Feature Distribution Matching by Optimal Transport for Effective and Robust Coreset Selection

Weiwei Xiao[1,2], Yongyong Chen[1], Qiben Shan[2], Yaowei Wang[2], Jingyong Su[1,2*]
[1]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen
[2]Pengcheng laboratory, Shenzhen
xiaowwhit@gmail.com, YongyongChen.cn@gmail.com, {shanqb,wangyw}@pcl.ac.cn, sujingyong@hit.edu.cn

Abstract

Training neural networks with good generalization requires large computational costs in many deep learning methods due to large-scale datasets and over-parameterized models. Despite the emergence of a number of coreset selection methods to reduce the computational costs, the problem of coreset distribution bias, i.e., the skewed distribution between the coreset and the entire dataset, has not been well studied. In this paper, we find that the closer the feature distribution of the coreset is to that of the entire dataset, the better the generalization performance of the coreset, particularly under extreme pruning. This motivates us to propose a simple yet effective method for coreset selection to alleviate the distribution bias between the coreset and the entire dataset, called feature distribution matching (FDMat). Unlike gradient-based methods, which selects samples with larger gradient values or approximates gradient values of the entire dataset, FDMat aims to select the coreset that is closest to the feature distribution of the entire dataset. Specifically, FDMat transfers coreset selection as an optimal transport problem from the coreset to the entire dataset in feature embedding spaces. Moreover, our method shows strong robustness due to the removal of samples far from the distribution, especially for the entire dataset containing noisy and class-imbalanced samples. Extensive experiments on multiple benchmarks show that FDMat can improve the performance of coreset selection than existing coreset methods. The code is available at https://github.com/successhaha/FDMat.

Introduction

Recently, foundation models in the visual and multi-modal domains have achieved unprecedented success (Ouyang et al. 2022). However, these high-performing models rely on hundreds of millions of data for training, resulting in significant computational costs. To explore the potential of achieving comparable performance for models on smaller dataset, there has been a growing interest in coreset selection methods. The objective of the coreset selection is to identify a subset of samples that serves as an effectively representative of the entire dataset.

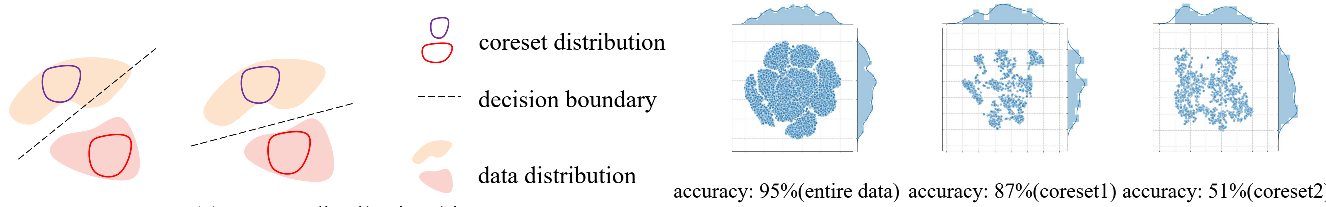
To identify samples that contribute to model learning, gradient-based methods (Killamsetty et al. 2021b; Paul, Ganguli, and Dziugaite 2021) typically assess the contribution of individual samples according to gradient values or gradient matching method (Killamsetty et al. 2021a) finds the coreset that is closest to the gradients of the entire dataset. Although gradient-based methods offer sufficient theoretical proofs, robustness is rarely guaranteed for datasets containing gradient-abnormal noise samples, due to the reliance on gradient judgments. As a result, gradient-based methods tend to exclude some samples with a high gradient to minimize perturbations from noise samples (Paul, Ganguli, and Dziugaite 2021). If the diversity of samples in coreset is not enough to represent the entire dataset or contains outlier samples that deviate from the true distribution, the coreset exhibits distribution bias, also known as sample selection bias (Cortes et al. 2008). Furthermore, the performance of the gradient-based method deteriorates significantly under extreme pruning conditions (such as retaining less than 40% of training samples) due to insufficient diversity of samples (Killamsetty et al. 2021b; Paul, Ganguli, and Dziugaite 2021).

In this paper, we delve into the distribution bias of coreset selection, which refers to the bias of feature distribution in feature embedding space. Intuitively, the coreset is a highly condensed version of the entire dataset, and its distribution should closely resemble that of the entire dataset. When the distribution of the coreset deviates significantly from that of the entire dataset, the generalization performance of the classifier may suffer. These phenomena are shown in Fig. 1(a). The distribution bias of the coreset could cause the decision boundary of the model to deviate, which in turn affects the generalization ability of the model on testing samples.

To validate our intuition, we randomly choose 5 classes from the MINIST (Lecun and Bottou 1998) as the same class and train a binary classification network. Then we train the network using two coresets : randomly sampling 10% and skewed sampling 10% of the entire dataset. In Fig. 1(b), we visualize the feature distribution of coresets and the entire dataset, and then evaluate the accuracy of coresets with different feature distributions. As expected, the less distribution bias and the more similar distribution between the coreset and the entire dataset in feature embedding spaces, the higher test accuracy of the coreset.

Our method is simple yet effective and exhibits strong robustness, especially when the entire dataset contains class-

*Corresponding Authors.



(a) coreset distribution bias

(b) coreset distribution

Figure 1: (a) The decision boundary is extremely sensitive to the position of coreset distribution in coreset selection, resulting in high performance fluctuations, especially when the coreset distribution is far from the centroid of the entire distribution. (b) The test accuracy of the model is higher, when the coreset distribution is closer to the entire distribution.

imbalanced and noisy samples, which removes samples far away from the distribution. Our contributions are summarized as follows:

- We propose an efficient coreset method to discover important samples by alleviating the distribution bias between the coreset and the entire dataset through optimal transport in deep learning scenarios.
- We propose the supremum of maximum mean discrepancy (MMD) between coreset and entire dataset can be solved by minimizing 1-Wasserstein distance in feature embedding space.
- Extensive experiments show that FDMat outperforms existing methods under extreme pruning and noise label datasets, which guarantees closest distribution of coreset to the entire dataset by feature distribution matching.

Related Work

Most current foundation models are computationally expensive due to training on large datasets, resulting in increased energy costs. As a result, there has been a growing interest in improving data-efficiency. In this paper, we primarily focus on coreset selection methods to reduce computational costs. Initially introduced in computational geometry (Agarwal et al. 2005), coresets have been quickly adopted later by machine learning community to address classical problems, such as Bayesian inference (Huggins, Campbell, and Broderick 2016), K-means (Har-Peled and Mazumdar 2004) and more.

The traditional coreset methods can be roughly divided into two groups, one is **sensitivity-based** important sampling coresets, and the other is **distribution-based** coresets under certain conditions. **Sensitivity-based methods**, such as the k-clustering problems take into account the importance of samples by approximate probability (Bachem, Lucic, and Lattanzi 2018; Bateni et al. 2014). **Distribution-based** methods typically require consideration of the underlying data distribution, such as designing the coreset based on Reproducing Kernel Hilbert Space (RKHS) theory (Chen, Welling, and Smola 2012) or utilizing the integral probability metric in the context of optimal transport theory (Claici, Genevay, and Solomon 2018). However, these traditional coreset methods (Feldman, Faulkner, and Krause 2011; Bachem, Lucic, and Krause 2015; Zhang et al. 2023) face challenges due to their high computational complexity and reliance on fixed data representations (which are seldom suited for image data). As a result, doubts have arisen regarding their effectiveness in deep learning scenarios.

With the increasing prominence of data-driven methods, coreset selection has emerged as a focal point of research in deep learning scenarios. In this context, coreset selection can be roughly categorized into two main groups: **gradient-based** and **decision boundary-based** methods.

Gradient-based methods aim to find the coreset in which the gradient aligns with minimal error to the gradient of the entire dataset, thereby ensuring that the model exhibits comparable generalization performance on the coreset. Some recent works include Craig (Mirzasoleiman, Bilmes, and Leskovec 2020), GradMatch (Killamsetty et al. 2021a), Glister (Killamsetty et al. 2021b), AdaCore (Pooladzandi, Davini, and Mirzasoleiman 2022), LCMat (Shin et al. 2023), etc. Craig (Mirzasoleiman, Bilmes, and Leskovec 2020) identifies the coreset by formulating gradient matching as an optimization problem of a monotonic submodular function (Fujishige 2005) under a first-order gradient error bound. Similar to Craig, GradMatch (Killamsetty et al. 2021a) incorporates L2 regularization on the basis of first-order error to reduce reliance on specific samples, and leverages orthogonal matching pursuit (OMP) (Elenberg et al. 2016) to select samples. Crust (Mirzasoleiman, Cao, and Leskovec 2020) extents the coreset selection method to handle noisy label scenarios while maintaining the first-order error. Building upon first-order error, AdaCore (Pooladzandi, Davini, and Mirzasoleiman 2022) extends the gradient error range to second-order, resulting in improved gradient matching through Hessian matrix computations. LCMat (Shin et al. 2023) further refines the second-order gradient error by introducing sharpness-aware minimization (Foret P. 2020). Gradient-based methods have a strong theoretical basis, and their convergence has been analyzed. However, the diversity of samples in coreset under extreme pruning conditions is not well guaranteed, which ignores the relationship between samples.

Decision boundary-based methods consider that samples that are difficult to distinguish near the decision boundary are beneficial to construct the coreset. Deepfool (Moosavi-Dezfooli, Fawzi, and Frossard 2016) constructs coreset based on the minimum perturbation required for samples, which applying perturbation until the predicted labels change. Similarly, Cal (Margatina et al. 2021) constructs the coreset by comparing the maximum deviation between the predicted likelihood of samples and their neighbors. Additionally, some methods implicitly use decision boundaries to select coreset in feature embedding spaces. Herding (Welling 2009) constructs the coreset using a distance-based approach that greedily chooses samples to minimize the distance between the centers of the coreset and the entire dataset in feature space. K-Center Greedy (Sener and Savarese 2017) further defines coreset selection as minimizing the maximum distance between the coreset and the entire dataset. However, these methods are slower to run on models with over-parameters, some methods use a proxy model to speed up the process of coreset selection (Coleman et al. 2020; Sachdeva, Wu, and McAuley 2021). SVP (Coleman et al. 2020) accelerates the coreset selection process by using a lighter model as a proxy, and selecting coreset on the proxy model using metrics such as decision boundary (margin) or uncertainty.

Although the gradient-based method has sufficient theoretical support, it is susceptible to noise samples and labels, leading to reduced robustness of the coreset. On the other hand, the decision boundary-based method can find samples far away from the distribution, but it lacks sufficient theoretical support. Therefore, we revisit the traditional distribution-based coreset methods (Claici, Genevay, and Solomon 2018; Chen, Welling, and Smola 2012) and design FDmat based on optimized transport theory for deep learning scenarios. In contrast to (Claici, Genevay, and Solomon 2018), we redefine the maximum mean discrepancy with the 1-Lipschitz constraint instead of RKHS, enabling it to be applied to feature space for deep learning methods. We then formulate it as a dual problem to solve for the 1-Wasserstein distance. Additionally, we expedite the computation of the 1-Wasserstein distance using the Sinkhorn (Cuturi 2013) method instead of stochastic gradient descent.

Methodology

In this section, we introduce the coreset selection preliminaries and the details of our method.

Preliminaries

We focus on dataset selection for classification task, which is a widely studied scenario in machine learning community. We are given a training set $\mathcal{U} = \{(x_i, y_i)\}_{i=1}^n$ and a coreset $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^m$ from unknown Borel probability distribution $\mathcal{P}$ and $\mathcal{Q}$, where $x_i \in \mathcal{X}$ represents the input, $y_i \in \mathcal{Y}$ represents the label of x_i, m and n are sample numbers. Let f_θ be the feature extractor and f_θ^c be a classifier of c classes. The goal of coreset selection is to find the most representative coreset $\mathcal{S}$ with the constraint $\mathcal{S} \subset \mathcal{U}$, so that the model $\theta^{\mathcal{S}}$ trained on $\mathcal{S}$ has closer generalization performance to the model $\theta^{\mathcal{U}}$ trained on the entire dataset $\mathcal{U}$.

Feature-Distribution Matching (FDMat)

As shown in Fig. 1(b), the more similar distribution between the coreset and the entire dataset, the higher accuracy of the coreset. Therefore, we devote to developing a coreset method that has a distribution closest to that of the entire dataset while maintaining strong robustness. In this section, we introduce the optimal objective of feature distribution matching and prove its supremum. Additionally, we elaborate how to select a coreset by iteratively solving feature distribution matching.

Optimal Objective of Feature Distribution Matching This section introduces an objective for coreset selection, called FDMat, which matches the feature distribution of $\mathcal{U}$ and $\mathcal{S}$ based on feature extractor f_θ pre-trained with $\mathcal{U}$. The discrepancy between the feature distribution of $\mathcal{U}$ and $\mathcal{S}$ is our primary focus.

Definition 3.1 Let the shorthand notation $E_{x\sim\mathcal{P}}[f_\theta(x)]$ and $E_{x\sim\mathcal{Q}}[f_\theta(x)]$ to denote expectation with respect to $\mathcal{P}$ and $\mathcal{Q}$, respectively.

The goal of feature distribution matching is to minimize the maximum mean discrepancy between the coreset $\mathcal{S}$ and the entire dataset $\mathcal{U}$ in feature embedding spaces. Following the optimization scheme, we formulate the primary objective as follows:

$$\underset{\mathcal{S}}{min}\, max(E_{x\sim\mathcal{P}}[f_\theta(x)] - E_{x\sim\mathcal{Q}}[f_\theta(x)]) \qquad (1)$$

In Eq. (1), our goal is first to find the supremum of the maximum mean discrepancy between $\mathcal{U}$ and $\mathcal{S}$, and then minimize this supremum on coreset $\mathcal{S}$.

Theorem 3.2 If f_θ is the 1-Lipschitz function, then the supremum of MMD can be obtained from the Kantorovich-Rubinstein duality (Hörmander, Totaro, and Waldschmidt 2006).(Proof in Appendix A.1 of the supplementary material)

$$\mathcal{W}_1(\mathcal{P}, \mathcal{Q}) = \underset{\|f_\theta\|_L \le 1}{sup} (E_{x\sim\mathcal{P}}[f_\theta(x)] - E_{x\sim\mathcal{Q}}[f_\theta(x)]) \quad (2)$$

where $\mathcal{W}_1$ is Wasserstein distance. Therefore, the goal can be defined as the optimal transport problem : finding a mapping route that minimizes the cost of distributed transmission from the coreset $\mathcal{S}$ to the entire dataset $\mathcal{U}$.

Wasserstein distance Let $\mathcal{P} = \sum_{k=1}^{n} p_i \delta_{\mathcal{S}_i}, \mathcal{Q} = \sum_{k=1}^{m} q_i \delta_{\mathcal{U}_i}$ be two discrete distributions, described by their supports $(\mathcal{S}_i)_{i=1}^n \in \mathbb{R}^{n\times d}$ and $(\mathcal{U}_i)_{i=1}^m \in \mathbb{R}^{m\times d}$ and weight vectors $p \in \triangle_n$ and $q \in \triangle_m$.

According to **Theorem 3.2**, we get the supremum of MMD. Therefore, the optimal solution can be defined as the case where Wasserstein distance corresponds to the first-order ground cost.

$$\mathcal{W}_1(\mathcal{P}, \mathcal{Q}) = \underset{P\in U(p,q)}{min} < P, M >= \sum_{i,j} P_{i,j} M_{i,j} \qquad (3)$$

where $U(p,q) = \{P \in \mathbb{R}_+^{n\times m} \colon P1_m = p, P^T 1_n = q\}$ represents the mapping transport from distribution p to distribution q, $P1_m = (\sum_j P_{i,j})_i \in \mathbb{R}^n$ and $P^T 1_n = (\sum_i P_{i,j})_j \in \mathbb{R}^m$ are the matrix-vectors, $M = (\|\mathcal{S}_i - \mathcal{U}_j\|)_{ij} \in \mathbb{R}^{n\times m}$ is the matrix of pairwise Euclidean distances between the supports, m and n are sample numbers.

Algorithm 1: FDMat:Feature Distribution Matching

Input: Training data $\mathcal{U} = \{x_i, y_j\}^n$, coreset size k, feature extractor f_θ, class w, hyperparameter λ
Output: coreset $\mathcal{S}$
Training Variables:feature extractor f_θ pre-trained on $\mathcal{U}$
for $(x_i, y_i) \in \mathcal{U}$ **do**
  Transform x_i with Tukey's Ladder of Powers by Eq. (4)
  Approximate feature distribution c_j by Eq. (5)
  Calculate the distance matrix $M_{i,j}$ by Eq. (6)
end
Initialize edge distribution : $p \leftarrow 1_n$, $q \leftarrow |c_j|1_w$
Optimal transport distance :
$d_M^\lambda(p,q) = Sinkhorn(M,p,q,\lambda)$ by Eq. (8)
return $\mathcal{S} = \underset{k}{topK}(\underset{i}{argmin}\ d_{M_{i,j}}^\lambda(p,q))$

Solving the Distribution Matching Problem

In this section, we provide a detailed description of how to solve the feature distribution matching and find the coreset.

Reduce the distribution skew As the distribution of entire dataset $\mathcal{U}$ is unknown, we use Tukey's Ladder of powers transformation (Keyfitz 1977) to correct the feature distribution and reduce the distribution skew. This technique is commonly used in few-shot learning to alleviate distribution bias (Wang et al. 2019; Tao et al. 2022). Tukey's Ladder of Powers transformation is formulated as:

$$f_\theta = \begin{cases} (f_\theta)^\beta / \|(f_\theta)^\beta\|_2 & \beta \neq 0 \\ log(f_\theta)/\|log(f_\theta)\|_2 & \beta = 0 \end{cases} \tag{4}$$

where β is a hyper-parameter to adjust the degree of distribution skew. When β is set to 1, the original feature can be recovered. Decreasing the value of β phases out the positive skew in the distribution, while increasing it has the opposite effect. We set β to an empirical value of 0.5.

Approximate the estimated feature distribution By applying distribution correction, we assume that the feature distribution of the entire dataset $\mathcal{U}$ follows a Gaussian-like distribution. We then approximate the feature distribution of the entire dataset $\mathcal{U}$ as follows:

$$c_j = \frac{\sum_{i=1}^{n_j} f_\theta(x_i)}{n_j} \tag{5}$$

where $f_\theta(x_i)$ denotes a feature vector of the i-th sample of class j in dataset $\mathcal{U}$ and n_j denotes the total number of samples in class j. Then, the distance matrix M in the optimal transport can be obtained as follows:

$$M_{i,j} = \|f_\theta(x_i) - c_j\| \tag{6}$$

Solve the feature distribution matching Computing the $\mathcal{W}_1$ in Eq. (3) requires solving a computationally expensive linear program. However, an approximate solution of the optimal transport problem can be obtained by entropy regularization (Peyré, Cuturi et al. 2019; Cuturi 2013).

$$d_{M_{i,j}}^\lambda(p,q) = \min_{P \in U(p,q)} \sum_{i,j} P_{i,j} M_{i,j} - \lambda H(P) \tag{7}$$

where $H(P) = -\sum_{i,j} P_{i,j}(log(P_{i,j}) - 1)$ represents the entropy regularization, $\lambda > 0$ represents the regularization parameter used to control the degree of the entropy regularization. Sinkhorn is used to iteratively solve Eq. (7) as follows:

$$d_{M_{i,j}}^\lambda(p,q) = Sinkhorn(M,p,q,\lambda) \tag{8}$$

Through Sinkhorn algorithm (Cuturi 2013), we get the $\mathcal{W}_1$ distance of optimal transport.

The entire algorithm is shown in Algorithm 1. FDMat mainly contains three steps :

- Step 1 : Using Tukey's Ladder to reduce the skewed distribution and then approximate the distribution of entire dataset $\mathcal{U}$;
- Step 2 : Calculating the optimal transport distance from the training samples to the approximate distribution of entire dataset $\mathcal{U}$ by Sinkhorn;
- Step 3 : Obtaining the coreset $\mathcal{S}$ based on the size of the coreset and the transport cost of samples.

Since FDMat selects the coreset that is closest to the class center, it can effectively eliminate noise samples that far away from the distribution.

Experiments

This section examines the effectiveness and robustness of FDMat method through various experiments.

Implementation Details and Baselines

Implementation details Most current coreset selection methods have been evaluated in different experimental settings, such as datasets, model architectures, coreset sizes, augmentations, training strategies. These differences may lead to unfair comparisons between different methods and unconvincing results. To investigate the effectiveness of coreset selection methods under a fair and unified framework, we follow the coreset selection scenario of these methods (Shin et al. 2023; Guo, Zhao, and Bai 2022).

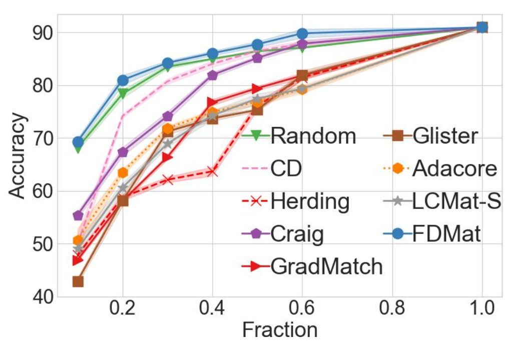
We evaluate FDMat on three widely-used datasets and one medical dataset for coreset selection : CIFAR10, CIFAR100 (Krizhevsky and Hinton 2009), Tiny-ImageNet (Russakovsky et al. 2015) and Path-MNIST (Yang, Shi, and Ni 2021). Datasets with varying levels of granularity, such as CIFAR100 and Tiny-ImageNet which respectively contain 100 and 200 classes of samples, may exhibit different feature spatial distributions(Liang and Zou 2022). The objective of our evaluation is to show the effectiveness and universality of our method across datasets with different distributions.

Baselines To demonstrate the effectiveness of FDMat under extreme pruning conditions, we compare various coreset methods in recent years. The comparison baselines can be divided into two primary categories: the first category is based on the forward-output of the model, including layer feature vector and softmax output, such as C-Div(Agarwal et al. 2020), Herding(Welling 2009), K-Center(Sener and Savarese 2017), Least Confidence(Coleman et al. 2020), Entropy(Coleman et al. 2020), and Margin(Coleman et al.

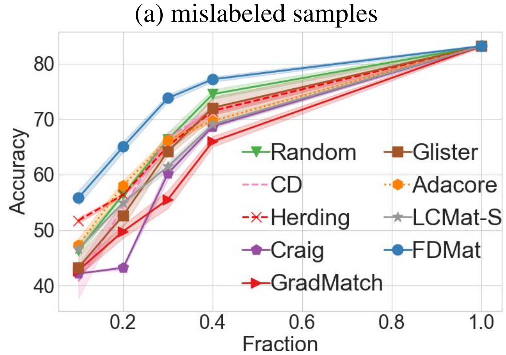
	CIFAR-10					CIFAR-100				
Fraction	10%	20%	30%	40%	100%	10%	20%	30%	40%	100%
Random	77.45±1.0	87.36±0.4	90.67±0.2	92.10±0.3		34.09±2.4	55.98±0.7	64.59±0.1	68.65±0.1	
C-Div	56.85±1.7	81.30±2.5	90.93±0.5	93.30±0.4		20.53±0.6	44.91±1.1	58.60±2.7	66.21±1.5	
Herding	63.04±2.5	74.10±2.5	79.93±1.5	85.20±0.9		26.47±0.2	42.83±1.9	52.14±1.4	60.99±0.5	
K-Center	72.12±1.7	87.10±0.3	90.83±0.3	92.80±0.1		27.37±1.5	52.10±0.8	63.74±0.7	68.88±0.2	
L-Conf	58.43±3.0	81.90±2.2	91.21±0.1	93.10±0.5		17.63±2.1	41.29±1.1	58.86±1.0	62.41±0.3	
Entropy	57.45±3.6	81.90±0.4	91.06±0.7	93.20±0.2		16.94±0.9	41.88±1.3	57.45±2.0	63.74±0.4	
Margin	59.90±6.7	81.70±3.2	90.92±0.4	92.90±0.2	**95.48±0.1**	20.70±1.1	46.36±2.7	59.45±2.2	66.56±0.3	**78.91±0.2**
Craig	51.73±4.6	79.60±3.1	87.25±0.8	90.80±1.4		20.32±0.6	32.23±0.2	47.09±1.4	66.94±0.5	
GradMatch	51.11±2.3	79.90±2.6	84.88±1.4	90.40±1.5		20.23±0.5	40.28±1.1	51.03±1.5	57.50±0.2	
Glister	56.89±2.7	84.80±0.9	89.37±0.4	93.10±0.2		24.07±0.4	44.42±1.4	56.81±1.2	64.90±0.6	
AdaCore	76.44±1.5	86.86±0.4	90.54±0.4	92.20±0.1		35.26±1.8	56.54±0.6	63.87±0.4	64.10±0.7	
LCMat-S	77.41±2.0	87.70±0.4	91.32±0.2	92.30±0.3		36.66±1.0	56.66±0.6	64.81±0.9	68.30±1.1	
FDMat	**80.29±2.1**	**88.40±0.5**	**91.70±0.4**	**93.80±0.3**		**38.25±1.2**	**57.73±1.6**	**66.48±0.2**	**70.16±0.2**	

Table 1: The accuracy of coreset selection methods on CIFAR-10 and CIFAR-100 using ResNet-18 over 5 different random seeds. The best and second-best results for each setting are highlighted in bold and underlined, respectively.

2020). The second category is based on the weight parameters of the model, such as Craig(Mirzasoleiman, Bilmes, and Leskovec 2020), GradMatch(Killamsetty et al. 2021a), Glister(Killamsetty et al. 2021b), AdaCore(Pooladzandi, Davini, and Mirzasoleiman 2022), LCMat(Shin et al. 2023). Moreover, as random selection is more adaptable in most scenarios, we have also included a comparison with the random selection method. For all baselines, we adopt class-balance setting when selecting coreset $\mathcal{S}$.



(a) mislabeled samples



(b) class-imbalanced samples

Figure 2: The performance of different coreset selection methods on the entire dataset $\mathcal{U}$ with (a) mislabeled samples and (b) class-imbalanced samples.

Evaluation on Benchmarks

We first evaluate the performance of the coreset under different pruning ratios on CIFAR10 and CIFAR100. Table 1 reports accuracy of various coreset methods with ResNet-18 (He et al. 2016). Interestingly, the results in Table 1 show that the random selection achieves competitive performance compared to most existing coreset selection methods under extreme pruning ($10\% - 30\%$), indicating weakness in the robustness of current methods. However, FDMat performs significant improvement over other coreset methods, as it guarantees the similarity of distribution between the coreset and the entire dataset. Notably, our FDMat achieves 3.7% and 4.3% improvement over the second-best method in terms of accuracy on CIFAR10 and CIFAR100, respectively, at the 10% pruning ratio in Table 1. Moreover, to evaluate the performance of coreset methods with a larger number of classes, we conduct experiments on Tiny-ImageNet, and the results are presented in Table 2. Table 2 shows that FDMat consistently outperforms other coreset selection baselines on Tiny-ImageNet. It should be noted that most existing coreset selection methods tend to perform poorly when faced with large datasets containing a great number of classes, such as Craig, Glister and GradMatch. To some extent, this experiment indicates that the exisiting coreset methods are weak in generalization and robustness.

Finally, we also evaluate the effectiveness of FDMat on the unnatural dataset, medical image dataset (Yang, Shi, and Ni 2021). Table 2 shows that FDMat method still has a good performance than other methods.

Robustness Analysis

In this section, we conduct robustness experiments of coreset methods in various situations including mislabeled samples, class imbalance samples, cross-architectures, and pretrained feature extractors.

Mislabeled Samples Similar to (Killamsetty et al. 2021b), we artificially construct a mislabeled samples dataset on CIFAR10 to investigate the impact of mislabeled sam-

	Tiny-ImageNet					Path-MNIST				
Fraction	10%	20%	30%	40%	100%	10%	20%	30%	40%	100%
Random	12.81±0.3	20.43±0.1	24.85±0.1	28.71±0.1		71.30±2.1	85.76±1.4	86.65±0.1	86.67±0.1	
C-Div	5.29±2.3	10.21±1.7	16.55±1.2	20.03±1.3		36.75±0.4	75.37±0.3	78.24±0.4	82.53±0.2	
Herding	11.52±1.4	18.12±0.9	24.01±0.7	28.00±0.7		38.05±1.2	80.06±0.3	80.47±0.4	82.17±0.2	
K-Center	8.26±1.8	18.12±0.9	24.01±0.7	28.00±0.7		77.45±0.4	83.18±0.4	86.51±0.2	87.29±0.2	
L-Conf	4.8±0.5	10.15±2.2	16.52±0.1	23.10±0.5		72.24±0.3	76.72±0.3	83.85±0.3	85.53±0.2	
Entropy	4.96±3.6	11.25±0.4	16.34±0.7	20.96±0.2		74.62±0.2	77.06±0.3	83.28±0.4	86.60±0.2	
Margin	7.92±6.7	13.52±3.2	17.90±0.4	22.36±0.3	**43.03±0.1**	63.73±0.3	71.75±0.3	84.86±0.2	86.45±0.2	**88.45±0.1**
Craig	8.34±1.1	11.13±1.2	17.10±0.9	18.01±0.7		69.33±0.4	74.41±0.4	81.04±0.4	82.17±0.2	
GradMatch	11.42±0.9	14.70±1.6	19.72±0.8	24.12±0.5		54.19±0.4	70.58±0.2	78.87±0.2	81.59±0.2	
Glister	11.30±0.6	16.49±1.1	20.08±0.5	23.03±0.3		60.34±0.3	75.52±0.2	76.45±0.2	81.17±0.2	
AdaCore	12.95±0.4	21.03±0.3	25.50±0.2	29.65±0.2		83.10±0.2	85.89±0.2	87.76±0.2	88.16±0.1	
LCMat-S	13.04±0.3	18.30±0.2	23.59±0.2	27.73±0.1		80.94±0.2	83.24±0.2	86.17±0.1	87.21±0.1	
FDMat	**13.88±0.6**	**21.60±0.3**	**26.04±0.4**	**30.13±0.3**		**83.96±0.2**	**87.14±0.3**	**88.10±0.2**	**88.40±0.1**	

Table 2: The accuracy of coreset selection methods on Tiny-ImageNet and Path-MNIST with ResNet 18 over 5 different random seeds. The best and second-best results for each setting are highlighted in bold and underlined, respectively.

ples. Fig. 2(a) reports the experimental results under mislabeled dataset. The results show that the performance of the gradient-based methods significantly decreases under extreme pruning, due to the abnormal gradient disturbance caused by the presence of noisy samples. In contrast, FDMat outperforms the other coreset methods by a significant margin, particularly under extreme pruning of a 10% fraction, where the performance is nearly 50% better than that of the second-best coreset method. Moreover, we find that FDMat can outperform the entire mislabeled dataset using only 60% of samples. This indicates that the gradient-based coreset selection methods could suffer from significant selection bias in the presence of noisy samples, whereas FDMat can effectively eliminate noisy samples that deviate from the underlying distribution. This experiment verifies the robustness of FDMat method from the empirical level to some extent.

Class imbalance samples We then study the effectiveness of coreset methods under class imbalance on CIFAR10. To this end, we manually conduct the class-imbalanced experiments, randomly selected three classes and save 50% of samples on CIFAR10. The experimental results are presented in Fig. 2(b). We observe that selecting a coreset using a feature extractor trained on a class-imbalanced dataset leads to more severe sample selection bias. Even effective random selection method exhibits significant performance degradation in multiple scenarios. However, our FDMat significantly outperforms other coreset methods. Overall, our results suggest that ensuring the similarity of distributions is more effective than considering the magnitude of gradient value when dealing with class-imbalanced datasets.

Robustness on cross-architecture We also study the generalization performance of FDMat under cross-architecture. Specifically, we use the ResNet-18 to select coreset samples, and then train the coreset samples on other architectures, including WRN-16-8 (Zagoruyko and Komodakis 2016), VGG-16 (Simonyan and Zisserman 2015) and Inception-v3 (Szegedy et al. 2016). Table 3 shows that the data selected by FDMat consistently achieve improved or competitive performance across different network structures compared with other coreset methods. Moreover, this experiment also reveals a certain degree of consistency in the performance of coreset samples selected by different architectures.

Fraction	Method	ResNet-18	VGG-16	Inception	WRN-16-8
10%	Random	33.9±1.1	35.6±0.7	**35.1±0.6**	47.3±0.4
	Craig	21.1±0.1	13.7±0.3	20.0±0.4	24.1±0.6
	GradMatch	23.9±0.5	23.4±0.4	23.5±0.3	33.1±0.3
	Glister	26.1±0.2	26.0±0.7	25.2±0.4	33.9±0.4
	AdaCore	35.6±0.3	35.7±0.3	34.1±0.4	48.3±0.7
	LCMat-S	35.8±0.4	33.2±1.2	34.2±0.6	48.4±0.5
	FDMat	**36.5±0.3**	**36.2±0.4**	34.4±0.6	**48.7±0.3**
20%	Random	55.5±0.4	53.1 ±0.3	53.2±1.1	61.8±1.4
	Craig	40.8±1.7	23.4±1.6	45.0±1.3	42.8±0.7
	GradMatch	39.6±1.9	35.9±0.4	41.4±1.2	50.7±0.6
	Glister	42.5±1.1	39.2±0.5	29.8±0.4	50.8±0.3
	AdaCore	54.9±0.9	53.5±0.6	51.3±1.3	62.2±0.4
	LCMat-S	55.6±0.7	52.6±0.3	55.3±0.8	62.1±0.8
	FDMat	**57.2±0.6**	**53.6±0.5**	**58.4±0.6**	**62.8±0.4**

Table 3: Cross-architecture generalization performance(%) on CIFAR-100 with ResNet-18.

Robustness on the pre-training We also evaluate the robustness of the coreset methods under different training hyperparameters. Specifically, we conduct multiple experiments with combinations of selected epochs [1, 5, 10, 15, 20]; weight decay [1e-4, 5e-4, 1e-3]; optimizers [SGD, Adam]; and 3 seeds, which result in a total of 90 cases on Tniy-ImageNet. Fig. 3 shows the number of times that each method beats the others in these experiments. Notably, FDMat is superior to other methods in most cases and exhibits strong robustness across different training parameters.

Continual Learning with Memory Replay

Memory-based continual learning methods store small representative instances and optimize their classifiers using the

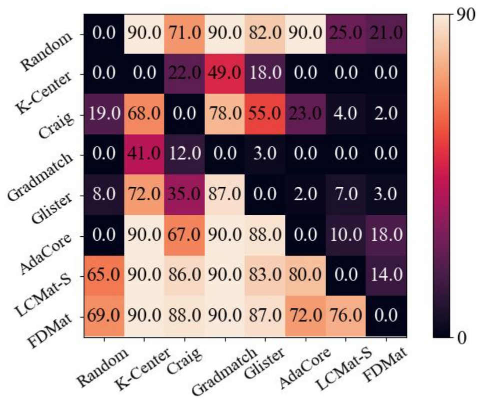


Figure 3: Performance of robustness. The comparison of robustness of different coreset selection methods under different training hyperparameters on Tiny-ImageNet dataset of 10% fraction.

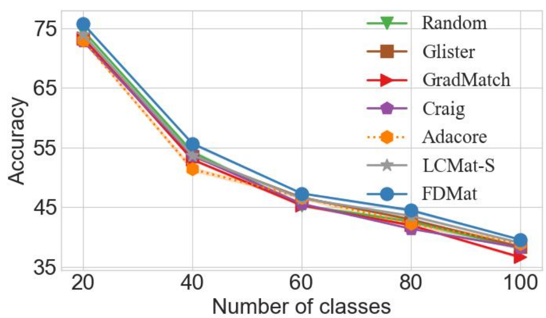


Figure 4: Memory replay. Test accuracy of coreset selection methods in continuous learning scenario on CIFAR100.

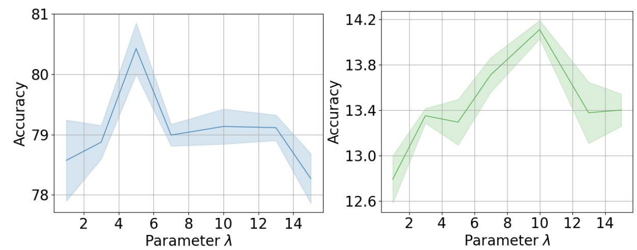
samples stored in memory to alleviate catastrophic forgetting of previously observed tasks. As an application, we use coreset $\mathcal{S}$ as a memory exemplar for previously seen classes under the class incremental setting of the method (Zhao, Mopuri, and Bilen 2020). In this setting, CIFAR-100 is divided into 5 sets of sub-classes, with a memory budget of 50 images per class. Each set of classes represents a separate task stage. The model is trained purely based on the latest memory at each task stage. Fig. 4 shows that FDMat outperforms existing coreset methods under memory replay setting, which represents that the sample selected by FDMat has an anti-catastrophic forgetting effect to a certain extent.

Efficiency Analysis

In Table 4, the selection efficiency of various coreset methods on CIFAR10 is shown. It can be seen from the experimental results that FDMat has certain advantages in selection efficiency compared with the method using submodular function, Gradmatch, Glister, Adacore, LCMat-s. The AdaCore and LCMat-S, which leverage second-order gradients and involve the computation of Hessian matrix, exhibit longer selection times. GradMatch employs an OMP algorithm with substantial computing consumption, resulting in an extremely time-consuming process. Our FDMat method

Fraction	10%	20%	30%
Herding	15.88	17.65	22.14
K-Center	48.52	49.11	54.04
L-Conf	22.51	22.55	22.68
Entropy	22.69	23.56	24.28
Margin	22.08	22.21	22.26
Craig	59.39	61.88	65.56
GradMatch	1395.73	2157.49	4085.69
Glister	98.35	172.84	394.85
Adacore	762.08	1407.62	2150.39
LCMat-S	803.45	1543.99	2259.23
FDMat	31.96	33.70	44.04

Table 4: Selection time analyses of coreset methods on CIFAR10(seconds)



(a) CIFAR10 (b) Tiny-ImageNet

Figure 5: The ablation experiments of hyperparameter λ with 10% selection ratio on (a) CIFAR10 and (b) Tiny-ImageNet.

strikes a balance between performance and efficiency, offering a reliable compromise between the two factors.

Ablation Study

When solving the optimal transport distance iteratively using Eq. (8), the hyperparameter λ can affect the choose of coreset samples. To evaluate the impact of the hyperparameter λ, we conduct a sensitivity analysis of λ. Fig. 5 illustrates that FDMat performs better with $\lambda = 5$ on CIFAR10 and $\lambda = 10$ on Tiny-ImageNet. Our sensitivity analysis reveals that the optimal value of λ for the coreset selection method typically falls between 5 and 10, as demonstrated by numerous experiments.

Conclusion

We proposed a novel objective for coreset selection, called FDMat, which aims to reduce the maximum mean discrepancy between coreset and entire dataset in feature embedding spaces. By ensuring that the coreset distribution is closest to the entire dataset, FDMat significantly outperforms existing coreset methods under extreme pruning. Notably, FDMat exhibits strong robustness in handling noise and class-imbalanced samples by eliminating samples far from the distribution. Moreover, FDMat shows clear performance merits for continual learning.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grants 62376068 and 62106063, by the Shenzhen Science and Technology Innovation Program under Grant JCYJ20220818102414031.

References

Agarwal, P. K.; Har-Peled, S.; Varadarajan, K. R.; et al. 2005. Geometric approximation via coresets. *Combinatorial and computational geometry*, 52(1): 1–30.

Agarwal, S.; Arora, H.; Anand, S.; and Arora, C. 2020. Contextual diversity for active learning. In *European Conference Computer Vision*.

Bachem, O.; Lucic, M.; and Krause, A. 2015. Coresets for nonparametric estimation-the case of DP-means. In *International Conference on Machine Learning*.

Bachem, O.; Lucic, M.; and Lattanzi, S. 2018. One-shot coresets: The case of k-clustering. In *International conference on artificial intelligence and statistics*, 784–792. PMLR.

Bateni, M.; Bhaskara, A.; Lattanzi, S.; and Mirrokni, V. 2014. Distributed balanced clustering via mapping coresets. In *Advances in Neural Information Processing Systems*.

Chen, Y.; Welling, M.; and Smola, A. 2012. Super-samples from kernel herding. In *Conference on Uncertainty in Artificial Intelligence*.

Claici, S.; Genevay, A.; and Solomon, J. 2018. Wasserstein measure coresets. *arXiv preprint arXiv:1805.07412*.

Coleman, C.; Yeh, C.; Mussmann, S.; Mirzasoleiman, B.; Bailis, P.; Liang, P.; Leskovec, J.; and Zaharia, M. 2020. Selection via proxy: Efficient data selection for deep learning. In *International Conference on Learning Representations*.

Cortes, C.; Mohri, M.; Riley, M.; and Rostamizadeh, A. 2008. Sample selection bias correction theory. In *International Conference on Algorithmic Learning Theory*.

Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*.

Elenberg, E. R.; Khanna, R.; Dimakis, A. G.; and Negahban, S. N. 2016. Restricted Strong Convexity Implies Weak Submodularity. *ArXiv*, abs/1612.00804.

Feldman, D.; Faulkner, M.; and Krause, A. 2011. Scalable training of mixture models via coresets. In *Advances in neural information processing systems*.

Foret P., A. M. H. N. B., Kleiner. 2020. Sharpness-Aware Minimization for Efficiently Improving Generalization. In *International Conference on Learning Representations*.

Fujishige, S. 2005. *Submodular functions and optimization*. Elsevier.

Guo, C.; Zhao, B.; and Bai, Y. 2022. Deepcore: A comprehensive library for coreset selection in deep learning. In *International Conference Database and Expert Systems Applications*.

Har-Peled, S.; and Mazumdar, S. 2004. On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, 291–300.

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

Hörmander, F.; Totaro, N.; and Waldschmidt, A. V. M. 2006. *Grundlehren der mathematischen wissenschaften*, volume 5.

Huggins, J.; Campbell, T.; and Broderick, T. 2016. Coresets for scalable Bayesian logistic regression. *Advances in neural information processing systems*, 29.

Keyfitz, N. 1977. Introduction to the mathematics of population: with revisions. *Addison-Wesley Series in Behavioral Science: Quantitative Methods*, 65(331): 1420.

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

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

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases*, 1(4).

Lecun, Y.; and Bottou, L. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Liang, W.; and Zou, J. 2022. MetaShift: A Dataset of Datasets for Evaluating Contextual Distribution Shifts and Training Conflicts. In *International Conference on Learning Representations*.

Margatina, K.; Vernikos, G.; Barrault, L.; and Aletras, N. 2021. Active learning by acquiring contrastive examples.

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

Mirzasoleiman, B.; Cao, K.; and Leskovec, J. 2020. Coresets for Robust Training of Neural Networks against Noisy Labels. In *Advances in Neural Information Processing Systems*.

Moosavi-Dezfooli, S.-M.; Fawzi, A.; and Frossard, P. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. In *In Advances in Neural Information Processing Systems*.

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

Peyré, G.; Cuturi, M.; et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends in Machine Learning*, 11(5-6): 355–607.

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

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252.

Sachdeva, N.; Wu, C.-J.; and McAuley, J. 2021. Svp-cf: Selection via proxy for collaborative filtering data. *arXiv preprint arXiv:2107.04984*.

Sener, O.; and Savarese, S. 2017. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *International Conference on Learning Representations*.

Shin, S.; Bae, H.; Shin, D.; Joo, W.; and Moon, I.-C. 2023. Loss-Curvature Matching for Dataset Selection and Condensation. In *International Conference on Artificial Intelligence and Statistics*.

Simonyan, K.; and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Tao, R.; Zhang, H.; Zheng, Y.; and Savvides, M. 2022. Powering Finetuning in Few-Shot Learning: Domain-Agnostic Bias Reduction with Selected Sampling.

Wang, Y.; Chao, W. L.; Weinberger, K. Q.; and Laurens, V. 2019. SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Welling, M. 2009. Herding dynamical weights to learn. In *International Conference on Machine Learning*.

Yang, J.; Shi, R.; and Ni, B. 2021. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 191–195. IEEE.

Yang, S.; Liu, L.; and Xu, M. 2021. Free Lunch for Few-shot Learning: Distribution Calibration. In *International Conference on Learning Representations*.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. In *British Machine Vision Conference*.

Zhang, J.; Meng, C.; Yu, J.; Zhang, M.; Zhong, W.; and Ma, P. 2023. An optimal transport approach for selecting a representative subsample with application in efficient kernel density estimation. *Journal of Computational and Graphical Statistics*, 32(1): 329–339.

Zhao, B.; Mopuri, K. R.; and Bilen, H. 2020. Dataset Condensation with Gradient Matching. In *International Conference on Learning Representations*.