Balancing Feature Similarity and Label Variability for Optimal Size-Aware One-shot Subset Selection

Abhinab Acharya [1] **Dayou Yu** [1] **Qi Yu** [1] **Xumin Liu** [1]

Abstract

Subset or core-set selection offers a data-efficient way for training deep learning models. One-shot subset selection poses additional challenges as subset selection is only performed once and full set data become unavailable after the selection. However, most existing methods tend to choose either diverse or difficult data samples, which fail to faithfully represent the joint data distribution that is comprised of both feature and label information. The selection is also performed independently from the subset size, which plays an essential role in choosing what types of samples. To address this critical gap, we propose to conduct Feature similarity and Label variability Balanced One-shot Subset Selection (BOSS), aiming to construct an optimal size-aware subset for data-efficient deep learning. We show that a novel balanced core-set loss bound theoretically justifies the need to simultaneously consider both diversity and difficulty to form an optimal subset. It also reveals how the subset size influences the bound. We further connect the inaccessible bound to a practical surrogate target which is tailored to subset sizes and varying levels of overall difficulty. We design a novel Beta-scoring importance function to delicately control the optimal balance of diversity and difficulty. Comprehensive experiments conducted on both synthetic and real data justify the important theoretical properties and demonstrate the superior performance of BOSS as compared with the competitive baselines.

1. Introduction

The success of deep learning (Brown et al., 2020; Liu et al., 2019; Ramesh et al., 2021; Dosovitskiy et al., 2021; Tan & Le, 2019; Chen et al., 2020) comes at the cost of large amounts of data and increased resource consumption (Schwartz et al., 2020; Strubell et al., 2019). Subset or core-set selection aims to find candidate data points from a large pool of data such that the model trained on the subset has comparable performance to that of the model trained on the full set (Feldman, 2020), which in turn helps decrease the resources consumed by training on large amounts of data.

Intuitively, subsets can be chosen dynamically during each training epoch (Mirzasoleiman et al., 2020; Killamsetty et al., 2021b;a; Pooladzandi et al., 2022). However, the selection algorithm is usually time-consuming and can significantly increase the overall training duration (Shin et al., 2023). Such a process also requires a forward pass through the entire dataset each time a subset is chosen, which incurs a high cost for a large dataset. Additionally, there are some important applications such as continual learning (Nguyen et al., 2018), where the full set data is only available once such that the dynamic subset selection is not possible. Thus, contrary to the dynamic selection, *one-shot subset selection* only picks the subset once and uses that subset for the entire training process (Zheng et al., 2023; Paul et al., 2021; Feldman & Zhang, 2020; Sorscher et al., 2022). While it may still be essential to initialize a model using the full dataset for a few epochs to obtain the training dynamics employed for subset selection, one-shot subset selection offers the advantage that the time required for this selection is accounted for only once. Furthermore, we are not required to store the large full-set data after selecting the subset.

Subset selection have been used for classical problems such as regression (Madigan et al., 2002), classification (Tsang et al., 2005), and clustering (Har-Peled & Kushal, 2005). Recent works have started exploring applications of core-set selection for data-efficient deep learning (Guo et al., 2022; Wan et al., 2022; Killamsetty et al., 2021c). Two categories of methods have been explored to incorporate the most important examples into the selected subset, including 1) diversity based, which selects a diverse set of samples to cover the entire feature (or gradient) space (Mirzasoleiman et al., 2020; Killamsetty et al., 2021a;b; Pooladzandi et al., 2022; Shin et al., 2023; Welling, 2009; Agarwal et al., 2020; Sener & Savarese, 2018), and 2) difficulty-based, which selects the most difficult samples to best characterize the

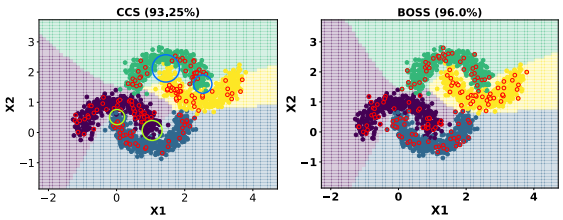
[1]Rochester Institute of Technology, Rochester, NY, USA. Correspondence to: Xumin Liu <xumin.liu@rit.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

decision boundary (Toneva et al., 2019; Feldman & Zhang, 2020; Paul et al., 2021; Sorscher et al., 2022). More specifically, diversity-based methods leverage the facility location objective (Farahani & Hekmatfar, 2009) to select the optimal subset such that the distance between the subset and full set in the feature (Sener & Savarese, 2018; Welling, 2009; Agarwal et al., 2020) or gradient space (Mirzasoleiman et al., 2020; Killamsetty et al., 2021a; Pooladzandi et al., 2022; Shin et al., 2023) is minimized. In contrast, difficulty-based methods (Toneva et al., 2019; Paul et al., 2021) score examples based on a difficulty metric where a high score corresponds to a higher difficulty. (Paul et al., 2021) shows that by removing the easier examples, a large portion (*i.e.,* 25%–50%) of a full dataset can be pruned without obviously compromising the model's generalization performance.

However, the existing methods as described above are fundamentally limited as they are inadequate to select the optimal subset of samples. This is due to the fact that the selection criterion does not align with the ultimate goal of subset selection, which is to represent a joint distribution $P(\mathbf{x}, \mathbf{y})$ (as instantiated by a full set) using a small subset of data samples. Consequently, solely relying on the feature (*i.e.,* $\mathbf{x}$) or the label side (*i.e.,* $\mathbf{y}$) will lead to a suboptimal selection result. Furthermore, the subset size also plays a crucial role in determining what types of samples should be selected, which has been largely ignored by most existing works. Some recent efforts try to explore different difficulty metrics or new sampling strategies to improve one-shot selection performance (Zheng et al., 2023; Xia et al., 2023). Nevertheless, a principled way is still lacking to properly balance diversity and difficulty given a subset size for choosing a subset that can faithfully represent the underlying joint distribution. As Figure 1 (a) shows, the subset chosen by CCS (Zheng et al., 2023), as highlighted in red, misses some critical regions in the full set, as annotated by the circles, leading to a suboptimal subset with a lower generalization performance (due to a less accurate decision boundary).

To address the key limitations of existing approaches, we propose to perform Feature similarity and Label variability **B**alanced **O**ne-shot **S**ubset **S**election (BOSS) aiming to construct an optimal subset to achieve data-efficient deep learning. BOSS performs subset selection guided by a balanced core-set loss bound that reveals an important trade-off between feature similarity (*i.e.,* diversity) and label variability (*i.e.,* difficulty). In particular, the balanced loss bound is comprised of two key components as a natural result of the joint impact from the feature and label sides, respectively. This theoretical result further confirms the need to properly model the joint data distribution in subset selection as solely relying on the feature or label sides will result in a significantly loose loss bound that will compromise the learning process. Furthermore, the novel loss bound also uncovers important relationship between the type of data samples to be selected (*i.e.,* diverse or difficult) and the size of the subset (as determined by the available computing budget). For a small subset size, focus should be placed on representative (*i.e.,* diverse) samples as feature similarity will dominate the bound. As the size increases, the large label variability from certain (*i.e.,* difficult) regions in the joint distribution will contribute more significantly to the overall loss bound. This will force the selection of samples from these regions so that the decision-boundary can be further refined to reduce the label loss. Since directly minimizing the bound in infeasible, we connect the inaccessible bound to a practical surrogate target which is tailored to subset sizes and varying levels of overall difficulty. Building on this connection, we design a novel Beta-scoring importance function to delicately control the optimal balance of diversity and difficulty.



(a) (b)

Figure 1: CCS (a) Vs. BOSS (b): Decision boundary learned using the selected subset shown in red circles by CCS and BOSS, where the subset is 10% of the full set. The recent methods like CCS miss critical regions as highlighted by green and blue circles. They do not consider the balance between difficulty and diversity with respect to the subset size. Such a size-agnostic control of the diversity-difficulty balance may result in a suboptimal selection which affects the decision boundary in some cases, such as the edge of the moon-shaped boundary.

Figure 1 (b) visualizes the subset chosen by the proposed BOSS method. As compared with CCS, BOSS adequately covers the entire feature space while attending to all critical regions, which ensures that an accurate decision boundary can be learned from the chosen subset with a much improved prediction performance than CCS. Our main contribution is threefold: (1) a novel balanced core-set loss bound which not only justifies the necessity of simultaneously considering both diversity and difficulty for subset selection but also unveils the key relationship between the type of data samples to be included in the subset and the subset size, (2) design of an expressive importance function to optimally balance diversity and difficulty for subset selection given the subset size, and (3) a comprehensive evaluation using both synthetic and real-world data to verify the key theoretical results and empirical performance of the proposed method.

2. Related Work

2.1. Diversity-Based Subset Selection

Gradient-based subset selection (GB-SS). GB-SS aims to

find a subset such that the difference between the sum of the gradients of the full set and the weighted sum of the gradients of the subset is minimized. As a representative GB-SS method, CRAIG (Mirzasoleiman et al., 2020) shows the gain for convex optimization or simple classification tasks but becomes less competitive for complex deep learning models and difficult learning tasks. GradMatch (Killamsetty et al., 2021a) improves on CRAIG by regularizing the weight values such that large weight values are penalized while selecting the subset. Adacore (Pooladzandi et al., 2022) leverages a Hessian pre-conditioned gradient to capture the curvature information of gradient and exponential moving average of gradients to smooth out the local gradient information. In addition to minimizing the gradient difference, LCMAT (Shin et al., 2023) also minimizes the difference of maximum eigenvalue obtained from the inverse of Hessian of full set and subset in order to capture the curvature information of the loss landscape. Although these methods improve the performance, the calculation of inverse Hessian approximation is time-consuming and computationally expensive (Pearlmutter, 1994).

Feature-based subset selection (FB-SS). FB-SS aims to find a representative subset in the feature space. K-center (Farahani & Hekmatfar, 2009) is a mini-max facility location problem where the subset is selected such that the maximum distance between a point in the original dataset closest to the chosen center is minimized. Herding (Welling, 2009) selects the subset such that the distance between the centroid of the full set and the subset is minimized. The centroid is found using the feature of the input. Contextual Diversity (Agarwal et al., 2020) improves the visual diversity in the feature space and uses KL divergence for calculating the pairwise distance. Although these methods leverage input features to select the subset, they do not consider the sample difficulty. However, the difficulty level of the samples is important because even if two samples are close in the feature space, they can have distinct difficulty scores, especially for those close to the decision boundary.

2.2. Difficulty-Based Subset Selection

Difficulty-based subset selection scores each example based on some difficulty metric that measures how difficult it is to learn the sample or how much impact the sample has on the generalization. (Toneva et al., 2019) count the number of times an example is learned and then forgotten to identify which examples are difficult. The most difficult samples are chosen as the subset. (Paul et al., 2021) introduce the EL2N score which stands for L_2 norm of a prediction error. Unlike forgetting scores, EL2N can be calculated early on during the training such that the time to find the subset is significantly lower. (Sorscher et al., 2022) compare the EL2N score with other scores such as the influence score (Feldman & Zhang, 2020) to select the subset. The influence score of a sample is the measure of how much the generalization performance of a model suffers if that sample is removed from the training dataset. Samples with high influence scores are deemed more difficult. However, this method is computationally expensive because it needs to train the model multiple times on the full dataset. Although difficulty-based methods prove to be effective for larger subset sizes, they tend to choose suboptimal solutions when the subset size is small. Our theoretical results reveal the key underlying reason for this behavior. One very recent work (Xia et al., 2023) defines a new difficulty metric based on the distance of each example with the center of the related class such that we can select the samples with smaller distances to their class center. However, it ignores the diversity in the feature space. Another recent work (Zheng et al., 2023) develops a new sampling method that can utilize different difficulty scores to achieve better performance compared to only selecting the most difficult samples. It selects samples randomly among different strata of difficulty scores and allocates an equal budget among the strata. Our theoretical results show that the diversity and difficult components need to be carefully balanced to avoid a loose loss bound that can misguide the subset selection process. (He et al., 2023) incorporates subset selection together with data condensation. However, it focuses on pruning already condensed data where the resulting final subset size is very small and also does not consider representative samples. In contrast, our method can perform size-aware subset selection, which is much more flexible and broadly applicable to more application scenarios.

3. Methodology

Consider a deep learning model with parameters $\boldsymbol{\theta}$ and a training dataset $\mathcal{V} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{|\mathcal{V}|}$ from which we want to select a subset $\mathcal{S} \subseteq \mathcal{V}$. We use one-hot vectors for the labels $\mathbf{y}$. The training objective is to find the set of parameters $\boldsymbol{\theta}$ that gives us the lowest training error $l = \frac{1}{|\mathcal{T}|}\sum_{n=1}^{|\mathcal{T}|} l_n(\boldsymbol{\eta}(\mathbf{x}_n), \mathbf{y}_n; \boldsymbol{\theta})$, where $\mathcal{T}$ could be either the full set or the subset and $\boldsymbol{\eta}(\mathbf{x}_n) = (\eta^{(1)}, ...\eta^{(K)})^\top$ is the model prediction for $\mathbf{x}_n$ given $\boldsymbol{\theta}$. To obtain the optimal model that can be trained over $\mathcal{S}$, we first train a model for a few epochs on the full dataset (*i.e.,* $\mathcal{T} = \mathcal{V}$) to select the subset $\mathcal{S}$ from $\mathcal{V}$ using the model information. A newly initialized model $\boldsymbol{\theta}_\mathcal{S}$ is then trained on the subset ($\mathcal{T} = \mathcal{S}$). The size $|\mathcal{S}|$ is limited by the amount of budget or resources available. We want to find a subset such that the model trained on the subset has a comparable generalization capability to that of the model trained on the full set.

3.1. Balanced core-set Loss Bound - Subset Size Matters

Our goal is to find the optimal subset that generalizes similarly to the model trained on the full set. Following the core-set based formulation (Sener & Savarese, 2018), the true generalization loss of the model $\theta_\mathcal{S}$ is closely related to

the full set loss:

$$\begin{aligned}&\mathbb{E}_{\mathbf{x},\mathbf{y}}\left[l(\boldsymbol{\eta}(\mathbf{x}),\mathbf{y};\boldsymbol{\theta})\right]\\&\leq\left|\mathbb{E}_{\mathbf{x},\mathbf{y}}\left[l(\boldsymbol{\eta}(\mathbf{x}),\mathbf{y};\boldsymbol{\theta})\right]-\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})\right|\\&\quad+\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})\end{aligned}\tag{1}$$

The first term in the above equation is the difference between true generalization loss and the full set empirical loss which is inaccessible. Thus, we focus on the full set loss given model $\boldsymbol{\theta}_{\mathcal{S}}$. We assume that for every input $\mathbf{x}_i$ in the full set, there exists an $\mathbf{x}_j$ in the subset such that the training loss on $\mathbf{x}_j$ is 0 due to the optimization of the model on $\mathcal{S}$.

Theorem 1 (Balanced Core-set Loss Bound)**.** *Given the full set $\mathcal{V}$ and the subset $\mathcal{S}$, for each $\mathbf{x}_i \in \mathcal{V}$, we can locate a corresponding $\mathbf{x}_j \in \mathcal{S}$, such that $\|\mathbf{x}_j - \mathbf{x}_i\| = \min_{\mathbf{x}_n\in\mathcal{S}}\|\mathbf{x}_n - \mathbf{x}_i\|$ and $l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_j) = 0$. Then, we have*

$$\begin{aligned}&\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i,\boldsymbol{\theta}_{\mathcal{S}})\\&\leq\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}(\lambda^{\boldsymbol{\eta}}\|\mathbf{x}_i-\mathbf{x}_j\|+\lambda^{y}\|\mathbf{y}_i-\mathbf{y}_j\|)+L\sqrt{\frac{\log(1/\gamma)}{2|\mathcal{V}|}}\end{aligned}\tag{2}$$

with the probability of $1-\gamma$, where $\lambda^{\boldsymbol{\eta}}$ and λ^{y} are Lipschitz parameters, L is the maximum possible loss and γ is the probability of the Hoeffding's bound not holding true.

Proof Sketch. To obtain the inequality, we utilize Hoeffding's bound. The problem then becomes finding the expectation of the full set loss ($\mathbb{E}[\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})]$). Note that unlike in the active learning case where the labels of the full set are unknown, we have access to both the inputs and labels in the subset selection scenario. Thus, we treat the model $\boldsymbol{\theta}_{\mathcal{S}}$ as the variable and convert all difference terms to $\|\mathbf{x}_i - \mathbf{x}_j\|$ or $\|\mathbf{y}_i - \mathbf{y}_j\|$ using Lipschitz conditions. The proof mainly involves the following step where we utilize the triangle inequality and the assumption that $l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_j;\boldsymbol{\theta}_{\mathcal{S}}) = 0, \forall \mathbf{x}_j \in \mathcal{S}$:

$$\begin{aligned}&\mathbb{E}\left[\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})\right]\\&=\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}\mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})-l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})\\&\quad+l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})-l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_j;\boldsymbol{\theta}_{\mathcal{S}})|]\\&\leq\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}\mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})-l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})|\\&\quad+|l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_i;\boldsymbol{\theta}_{\mathcal{S}})-l(\boldsymbol{\eta}(\mathbf{x}_j),\mathbf{y}_j;\boldsymbol{\theta}_{\mathcal{S}})|]\end{aligned}\tag{3}$$

□

Remark. Theorem 1 provides an upper bound of the training loss. Ideally, for a subset with a fixed size, a good subset should keep both terms small in order to obtain a tight loss bound. However, this bound can not be directly evaluated due to the unknown Lipschitz parameters associated with the feature difference (*i.e.,* $\|\mathbf{x}_i - \mathbf{x}_j\|$) and the label variability (*i.e.,* $\|\mathbf{y}_i - \mathbf{y}_j\|$), respectively. To overcome this hindrance, we aim to find a practical surrogate target instead of directly minimizing the l.h.s. of (2). Besides the dataset itself, an important known factor in the selection process is the desired size of the subset $|\mathcal{S}|$. A fundamental property of the feature similarity is that $\sum_{i\in\mathcal{V}}\|\mathbf{x}_i - \mathbf{x}_j\|$ monotonically decreases as the size $|\mathcal{S}|$ increases. This allows us to build upon the diversity-based approach, which naturally minimizes the feature objective. The challenge lies in how to further integrate the label variability objective that can adapt to the size of the subset. To this end, we first show the theoretical connection between the label variability and difficulty score and then introduce a novel importance function to adapt the label variability according to the subset size.

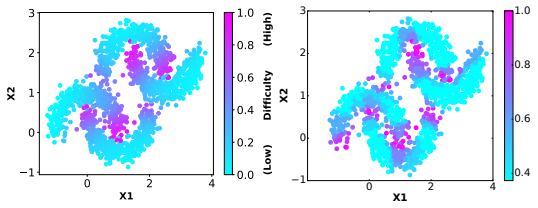
Bridging label variability with difficulty score. Here, we show that a difficulty-based approach can account for the label objective. For the purpose of integrating the label variability objective into the diversity-based approach in a size-aware manner, we look for a surrogate that can be evaluated easily for $\mathcal{V}$ and characterized given varying subset sizes. In fact, a difficulty score such as the EL2N score (defined as $\mathbb{E}\left[\|\boldsymbol{\eta}(\mathbf{x}) - \mathbf{y}\|\right]$) lower bounds the $\|\mathbf{y}_i - \mathbf{y}_j\|$ objective in the difficult regions of the joint data distribution, as we show in the following theorem:

Theorem 2 (EL2N lower bounds the label variability)**.** *Assuming a subset sample $(\mathbf{x}_j, \mathbf{y}_j) \in \mathcal{S}$ is located in a difficult region (*e.g.*, near the decision boundary), where (i) the neighborhood $\mathcal{N}_j$ is dense ($\|\mathbf{x}_j - \mathbf{x}_i\| \leq \delta_x, \forall(\mathbf{x}_i,\mathbf{y}_i) \in \mathcal{N}_j$ for $|\mathcal{N}_j|$ closest points) and (ii) the label variability is high ($p(\|\mathbf{y}_i - \mathbf{y}_j\| > 0) \geq \xi$), the EL2N score produced by a smooth model (*e.g.*, the initial model $\boldsymbol{\eta}_0(x;\mathcal{V})$) will lower bound the label variability in this neighborhood $\mathcal{N}_j$.*

Proof. For the initial model trained for a few epochs on the full set $\mathcal{V}$, we denote it as $\boldsymbol{\eta}_0$. Given a difficult region as specified by the theorem, we consider the closest neighbors $\mathbf{x}_i$ and $\mathbf{x}_j$, which implies $\delta_x \approx 0$. Assume that $\mathbf{x}_j$ is correctly predicted: $\|\mathbf{y}_j - \boldsymbol{\eta}_0(\mathbf{x}_j)\| \approx 0$. Then, we have

$$\begin{aligned}\|\mathbf{y}_i-\mathbf{y}_j\|&\approx\|\mathbf{y}_i-\mathbf{y}_j\|+\lambda^{\boldsymbol{\eta}_0}\delta_x\\&\geq\|\mathbf{y}_i-\mathbf{y}_j-\boldsymbol{\eta}_0(\mathbf{x}_i)+\boldsymbol{\eta}_0(\mathbf{x}_j)\|\\&=\|(\mathbf{y}_i-\boldsymbol{\eta}_0(\mathbf{x}_i))-(\mathbf{y}_j-\boldsymbol{\eta}_0(\mathbf{x}_j))\|\\&\approx\|\mathbf{y}_i-\boldsymbol{\eta}_0(\mathbf{x}_i)\|\end{aligned}\tag{4}$$

If $\mathbf{x}_j$ is from a difficult region as specified by the theorem, then two conditions (i) and (ii) are satisfied. Consider another data sample $\mathbf{x}_j$ from the dense neighborhood where



(a) Difficulty Score (b) Label Variability

Figure 2: (a) The difficulty level of the samples for the synthetic dataset is computed using the EL2N score. The darker points refers to the difficult samples with higher value of EL2N score. The EL2N score is computed at epoch 10. (b) The expected label variability $\|\mathbf{y}_i - \mathbf{y}_j\|$ in the 10-sample neighborhood (scaled value).

$\|\mathbf{x}_j - \mathbf{x}_i\| \leq \delta_x$, we have $\|\boldsymbol{\eta}_0(\mathbf{x}_j) - \boldsymbol{\eta}_0(\mathbf{x}_i)\| \leq \lambda^{\boldsymbol{\eta}_0}\delta_x$. Since $\delta_x \to 0$, we have $\boldsymbol{\eta}_0(\mathbf{x}_j) \approx \boldsymbol{\eta}_0(\mathbf{x}_i)$. On the other hand, condition (ii) implies $\mathbf{y}_i \neq \mathbf{y}_j$, so $\mathbf{x}_i$ is likely to be wrongly predicted by the model. Then, the $\|\mathbf{y}_i - \mathbf{y}_j\|$ term can be lower bounded by the difference between the model prediction and label of the wrongly classified sample for the pair $(\mathbf{x}_i, \mathbf{x}_j)$: $\|\mathbf{y}_j - \boldsymbol{\eta}_0(\mathbf{x}_j)\|$. This way, in the most difficult region, we can approximate the overall $\|\mathbf{y}_i - \mathbf{y}_j\|$ by the expected difference between the prediction and the label. More importantly, if the subset is populated in the most difficult region, this lower bound will not change if we permute $(\mathbf{x}_j, \mathbf{y}_j)$ and $(\mathbf{x}_i, \mathbf{y}_i)$ in the same neighborhood $\mathcal{N}_j$ (as long as $p(\|\mathbf{y}_i - \mathbf{y}_j\| > 0) \geq \xi$) even if the wrongly classified samples are exchanged. We can then average the expected difference between the prediction and the label, which resembles the definition of the EL2N score $EL2N = \mathbb{E}_t[\|\boldsymbol{\eta}_0(x) - \mathbf{y}\|]$, where the expectation is taken over several undertrained initial models $\boldsymbol{\theta}_{\mathcal{V}}^{(t)}$. □

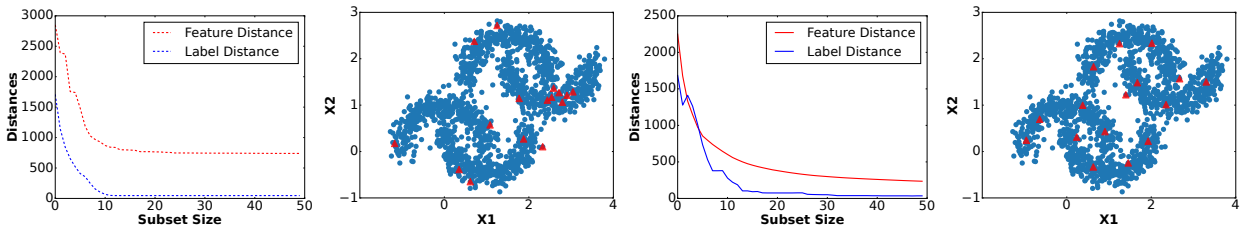
Remark. The connection between the EL2N score and the label variability is consistent with the typical behavior, where data samples with high difficulty scores are distributed near the difficult region of the decision boundary. Figure 2 demonstrates this behavior. The dataset consists of four moon-shaped classes, with slight overlapping (noises) as visualized in the figure. Figure 2 (a) shows the EL2N scores with a color map and the boundary points have the highest difficulty scores. In Figure 2 (b), we show the averaged label variability $|\sum_{i\in\mathcal{N}_j}\|\mathbf{y}_i - \mathbf{y}_j\|| / |\mathcal{N}_j|$ in a random 10-sample neighborhood setting (the values are scaled for visualization purpose). We can see that the trend of label variability matches the EL2N scores well in the difficult regions as shown in Figure 2 (a).

3.2. Feature Similarity-Label Variability Balanced One-shot Subset Selection

We have shown that we can account for the feature similarity objective by building upon the diversity-based approach and also establish lower bound for label variability via the difficulty score. However, as the subset size changes, the contribution from the two components may vary significantly, which in turn will affect the optimal balancing mechanism. In particular, when the subset size is small, the feature term tends to dominate the entire bound because if some major clusters in the data distribution are completely missed, then all the data samples in the entire cluster will be represented by some dissimilar data samples from different clusters. This will accumulate a large feature difference that leads to a very loose bound. As the subset size increases and representative samples are properly chosen from all major clusters, the label variability starts to make a more obvious contribution to the overall bound. As revealed in our proof of Theorem 2, completely missing a difficult region will lead to a large label variability, which will result in a larger loss bound. Intuitively, missing samples from these regions makes the model lose the opportunity to learn a fine-grained decision boundary to further improve the generalization performance.

Impact of the subset size to the balanced core-set bound. To illustrate the impact of subset size on the balanced core-set loss bound, we quantify and visualize the two major components in the loss bound: $\sum_i ||\mathbf{x}_i - \mathbf{x}_j||$ and $\sum_i ||\mathbf{y}_i - \mathbf{y}_j||$, which essentially captures the feature distance and label distance between the selected subset and the full set, respectively. As shown in Figure 3 (a), for a small subset size, choosing the subset based on the label variability (or difficulty) can help to quickly reduce the label distance. However, it also leads to a large feature distance, making the overall bound large. Figure 3 (b) further confirms this because the selected samples miss major data distribution regions. In contrast, focusing on the first component (*i.e.,* diversity), the feature distance drops significantly as shown in Figure 3 (c), which implies that the selected subset can represent the entire data distribution well. Figure 3 (d), visualizes the distribution of the selected data samples based on feature distance. As more samples are selected, they start to cover the difficult regions, which effectively reduces the label distance as shown in Figure 3 (c).

Balancing diversity-difficulty through an importance sampling function. While the desired behavior of a selected subset given a fixed size can be derived from the balanced loss bound, the exact contribution of each of the two components with respect to the subset size remains unknown in practice. In order to achieve a fine-grained balance between them, we utilize the two key pieces of information: subset size $|\mathcal{S}|$ and the overall level of difficulty that can be evaluated using the average difficulty score $\bar{D}$ from the full set. We leverage $|\mathcal{S}|$ and $\bar{D}$ to construct a special `Beta` distribution as an importance sampling function, where $\bar{D}$ and $|\mathcal{S}|$ help to determine the *Peak* and the *Sharpness* of the



(a) (b) (c) (d)

Figure 3: Trend of the two major components in the balanced core-set loss bound: (a) feature and label distance trends by label variability based selection; (b) first 16 samples selected based on label variability; (c) feature and label distance trends by focusing on diversity first; (d) first 16 samples selected based on feature similarity.

desired sampling distribution, respectively:

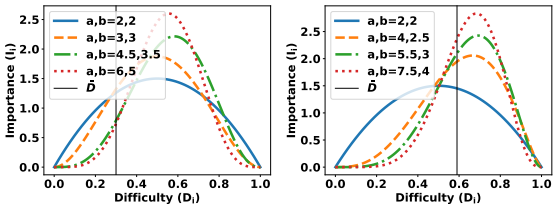
$$\mathcal{I}(\mathbf{x}_j, \mathbf{y}_j) = \texttt{Beta}(D_j|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} D_j^{a-1}(1-D_j)^{b-1} \quad (5)$$

where $D_j \in [0,1]$ is the difficulty score such that $D_j \to 0$ (resp. $D_j \to 1$) denotes easy (resp. difficult) samples, and a, b are parameters of the beta distribution $\texttt{Beta}(\cdot|a,b)$ determined by $|\mathcal{S}|$ and $\bar{D}$ to meet some important properties that are summarized in the following proposition.

Proposition 1 (Setting a and b for desired `Mode` and `Variance` for the importance sampling function)**.** *By setting $a = 1 + \bar{D} + c_a|\mathcal{S}|$ and $b = 2 + c_b|\mathcal{S}|$, where $c_a > c_b > 0$, the importance function meets the following three properties:*

- *P_1: `Mode` increases with $|\mathcal{S}|$ and $\bar{D}$;*
- *P_2: `Mode` $> \bar{D}$ generally holds true;*
- *P_3: `Variance` decreases with $|\mathcal{S}|$ and $\bar{D}$ under mild conditions ($c_a < c_b b$).*

Proof of the proposition is given in Appendix B.4. The three key properties of the specially designed `Beta` distribution allow us to assign desired importance scores to support subset selection according to the subset size. In particular, P_1 ensures that with the increase of the subset size and the general difficulty of the dataset, data samples with a high difficulty score will more likely to be selected as being close to the mode of the distribution. P_2 guarantees that trivial data samples (*i.e.,* those very easy ones) are less likely to be included into the subset as they may not significantly contribute to the learning process. Finally, P_3 shows that a small subset (*i.e.,* $|\mathcal{S}|$ is small) leads to a more flat distribution since the variance is large. As a result, more diverse samples can be selected. In contrast, with the increase of $|\mathcal{S}|$, the distribution becomes more concentrated and the mode also moves along with the average difficulty score $\bar{D}$, which allows the subset to select data samples with a higher difficulty score to learn a more challenging decision boundary. While determining a universal set of c_a and c_b may seem difficult, we further show that as long as we increase a and b in an approximately linear fashion with some trivial constraints such as $c_a|\mathcal{S}| > 1 - \bar{D}, c_b \sim (1-\bar{D})c_a$, all these properties almost always hold true in practice.

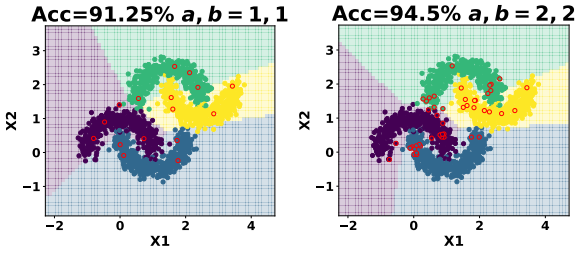


(a) CIFAR 10 (b) TinyImageNet

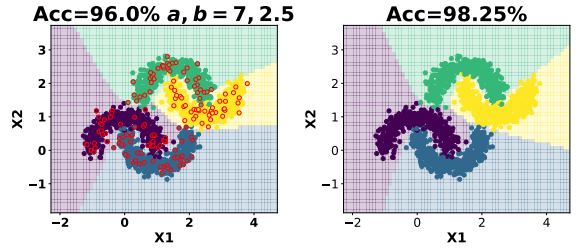
Figure 4: Importance functions with different combination of a and b. The corresponding subset sizes are Blue (solid) = 10%, Orange (dash) = 20%, Green (dash-dot) = 30%, and Red (dot) = 50%

Figure 4 visualizes the importance sampling function for both CIFAR10 and TinyImageNet. In both cases, as the subset sizes increase, the mode of the function shifts to the right that allows the selection of more difficult samples. Meanwhile, the mode is on the right side of $\bar{D}$, which avoids choosing relatively trivial samples. As the subset size increases, the distribution becomes more concentrated that further improves the chance of choosing difficult samples. Finally, since TinyImageNet is a more difficult dataset than CIFAR10 (with a higher $\bar{D}$), the mode of its importance sampling function is further shifted towards the right as compared with CIFAR10. Consequently, more samples with a higher difficulty scores will be included into the subset.

We also conduct experiments on a synthetic dataset to demonstrate how the importance function can perform an optimally balanced subset selection as the size of the subset varies. The results are shown in Figure 5. Given the extremely small subset size (1%), it is preferred to let the model choose more diverse (and representative) samples to cover a wide range of the data space by setting $a, b = 1, 1$. As the subset increases (3% $\to$ 10%), the peak of $\mathcal{I}$ can be shifted to higher difficulty levels by increasing both a and b where $a > b$. As can be seen, by training the model using



(a) 1% Subset (b) 3% Subset



(c) 10% Subset (d) Full Set

Figure 5: Visualization of subset selection for a synthetic moon-shaped dataset using BOSS. For different subset sizes, the respective test performance and the optimal values for parameters a and b are shown. The decision boundary learned for different subset sizes is compared with the full set.

a subset that is only 10% of the full set, it can discover a decision boundary as shown in (c) really close to the one using a model trained using the full set, as shown in (d).

Balanced subset selection Function. Combining the minimization of the diversity objective using the maximization of the similarity between the full set and the subset, and the minimization of the difficulty objective using our controllable importance function, we propose the balanced subset selection function as:

$$F(\mathcal{S}) = \sum_{i \in \mathcal{V}} \max_{j \in \mathcal{S}} \texttt{Sim}(\mathbf{x}_i, \mathbf{x}_j) \mathcal{I}(\mathbf{x}_j, \mathbf{y}_j) \tag{6}$$

where we use multiplication since we remain agnostic about the Lipschitz coefficients. Even with our fine-grained difficulty control, there is still the risk of selecting noises especially when we target the most difficult. To this end, we will adopt the cutoff mechanism as in (Zheng et al., 2023). In Eq. (6), the range of the beta distribution ensures the non-negativity of $\mathcal{I}$. Thus, $F(\mathcal{S})$ is a monotonically increasing function and can be shown to be submodular. This allows us to use a lazy greedy algorithm to approximate the optimum subset that can minimize $F(\mathcal{S})$. The greedy algorithm starts with an empty set $\mathcal{S} = \phi$ and keeps on adding samples $(\mathbf{x}_j, \mathbf{y}_j)$ to subset $\mathcal{S}$ that maximizes the gain:

$$F((\mathbf{x}_j, \mathbf{y}_j)|\mathcal{S}) = F(\mathcal{S} \cup (\mathbf{x}_j, \mathbf{y}_j)) - F(\mathcal{S}) \tag{7}$$

The pseudo code summarizing our implementation is described in Appendix C.

4. Experiments

We conducted experiments on both synthetic and real data, aiming to verify our proposed theoretical results through the former and demonstrate the superior empirical performance through the latter. Limited by space, the synthetic experimental results are presented in Appendix D.1. Our real data experiments are conducted using four datasets: SVHN, CIFAR10, CIFAR100, and Tiny-ImageNet. We present both comparison result and a detailed ablation study.

Comparison baselines. We compare BOSS with eight baselines: **1) Random**: The samples are selected uniformly. **2) CRAIG**: CRAIG is one of the first representative-based subset selections developed for classical models as well as deep learning models. It selects the subset by matching the gradient update signals of the full set and the subset (Mirzasoleiman et al., 2020). **3) GradMatch**: GradMatch uses orthogonal matching pursuit algorithm to match the gradient of subset and training set (Killamsetty et al., 2021a). **4) Adacore**: Adacore uses hessian preconditioned gradient instead of gradient (Pooladzandi et al., 2022). **5) LCMAT**: LCMAT selects the subset such that they match the loss curvature of the full set and the subset by matching the gradient and maximum eigenvalue of hessian between the full set and the subset (Shin et al., 2023). **6) Moderate**: Moderate core-set introduces distance-based scores such that samples with features closer to the median of the features of the related class is more important such that they keep the most important samples and prune the unimportant ones (Xia et al., 2023). **7) CCS**: CCS is coverage-centric core-set selection, which choose data samples randomly across different strata of importance scores giving priority to sparse strata (Zheng et al., 2023). **8) YOCO**: YOCO selects a subset using a Logit-Based Prediction Error (LBPE) score to give importance to easier samples (He et al., 2023). As it primarily performs subset selection on already condensed data resulting in a much smaller subset size, we compare with YOCO separately in Appendix D.3.

Experimental setup. Our experiment setup follows existing approaches, such as (Shin et al., 2023; Guo et al., 2022; Zheng et al., 2023), where to select the subset, we first initialize a model by training it using the full dataset for a limited number of epochs. Then using the training dynamics obtained from the initialized model, we obtain the difficulty score for each sample which is used to select the subset. We then evaluate the selected subset by keeping the subset fixed and using the subset to train a new randomly initialized model. For the difficulty score, we mainly experiment using the EL2N score because it can be computed efficiently early on during the training. The features, gradients, and Hessians are computed from the second-last layer of the network. The baselines vary in the way they select the subset. For the model, we train the ResNet18 model (He et al., 2016) using SGD with a learning rate decay of

Table 1: Comparison results on subsets with different sizes

Dataset	Subset	Random	CRAIG	GradMatch	Adacore	LCMAT	Moderate	CCS	BOSS(Ours)
Tiny ImageNet	10%	24.11±1.9	24.61±0.9	23.68 ±1.5	24.12±1.5	23.26±1.9	24.16±1.3	29.59±0.9	**32.54**±0.9
	20%	37.67±0.3	37.76±0.6	38.20 ±1.3	37.94±0.6	36.71±0.8	37.57±1.1	40.42±0.6	**44.49**±0.2
	30%	45.12±0.9	44.63±0.5	44.93 ±0.6	44.72±0.5	44.06±0.38	45.30±0.4	47.11±0.5	**51.21**±0.3
	50%	53.07±0.7	53.03±0.6	53.81 ±0.2	53.37±0.4	53.10±0.4	53.31±0.4	55.11±0.3	**57.77**±0.5
CIFAR 100	10%	37.35±1.9	38.67±1.3	36.68 ±0.6	37.65±0.8	37.23±0.8	37.76±0.9	40.26±1.6	**46.54**±0.9
	20%	51.55±2.6	51.44±1.7	53.16 ±2.2	52.79±0.8	53.11±0.3	50.90±1.9	55.48±1.8	**61.76**±0.5
	30%	62.89±0.6	62.92±0.7	63.02 ±1.2	62.28±1.2	62.25±0.8	62.55±0.6	64.61±0.5	**67.73**±0.01
	50%	70.67±0.3	70.69±0.5	70.68 ±0.4	71.19±0.3	70.53±0.4	71.13±0.2	71.53±0.3	**73.93**±0.2
CIFAR 10	10%	70.69±1.2	70.96±1.6	72.26 ±0.5	72.65±0.9	71.03±2.6	72.04±0.7	74.78±1.8	**78.27**±0.9
	20%	83.27±1.2	83.36±1.5	84.30 ±0.9	84.30±1.2	83.98±1.3	83.64±0.8	86.45±2.1	**88.14**±1.2
	30%	88.89±0.6	88.98±1.2	88.47 ±0.6	88.37±1.2	88.54±0.7	88.46±0.5	91.49±0.5	**92.14**±0.2
	50%	92.69±0.2	92.75±0.3	91.89 ±0.4	92.67±0.5	92.58±0.2	92.61±0.2	93.45±0.5	**94.46**±0.1
SVHN	8%	84.98±1.9	84.30±1.1	84.31 ±1.8	82.31±2.6	84.05±1.8	84.51±0.7	86.69±1.5	**88.83**±1.8
	12%	87.16±2.4	88.49±0.4	88.99 ±1.0	88.41±1.3	87.49±1.3	88.97±0.6	92.16±0.9	**93.16**±0.9
	16%	90.47±0.7	89.92±0.9	90.42 ±0.8	90.34±0.8	90.16±0.6	90.35±1.1	93.87±0.5	**94.51**±0.5
	20%	91.64±0.7	92.13±0.3	91.56 ±0.4	91.95±0.8	91.36±0.4	91.30±0.9	94.38±0.5	**95.15**±0.2

5×10^{-4}, starting learning rate of 0.1, and momentum of 0.9. We use ResNet34 for the Tiny ImageNet dataset. We use a batch size of 256. To compute the EL2N score, we use the training dynamics up to the first 10 epochs of the initial training. Similarly, we use the feature representations, gradients, and Hessians of epoch 10 of the initial training. The reported results are averaged over five runs. For our method, we sample the subset in a class-balanced fashion. Additional details can be found in the Appendix.

Performance comparison. Table 1 show the result for the four datasets as compared to the baselines. Our method systematically integrates both diversity and difficulty while performing a balanced selection according to the subset size and the nature of dataset. It significantly outperforms all the competitive baselines, especially on the low budget regime. The performance difference decreases as the subset size increases because there is less room for improvement.

To show the behavior of the importance function, we run experiments over different values of a and b and present the optimal values in Figure 6 (a) and (b). These values depend on the subset size $\mathcal{S}$ and the complexity of the dataset $\bar{D}$ according to the theoretical analysis. To satisfy the general constraints, c_a and c_b should not make a and b deviate too much from 2. Given the total sizes of the real-world datasets, the optimal c_a lies close to 0.0002, while the optimal c_b is around 0.0001. This also satisfies the condition $c_a > c_b$, as we want to place the mode of the specially designed beta distribution greater than $\bar{D}$. For each dataset, the ratio $\frac{c_a}{c_b}$ is actually close to $\frac{1}{1-\bar{D}}$, which also aligns with our analysis for the variance of the beta distribution.

To ensure a fair comparison with CCS, we also leverage the cutoff rate parameter β. Figure 6(c) shows that β should be set higher for a small subset size to avoid choosing noisy or outlier samples. This can ensure a more robust subset of data samples to be selected.

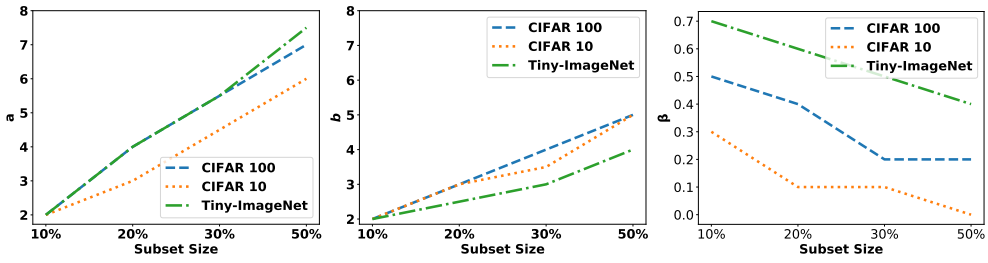
Ablation study. Our ablation study investigates the following parts: 1) the Diversity component, where we minimize the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$; 2) the Difficulty component, where we select samples based on their difficulty scores; 3) Diversity+Difficulty, which performs sample selection based on the proposed balanced subset selection function F; and 4) Diversity+Difficulty+Cutoff, where we further prune the potential noisy examples while balancing diversity and difficulty. Table 2 shows the ablation study results. Only using the Diversity component, which selects the representative samples has sub-optimal performance since it does not consider any difficulty-level information of the datasets. Furthermore, only using the Difficulty component has the worst performance, especially for the low budget regime. This is because the selection is highly biased towards those difficult samples, which causes a large feature difference, leading to a very loose loss bound, as our analysis shows. When combining difficulty and diversity through the proposed importance function, the performance improves by a large margin. Integrating the cutoff mechanism can slightly improve the performance, especially for those more complex datasets, such as Tiny ImageNet. This is because those datasets may likely contain noisy or outlier samples, which if selected, could negatively impact the quality of the subset.

Comparison with Additional Submodular Functions. In Table 3, we compare two additional submodular functions (Kaushal et al., 2022) with BOSS. BOSS (Graph Cut) indicates using Graph Cut as the submodular function for our method and is similar for Log Determinant and Facility Location. As the Log Determinant submodular function focuses on diversity, we suspect it is susceptible to selecting difficult points making it suffer for a smaller subset size.

Other difficulty metrics. Table 4 shows the comparison between CCS and BOSS while using EL2N (Paul et al., 2021), Forgetting (Toneva et al., 2019), and Accumulated Margin(AUM) (Pleiss et al., 2020) for the difficulty met-

Table 2: Ablation study results

Dataset	Subset	Diversity	Difficulty (EL2N)	Diversity + Difficulty	Diversity + Difficulty + Cutoff
Tiny ImageNet	10%	24.04±1.1	3.39±0.3	31.23±0.9	**32.54**±0.9
	20%	37.63±0.9	7.75±0.5	41.56±0.4	**44.49**±0.2
	30%	44.47±0.9	20.92±1.9	46.30±0.4	**51.21**±0.3
	50%	52.78±0.2	44.42±0.6	52.51±0.4	**57.77**±0.5
CIFAR 100	10%	36.69±0.6	7.11±0.4	**47.41**±0.6	46.54±0.9
	20%	52.04±0.9	14.78±0.5	60.01±0.4	**61.76**±0.5
	30%	62.41±0.3	31.99±1.1	66.40±0.5	**67.73**±0.01
	50%	70.18±0.1	65.73±1.0	71.97±0.5	**73.93**±0.2
CIFAR 10	10%	72.17±0.9	22.26±0.4	76.19±2.3	**78.27**±0.9
	20%	84.10±1.0	41.95±1.9	87.09±0.5	**88.14**±1.2
	30%	88.63±0.4	78.75±6.4	89.14±0.3	**92.14**±0.2
	50%	92.52±0.5	94.41±0.2	94.23±0.1	**94.46**±0.1
SVHN	8%	83.96±2.5	63.00±1.9	85.86±1.8	**88.83**±1.8
	12%	89.02±1.2	77.68±1.2	89.14±1.5	**93.16**±0.9
	16%	89.83±0.9	81.83±1.7	91.32±0.5	**94.51**±0.5
	20%	91.27±0.6	84.84±1.1	92.82±0.7	**95.15**±0.2



(a) a (b) b (c) β

Figure 6: The change in the best value of key parameters a (left), b (middle), and β (right) with respect to the subset size for CIFAR 100 (blue), CIFAR 10 (orange), and Tiny-ImageNet (green) dataset.

Table 3: CIFAR-100 Submodular Functions Comparison

Subset Size	Graph Cut	BOSS (Graph Cut)	Log Determinant	BOSS (Log Determinant)	BOSS (Facility Location)
10%	35.98±0.8	46.54±1.5	12.61±0.8	40.78±1.1	**46.54**±0.9
20%	52.59±0.9	61.35±0.5	24.80±0.5	59.30±0.9	**61.76**±0.5
30%	61.67±1.7	67.52±1.2	43.22±0.9	65.74±0.3	**67.73**±0.01
50%	70.34±0.4	72.31±0.2	66.99±0.6	71.89±0.3	**73.93**±0.2

ric. For a fair comparison, we compare CCS and BOSS for each difficulty metric separately. Our method (BOSS) outperforms CCS for every difficulty metric.

Additional ablation is in Appendix D.4 including imbalanced dataset, other models, and computational cost.

5. Conclusion

Subset selection is a promising direction in solving the problem of increasing resource consumption by large deep learning models. Existing subset selection methods have limitations because their selection criteria do not consider a joint distribution of data diversity and difficulty. We propose a novel subset selection strategy that systematically integrates both diversity and difficulty supported by a balanced coreset loss bound. The novel loss bound also suggests important relationship between the difficulty of the selected sample and the subset size, which leads to an expressive importance function that enables us to select appropriate samples according to the subset size. Our theoretical analysis along with the empirical results on synthetic and real-world data demonstrate the greater effectiveness of BOSS compared with the competitive baselines.

Table 4: Comparison of CCS and BOSS for Tiny ImageNet while using different difficulty metrics

	EL2N		Forgetting		AUM	
Subset	CCS	BOSS	CCS	BOSS	CCS	BOSS
10%	29.59±0.9	**32.54**±0.9	30.44±1.7	**33.78**±1.3	31.51±1.2	**33.47**±0.7
20%	40.42±0.6	**44.49**±0.2	42.75±0.9	**45.56**±0.6	42.05±0.4	**45.80**±0.6
30%	47.11±0.5	**51.21**±0.3	48.61±0.7	**51.81**±0.2	48.92±0.1	**52.11**±0.3
50%	55.11±0.3	**57.77**±0.5	55.91±0.5	**57.82**±0.3	55.74±0.4	**57.79**±0.3

Acknowledgement

This research was supported in part by an NSF IIS award IIS-1814450. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agency.

Impact Statement

In this paper, we have proposed a balanced one-shot subset selection method (BOSS). While using the subset reduces the resource consumption by large deep learning models, we should consider the change of information from the full set to the subset and the potential biases the selection might introduce. We should also carefully deploy the balanced selection method as data diversity and informativeness (difficulty) are being explicitly controlled. In real-world applications, only preserving the subset could be beneficial if the above aspects are well-considered, and can prove to be useful in scenarios such as continual learning and various socially impactful deep learning applications.

References

Agarwal, S., Arora, H., Anand, S., and Arora, C. Contextual diversity for active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pp. 137–153. Springer, 2020.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. E. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Farahani, R. Z. and Hekmatfar, M. *Facility location: concepts, models, algorithms and case studies*. Springer Science & Business Media, 2009.

Feldman, D. Introduction to core-sets: an updated survey. *arXiv preprint arXiv:2011.09384*, 2020.

Feldman, V. and Zhang, C. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891, 2020.

Guo, C., Zhao, B., and Bai, Y. Deepcore: A comprehensive library for coreset selection in deep learning. In *Database and Expert Systems Applications: 33rd International Conference, DEXA 2022, Vienna, Austria, August 22–24, 2022, Proceedings, Part I*, pp. 181–195. Springer, 2022.

Har-Peled, S. and Kushal, A. Smaller coresets for k-median and k-means clustering. In *Proceedings of the twenty-first annual symposium on Computational geometry*, pp. 126–134, 2005.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, Y., Xiao, L., and Zhou, J. T. You only condense once: Two rules for pruning condensed datasets. *Advances in Neural Information Processing Systems*, 36, 2023.

Kaushal, V., Ramakrishnan, G., and Iyer, R. Submodlib: A submodular optimization library. *arXiv preprint arXiv:2202.10680*, 2022.

Killamsetty, K., Durga, S., Ramakrishnan, G., De, A., and Iyer, R. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning*, pp. 5464–5474. PMLR, 2021a.

Killamsetty, K., Sivasubramanian, D., Ramakrishnan, G., and Iyer, R. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8110–8118, 2021b.

Killamsetty, K., Zhao, X., Chen, F., and Iyer, R. Retrieve: Coreset selection for efficient and robust semi-supervised learning. *Advances in Neural Information Processing Systems*, 34:14488–14501, 2021c.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Madigan, D., Raghavan, N., DuMouchel, W., Nason, M., Posse, C., and Ridgeway, G. Likelihood-based data squashing: A modeling approach to instance construction. *Data Mining and Knowledge Discovery*, 6:173–190, 2002.

Mirzasoleiman, B., Bilmes, J., and Leskovec, J. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, pp. 6950–6960. PMLR, 2020.

Nguyen, C. V., Li, Y., Bui, T. D., and Turner, R. E. Variational continual learning. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BkQqq0gRb.

Paul, M., Ganguli, S., and Dziugaite, G. K. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34: 20596–20607, 2021.

Pearlmutter, B. A. Fast exact multiplication by the hessian. *Neural computation*, 6(1):147–160, 1994.

Pleiss, G., Zhang, T., Elenberg, E., and Weinberger, K. Q. Identifying mislabeled data using the area under the margin ranking. *Advances in Neural Information Processing Systems*, 33:17044–17056, 2020.

Pooladzandi, O., Davini, D., and Mirzasoleiman, B. Adaptive second order coresets for data-efficient machine learning. In *International Conference on Machine Learning*, pp. 17848–17869. PMLR, 2022.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Schwartz, R., Dodge, J., Smith, N. A., and Etzioni, O. Green ai. *Communications of the ACM*, 63(12):54–63, 2020.

Sener, O. and Savarese, S. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1aIuk-RW.

Shin, S., Bae, H., Shin, D., Joo, W., and Moon, I.-C. Loss-curvature matching for dataset selection and condensation. In *International Conference on Artificial Intelligence and Statistics*, pp. 8606–8628. PMLR, 2023.

Sorscher, B., Geirhos, R., Shekhar, S., Ganguli, S., and Morcos, A. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

Strubell, E., Ganesh, A., and McCallum, A. Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*, 2019.

Tan, M. and Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pp. 6105–6114. PMLR, 2019.

Toneva, M., Sordoni, A., des Combes, R. T., Trischler, A., Bengio, Y., and Gordon, G. J. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJlxm30cKm.

Tsang, I. W., Kwok, J. T., Cheung, P.-M., and Cristianini, N. Core vector machines: Fast svm training on very large data sets. *Journal of Machine Learning Research*, 6(4), 2005.

Wan, Z., Wang, Z., Chung, C., and Wang, Z. A survey of data optimization for problems in computer vision datasets. *arXiv preprint arXiv:2210.11717*, 2022.

Welling, M. Herding dynamical weights to learn. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 1121–1128, 2009.

Xia, X., Liu, J., Yu, J., Shen, X., Han, B., and Liu, T. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=7D5EECbOaf9.

Zheng, H., Liu, R., Lai, F., and Prakash, A. Coverage-centric coreset selection for high pruning rates. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=QwKvL6wC8Yi.

Appendix

Appendix A summarizes the major notations used in the main paper. Appendix B provides detailed proofs of our main theoretical results. Appendix C describes our subset selection algorithm. Appendix D provides additional experimental details, link to the source code, results for synthetic data, and ablation studies on real-world data.

A. Summary of Notations

Table 5: Summary of Notations

Symbol	Description
$\mathcal{V}$	Set of all the training samples (Full Set)
$\mathcal{S}$	Set of samples that are selected (Subset)
λ	Lipschitz parameter
$\boldsymbol{\eta}$	Neural network regression function
$\mathbf{x}_i$	Input feature of a sample in the Full-set
$\mathbf{y}_i$	True label of a sample in the Full-set
$\mathbf{x}_j$	Input feature of a sample in the Subset
$\mathbf{y}_j$	True label of a sample in the Subset
$\boldsymbol{\theta}_{\mathcal{S}}$	Model trained on the Subset
$l(\cdot)$	Loss function
γ	Probability for Hoeffding's inequality
L	Upper bound for the loss function
C	Number of classes
$F(\cdot)$	Balanced subset selection function
Sim	Similarity function
$\mathcal{I}(\mathbf{x}_j, \mathbf{y}_j)$	Importance score of sample j
D_j	Difficulty score of sample j
c	Parameter controlling peak of importance function
α	Parameter controlling the sharpness of importance function
β	Hard cut-off rate

B. Proofs of Theoretical Results

In this section, we provide detailed proofs for the proposed theorems in the main paper. We have introduced the balanced core-set loss bound in Section 3.1. Here we first expand the loss bound derivation and analysis and then provide detailed proofs for Theorem 1.

B.1. Our Take on the core-set Loss Bound

In (Sener & Savarese, 2018), the authors proposed the classic core-set cover loss bound, in the active learning setting. The first step is to upper bound the true expectation of generalization loss by the full set loss, which is shown in:

$$\begin{aligned}\mathbb{E}_{x,y}\left[l(\boldsymbol{\eta}(\mathbf{x}), \mathbf{y}; \boldsymbol{\theta})\right] &= \mathbb{E}_{x,y}\left[l(\boldsymbol{\eta}(\mathbf{x}), \mathbf{y}; \boldsymbol{\theta})\right] - \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_{\mathcal{S}}) + \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_{\mathcal{S}}) \\ &\leq \left|\mathbb{E}_{x,y}\left[l(\boldsymbol{\eta}(\mathbf{x}), \mathbf{y}; \boldsymbol{\theta})\right] - \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_{\mathcal{S}})\right| + \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_{\mathcal{S}}) \end{aligned} \tag{8}$$

Same as (Sener & Savarese, 2018) and (Zheng et al., 2023), we adopt this approach and focus on the expectation of the full set loss. However, differently from their approach, we formally tailor the problem in the known full set setting. In our case, all data samples $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{V}$ are treated as known, and the unknown is the model $\boldsymbol{\theta}_{\mathcal{S}}$ trained on the subset (and

the corresponding outputs $\boldsymbol{\eta}(\mathbf{x}_i)$), which corresponds accurately to the subset selection problem. For the same reason, we denote the loss function by $l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})$. Next, we also apply the Hoeffding's bound to analyze the full set loss. However, unlike (Sener & Savarese, 2018) and (Zheng et al., 2023) which reached loose conclusions by bounding the feature difference $\|\mathbf{x}_i - \mathbf{x}_j\|$ using the core-set cover, we consider the joint effect over both the feature and the label, arriving at a balanced core-set loss bound which will be explained below.

B.2. Proof for Theorem 1

Proof. We obtain the inequality in Eq. (2) by applying the Hoeffding's bound:

If $X_1, X_2, ..., X_n$ are independent and $a_i \leq X_i \leq b_i$ almost surely, then the sum $S_n = X_1 + ... + X_n$ satisfy

$$P(S_n - \mathbb{E}[S_n] \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right) \tag{9}$$

In Theorem 1, we apply the above inequality to the full set loss $\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})$ by substituting $S_n = \sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})$ and using $0 \leq l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) \leq L$ (L being the maximum loss value):

$$P\left(\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - \mathbb{E}\left[\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})\right] \geq \frac{t}{|\mathcal{V}|}\right) \leq \exp\left(-\frac{2t^2}{|\mathcal{V}|L^2}\right) \tag{10}$$

Let $\gamma = \exp\left(-\frac{2t^2}{|\mathcal{V}|L^2}\right)$, then with probability $1-\gamma$,

$$\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - \mathbb{E}\left[\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})\right] \leq \frac{t}{|\mathcal{V}|} \tag{11}$$

Rearranging and solving for t, we get

$$\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - \mathbb{E}\left[\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})\right] \leq L\sqrt{\frac{\log(1/\gamma)}{2|\mathcal{V}|}} \tag{12}$$

Next, we explain the expectation of the full set loss and obtain the balanced combination result:

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})\right] \\
&= \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} \mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) + l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_j; \boldsymbol{\theta}_\mathcal{S})|] \\
&\leq \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} \mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})| + |l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_j; \boldsymbol{\theta}_\mathcal{S})|] \\
&= \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}}\Big(\mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})|] \\
&\quad + \mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_j; \boldsymbol{\theta}_\mathcal{S})|]\Big)
\end{aligned} \tag{13}$$

which has been broken into two terms.

For the first term $\mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})|]$, we utilize the Lipschitzness of the model combined with the Lipschitzness of the loss function to get $\mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_i), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S})|] \leq \mathbb{E}[\lambda^{\boldsymbol{\eta}}\|\mathbf{x}_i - \mathbf{x}_j\|]$. The expectation should be taken over the model prediction $\eta^{(k)}(\mathbf{x}_i)$, and will result in a class-irrelevant term for $\sum_k \eta^{(k)}(\mathbf{x}_i) = 1$ if we use loss functions like the cross-entropy loss so that $\lambda^{\eta^{(k)}}$ is the same for all k.

For the second term $\mathbb{E}[|l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_i; \boldsymbol{\theta}_\mathcal{S}) - l(\boldsymbol{\eta}(\mathbf{x}_j), \mathbf{y}_j; \boldsymbol{\theta}_\mathcal{S})|]$ we directly use the Lipschitzness of the loss function w.r.t. $\mathbf{y}$ and it is independent from $\boldsymbol{\eta}(\mathbf{x}_i)$ so we have $\lambda^y\|\mathbf{y}_i - \mathbf{y}_j\|$.

Substituting all the above terms back and we will obtain Eq. (2). □

B.3. Analysis of Theorem 2

In Theorem 2, we connect the EL2N score to the expected label variability in a neighborhood $\mathcal{N}_j$ of a subset point $(\mathbf{x}_j, \mathbf{y}_j)$. Unlike Theorem 1, which is more general about the loss on full set $\mathcal{V}$, this connection is specifically made in the difficult region.

Remark (2.1). *Why is it important to consider the difficult region?*

We have defined the difficult region as dense and having high label variability. This is because we really focus on the disadvantage of only considering the diversity aspect or the difficulty aspect. Intuitively, if the true distribution of $(\mathbf{x}, \mathbf{y})$ is clearly separated and smooth, using a few samples can perfectly explain the classification problem as long as all classes are represented. However, if there exists a difficult region such that $\|\mathbf{x}_j - \mathbf{x}_i\| \leq \delta_x$ and $p(\|\mathbf{y}_i - \mathbf{y}_j\| > 0) \geq \xi$, then it poses a challenge for both single-sided approaches: a diversity-only approach can not identify informative data samples that help learn the decision boundary in this difficult region, while a difficulty-only approach will be highly biased towards this region and won't be able to efficiently represent the majority of samples which are easy to classify. We will present more visualizations using the synthetic dataset in Appendix D.1. Thus, it takes a delicate balancing to improve the overall objective in (2) when the difficult region exists.

Remark (2.2). *How do we utilize the EL2N score to explain the label variability?*

In the proof of Theorem 2, we present an approximately lower bounding relationship between the label difference between $\mathbf{y}_i$ and $\mathbf{y}_j$ and the EL2N score of the wrongly classified sample $(\mathbf{x}_j, \mathbf{y}_j)$. If we assume that the neighborhood $\mathcal{N}_j$ includes $(\mathbf{x}_i, \mathbf{y}_i)$, $(\mathbf{x}_j, \mathbf{y}_j)$, and $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^{|\mathcal{N}_j|-2}$. With $p \geq \xi$, we have $\mathbf{y}_i \neq \mathbf{y}_j$, thus flipping them will likely flip the model prediction for all $\mathbf{x}_n$. When we take the expectation over all samples in $\mathcal{N}_j$, the averaged $\|\mathbf{y}_i - \mathbf{y}_j\|$ will be connected to the distribution of the EL2N scores of these data samples which is the expectation over a series of models $\boldsymbol{\theta}_{\mathcal{V}}^{(t)}$.

B.4. Proof and analysis of Proposition 1

Proof. For the Beta function, we have

$$\texttt{Mode} = \frac{a-1}{a+b-2}, \quad \texttt{Variance} = \frac{ab}{(a+b)^2(a+b+1)} \tag{14}$$

Given the setting of a and b, we have

$$\texttt{Mode} = \frac{\bar{D} + c_a|\mathcal{S}|}{E}, \quad \texttt{Variance} = \frac{(1+\bar{D}+c_a|\mathcal{S}|)(2+c_b|\mathcal{S}|)}{(E+2)^2(E+3)} \tag{15}$$

where $E = D + (c_a + c_b)|\mathcal{S}| + 1$. By taking the first derivative w.r.t $\bar{D}$ and $|\mathcal{S}|$, we get $\frac{\partial \texttt{Mode}}{\partial \bar{D}} = \frac{c_b|\mathcal{S}|}{E^2} > 0$ and $\frac{\partial \texttt{Mode}}{\partial |\mathcal{S}|} = \frac{c_a - c_b\bar{D}}{E^2} > 0$. Thus, Mode increases as either $\bar{D}$ or $|\mathcal{S}|$ increases.

As for Variance, it is related to how the numerator and denominator increase with $\bar{D}$ and $|\mathcal{S}|$. Generally speaking, for a Beta distribution Beta(a, b), if $a \uparrow, b \uparrow, \frac{a}{b} \sim$ Constant, Variance clearly decreases because the numerator is second-order dependent on a and b while the denominator is third-order dependent. In our setting, the dependency on $\bar{D}$ is clear as it only appears once in the numerator and three times in the denominator. For $|\mathcal{S}|$, it also generally holds true because $a \uparrow, b \uparrow$ as $|\mathcal{S}| \uparrow$ linearly. The only exception is when $\frac{a}{b}$ drastically changes and goes to extreme values, for example, if $c_a > c_b b$. Adding the constraint $c_a < c_b b$, we ensure that Variance should steadily decrease. □

Remark (3.1). *In most cases,* Mode $> \bar{D}$ *can be inferred from the function shape.*

We can investigate this problem by defining a function of $\bar{D}$: $f(\bar{D}) = \texttt{Mode} - \bar{D} = \frac{c_a\mathcal{S} - \bar{D}^2 - (c_a+c_b)\mathcal{S}\bar{D}}{E}$. Since we only care about the sign of $f(\bar{D})$, we focus on the numerator, which is a quadratic function of $\bar{D}$ with parameters $\mathcal{S}, c_a, c_b$. On a real-world dataset, the mean difficulty score $\bar{D}$ usually lies between 0.3 and 0.7 (CIFAR-10: 0.311986, CIFAR-100: 0.444376, Tiny-ImageNet: 0.597217). In our experiments, we have $1 \leq c_a\mathcal{S} \leq 8, 1 \leq c_b\mathcal{S} \leq 5$. In these ranges, we can see that $f(\bar{D}) > 0$ holds true. Mathematically, if we set $c_b = (1 - \bar{D})c_a$, we can also show that it even holds true over $\bar{D} \in [0, 1]$. Combining with the fact that $\bar{D} \uparrow$, Mode $\uparrow$, we ensure the control of the selection *peak* given $\bar{D}$.

C. Algorithm

Algorithm 1 BOSS (Balanced One-Shot Subset Selection)

Initial Training

Input: Dataset $\mathcal{V}$
Output: Difficulty score D_i, feature vector $\mathbf{x}_i$

Initialize full set model $\boldsymbol{\theta}_{\mathcal{V}}$
Train $\boldsymbol{\theta}_{\mathcal{V}}$ on $\mathcal{V}$
From $\boldsymbol{\theta}_{\mathcal{V}}$ obtain $\boldsymbol{\eta}(\mathbf{x}_i^t)$, $\mathbf{y}_i^t$ for each epoch $t \in [1, T_0]$
From $\boldsymbol{\theta}_{\mathcal{V}}$ obtain $\mathbf{x}_i$ for epoch T_0.
Compute EL2N Score D_i using $\boldsymbol{\eta}(\mathbf{x}_i^t)$, $\mathbf{y}_i^t$
return D_i, $\mathbf{x}_i$

Subset Selection

Input: Dataset $\mathcal{V}$, Subset size B, difficulty score D_i, input feature x_i
Parameter: Hard example prune rate β, importance function parameters a and b
Output: Subset $\mathcal{S}$

$I_i \leftarrow \texttt{Beta}(D_i, a, b)$ {Convert difficulty score to importance score}
$\mathcal{V}' \leftarrow \mathcal{V} \setminus \{|\mathcal{V}| * \beta \quad \texttt{hardest samples}\}$ {Prune hardest samples}
$\mathcal{S} \leftarrow \phi$ {Start with an empty subset and add to the subset until we reach the budget for the subset:}
while $|\mathcal{S}| < B * |\mathcal{V}|$ **do**
    $F(\mathcal{S}) \leftarrow \sum_{i \in \mathcal{V}} \max_{j \in \mathcal{S}} \texttt{Sim}(\mathbf{x}_i, \mathbf{x}_j) I_j$ {Compute the facility location function from the feature similarity and sample importance}
    $j \in \texttt{argmax}_{e \in \mathcal{V} \setminus \mathcal{S}} F(e|\mathcal{S})$ {Using lazy-greedy algorithm, select sample j which gives us the maximum conditional gain $F(e|\mathcal{S})$}
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{j\}$ {Update the subset with the new element}
end while
return $\mathcal{S}$

Our method has three main components, 1) Initial training where we first train a model to generate training dynamics from which we can compute the importance scores, 2) Generating importance score from the training dynamics, and 3) Selecting Subset and evaluating the subset by training a new model using the selected subset. Details are provided in Algorithm 1.

D. Additional Experimental Details and Results

We perform our experiments on machines with GPUs: A100, V100, and P4. In our experiments, when combining the difficulty and diversity, we show the results for using the EL2N score (Paul et al., 2021) as a difficulty metric. The EL2N score is calculated in the initial training phase by averaging the error norm over the first 10 epochs. To leverage the cutoff, the Accumulated Margin (AUM) metric (Pleiss et al., 2020) is leveraged, for which we need to train the model for the full epoch (200 epoch for CIFAR10 and CIFAR100, and 100 epoch for Tiny ImageNet and SVHN). Our implementation details and source code can be found here.

D.1. Synthetic Data Experiments

We create the synthetic data to include four moons with two input features such that we can visualize and simulate the complex decision boundary. The synthetic data is visualized in the Figure 4. The dataset has 2000 samples which are split into 80/20 train/test sets. For the model, we use a fully connected neural network with two hidden layers each containing 100 neurons. To train the neural network we use Adam optimizer with a learning rate of 0.001, ϵ = 1e-08, and weight decay = 0. For the full set, we train the model for 100 epochs. The model reaches a test accuracy of 98.25% while training on the full data.

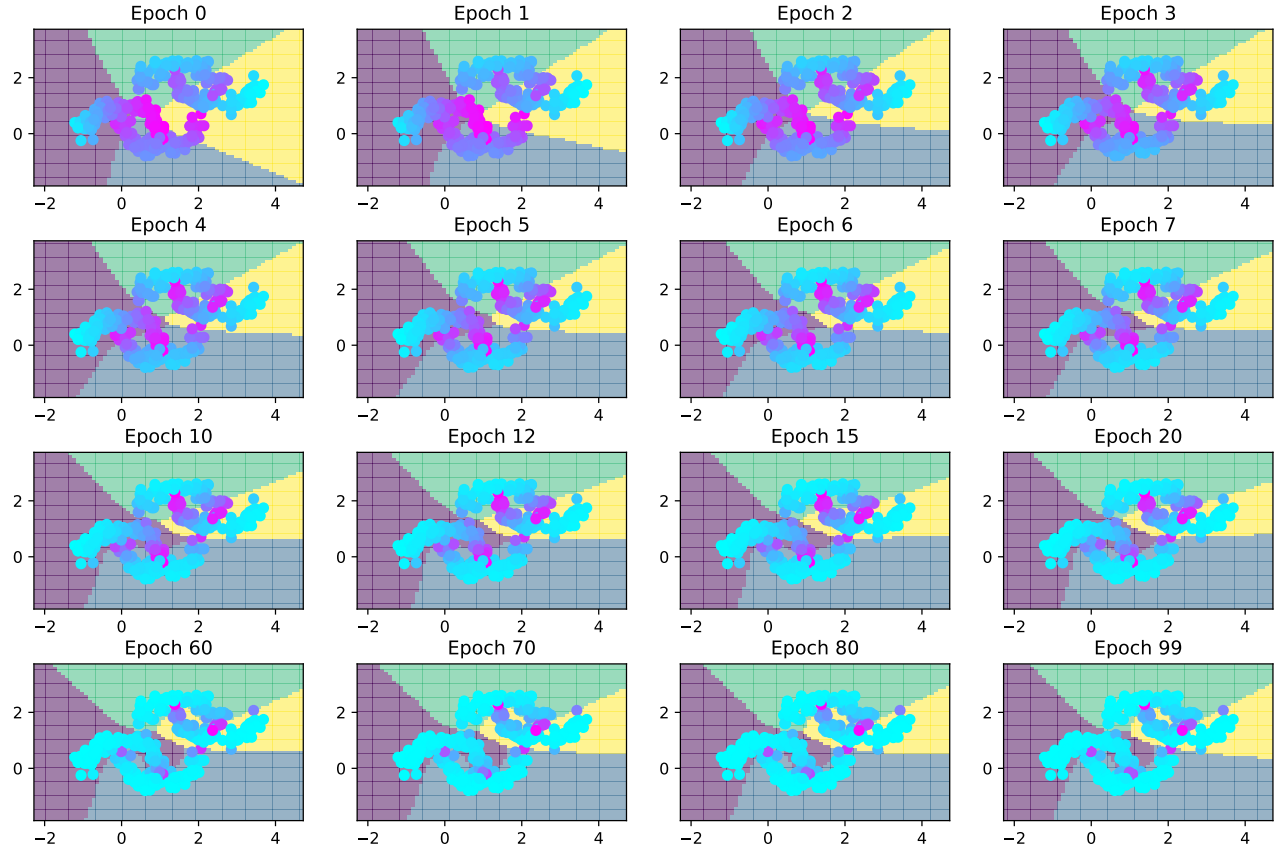
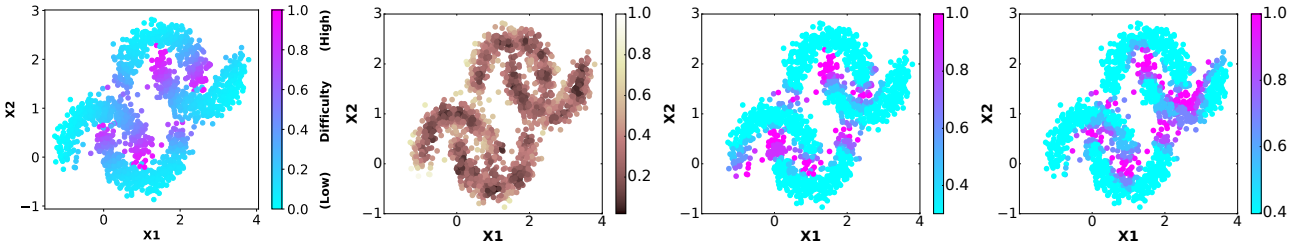


Figure 7: Decision Boundary Evolution

Decision boundary evolution. Figure 7 shows the evolution of the decision boundary along with the difficulty level of each data point for every epoch. The difficulty is calculated using the EL2N score which is the L2 norm of prediction and the onehot label and averaged over the previous epochs. As we train the model for a higher number of epochs, the model is able the learn complex decision boundaries or the curved region and most of the samples become easier or has low EL2N score. However, the difficulty score computed at the earlier epochs, for instance at epoch 10, truly captures the difficulty level of samples along the decision boundary. This agrees with the past methods which compute the EL2N score at epoch 10.



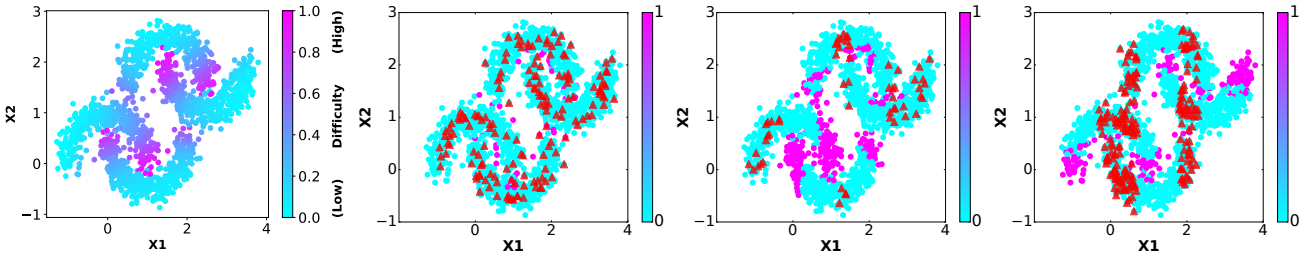
(a) Difficulty Score (b) Feature Distance (c) Label Variability 1 (d) Label Variability 2

Figure 8: Difficulty score, feature distance, and label variability comparison. All values are scaled to 0 to 1 for better visualization. (c) Label variability 1 and (d) Label variability 2 are permutations of Figure 2(b) by randomly changing the data samples included in the 10-sample neighborhood.

Label variability visualization Following Appendix B.3, we visualize the terms that have been discussed in our theoretical results using the synthetic dataset.

In Figure 8, we visualize the difficulty score, feature distance, and label variability with random permutations to the anchor

points being used as $(\mathbf{x}_j, \mathbf{y}_j)$ in the 10-sample neighborhood case. From Figure 8 (a) and (b), we see that the difficulty score does not correlate with the feature distance, and the feature distance is not informative in the difficult region, where the distance is consistently low because it is the denser area. From Figure 2 (b) and Figure 8 (c) and (d), we can see that even with different permutations, the label variability shares the same trend as the difficulty score near the decision boundary.



(a) Difficulty Score (b) Subset 1 (c) Subset 2 (d) Subset 3

Figure 9: Difficulty score and $\mathbb{1}(\mathbf{y}_i \neq \mathbf{y}_j; j = \arg\max_{n \in \mathcal{S}} \texttt{Sim}(\mathbf{x}_n, \mathbf{x}_i))$ comparison, where red triangles represent data samples in $\mathcal{S}$.

In Figure 9, we show a different visualization presenting the actual label differences between the full set and the subset if we choose from different regions. Figure 9 (b) shows a diverse selection, while Figure 9 (c) and (d) show two different biased selections. In all cases, the data samples near the decision boundary have a different label from the nearest sample in $\mathcal{S}$ ($\mathbb{1}(\mathbf{y}_i \neq \mathbf{y}_j; j = \arg\max_{n \in \mathcal{S}} \texttt{Sim}(\mathbf{x}_n, \mathbf{x}_i))$=1). This further supports our motivation as allocating the budget to cover the more difficult region does not guarantee the reduction of the label variability objective in the loss bound given in (2) unless we can cover all these samples. Thus, it is important that we propose the balanced selection function.

BOSS Subset selection comparison. Here we include the subset selection visualization and comparison for the synthetic dataset. The red circles are the samples selected in the subset. We already saw the comparison of CSS and BOSS for the 10% subset in Figure 1. Here we further compare these methods for 1% and 3% subset sizes. In all the cases, BOSS outperforms CCS.

Figure 10 compares the subset selected by CCS and BOSS for 1% subset size. In this case, BOSS can select the diverse samples by setting $a = b = 1$. Although the model cannot learn the complex decision boundary because of the lack of enough data, the CSS misses samples from the important regions and learns an even worse decision boundary.

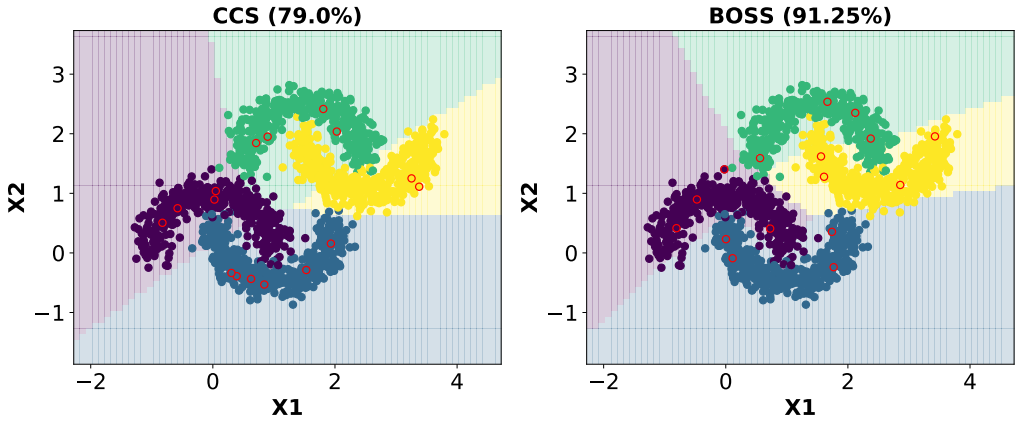


Figure 10: CCS vs BOSS for 1% Subset Size for the synthetic data. For CCS, $\beta = 0.1$. For BOSS, $\beta = 0$ and $a = b = 1$ which is the same as only using representative-based selection.

Similarly, Figure 11 compares CCS and BOSS for a 3% subset size. This figure also verifies that the CCS misses the samples from the critical region that our method is able to capture. In turn our method learns the complex decision boundary to achieve better performance than the CCS baseline.

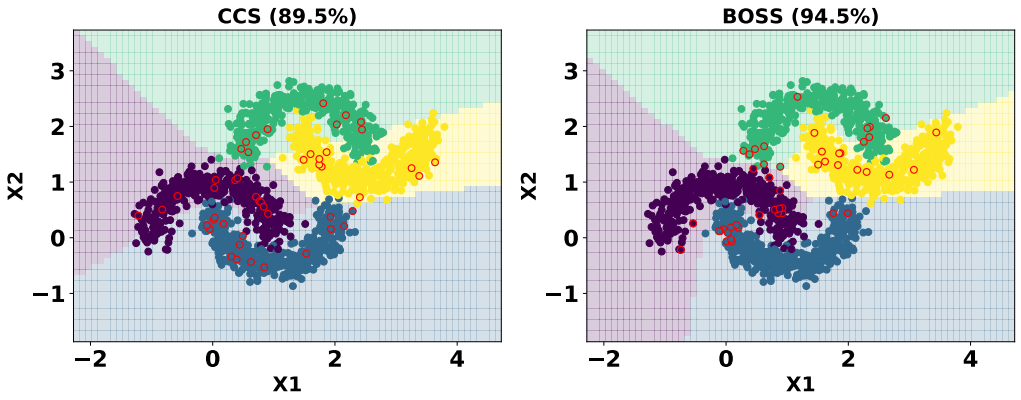


Figure 11: CCS vs BOSS for 3% Subset Size for the synthetic data. For CCS, $\beta = 0.1$. For BOSS, $\beta = 0$, $a = b = 2$

Figure 12 shows the comparison of the subset selected by the representative-based method which matches the feature of the subset and the full set (Diverse) compared with the subset selected by our method. The representative-based subset selection does not consider the sample difficulty which leads it to ignore samples from a very difficult region. However, our method is able to give more emphasis on the difficult region to better learn the decision boundary.

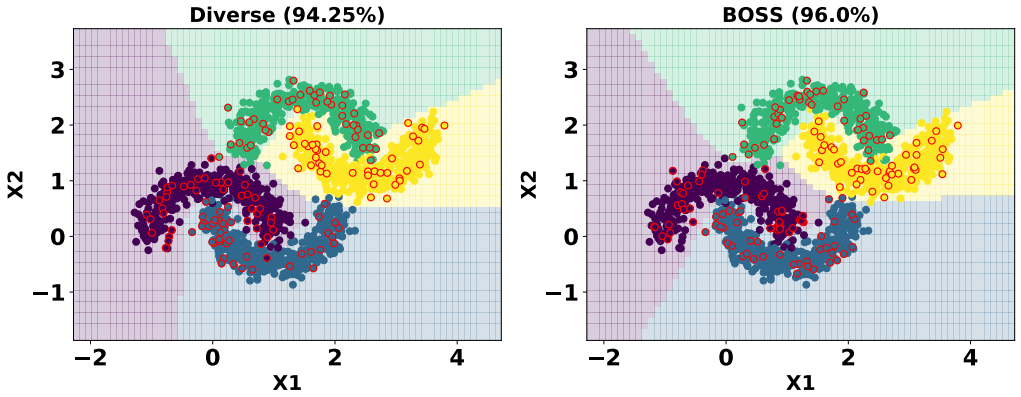


Figure 12: Visualization of representative-based subset selection compared with our method. The red circles are the points selected for the subset. The subset size is 10% and $\beta = 0$. For Diverse, $a = b = 1$ and for BOSS, $a = 7, b = 2.5$

D.2. Comparison with YOCO

We conducted additional experiments to compare our model with YOCO (He et al., 2023) on CIFAR100 and ImageNet-1k datasets. Tables 6 and 8 show the result for CIFAR-100 and ImageNet-1k (described in D.3) respectively. As can be seen, our model has a clear advantage over YOCO. The improvement is mainly because YOCO only gives priority to the easiest samples but does not consider the diversity of samples. However, our method considers both difficulty and diversity.

Table 6: CIFAR100 Comparison with YOCO

Subset Size	EL2N	CCS	YOCO	BOSS (Ours)
10%	7.11$_{\pm 0.4}$	40.26$_{\pm 1.6}$	38.86$_{\pm 0.6}$	**46.54**$_{\pm 0.9}$
20%	14.78$_{\pm 0.5}$	55.48$_{\pm 1.8}$	53.53$_{\pm 0.5}$	**61.76**$_{\pm 0.5}$
30%	31.99$_{\pm 1.1}$	64.61$_{\pm 0.5}$	62.98$_{\pm 0.5}$	**67.73**$_{\pm 0.01}$
50%	65.73$_{\pm 1.0}$	71.53$_{\pm 0.3}$	70.59$_{\pm 0.02}$	**73.93**$_{\pm 0.2}$

In Tables 7 and 9 we further compare our model with the YOCO baseline in their paper's experimental setting. Here, the dataset is first condensed to the point where we have $\texttt{IPC}_{\texttt{F}}$ number of images per class. Then the data is further pruned into

$\texttt{IPC}_\texttt{T}$ number of images per class. Here we compare our pruning method (BOSS) with three other pruning methods: EL2N, CCS, and YOCO. We select two major datasets, CIFAR-100 and ImageNet-10 as used by the YOCO baseline. We follow the same experiment settings as that of the YOCO baseline for a fair comparison. We show that even for a very small subset size, our method is competitive or better compared to the recent baseline YOCO.

Table 7: CIFAR100 YOCO's setting

Subset Size $\texttt{IPC}_\texttt{F}\rightarrow\texttt{IPC}_\texttt{T}$	EL2N	CCS	YOCO	BOSS (Ours)
50→1	4.21±0.02	19.05±0.08	23.47±0.11	**24.71**±0.1
50→2	5.01±0.05	24.32±0.07	29.59±0.11	**30.69**±0.3
50→5	7.24±0.11	31.93±0.06	37.52±0.00	**38.86**±0.16
50→10	11.72±0.06	38.05±0.09	42.79±0.06	**43.16**±0.07

D.3. Experiments on large scale dataset

We conducted additional experiments to compare our model with the recent and most competitive methods on the ImageNet-1k dataset which is shown in Table 8. For ImageNet-1k, we use the pre-trained EffecientNet-B0 model to generate the embedding. We freeze the entire network before and up to the second last layer and only train the final classification layer for the experiments. As can be seen, the proposed method clearly outperforms all the baselines on this large-scale dataset, which further confirms its effectiveness.

Table 8: Experiments on large scale ImageNet-1K dataset

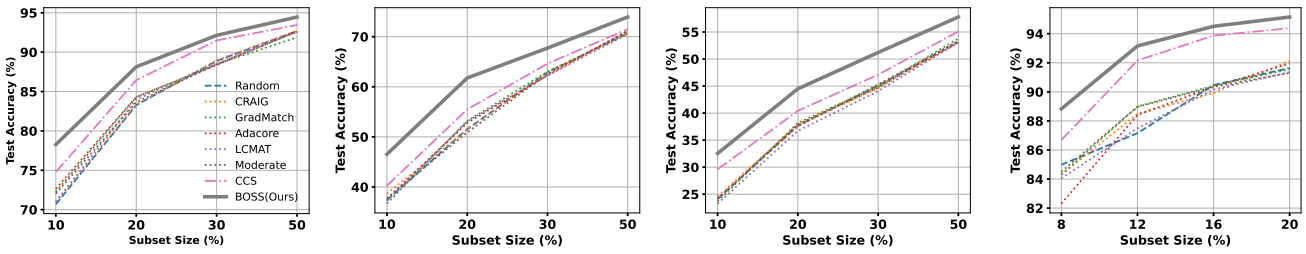
Subset Size	EL2N	CCS	YOCO	BOSS (Ours)
10%	30.78±0.03	59.10±0.01	54.93±0.04	**68.53**±0.1
20%	54.91±0.1	64.77±0.01	62.19±0.01	**69.74**±0.02
30%	66.41±0.1	67.93±0.02	65.71±0.01	**70.54**±0.03
50%	73.79±0.04	73.95±0.04	69.76±0.02	**74.28**±0.02

Table 9: ImageNet-10 YOCO's setting

Subset Size $\texttt{IPC}_\texttt{F}\rightarrow\texttt{IPC}_\texttt{T}$	EL2N	CCS	YOCO	BOSS (Ours)
20→1	24.09±0.7	34.64±0.2	53.07±0.3	**53.80**±0.3
20→2	33.16±0.4	42.22±0.2	**58.96**±0.4	58.93±1.5
20→5	46.02±0.2	57.11±0.2	64.38±0.4	**65.73**±1.1

D.4. Additional Ablation Study

Figure 13 further presents a performance-subset size plot for our proposed method and baseline methods to demonstrate the size-aware nature of our method.



(a) CIFAR10 (b) CIFAR100 (c) TinyImageNet (d) SVHN

Figure 13: Demonstration of the size-aware nature of our main result compared to the baselines.

Table 10: EfficientNet-B0 results for CIFAR100

Subset	Random	Moderate	CCS	BOSS
10%	30.51±1.0	32.59±1.3	36.91±2.2	**42.64**±0.6
20%	43.52±1.9	42.04±2.2	46.53±3.7	**53.39**±0.3
30%	55.48±0.7	55.26±1.7	56.89±0.3	**60.37**±0.4
50%	64.05±0.7	63.91±0.3	63.59±0.5	**68.27**±0.5

Table 11: ViT-B16 results for CIFAR100

Subset	Random	Moderate	CCS	BOSS
10%	78.49±0.7	50.41±0.7	78.62±0.3	**79.97**±0.4
20%	81.87±0.7	69.81±0.5	81.95±0.8	**83.19**±0.1
30%	83.98±0.2	77.66±0.5	84.93±0.1	**85.08**±0.1
50%	85.88±0.1	84.19±0.0	85.88±0.1	**86.55**±0.1

D.4.1. Results for Other Models

In Tables 10 and 11, we evaluate our method on two additional models: EfficientNet-B0 (Tan & Le, 2019) and a vision transformer ViT-B16 (Dosovitskiy et al., 2021). For EfficientNet, we use our previously mentioned setting. For ViT, we use pre-trained weights, batch size of 128, learning rate of 0.01, and train for 12 epochs. We compare with the two most recent and competitive baselines: CCS (Zheng et al., 2023), Moderate (Xia et al., 2023) and Random. Our method performs better than the baselines for both models. The performance margin for ViT is lower because we are using pre-trained weights and the room for improvement is small. Nonetheless, we show the usefulness of our method for other models than ResNet.

D.4.2. Imbalanced Datasets

Table 12 and 13 presents evaluation on imbalanced data. We consider two different methods to generate an imbalanced dataset. The first one is an exponential imbalance where the number of samples per class decreases with the factor of $N_{c_i} \times e^{-0.01i}$ and N_{c_i} is the number of samples for class c_i. The second one is the step-wise imbalance where 80% of data is removed from the 20% of classes. We compare our method with the most recent and competitive baselines such as CCS and Moderate. Our method can consistently outperform under both class imbalance settings.

Table 12: Imbalanced data result for CIFAR100 (Exponential)

Subset	Random	Moderate	CCS	BOSS
10%	27.39±0.9	25.37±1.7	29.41±0.5	**35.98**±0.8
20%	42.82±1.1	40.57±0.6	44.36±1.7	**49.74**±0.4
30%	52.51±0.7	50.00±3.0	50.87±1.4	**55.34**±1.8
50%	63.03±0.6	61.76±0.2	61.86±0.5	**66.37**±0.4

Table 13: Imbalanced data result for CIFAR100 (Step)

Subset	Random	Moderate	CCS	BOSS
10%	31.66±0.7	27.02±0.7	33.60±0.9	**40.41**±0.3
20%	47.36±0.9	41.77±2.9	46.82±0.6	**54.62**±0.7
30%	56.64±0.2	52.31±0.6	52.81±1.1	**58.59**±0.1
50%	62.19±0.9	59.96±0.7	60.25±0.2	**65.71**±0.4

D.5. Time Comparison

In Table 14, we compare the time taken by our method for *Subset Selection* and *Subset Training*. The *Subset Selection* time is shown for both *One-Shot* and *Dynamic* subset selection. The *One-Shot Selection* consists of time for initial training for 10 epochs on the full data which is fixed, and the time for the lazy greedy algorithm to select the subset that changes with the

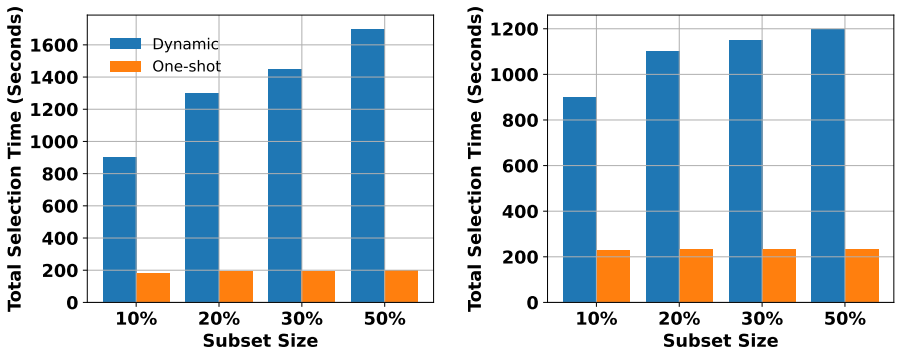
Table 14: Time comparison in seconds

Dataset	Subset Size	One-Shot Selection (Initial Training)	One-Shot Selection (Selection)	Dynamic Selection (Training Dynamics)	Dynamic Selection (Selection)	Subset Training	Full Set Training
CIFAR100	10%	219	9	450	450	346	4387
	20%		13		650	587	
	30%		14		700	800	
	50%		15		750	1816	
CIFAR10	10%	173	11	350	550	342	3468
	20%		19		950	571	
	30%		22		1100	801	
	50%		27		1350	1244	

subset size. The *One-Shot Selection* time is shorter compared to training on the subset (*Subset Training*) and also takes a very short time compared to training on the full set (*Full Set Training*). In the case of *One-Shot Selection*, the time for the subset selection algorithm (time excluding the initial training) is significantly small compared to the initial training time and also does not require GPU computation.

In contrast, there is no initial training for Dynamic Selection but the subset is selected every n epoch. For the comparison, we assume the subset is selected every 4 epochs for 200 epochs. Thus, the time required to get the features (*Training Dynamics*) using a single forward pass over the entire dataset along with the time required to select the subset (*Selection*) is multiplied by 200. The time comparison is further visualized in Figure 14. Given these results, dynamic selection is much more inefficient compared to one-shot since tens of selections can amount to a similar time cost to full training.

We measure the time in seconds using NVIDIA RTX A6000 GPU for CIFAR10 and CIFAR100 datasets.



(a) CIFAR10 (b) CIFAR100

Figure 14: Total selection time comparison between dynamic and one-shot subset selection.