 87.1–87.12. BMVA Press, September 2016. ISBN 1-901725-59-6. doi: 10.5244/C.30.87. URL https://dx.doi.org/10.5244/C.30.87.

Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. State-Relabeling Adversarial Active Learning. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8753–8762, 2020. doi: 10.1109/CVPR42600.2020.00878.

Xiao-Yu Zhang, Shupeng Wang, and Xiaochun Yun. Bidirectional Active Learning: A Two-Way Exploration Into Unlabeled and Labeled Data Set. IEEE Transactions on Neural Networks and Learning Systems, 26(12):3034–3044, 2015. doi: 10.1109/TNNLS.2015.2401595.

Guang Zhao, Edward Dougherty, Byung-Jun Yoon, Francis Alexander, and Xiaoning Qian. Efficient Active Learning for Gaussian Process Classification by Error Reduction. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), Advances in Neural Information Processing Systems, volume 34, pp. 9734–9746. Curran Associates, Inc., 2021a. URL https://proceedings.neurips.cc/paper/2021/file/50d2e70cdf7dd05be85e1b8df3f8ced4-Paper.pdf.

Guang Zhao, Edward Dougherty, Byung-Jun Yoon, Francis Alexander, and Xiaoning Qian. Uncertainty-aware Active Learning for Optimal Bayesian Classifier. In International Conference on Learning Representations, 2021b. URL https://openreview.net/forum?id=Mu2ZxFctAI.

A PROOF OF THEOREM 1

We consider multi-class classification, which we recall here from Section 2. Let $\mathcal{X}$ and $\mathcal{Y}$ be the instance and label space with $\mathcal{Y} = \{\boldsymbol{e}_i\}_{i=1}^C$, where $\boldsymbol{e}_i$ is the i^{th} standard basis vector of $\mathbb{R}^C$ (i.e. one-hot encoding of the label i), and $\mathcal{H}$ be the hypothesis space of $h : \mathcal{X} \to \mathcal{Y}$. We also recall some definitions: $\mathcal{H}^{\hat{h},\boldsymbol{x}_0} = \{h \in \mathcal{H} \mid h(\boldsymbol{x}_0) \neq \hat{h}(\boldsymbol{x}_0)\}$ and $\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0} := \left\{h \in \mathcal{H}_N \mid h \in \mathcal{H}^{\hat{h},\boldsymbol{x}_0}\right\}$, where for computation purpose $\mathcal{H}$ is replaced with $\mathcal{H}_N$ of cardinality N. Lastly, we use S samples $X_i \sim \mathcal{D}_{\mathcal{X}}$ to construct the Monte-Carlo estimate of $\rho(h, \hat{h})$.

Let $S(N) \to \infty$ be an arbitrary monotone increasing sequence in N such that $S(N) = \omega(\log(CN))$. By the triangle inequality, we have that

$$|L_{N,S} - L| \leq \underbrace{\left|L_{N,S} - \widetilde{L}_N\right|}_{\triangleq \Delta_1(N)} + \underbrace{\left|\widetilde{L}_N - L\right|}_{\triangleq \Delta_2(N)},$$

where we denote $\widetilde{L}_N := \min_{h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h, \hat{h})$.

$\underline{\Delta_1(N) \xrightarrow{\mathbb{P}} 0}$
By definition,

$$\begin{aligned}
\Delta_1(N) &= \left| \inf_{h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho_{S(N)}(h, \hat{h}) - \inf_{h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h, \hat{h}) \right| \\
&= \left| \inf_{h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \frac{1}{S(N)} \sum_{i=1}^{S(N)} \mathbb{I}\left[h(X_i) \neq \hat{h}(X_i)\right] - \inf_{h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \mathbb{E}_{X \sim D_{\mathcal{X}}}[\mathbb{I}[h(X) \neq \hat{h}(X)]] \right|
\end{aligned}$$

As $\Delta_1(N)$ is a difference of infimums of a sequence of functions, over a sequence of sets, we need to establish some uniform convergence-type result. This is done by invoking "general" Glivenko-Cantelli Theorem, which we recall here:

Theorem 4 (Theorem 4.2 of Wainwright (2019))**.** *Let $Y_1, \cdots, Y_n \overset{i.i.d.}{\sim} \mathbb{P}$ for some distribution $\mathbb{P}$ over $\mathcal{X}$. For any b-uniformly bounded function class $\mathcal{F}$, any positive integer $n \geq 1$ and any scalar $\delta \geq 0$, we have*

$$\sup_{f \in \mathcal{F}} \left| \frac{1}{n} \sum_{i=1}^n f(Y_i) - \mathbb{E}[f(Y)] \right| \leq 2\mathcal{R}_n(\mathcal{F}) + \delta$$

with $\mathbb{P}$-probability at least $1 - \exp\left(-\frac{n\delta^2}{8b}\right)$. Here, $\mathcal{R}_n(\mathcal{F})$ is the Rademacher complexity of $\mathcal{F}$.

In our case, $n = S(N)$, $\mathbb{P} = \mathcal{D}_{\mathcal{X}}$, $b = 1$, and

$$\mathcal{F} = \left\{ f(\boldsymbol{x}) = \mathbb{I}[h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})] \mid h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0} \right\}$$

Here, the usual Rademacher identities (e.g. see Section 4.2 of Wainwright (2019)) does not apply, as we are considering the composition of vector-valued functions and vector-input functions. But for our setting, we provide a simple yet effective upper bound on the empirical Rademacher complexity for multi-class:

Lemma 5.

$$\mathcal{R}_{S(N)}(\mathcal{F}) \leq \sqrt{\frac{2\log(CN)}{S(N)}}. \tag{7}$$

Proof. For simplicity, we denote $\mathbb{E} \triangleq \mathbb{E}_{\{X_i\}_{i=1}^{S(N)}, \boldsymbol{\sigma}}$, where the expectation is w.r.t. $X_i \sim \mathcal{D}_{\mathcal{X}}$ i.i.d., and $\boldsymbol{\sigma}$ is the $S(N)$-dimensional Rademacher variable. Also, let $l : [S(N)] \to [C]$ be the labeling function for fixed samples $\{X_i\}_{i=1}^{S(N)}$ i.e. $l(i) = \arg\max_{c \in [C]} \left[\hat{h}(X_i)\right]_c$. As $\hat{h}$ outputs one-hot

encoding for each i, $l(i)$ is unique and thus well-defined. Thus, we have that $f(X_i) = \mathbb{I}[h(X_i) \neq \hat{h}(X_i)] = 1 - h_{l(i)}(X_i)$, where we denote $h = (h_1, h_2, \cdots, h_C)$ with $h_j : \mathcal{X} \to \{0, 1\}$.

By definition,

$$\begin{aligned}
\mathcal{R}_{S(N)}(\mathcal{F}) &= \mathbb{E}\left[\sup_{f\in\mathcal{F}} \frac{1}{S(N)} \sum_{i=1}^{S(N)} \sigma_i f(X_i)\right] \\
&= \mathbb{E}\left[\sup_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \frac{1}{S(N)} \sum_{i=1}^{S(N)} \sigma_i (1 - h_{l(i)}(X_i))\right] \\
&= \mathbb{E}\left[\sup_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \frac{1}{S(N)} \sum_{i=1}^{S(N)} \sigma_i h_{l(i)}(X_i)\right] \\
&\leq \mathbb{E}\left[\sup_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}, l\in[C]} \frac{1}{S(N)} \sum_{i=1}^{S(N)} \sigma_i h_l(X_i)\right] \\
&= \mathcal{R}_{S(N)}\left(\left\{h_l \mid h \in \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}, l \in [C]\right\}\right) \\
&\leq \sqrt{\frac{2\log(CN)}{S(N)}},
\end{aligned}$$

where the last inequality follows from Massart's Lemma (Massart, 2000) and the fact that $\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}$ is a finite set of cardinality at most N. □

Choosing $\delta = \sqrt{\frac{8\log(CN)}{S(N)}}$, we have that with probability at least $1 - \frac{1}{CN}$,

$$\sup_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \left|\rho_{S(N)}(h, \hat{h}) - \rho(h, \hat{h})\right| \leq 4\sqrt{\frac{2\log(CN)}{S(N)}}. \tag{8}$$

The following lemma completes the proof of the first part:

Lemma 6.

$$\inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho_{S(N)}(h, \hat{h}) - \inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h, \hat{h}) \xrightarrow[N\to\infty]{\mathbb{P}} 0$$

Proof. Let $\delta > 0$ be arbitrary, and choose any N such that $N > \frac{1}{C\delta}$ and $4\sqrt{\frac{2\log(CN)}{S(N)}} < \delta$.

From Eqn. (8) we have: with probability at least $1 - \delta$, $\sup_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \left|\rho_{S(N)}(h, \hat{h}) - \rho(h, \hat{h})\right| < \delta$.

First, we choose a sequence $\{h_j\} \subset \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}$ such that $\rho(h_j, \hat{h}) \to \inf_{h\in\mathcal{H}_N} \rho(h, \hat{h})$ for $j \to \infty$. Then,

$$\inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h, \hat{h}) = \lim_{j\to\infty} \rho(h_j, \hat{h}) > \lim_{j\to\infty} \rho_{S(N)}(h_j, \hat{h}) - \delta > \inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho_{S(N)}(h, \hat{h}) - \delta.$$

Similarly, by choosing a sequence $\{g_j\} \subset \mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}$ such that $\rho_{S(N)}(g_j, \hat{h}) \to \inf_{h\in\mathcal{H}_N} \rho_{S(N)}(h, \hat{h})$, we have that

$$\inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho_{S(N)}(h, \hat{h}) = \lim_{j\to\infty} \rho_{S(N)}(g_j, \hat{h}) > \lim_{j\to\infty} \rho(g_j, \hat{h}) - \delta > \inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h, \hat{h}) - \delta.$$

Combining them, we have that for any $\delta > 0$, there exists some $M(\delta)$ such that for any $N > M(\delta)$,

$$\mathbb{P}\left[\left|\inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h, \hat{h}) - \inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho_{S(N)}(h, \hat{h})\right| < \delta\right] \geq 1 - \delta.$$

Denoting $G_N \triangleq \inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho(h,\hat{h}) - \inf_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} \rho_{S(N)}(h,\hat{h})$, above is equivalent to

$$\forall\delta>0,\ \exists M(\delta) \text{ s.t. } \forall N\geq M(\delta),\ d_{KF}(G_N,0)\leq\delta,$$

where $d_{KF}(X,Y)=\inf\{\delta\geq 0 \mid \mathbb{P}[|X-Y|\geq\delta]\leq\delta\}$ is the *Ky-Fan metric*, which induces a metric structure on the given probability space with the convergence in probability (see Section 9.2 of Dudley (2002)). In other words, as $d_{KF}(G_N,0)\to 0$, we conclude that $G_N \xrightarrow{\mathbb{P}} 0$. □

$\lim_{N\to\infty}\mathbb{P}[\Delta_2(N)\leq\varepsilon]=1$

Let $\gamma>0$ be arbitrary, and denote

$$Z_N^{\varepsilon} := \inf_{\substack{h^*\in\mathcal{H}^{\hat{h},\boldsymbol{x}_0}\\ \rho(h^*,\hat{h})-\rho^\star<\varepsilon}} \min_{h\in\mathcal{H}_N} d_{\mathcal{H}}(h^*,h).$$

In this section, any convergence is w.r.t. the limit $N\to\infty$ with ε fixed. It's easy to see that Z_N^{ε} is monotone decreasing i.e. $Z_1^{\varepsilon}\geq Z_2^{\varepsilon}\geq\cdots$. Defining the events $E_N:=\{Z_N^{\varepsilon}<\gamma/B\}$, we have that $E_1\subseteq E_2\subseteq\cdots$. From our assumption and the fact that probability measure is continuous along any monotone sequence of events, we have that

$$1=\lim_{N\to\infty}\mathbb{P}[E_N]=\mathbb{P}\left[\lim_{N\to\infty}E_N\right]=\mathbb{P}\left[\lim_{N\to\infty}\inf_{\substack{h^*\in\mathcal{H}^{\hat{h},\boldsymbol{x}_0}\\ \rho(h^*,\hat{h})-L(\hat{h},\boldsymbol{x}_0)<\varepsilon}}\min_{h\in\mathcal{H}_N} d_{\mathcal{H}}(h^*,h)\leq\frac{\gamma}{B}\right].$$

This implies that w.p. 1 the following holds: for any $\zeta>0$ there exists N_0 s.t. for all $N\geq N_0$, there exists $h\in\mathcal{H}_N$ and $h^*\in\mathcal{H}^{\hat{h},\boldsymbol{x}_0}$ with $\rho(h^*,\hat{h})-L(\hat{h},\boldsymbol{x}_0)<\varepsilon$ s.t. $d_{\mathcal{H}}(h^*,h)\leq\frac{\gamma+\zeta}{B}$.

Thus,

$$\begin{aligned}
\Delta_2(N) &= \left|\min_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}}\rho(h,\hat{h})-\inf_{h\in\mathcal{H}^{\hat{h},\boldsymbol{x}_0}}\rho(h,\hat{h})\right|\\
&\leq \left|\min_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}}\rho(h,\hat{h})-\rho(h^*,\hat{h})\right|+\left|\rho(h^*,\hat{h})-\rho^\star\right|\\
&\leq \left|\min_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}}\rho(h^*,h)\right|+\varepsilon\\
&\leq B\min_{h\in\mathcal{H}_N^{\hat{h},\boldsymbol{x}_0}} d_{\mathcal{H}}(h,h^*)+\varepsilon\leq\gamma+\zeta+\varepsilon.
\end{aligned}$$

As γ and ζ were arbitrary, we conclude that w.p. 1, $\lim_{N\to\infty}\Delta_2(N)\leq\varepsilon$, and we are done.

B Theoretical Verifications of Intuitions

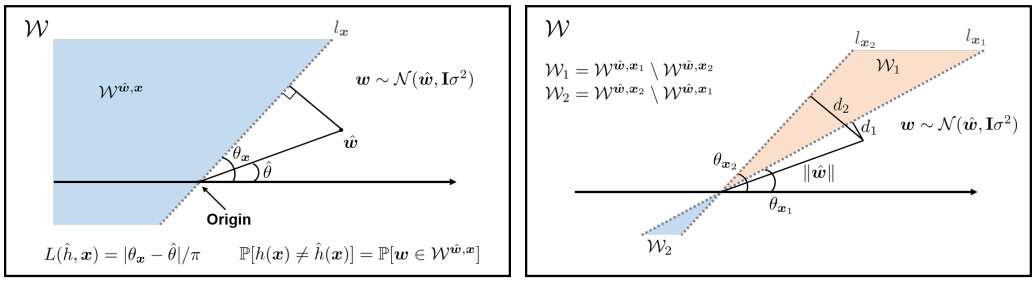
Here, we consider two-dimensional binary classification with a set of linear classifiers, $\mathcal{H}=\{h_{\boldsymbol{w}}: h_{\boldsymbol{w}}(x)=\operatorname{sgn}(\boldsymbol{x}^{\mathsf{T}}\boldsymbol{w}),\boldsymbol{w}\in\mathbb{R}^2\}$. We assume that $\boldsymbol{x}$ is uniformly distributed on $\mathcal{X}=\{\boldsymbol{x}:\|\boldsymbol{x}\|\leq 1\}\subset\mathbb{R}^2$. As we parametrize $\mathcal{H}$ using $\boldsymbol{w}$, following our Algorithm 1, we set $\mathcal{D}_{\mathcal{H}}(\hat{h})=\mathcal{N}(\hat{\boldsymbol{w}},\boldsymbol{I}\sigma^2)$, where $\hat{h}=h_{\hat{\boldsymbol{w}}}$. For simplicity, we omit the dependency on $\boldsymbol{w}$ and $\hat{\boldsymbol{w}}$.

B.1 LDM as an Uncertainty Measure

Recall from Section 2.1 that the intuition behind a sample with a small LDM indicates that even a small perturbation in the predictor can alter sample prediction. We theoretically prove this intuition

Proposition 7. *Suppose that $h=h_{\boldsymbol{w}}$ is sampled with $\boldsymbol{w}\sim\mathcal{N}(\hat{\boldsymbol{w}},\mathbf{I}\sigma^2)$. Then,*

$$L(\hat{h},\boldsymbol{x}_1)<L(\hat{h},\boldsymbol{x}_2)\iff\mathbb{P}[h(\boldsymbol{x}_1)\neq\hat{h}(\boldsymbol{x}_1)]>\mathbb{P}[h(\boldsymbol{x}_2)\neq\hat{h}(\boldsymbol{x}_2)].$$



(a) (b)

Figure 6: Proof of Proposition 7.

Proof of Proposition 7. One important observation is that by the duality between $\boldsymbol{w}$ and $\boldsymbol{x}$ (Tong & Chang, 2001), in $\mathbb{R}^2$, $\boldsymbol{w}$ is a point and $\boldsymbol{x}$ is represented by the hyperplane, $l_{\boldsymbol{x}} = \{\boldsymbol{w} \in \mathbb{R}^2 : \text{sgn}(\boldsymbol{x}^\mathsf{T}\boldsymbol{w}) = 0\}$. Suppose that h is sampled with $\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma^2)$, and let $\hat{\theta}$ be the angle of $\hat{\boldsymbol{w}}$, $\theta_{\boldsymbol{x}}$ be the angle between $l_{\boldsymbol{x}}$ and positive x-axis, and $\mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}}$ be the half-plane divided by $l_{\boldsymbol{x}}$ which does not contain $\hat{\boldsymbol{w}}$:

$$\mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}} = \{\boldsymbol{w}' \in \mathcal{W} \mid h'(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})\}$$

as in Figure 6a. Then, $L(\hat{h}, \boldsymbol{x}) = |\theta_{\boldsymbol{x}} - \hat{\theta}|/\pi$ and $\mathbb{P}[h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})] = \mathbb{P}[\boldsymbol{w} \in \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}}]$.

Let d_1, d_2 be the distances between $\hat{\boldsymbol{w}}$ and $l_{\boldsymbol{x}_1}, l_{\boldsymbol{x}_2}$ respectively, and

$$\mathcal{W}_1 = \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1} \setminus \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2}, \quad \mathcal{W}_2 = \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2} \setminus \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1}$$

as in Figure 6b. Suppose that $d_1 < d_2$, then $|\theta_{\boldsymbol{x}_1} - \hat{\theta}| < |\theta_{\boldsymbol{x}_2} - \hat{\theta}|$ since $d_i = \|\hat{\boldsymbol{w}}\| \sin|\theta_{\boldsymbol{x}_i} - \hat{\theta}|$, and

$$\mathbb{P}[\boldsymbol{w} \in \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1}] - \mathbb{P}[\boldsymbol{w} \in \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2}] = \mathbb{P}[\boldsymbol{w} \in \mathcal{W}_1] - \mathbb{P}[\boldsymbol{w} \in \mathcal{W}_2] > 0$$

by the followings:

$$\mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1} = \mathcal{W}_1 \cup (\mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1} \cap \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2}), \quad \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2} = \mathcal{W}_2 \cup (\mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1} \cap \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2})$$

where $\mathcal{W}_1, \mathcal{W}_2$, and $\mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_1} \cap \mathcal{W}^{\hat{\boldsymbol{w}},\boldsymbol{x}_2}$ are disjoint. Note that $\mathcal{W}_1$ and $\mathcal{W}_2$ are one-to-one mapped by the symmetry at origin, but the probabilities are different by the biased location of $\hat{w}$, i.e, $\phi(\boldsymbol{w}_1|\hat{\boldsymbol{w}}, \sigma^2) > \phi(\boldsymbol{w}_2|\hat{\boldsymbol{w}}, \sigma^2)$ for all pairs of $(\boldsymbol{w}_1, \boldsymbol{w}_2) \in \mathcal{W}_1 \times \mathcal{W}_2$ that are symmetric at the origin. Here $\phi(\cdot|\hat{\boldsymbol{w}}, \sigma^2)$ is the probability density function of the bivariate normal distribution with mean $\hat{\boldsymbol{w}}$ and covariance $\mathbf{I}\sigma^2$. Thus,

$$L(\hat{h}, \boldsymbol{x}_1) < L(\hat{h}, \boldsymbol{x}_2) \iff d_1 < d_2 \iff \mathbb{P}[h(\boldsymbol{x}_1) \neq \hat{h}(\boldsymbol{x}_1)] > \mathbb{P}[h(\boldsymbol{x}_2) \neq \hat{h}(\boldsymbol{x}_2)].$$

□

B.2 Varying σ^2 in Algorithm 1

Recall from Section 2.2 that the intuition behind using multiple σ^2 was that it controls the trade-off between the probability of obtaining a hypothesis with a different prediction than that of $\hat{h}$ and the scale of $\rho_{\mathcal{S}}(h, \hat{h})$. Theoretically, we show the following for two-dimensional binary classification with linear classifiers:

Proposition 8. *Suppose that h is sampled with $\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma^2)$ where $\boldsymbol{w}, \hat{\boldsymbol{w}} \in \mathcal{W}$ are parameters of $h, \hat{h}$ respectively, then $\mathbb{E}[\rho(h, \hat{h})]$ is continuous and strictly increasing with σ.*

We now empirically show that this intuition holds for general deep learning architectures. Figure 7 shows the relationship between $\mathbb{E}[\rho_{\mathcal{S}}(h, \hat{h})]$ and $\log\sigma$ for MNIST, CIFAR10, SVNH, CIFAR100, Tiny ImageNet, and FOOD101 datasets where h is sampled with $\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma^2)$. The $\mathbb{E}[\rho_{\mathcal{S}}(h, \hat{h})]$ is monotonically increasing with $\log\sigma$ in all experimental settings.

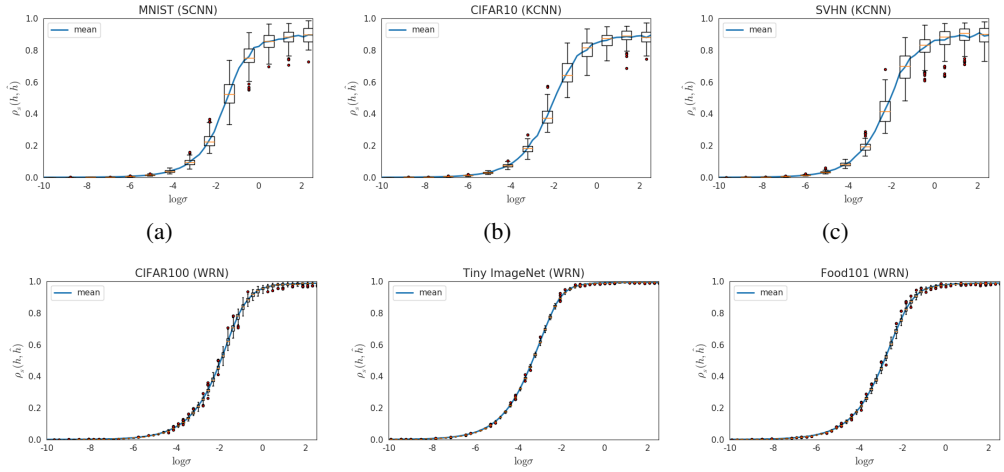


Figure 7: The relationship between the disagree metric and perturbation strength for MNIST (a), CIFAR10 (b), SVHN (c), CIFAR100 (d), Tiny ImageNet (e), and FOOD101 (f) datasets. $\mathbb{E}[\rho_S(h,\hat{h})]$ is monotonically increasing with the perturbation strength in all experimental settings.

Proof of Proposition 8. By the duality between $\boldsymbol{w}$ and $\boldsymbol{x}$, in $\mathcal{W}$, $\boldsymbol{w}$ is a point and $\boldsymbol{x}$ is represented by the hyperplane, $l_{\boldsymbol{x}} = \{\boldsymbol{w} \in \mathcal{W} : \operatorname{sgn}(\boldsymbol{x}^\mathsf{T}\boldsymbol{w}) = 0\}$. Let h be a sampled hypothesis with $\boldsymbol{w} \sim \mathcal{N}(\hat{\boldsymbol{w}}, \mathbf{I}\sigma^2)$, $\hat{\theta}$ be the angle of $\hat{\boldsymbol{w}} = (\hat{w}_1, \hat{w}_2)^\mathsf{T}$, i.e., $\tan\hat{\theta} = \hat{w}_2/\hat{w}_1$, θ be the angle of $\boldsymbol{w} = (w_1, w_2)^\mathsf{T}$, i.e., $\tan\theta = w_2/w_1$, and $\theta_{\boldsymbol{x}}$ be the angle between $l_{\boldsymbol{x}}$ and positive x-axis. Here, $\theta, \theta_{\boldsymbol{x}} \in [-\pi + \hat{\theta}, \pi + \hat{\theta}]$ in convenience. When $\theta_{\boldsymbol{x}}$ or $\pi + \theta_{\boldsymbol{x}}$ is between θ and $\hat{\theta}$, $h(\boldsymbol{x}) \neq \hat{h}(\boldsymbol{x})$, otherwise $h(\boldsymbol{x}) = \hat{h}(\boldsymbol{x})$. Thus, $\rho(h, \hat{h}) = |\theta - \hat{\theta}|/\pi$.

Using Box-Muller transform (Box & Muller, 1958), $\boldsymbol{w}$ can be generated by

$$w_1 = \hat{w}_1 + \sigma\sqrt{-2\log u}\cos(2\pi v),\ w_2 = \hat{w}_2 + \sigma\sqrt{-2\log u}\sin(2\pi v)$$

where u and v are independent uniform random variables on $[0, 1]$. Then, $\|\boldsymbol{w} - \hat{\boldsymbol{w}}\| = \sigma\sqrt{-2\log u}$ and $(w_2 - \hat{w}_2)/(w_1 - \hat{w}_1) = \tan(2\pi v)$, i.e., the angle of $\boldsymbol{w} - \hat{\boldsymbol{w}}$ is $2\pi v$. Here,

$$\|\hat{\boldsymbol{w}}\|\sin(\theta - \hat{\theta}) = \sigma\sqrt{-2\log u}\sin(2\pi v - \theta) \tag{9}$$

by using the perpendicular line from $\hat{\boldsymbol{w}}$ to the line passing through the origin and $\boldsymbol{w}$ (see the Figure 8 for its geometry), and Eq. 9 is satisfied for all θ. For given u and v, θ is continuous and the derivative of θ with respect to σ is

$$\frac{d\theta}{d\sigma} = \frac{\sqrt{-2\log u}\sin^2(2\pi v - \theta)}{\|\hat{\boldsymbol{w}}\|\sin(2\pi v - \hat{\theta})},$$

thus

$$\begin{cases} \frac{d\theta}{d\sigma} > 0, & v \in (\frac{\hat{\theta}}{2\pi}, \frac{\pi+\hat{\theta}}{2\pi}) \\ \frac{d\theta}{d\sigma} < 0, & v \in [0,1] \setminus [\frac{\hat{\theta}}{2\pi}, \frac{\pi+\hat{\theta}}{2\pi}] \end{cases}.$$

Then,

$$\frac{d\rho(h,\hat{h})}{d\sigma} = \operatorname{sgn}(\theta - \hat{\theta})\frac{d\theta}{d\sigma} > 0 \quad \text{where} \quad v \notin \left\{\frac{\hat{\theta}}{2\pi}, \frac{\pi+\hat{\theta}}{2\pi}\right\}.$$

Thus, $\rho(h, \hat{h})$ is continuous and strictly increasing with σ when $v \neq \hat{\theta}/2\pi$ and $v \neq (\pi + \hat{\theta})/2\pi$. Let $\rho(h, \hat{h}) = g(\sigma, u, v)$, then $\mathbb{E}[\rho(h, \hat{h})] = \int g(\sigma, u, v)f(u)f(v)dudv$ where $h(u) = \mathbb{I}[0 < u < 1]$ and $h(v) = \mathbb{I}[0 < v < 1]$. For $0 < \sigma_1 < \sigma_2$,

$$\mathbb{E}[g(\sigma_2, u, v)] - \mathbb{E}[g(\sigma_1, u, v)] = \int \left(g(\sigma_2, u, v) - g(\sigma_1, u, v)\right) f(u)f(v)dudv > 0$$

□

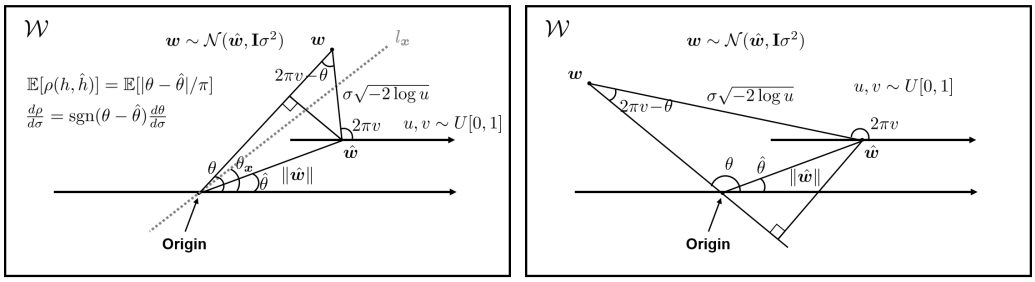


Figure 8: Proof of Proposition 8.

C Datasets, Networks and Experimental Settings

C.1 Benchmark Datasets

MNIST (Lecun et al., 1998) is a handwritten digit dataset which has $60,000$ training samples and $10,000$ test samples in 10 classes. Each sample is a black and white image and 28×28 in size.

CIFAR10 and **CIFAR100** (Krizhevsky, 2009) are tiny image datasets which has $50,000$ training samples and $10,000$ test samples in 10 and 100 classes respectively. Each sample is a color image and 32×32 in size.

SVHN (Netzer et al., 2011) is a real-world digit dataset which has $73,257$ training samples and $26,032$ test samples in 10 classes. Each sample is a color image and 32×32 in size.

Tiny ImageNet (Le & Yang, 2015) is a subset of the ILSVRC (Russakovsky et al., 2015) dataset which has $100,000$ samples in 200 classes. Each sample is a color image and 64×64 in size. In experiments, Tiny ImageNet is split into two parts: $90,000$ samples for training and $10,000$ samples for test.

Food101 (Bossard et al., 2014) is a fine-grained food image dataset which has $75,750$ training samples and $25,250$ test samples in 101 classes. Each sample is a color image and resized to 75×75.

All datasets are used without any preprocessing of images.

C.2 Deep Networks

S-CNN (Chollet et al., 2015) consists of [3×3×32 conv $-$ 3×3× 64 conv $-$ 2×2 maxpool $-$ dropout (0.25) $-$ 128 dense $-$ dropout (0.5) $-$ # class dense $-$ softmax] layers, and it is used for MNIST.

K-CNN (Chollet et al., 2015) consists of [two 3×3×32 conv $-$ 2×2 maxpool - dropout (0.25) $-$ two 3×3×64 conv $-$ 2×2 maxpool - dropout (0.25) $-$ 512 dense $-$ dropout (0.5) $-$ # class dense - softmax] layers, and it is used for CIFAR10, SVHN, and CIFAR100.

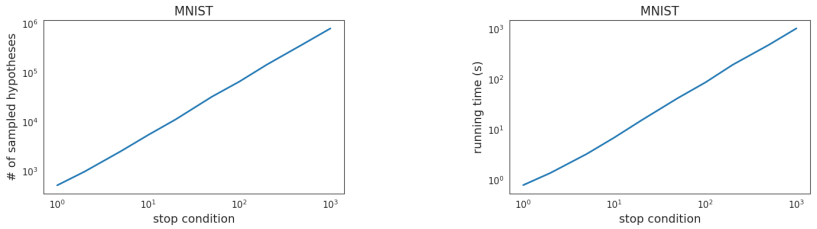
WRN-16-8 (Zagoruyko & Komodakis, 2016) is a wide residual network that has 16 convolutional layers and a widening factor 8, and it is used for CIFAR100 and Tiny ImageNet.

C.3 Experimental Settings

Training settings regarding a number of epochs, batch size, optimizer, learning rate, and learning rate schedule are summarized in Table 3. The model parameters are initialized with He normal initialization (He et al., 2015) for all experimental settings. For all experiments, the initial labeled samples for each repetition are randomly sampled according to the distribution of the training set.

Table 3: Settings for training.

Dataset	Model	Epochs	Batch size	Optimizer	Learning Rate	Learning Rate Schedule ×decay [epoch schedule]
MNIST	S-CNN	50	32	Adam	0.001	-
CIFAR10	K-CNN	150	64	Adam	0.0001	-
SVHN	K-CNN	150	64	Adam	0.0001	-
CIFAR100	WRN-16-8	100	128	Nesterov	0.05	×0.2 [60, 80]
Tiny ImageNet	WRN-16-8	200	128	Nesterov	0.1	×0.2 [60, 120, 160]
FOOD101	WRN-16-8	200	128	Nesterov	0.1	×0.2 [60, 120, 160]



(a) (b)

Figure 9: Th number of sampled hypotheses (a) and the runtime (b) w.r.t. the stop condition for evaluating empirical LDM using Algorithm 1 of pool data (MNIST). Both are monotonically increasing with the stop condition.

D Ablation Study

D.1 Stop Condition

Figure 9 shows the number of sampled hypotheses and the runtime for evaluating empirical LDM of pool data in MNIST, as the stop condition varies. Both are monotonically increasing with the stop condition, which implies that large stop condition, although it may result in empirical LDM closer to the true LDM, potentially requires huge computing power, which is very inefficient.

D.2 Need for Diversity in Batch Active Learning

In the batch selection setting where samples are selected in the order of the smallest empirical LDM, there may be some overlap in the information of the selected samples' features, thereby reducing the performance. In other words, querying unlabeled samples with the smallest LDMs may not lead to the best performance. To confirm this phenomenon, we query the k^{th} batch of size $q = 20$ for $k \in [50]$ from MNIST sorted in ascending order of LDM and compare the improvements in test accuracy. As shown in Figure 10a where 100 samples are labeled, the smallest LDM leads to the best performance, whereas in Figure 10d where 300 samples are labeled, the smallest LDM does *not* lead to the best performance. To see why this is the case, we've plotted the t-SNE plots (van der Maaten & Hinton, 2008) of the first and eighth batches for each case. In the first case, as shown in Figure 10b–10c, the samples of the first and eighth batches are all spread out, so there is no overlap of information between the samples in each batch; taking a closer look, it seems that smaller LDM leads to the samples being more spread out. However, in the second case, as shown in Figure 10e–10f, the samples of the first batch are close to one another i.e. there is a significant overlap of information between the samples in that batch. Surprisingly, this is not the case for the eighth batch, which consists of samples of larger LDMs. From this, we conclude that in the batch selection setting, even when using the LDM-based approach, diversity should be taken into consideration.

D.3 Comparing with Other Uncertainty Methods with Seeding

To clarify whether the gains of LDM-S over the standard uncertainty methods are due to weighted seeding or due to the superiority of LDM, the performance of LDM-S is compared with those meth-

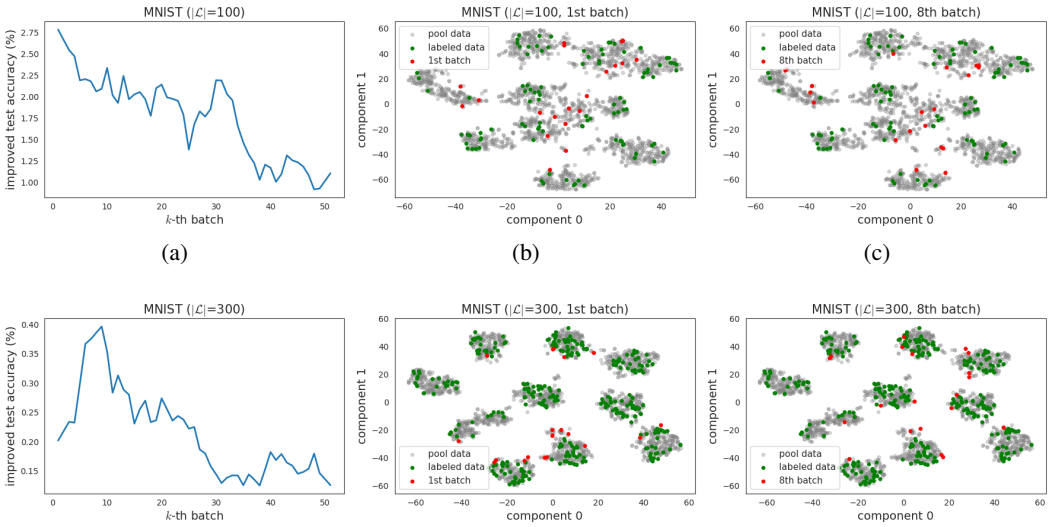


Figure 10: The improved test accuracy by labeling the k^{th} batch of size q from pool data sorted in ascending order of LDM when the number of labeled samples is 100 (a) or 300 (d), and t-SNE plots of the first and eighth batches for each case (b-c, e-f) on MNIST. There exists a case where the smallest LDM does not lead to the best performance due to the overlap of information in the batch.

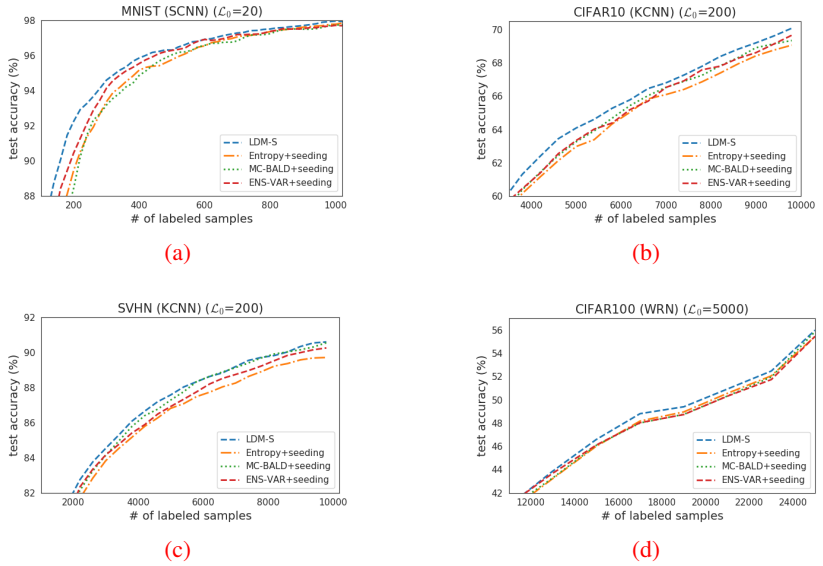


Figure 11: The performance comparison of LDM-S with the standard uncertainty methods to which weighted seeding is applied on MNIST (a), CIFAR10 (b), SVHN (c), and CIFAR100 (d). Even if weighted seeding is applied to the standard uncertainty methods, LDM-S performs better.

ods to which weighted seeding is applied. Figure 11 shows the test accuracy with respect to the number of labeled samples on MNIST, CIFAR10, SVHN, and CIFAR100 datasets. Overall, even when weighted seeding is applied to the standard uncertainty methods, LDM-S still performs better on all datasets. Therefore, the performance gains of LDM-S can be attributed to LDM's superiority over the standard uncertainty measures.

Table 4: The mean of runtime (min) for each algorithm and each dataset. The value in parentheses is the ratio of runtime for each algorithm to that of Entropy. We observe that LDM-S operates as fast as Entropy on almost all datasets.

	MNIST	CIFAR10	SVHN	CIFAR100	T. ImageNet	FOOD101
LDM-S	17.6 (170%)	106 (106%)	69 (106%)	406 (103%)	4,609 (103%)	4,465 (103%)
Entropy	10.4 (100%)	100 (100%)	65 (100%)	395 (100%)	4,466 (100%)	4,340 (100%)
Coreset	11.3 (109%)	106 (106%)	71 (109%)	430 (109%)	4,707 (105%)	4,476 (103%)
MC-BALD	12.1 (117%)	108 (108%)	106 (162%)	448 (113%)	4,829 (108%)	4,727 (109%)
ENS-VarR	49.6 (478%)	496 (496%)	325 (499%)	1,952 (494%)	19,356 (433%)	18,903 (436%)
BADGE	12.5 (120%)	102 (102%)	70 (108%)	445 (113%)	5,152 (115%)	4,704 (108%)

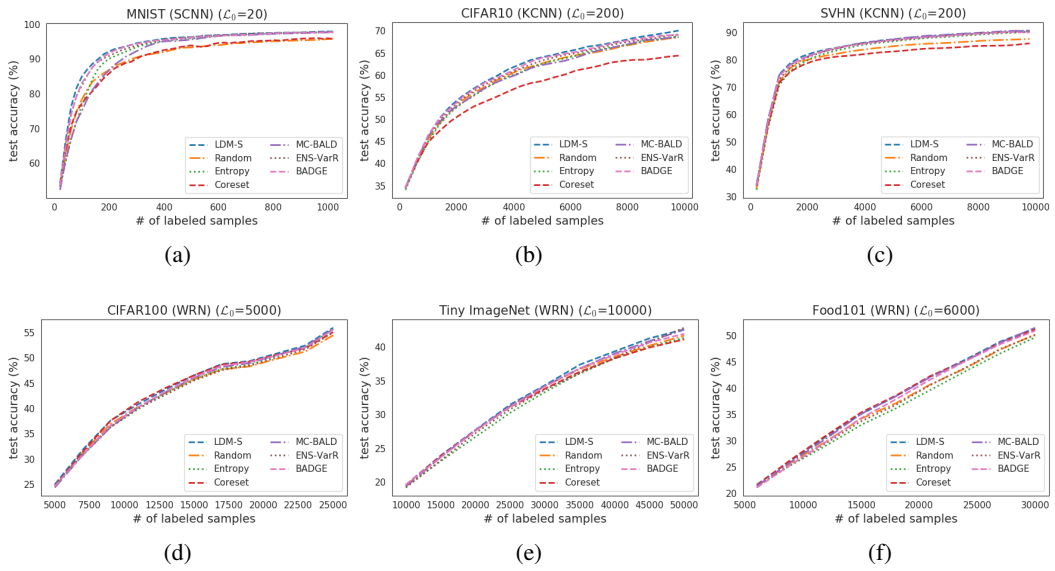


Figure 12: The test accuracy with respect to the number of labeled samples from initial to final step for all experimental settings.

E Additional Results

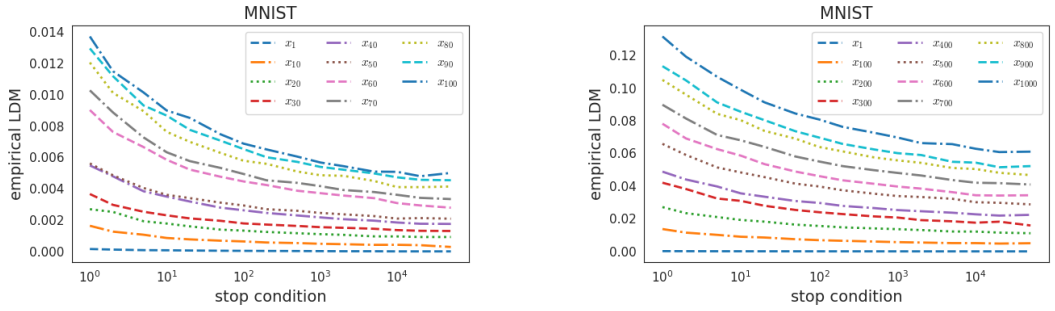
E.1 Runtime

Table 4 present the mean of runtime (min) to perform active learning for each algorithm and each dataset. The value in parentheses is the ratio of the runtime for each algorithm to that of Entropy. Overall, the runtime of LDM-S increased by only $3 \sim 6\%$ compared to Entropy, and it is a little faster than MC-BALD, Coreset, and BADGE. ENS-VarR requires about 5 times more computational load than Entropy, as all networks in the ensemble are individually trained in that method. The only exception is MNIST in which the runtime of LDM-S increases by 70% compared to Entropy; this can be attributed to the relatively small training time compared to acquisition time, which is due to the simplicity of MNIST.

E.2 Results for Test Accuracy

Figure 12 shows the test accuracy with respect to the number of labeled samples from initial to final step for all experimental settings.

F EXTENDED RESULTS FOR EMPIRICAL LDMS



(a) (b)

Figure 13: The extended results for empirical LDMs of MNIST samples with a four-layered CNN. The LDM of each sample tends to converge to a specific value.

Additional experiments are conducted for evaluating empirical LDMSs of MNIST samples with $s = \{2000, 5000, 10000, 20000, 50000\}$. Due to the limitation of computing time, the number of repetitions is to 10 or less, but the results are hardly fluctuated since the variance decreases as s increases. Figure 13 shows that the LDM of each sample tends to converge to a specific value.