Balancing Global and Local: Representative Sampling for Large-Scale Vector Data

ZHENG WU, Hong Kong Baptist University, China
YITONG SONG*, Hong Kong Baptist University, China
XULIANG ZHU, Antai College of Economics and Management, Shanghai Jiao Tong University, China
HUILING LI, Hong Kong Baptist University, China
JIANLIANG XU, Hong Kong Baptist University, China
XIN HUANG*, Hong Kong Baptist University, China

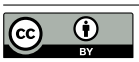
Representative sampling, which extracts a small subset of representative instances from massive, high-dimensional vector data, is useful for many applications, such as visualization, model prototyping, and data exploration. Existing methods often suffer from either *over-representing dense regions while neglecting sparse ones* or *failing to capture fine-grained data distributions*. To address these challenges, we study representative sampling with two objectives: *global coverage* and *local fidelity*. Specifically, given a dataset D and a sampling ratio $\rho \in (0, 1]$, we seek a subset $S \subseteq D$ with $|S| = \rho|D|$ that minimizes the local-fidelity cost subject to a global-coverage constraint. Here, the global coverage of S measures the average distance from data points to their nearest samples in S. We define local fidelity by comparing local intrinsic dimensionality (LID) estimates computed on S and on the full dataset D at a matched evaluation scale, which measures how well fine-grained local distributions are preserved. Unfortunately, efficiently computing an optimal solution is challenging; we prove that the representative sampling problem is NP-hard.

To address this challenge, we first propose LASS-Lite, a heuristic that improves global coverage by selecting widely separated initial samples across diverse LID levels and then enhances local fidelity by adding their nearby neighbors. Since LASS-Lite selects neighbors for each sample independently, we further propose LASS-NA, which employs a more effective joint selection strategy by prioritizing points that can serve as shared neighbors for multiple samples. This approach better utilizes the sampling budget to improve overall local fidelity. We formulate this strategy as a submodular maximization problem, yielding an efficient greedy algorithm with a provable $(1 - 1/e)$-approximation guarantee. To validate the effectiveness of our methods, we conduct extensive experiments on four large-scale, real-world vector datasets. In terms of local fidelity, our algorithms use only 5.0% of the data and reduce the local fidelity cost by 58.6%–62.7% compared to three state-of-the-art competitors at similar levels of global coverage. In addition, we demonstrate the practical utility of our representative sampling in data visualization, self-supervised model prototyping, graph-based approximate nearest neighbor search (ANNS) and label-efficient model selection. Our source code is available at https://github.com/i11ume/LASS.

CCS Concepts: • **Information systems → Data management systems**; **Nearest-neighbor search**; • **Theory of computation → Sketching and sampling**; *Approximation algorithms analysis.*

*Yitong Song and Xin Huang are co-corresponding authors.

Authors' Contact Information: Zheng Wu, cszhengwu@comp.hkbu.edu.hk, Hong Kong Baptist University, Hong Kong, China; Yitong Song, Hong Kong Baptist University, Hong Kong, China, yitong_song@hkbu.edu.hk; Xuliang ZHU, Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai, China, zhu.xl@sjtu.edu.cn; Huiling Li, Hong Kong Baptist University, Hong Kong, China, cshlli@comp.hkbu.edu.hk; Jianliang XU, Hong Kong Baptist University, Hong Kong, China, xujl@comp.hkbu.edu.hk; Xin HUANG, Hong Kong Baptist University, Hong Kong, China, xinhuang@comp.hkbu.edu.hk.



ACM 2836-6573/2026/6-ART142
https://doi.org/10.1145/3802019

Additional Key Words and Phrases: Representative sampling, High-dimensional vector data, Local intrinsic dimensionality, Submodular optimization, Approximate nearest neighbor search

ACM Reference Format:
Zheng Wu, Yitong Song, Xuliang ZHU, Huiling Li, Jianliang XU, and Xin HUANG. 2026. Balancing Global and Local: Representative Sampling for Large-Scale Vector Data. *Proc. ACM Manag. Data* 4, 3 (SIGMOD), Article 142 (June 2026), 27 pages. https://doi.org/10.1145/3802019

1 Introduction

Data sampling is a fundamental and widely used technique in modern data analysis. From interactive visualization and exploratory querying [1, 19] to fast prototyping of machine learning models [27, 42, 46, 56], many applications rely on creating a small but representative subset of a large dataset for efficient analysis, rapid experimentation, and better understanding of the underlying data patterns. Such a subset should be compact enough for efficient computation while still preserving the key characteristics of the original data distributions. Many types of unstructured data can be transformed into high-dimensional vector representations in modern AI applications. For example, in machine learning model development, data sampling is crucial not only for managing the scale of training data [8, 36, 57], but also for providing high-quality, small-scale data for cost-effective prototyping before deploying expensive jobs. Similarly, in billion-scale vector retrieval systems that support applications such as Retrieval-Augmented Generation (RAG) [55, 62, 63], data sampling over high-dimensional vectors is commonly employed to enhance search efficiency [22, 53]. These systems leverage a small set of representative samples to construct in-memory "navigators", which provide efficient entry points into the massive on-disk index [23, 38, 53]. Given its broad applicability, determining how to effectively select a representative subset from large-scale datasets has become a fundamental research problem.

Existing sampling approaches for vector data can be broadly categorized into three types: random sampling, cluster-based sampling, and difference-based sampling. As illustrated in Figure 1(a)–(c), these three approaches yield different sampling outcomes, but none of them adequately represent the entire dataset. Random sampling [9] (red points), governed by data density, heavily over-samples the dense regions while neglecting the sparse ones, resulting in poor global coverage. Cluster-based sampling [4, 37] (green points) enhances global coverage by ensuring that samples are drawn from each clustered region. However, it fails to capture the fine-grained data distribution within an individual cluster (e.g., the highlighted yellow region), as it treats all data points in a cluster uniformly and randomly selects a number of samples proportional to the cluster's size. Difference-based sampling [31, 60] (orange points) seeks to select samples that are maximally distant from one another. While this approach facilitates sample diversity and global coverage, it overemphasizes isolated or outlier points, making it difficult to capture the true distribution of dense regions (e.g., the yellow region).

Given these limitations, existing methods tend to emphasize data density, coverage, or diversity, but often overlook the subtle variability among nearby points, leading to inadequate preservation of local data distributions. We argue that an effective sample set should jointly meet the following conditions: (i) *Global Coverage:* every region of the data space should be sufficiently represented to comprehensively capture the overall data distribution; and (ii) *Local Fidelity:* within densely populated regions, the selected samples should capture the inherent variability among neighboring points to faithfully preserve the local distribution of the original dataset.[1]

In this paper, our goal is to construct a sample set $\mathcal{S}$ that achieves both comprehensive global coverage and high local fidelity. To this end, we first introduce two quantitative metrics that

[1]The intuition is general: coverage determines how well the samples span the entire space, while fidelity measures how faithfully the local distribution is preserved.

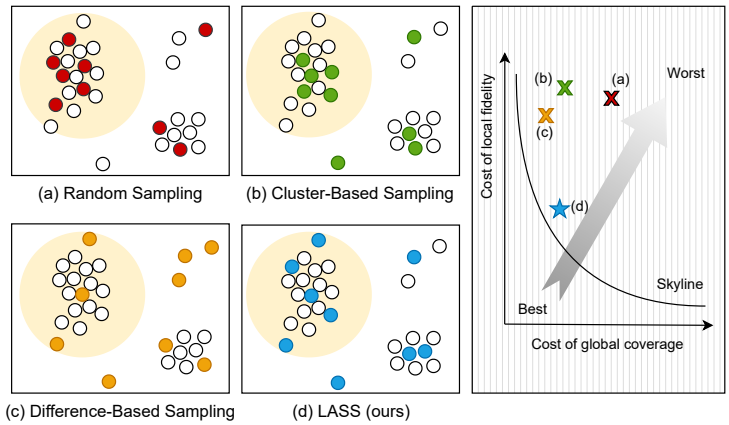


Fig. 1. A comparison of sampling methods.

respectively measure the degree of global coverage and local fidelity, laying the foundation for our sampling framework. The degree of global coverage is quantified using the classic facility-location cost function [7, 17, 30, 33], $\text{Cost}_{\text{cov}}(\mathcal{S}, D)$[2], which measures the average distance from each data point to its nearest representative. Minimizing this cost ensures that no region of the data is overlooked. The degree of local fidelity is quantified by a novel scale-aligned loss function, $\text{Cost}^{\phi}_{\text{LID}}(\mathcal{S}, D) \in [0, 1]$, whose minimization encourages the sampled subset to reflect the local data distributions of the original dataset. The main challenge in defining such a metric lies in ensuring a fair comparison between the distribution of a sparse sample and that of the dense full dataset. We address this by introducing a unified reference radius $R_D(s)$ for each sample s, enabling evaluation at a consistent local scale. Within this shared scale, we employ Local Intrinsic Dimensionality (LID) [2, 20] as a robust indicator of local distributional complexity. LID characterizes the growth rate of distances to a point's k nearest neighbors, thereby capturing how densely data are distributed around that point. The local fidelity cost then measures the discrepancy between the LID estimated from the sample set and the ground-truth LID computed from the full dataset.

Given these two metrics, we formulate representative sampling as a constrained optimization problem, aiming to find a sample set $\mathcal{S}$ that minimizes the local fidelity cost while satisfying a given constraint on global coverage. However, this formulation introduces several challenges. *First*, we prove that the optimization problem is NP-hard, making an exact solution computationally intractable. *Second*, global coverage and local fidelity inherently trade off against each other: spreading samples widely improves coverage but may harm the preservation of local distributions, while concentrating samples locally enhances fidelity at the expense of global representativeness. *Third*, it is nontrivial to determine which limited samples can best maintain local distributions. Choosing points that are too close fails to capture distribution diversity, whereas choosing points that are too far apart distorts local density.

To tackle the above challenges, we propose two near-linear time samplers, **LASS-Lite** and **LASS-NA**, whose sampling results are illustrated in Figure 1(d). LASS-Lite employs a two-phase framework that first improves global coverage by selecting widely separated samples from diverse LID levels, and then enhances local fidelity by choosing samples with their nearby neighbors. To further strengthen local fidelity, LASS-NA extends LASS-Lite by jointly considering shared neighbors among multiple samples and selecting the point that yields the greatest cost reduction. We also provide a theoretical guarantee for LASS-NA, framing the neighbor selection problem as a

[2]Table 1 summarizes all notation.

Table 1. Frequently used notations.

Notation	Description
$\mathcal{D}, \mathcal{S}$	Full dataset, $\{x_1, \ldots, x_n\} \subset \mathbb{R}^d$; sampled subset, $\mathcal{S} \subset \mathcal{D}$
n, d, ρ	Number of points; dimension; sampling ratio, $\lvert\mathcal{S}\rvert = \lfloor n \cdot \rho \rfloor$
$\mathrm{dist}(\cdot, \cdot)$	Euclidean distance; $\mathrm{dist}(x, \mathcal{S}) = \min_{s \in \mathcal{S}} \mathrm{dist}(x, s)$
$B(x, r)$	Closed ball $\{y : \mathrm{dist}(x, y) \le r\}$
q	Reference point for local analysis (e.g., a query)
$\mathrm{NN}_i(q; \mathcal{D})$	i-th nearest neighbor of q in $\mathcal{D}$; $\mathcal{N}_k(q; \mathcal{D})$ is exact k-NN set
k_{ref}	Reference neighbor count (scale alignment)
$R_{\mathcal{D}}(s)$	Reference radius at s (to the k_{ref}-th NN in $\mathcal{D}$)
$m_{\mathcal{S}}(s)$	# of sampled neighbors of s in $B(s, R_{\mathcal{D}}(s))$
$\widehat{\mathrm{LID}}_{\mathcal{D}}(s)$	LID of s estimated from $\mathcal{D}$ at radius $R_{\mathcal{D}}(s)$
$\widehat{\mathrm{LID}}_{\mathcal{S}}(s)$	LID of s estimated from $\mathcal{S}$ within $R_{\mathcal{D}}(s)$ (defined for $m_{\mathcal{S}}(s) \ge 1$)
$\mathrm{Cost}_{\mathrm{cov}}$	Global coverage cost (average nearest-sample distance)
$\mathrm{Cost}_{\mathrm{LID}}^{\phi}$	Local fidelity cost (preservation of local data distribution); $[0, 1]$

submodular optimization, which provides a $(1 - 1/e)$ approximation guarantee for its objective of maximizing the reduction in LID estimation error.

In summary, our contributions are as follows:

- **A novel and principled metric of local fidelity.** We identify the critical role of local fidelity in representative sampling and introduce a novel scale-aligned metric to quantify it. Furthermore, we provide a theorem showing that maximizing local fidelity exponentially increases the probability of successful search progress in downstream neighborhood-based algorithms (e.g., graph-based ANNS).
- **A constrained optimization model for representative sampling.** To the best of our knowledge, we are the first to formalize representative sampling as a constrained optimization problem that minimizes the local fidelity cost while satisfying a given global coverage constraint. We also prove that this problem is NP-hard.
- **Efficient samplers with theoretical guarantees.** To address this challenging problem, we propose two near-linear-time samplers, **LASS-Lite** and **LASS-NA**, with the latter offering provable approximation guarantees.
- **Extensive experimental validation.** Across four million-scale benchmarks, our methods reduce local fidelity cost ($\mathrm{Cost}_{\mathrm{LID}}^{\phi}$) by up to 70% over strong baselines while maintaining comparable global coverage ($\mathrm{Cost}_{\mathrm{cov}}$). We further validate robustness and scalability up to 100M vectors, and demonstrate practical gains in UMAP visualization, self-supervised prototyping, seed selection in ANNS, and label-efficient model selection.

2 Preliminaries

This section formalizes *representative sampling* for large vector data. Given a sampling ratio $\rho \in (0, 1]$, our goal is to select a small subset $\mathcal{S} \subset \mathcal{D}$ that is simultaneously (i) *spatially covering* and (ii) *faithful to the local data distribution*. We first introduce the necessary preliminaries, then define our two core objectives: **Global Coverage** and **Local Fidelity**. Finally, we combine these objectives into a constrained optimization problem. Table 1 lists the core notation used throughout this section.

2.1 Foundational Metrics and Notation

Let $\mathcal{D} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ be n points in d dimensions. We use Euclidean distance and write $\mathrm{dist}(\cdot, \cdot)$. The distance from a point x to a set $\mathcal{S}$ is its nearest-neighbor distance:

$$\mathrm{dist}(x, \mathcal{S}) = \min_{s \in \mathcal{S}} \mathrm{dist}(x, s).$$

We denote the (closed) ball of radius r centered at x by $B(x, r) = \{y \in \mathbb{R}^d \mid \mathrm{dist}(x, y) \le r\}$.

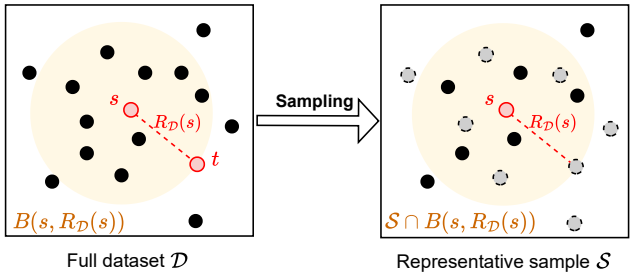


Fig. 2. Illustration of sample-based LID for $s, t \in \mathcal{S}, t = \mathrm{NN}_{10}(s; \mathcal{D})$ at the aligned external radius $R_{\mathcal{D}}(s)$, which equals the k_{ref}-NN distance in $\mathcal{D}$ (here $k_{\mathrm{ref}} = 10$, $m_{\mathcal{S}}(s) = 5$).

k-nearest neighbor structures. Given a dataset $\mathcal{D}$ and a query point $q \in \mathbb{R}^d$, the task of k-nearest neighbors (k-NN) search is to identify the k points in $\mathcal{D}$ closest to q. We write the i-th nearest neighbor of q in $\mathcal{D}$ as $\mathrm{NN}_i(q; \mathcal{D})$, and the exact k-NN set as

$$\mathcal{N}_k(q; \mathcal{D}) = \{\mathrm{NN}_1(q; \mathcal{D}), \ldots, \mathrm{NN}_k(q; \mathcal{D})\}.$$

At large scales, practical systems often use approximate procedures to query neighbors efficiently.
Local intrinsic dimensionality (LID) [2, 20]. LID characterizes the complexity of the local data distribution around a point and is widely used to quantify the difficulty of queries and datasets. Intuitively, the LID of a point $x \in \mathcal{D}$ describes the rate at which the number of neighbors grows as the radius increases; higher LID indicates a more complex local structure. To compute the LID of a point x, we use the maximum-likelihood estimator (MLE), a classic and widely adopted method [2]. Let k be the number of nearest neighbors used for estimation. The LID of x is estimated by:

$$\widehat{\mathrm{LID}}(x) = \left(-\frac{1}{k} \sum_{i=1}^{k} \ln \frac{\mathrm{dist}\big(x, \mathrm{NN}_i(x; \mathcal{D} \setminus \{x\})\big)}{\mathrm{dist}\big(x, \mathrm{NN}_k(x; \mathcal{D} \setminus \{x\})\big)} \right)^{-1},$$

where ln denotes the natural logarithm.

2.2 Global Coverage and Local Fidelity

We introduce two competing objectives for evaluating the quality of the samples. First, we address **Global Coverage**, which promotes a broad spatial distribution of the sample. Then, we introduce **Local Fidelity** to preserve fine-grained local data distributions. This latter objective is challenging, as it requires a formulation that explicitly accounts for the inherent density difference between the full dataset $\mathcal{D}$ and the sparse sample $\mathcal{S} \subseteq \mathcal{D}$.
Global coverage. This metric quantifies how well $\mathcal{S}$ covers the space occupied by $\mathcal{D}$. A good sample set $\mathcal{S}$ ensures that every point in $\mathcal{D}$ is sufficiently close to at least one representative in $\mathcal{S}$. Following the classical k-median and facility-location formulation [7, 17, 30, 33, 37], we define the coverage cost as:

$$\mathrm{Cost}_{\mathrm{cov}}(\mathcal{S}, \mathcal{D}) = \frac{1}{n} \sum_{x \in \mathcal{D}} \mathrm{dist}(x, \mathcal{S}).$$

A smaller $\mathrm{Cost}_{\mathrm{cov}}(\mathcal{S}, \mathcal{D})$ indicates better spatial coverage.
Local fidelity. This metric measures how well local data distributions are preserved. To formalize local fidelity for a representative $s \in \mathcal{S}$, our approach is three-fold. First, to ensure a fair comparison, we establish a common scale by defining a reference radius $R_{\mathcal{D}}(s)$ based on its neighbors in the full dataset $\mathcal{D}$. Second, within this fixed radius, we compute two LID estimates: a "ground-truth" estimate using all points from $\mathcal{D}$, and a sample-based estimate using only points from $\mathcal{S}$. Finally, we

measure the stabilized relative error between these two estimates and aggregate this error across all representatives to define our fidelity cost.

Fix a reference neighbor count $k_{\text{ref}} \geq 3$. For each representative $s \in \mathcal{S}$, we define its reference radius as the distance to its k_{ref}-th nearest neighbor in the full dataset:

$$R_{\mathcal{D}}(s) = \operatorname{dist}\big(s, \operatorname{NN}_{k_{\text{ref}}}(s; \mathcal{D} \setminus \{s\})\big).$$

At this aligned scale, we form two LID estimates. The first is the ground-truth LID at s:

$$\widehat{\operatorname{LID}}_{\mathcal{D}}(s) = \Big(-\tfrac{1}{k_{\text{ref}}} \sum_{i=1}^{k_{\text{ref}}} \ln \frac{\operatorname{dist}\big(s, \operatorname{NN}_i(s; \mathcal{D} \setminus \{s\})\big)}{R_{\mathcal{D}}(s)} \Big)^{-1},$$

and the second is the sample-based LID.

Let $\mathcal{M}_{\mathcal{S}}(s) = \big(\mathcal{S} \cap B(s, R_{\mathcal{D}}(s))\big) \setminus \{s\}$ be the set of sampled neighbors of s within the reference ball, and let $m_{\mathcal{S}}(s) = |\mathcal{M}_{\mathcal{S}}(s)|$ be their count. Then,

$$\widehat{\operatorname{LID}}_{\mathcal{S}}(s) = \Big(-\tfrac{1}{m_{\mathcal{S}}(s)} \sum_{s_j \in \mathcal{M}_{\mathcal{S}}(s)} \ln \frac{\operatorname{dist}(s, s_j)}{R_{\mathcal{D}}(s)} \Big)^{-1} \quad (m_{\mathcal{S}}(s) \geq 1).$$

Using a common radius $R_{\mathcal{D}}(s)$ for both estimates is key, as it removes the difference of scale between the sample and full dataset.

For $m_{\mathcal{S}}(s) \geq 1$, we define the stabilized relative error as:

$$r_s(\mathcal{S}) = \frac{\big|\widehat{\operatorname{LID}}_{\mathcal{S}}(s) - \widehat{\operatorname{LID}}_{\mathcal{D}}(s)\big|}{\max\{\widehat{\operatorname{LID}}_{\mathcal{D}}(s),\ \tau\}},$$

where τ is a small positive stabilizer to prevent division by zero in regions of very low geometric complexity (i.e., when $\widehat{\operatorname{LID}}_{\mathcal{D}}(s) \approx 0$). We then map this error via the concave saturating function $\phi(u) = u/(1+u) \in [0, 1)$. This crucial step bounds the per-point loss, preventing any single point with a large estimation error from dominating the total cost, thus making our metric robust to outliers. The per-point loss is:

$$\ell_s(\mathcal{S}) = \begin{cases} \phi(r_s(\mathcal{S})), & m_{\mathcal{S}}(s) \geq 1, \\ 1, & m_{\mathcal{S}}(s) = 0. \end{cases}$$

This assigns the maximum loss of 1 when a representative s is isolated (i.e., has no sampled neighbors within its reference radius), signifying a complete failure to capture its local data distribution. Aggregating over all representatives yields our final fidelity cost and the no-neighbor rate:

$$\operatorname{Cost}^{\phi}_{\text{LID}}(\mathcal{S}, \mathcal{D}) = \tfrac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \ell_s(\mathcal{S}), \qquad \rho_0(\mathcal{S}) = \tfrac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \mathbf{1}\{m_{\mathcal{S}}(s) = 0\},$$

where, trivially, $\rho_0(\mathcal{S}) \leq \operatorname{Cost}^{\phi}_{\text{LID}}(\mathcal{S}, \mathcal{D}) \leq 1$. To ensure consistency across different sample sets, we anchor the stabilizer τ on a fixed evaluation set $\mathcal{E}$ (e.g., $\mathcal{E} = \mathcal{D}$):

$$\tau = \max\Big\{ \operatorname{Quantile}_p\big(\{\widehat{\operatorname{LID}}_{\mathcal{D}}(x) : x \in \mathcal{E}\}\big),\ \varepsilon \Big\}, \quad \text{where } p \in [0.02, 0.10] \ \text{ (default 0.05)}, \ \ \varepsilon = 10^{-6}.$$

Remark. Our theoretical analysis in Section 3 instantiates $\operatorname{dist}(\cdot, \cdot)$ as the Euclidean distance on $\mathbb{R}^d$ to exploit metric properties (e.g., the triangle inequality and ball geometry). Algorithmically, however, the definitions of $\operatorname{Cost}_{\text{cov}}$ and $\operatorname{Cost}^{\phi}_{\text{LID}}$ only require a dissimilarity that supports k-NN queries, and can therefore be applied in spaces equipped with cosine or inner-product similarity. In practice, one can either use standard reductions to Euclidean distance (e.g., ℓ_2 normalization plus angular/Euclidean distance) or treat a monotone transform of similarity (such as $1 - \cos\theta$) as the distance in our metrics. Since the LID estimator is based on ratios of neighbor radii, it is invariant to

global rescaling and empirically robust to such monotone transformations. In general non-metric or learned similarity spaces, our LID-based fidelity metric remains a useful heuristic for measuring how well local neighborhoods are preserved, whereas the formal guarantees in Section 3.1 strictly apply to the Euclidean instantiation.

2.3 Problem Formulation

Building on the two objectives above, we formulate representative sampling as a constrained optimization problem. We treat local fidelity as the primary optimization target, and enforce global coverage as a critical constraint given the inherent trade-off between these two metrics.

Problem 1 (Representative Sampling). *Given a dataset $\mathcal{D} \subset \mathbb{R}^d$ with $|\mathcal{D}| = n$, a sampling ratio $\rho \in (0, 1]$, and a maximum tolerable coverage cost $C_{\max}$, find a subset $\mathcal{S} \subset \mathcal{D}$ that solves:*

$$\underset{\mathcal{S} \subset \mathcal{D}}{\text{minimize}}\ \text{Cost}^{\phi}_{\text{LID}}(\mathcal{S}, \mathcal{D}) \text{ subject to } \text{Cost}_{\text{cov}}(\mathcal{S}, \mathcal{D}) \leq C_{\max}, \text{ where } |\mathcal{S}| = \lfloor n \cdot \rho \rfloor. \tag{1}$$

Remark. In this problem, the goal is to find a sample $\mathcal{S}$ that achieves the best possible local fidelity among all samples of a given size that satisfy a minimum standard of spatial coverage. The parameter $C_{\max}$ serves as an intuitive coverage budget, allowing a user to specify an average distance budget they are willing to accept. This constrained formulation is computationally challenging; we formally establish in Theorem 2 that the problem is NP-hard. Our proposed algorithms, LASS-Lite and LASS-NA, are therefore designed as efficient heuristics to find high-quality samples without taking $C_{\max}$ as an input. They are designed to efficiently trace out a superior solution frontier.

3 Motivation and Challenges

This section motivates our two objectives and discusses the key challenges in representative sampling.

Global coverage is a fundamental objective in data sampling and has been widely used in several data sampling methods [4, 7, 33]. Intuitively, it aims to ensure that each region of the feature space is sufficiently represented, enabling the sampled subset to faithfully capture the overall data distribution. In contrast, the preservation of *local fidelity* is often neglected in existing approaches. However, for many downstream tasks, retaining the local data distribution among nearby data points is essential. For example, in manifold learning algorithms such as UMAP [39], insufficient local fidelity can fragment the learned manifold into disconnected components. Similarly, in high-dimensional ANNS [12, 61], it may lead to a poor entry point for queries, thereby degrading search efficiency. Therefore, preserving relative neighborhood relationships in the sample set plays a vital role in ensuring model quality, interpretability, and search accuracy. Next, we provide a theoretical analysis highlighting the importance of local fidelity.

3.1 Theoretical Motivation for Local Fidelity

To formalize the importance of *local fidelity*, we now quantitatively analyze its impact on downstream tasks. We instantiate our analysis in the context of **graph traversal**, a fundamental routine for exploring neighborhood structures common in ANNS. Our central thesis is that preserving local fidelity, i.e., minimizing $\text{Cost}^{\phi}_{\text{LID}}$, is equivalent to ensuring that each sample point has a sufficient number of local sampled neighbors. As we will demonstrate, this property exponentially increases the success probability of any downstream algorithm that relies on neighborhood exploration.

Local modeling assumptions. Fix $s \in \mathcal{S}$ with reference radius $R_{\mathcal{D}}(s)$. Within $B(s, R_{\mathcal{D}}(s))$ we adopt:

(A1) **Local homogeneity and isotropy.** Neighbors are i.i.d. with radial CDF $F(r) = (r/R_{\mathcal{D}}(s))^{D(s)}$ and independent uniform directions on $\mathbb{S}^{k(s)-1}$, with $k(s) = \lceil D(s) \rceil$.[3]
(A2) **Locally distribution-agnostic thinning.** Conditioned on $m_{\mathcal{S}}(s)$, the $m_{\mathcal{S}}(s)$ sampled neighbors retained by $\mathcal{S}$ in $B(s, R_{\mathcal{D}}(s))$ are i.i.d. draws from the base local model in (A1).[4]
(A3) **Availability.** Each geometric neighbor, independently of others and of geometry, becomes an available graph neighbor with probability at least $\alpha \in (0, 1]$.

For beam search, we assess progress using the current best pivot $s^\star = \arg\min_{u \in \mathcal{B}} \mathrm{dist}(u, q)$.

Lemma 1. *Under (A1)-(A2) at $s \in \mathcal{S}$, let $D = D(s)$ and $m = m_{\mathcal{S}}(s)$. The sample-based LID estimator admits the representation*

$$\widehat{\mathrm{LID}}_{\mathcal{S}}(s) = D \cdot \Big(\tfrac{1}{m} \sum_{j=1}^{m} Y_j \Big)^{-1}, \quad Y_j \overset{\text{i.i.d.}}{\sim} \mathrm{Exp}(1),$$

with the following moments:

- *For $m > 1$, $\mathbb{E}\big[\widehat{\mathrm{LID}}_{\mathcal{S}}(s) \mid D, m\big] = D \cdot \frac{m}{m-1}$ (bias $= \frac{D}{m-1}$); for $m = 1$, the expectation is infinite.*
- *For $m > 2$, $\mathrm{Var}\big(\widehat{\mathrm{LID}}_{\mathcal{S}}(s) \mid D, m\big) = D^2 \frac{m^2}{(m-1)^2(m-2)}$; for $m = 2$, the variance is infinite.*

Consequently, for $m \geq 3$,

$$\mathrm{MSE}(m) = \mathbb{E}\big[(\widehat{\mathrm{LID}}_{\mathcal{S}}(s) - D)^2 \mid D, m\big] = D^2 \frac{m+2}{(m-1)(m-2)}$$

is strictly decreasing in m; consequently, an upper bound on $\mathbb{E}\big|\widehat{\mathrm{LID}}_{\mathcal{D}}(s) - \widehat{\mathrm{LID}}_{\mathcal{S}}(s)\big|$ is monotonically non-increasing in m.

Proof. By (A1)-(A2), $U_j = (\mathrm{dist}(s, s_j)/R_{\mathcal{D}}(s))^D \sim \mathrm{Unif}[0, 1]$, hence $Y_j = -\ln U_j \sim \mathrm{Exp}(1)$ i.i.d. Let $Z = \sum_{j=1}^{m} Y_j \sim \Gamma(m, 1)$. Then $\mathbb{E}[\widehat{\mathrm{LID}}_{\mathcal{S}}(s)] = D\,\mathbb{E}[m/Z] = D \cdot m/(m-1)$ for $m > 1$, and $\mathrm{Var}(\widehat{\mathrm{LID}}_{\mathcal{S}}(s)) = (Dm)^2 \mathrm{Var}(1/Z) = D^2 m^2/[(m-1)^2(m-2)]$ for $m > 2$ (standard identities for $Z \sim \Gamma(m, 1)$). The MSE expression follows by Var + bias^2, and its monotonicity in m is immediate. Finally, by the triangle inequality, $\mathbb{E}|\widehat{\mathrm{LID}}_{\mathcal{D}}(s) - \widehat{\mathrm{LID}}_{\mathcal{S}}(s)| \leq \mathbb{E}|\widehat{\mathrm{LID}}_{\mathcal{D}}(s) - D| + \mathbb{E}|\widehat{\mathrm{LID}}_{\mathcal{S}}(s) - D|$, where the first term depends on $k_{\mathrm{ref}} > 2$ but not on m, and the second term is upper-bounded by $\sqrt{\mathrm{MSE}(m)}$, decreasing in m for $m \geq 3$. □

Large deviations via a squared-error bound. For any $\epsilon > 0$ and $m \geq 3$, Markov applied to the squared error gives

$$\mathbb{P}\Big(\big|\widehat{\mathrm{LID}}_{\mathcal{S}}(s) - D\big| \geq \epsilon \;\Big|\; m\Big) \leq \frac{\mathrm{MSE}(m)}{\epsilon^2} = \frac{D^2(m+2)}{\epsilon^2\,(m-1)(m-2)}.$$

By the triangle inequality and a union bound, for any $\epsilon > 0$,

$$\begin{aligned}\mathbb{P}\Big(\big|\widehat{\mathrm{LID}}_{\mathcal{S}}(s) - \widehat{\mathrm{LID}}_{\mathcal{D}}(s)\big| \geq \epsilon \;\Big|\; m\Big) &\leq \mathbb{P}\Big(\big|\widehat{\mathrm{LID}}_{\mathcal{S}}(s) - D\big| \geq \tfrac{\epsilon}{2} \;\Big|\; m\Big) + \mathbb{P}\Big(\big|\widehat{\mathrm{LID}}_{\mathcal{D}}(s) - D\big| \geq \tfrac{\epsilon}{2}\Big) \\ &\leq \frac{4\,\mathrm{MSE}(m)}{\epsilon^2} \;+\; p_{\mathrm{ref}}\big(\tfrac{\epsilon}{2};\, k_{\mathrm{ref}}, D\big),\end{aligned}$$

where p_{ref} is the tail probability of the reference estimator (made negligible by a sufficiently large k_{ref}).

[3] Alternatively, using the ambient dimension d for angular measures yields slightly looser but still valid bounds.
[4] This does not require global uniform sampling; it only postulates that, at the aligned scale $R_{\mathcal{D}}(s)$, the inclusion of nearby points is independent of their local radius/direction.

Lemma 2. *Let* $\mathrm{MSE}(m)$ *be as in Lemma 1, and let* $\mathrm{MSE}_{\mathrm{ref}}$ *denote the MSE of* $\widehat{\mathrm{LID}}_{\mathcal{D}}(s)$ *for* $k_{\mathrm{ref}} > 2$. *For* $\tau > 0$ *and* $\phi(u) = u/(1+u)$,

$$\mathbb{E}\big[\ell_s(\mathcal{S}) \mid m\big] \;\le\; \phi\left(\frac{\sqrt{\mathrm{MSE}(m)} + \sqrt{\mathrm{MSE}_{\mathrm{ref}}}}{\tau}\right), \qquad m \ge 3,$$

and the RHS is strictly decreasing in m.

Proof. Write $e_{\mathcal{S}} = \widehat{\mathrm{LID}}_{\mathcal{S}}(s) - D$ and $e_{\mathcal{D}} = \widehat{\mathrm{LID}}_{\mathcal{D}}(s) - D$. Then $|\widehat{\mathrm{LID}}_{\mathcal{S}}(s) - \widehat{\mathrm{LID}}_{\mathcal{D}}(s)| \le |e_{\mathcal{S}}| + |e_{\mathcal{D}}|$. Since $\max\{\widehat{\mathrm{LID}}_{\mathcal{D}}(s), \tau\} \ge \tau$,

$$\mathbb{E}[r_s \mid m] \;\le\; \frac{\mathbb{E}|e_{\mathcal{S}}| \;+\; \mathbb{E}|e_{\mathcal{D}}|}{\tau} \;\le\; \frac{\sqrt{\mathrm{MSE}(m)} + \sqrt{\mathrm{MSE}_{\mathrm{ref}}}}{\tau},$$

where we used Cauchy-Schwarz for $\mathbb{E}|e| \le \sqrt{\mathbb{E}e^2}$. ϕ is increasing and concave on $[0,\infty)$, hence by Jensen,

$$\mathbb{E}[\ell_s(\mathcal{S}) \mid m] = \mathbb{E}[\phi(r_s) \mid m] \;\le\; \phi(\mathbb{E}[r_s \mid m]),$$

giving the claim. Monotonicity in m follows from the strict decrease of $\mathrm{MSE}(m)$ for $m \ge 3$. □

Success region. Let $d_s = \mathrm{dist}(s,q)$ and define the geometric radius

$$\tau_{\mathrm{geo}}(s,q,\varphi) = \min\{2\, d_s \cos\varphi,\; R_{\mathcal{D}}(s)\}, \qquad \varphi \in (0, \pi/2].$$

By the law of cosines, any s' inside the cone at s with half-angle φ and radius $< \tau_{\mathrm{geo}}(s,q,\varphi)$ is strictly closer to q than s (boundary cases have measure zero). For the angular measure on $\mathbb{S}^{k(s)-1}$, let $\mu_{k(s)}(\varphi)$ denote the normalized surface area of a spherical cap of half-angle φ.

Lemma 3. *Under (A1)-(A2), for* $m = m_{\mathcal{S}}(s)$ *i.i.d. neighbors with uniform directions on* $\mathbb{S}^{k(s)-1}$, *the probability that none falls into the cap of half-angle* φ *equals* $(1 - \mu_{k(s)}(\varphi))^m$ *and is bounded by* $\exp(-m\,\mu_{k(s)}(\varphi))$.

Proof. Each neighbor independently falls in the cap with probability $p = \mu_{k(s)}(\varphi)$, hence the empty-cap probability is $(1-p)^m \le e^{-mp}$. □

Theorem 1. *Let* $D(s)$ *denote the local intrinsic dimension at* s, $k(s) = \lceil D(s) \rceil$, *and* $d_s = \mathrm{dist}(s,q)$. *Define*

$$p_{\mathrm{geom}}(s,q,\varphi) = \mu_{k(s)}(\varphi) \left(\frac{\tau_{\mathrm{geo}}(s,q,\varphi)}{R_{\mathcal{D}}(s)}\right)^{D(s)}.$$

Under (A1)-(A3), the one-layer progress probability at pivot s

$$p_{\mathrm{prog}}(s;q,\varphi) \triangleq \mathbb{P}\big(\exists\ \textit{available neighbor } s' :\ \mathrm{dist}(s',q) < d_s\big)$$

admits the lower bound

$$p_{\mathrm{prog}}(s;q,\varphi) \;\ge\; 1 - \exp\big(-\alpha\, m_{\mathcal{S}}(s)\, p_{\mathrm{geom}}(s,q,\varphi)\big).$$

For a beam/frontier $\mathcal{B}$ *with best pivot* $s^\star = \arg\min_{u\in\mathcal{B}} \mathrm{dist}(u,q)$ *and* $d^\star = \mathrm{dist}(s^\star,q)$, *the beam-wise progress probability*

$$p_{\mathrm{prog}}(\mathcal{B};q) \;\triangleq\; \mathbb{P}\Big(\exists\, s' \textit{ from some } s \in \mathcal{B} :\ \mathrm{dist}(s',q) < d^\star\Big)$$

satisfies the conservative lower bound

$$p_{\mathrm{prog}}(\mathcal{B};q) \;\ge\; 1 - \exp\big(-\alpha\, m_{\mathcal{S}}(s^\star)\, p_{\mathrm{geom}}(s^\star,q,\varphi_{s^\star})\big),$$

for any choice of $\varphi_{s^\star} \in (0, \pi/2]$.

Proof. For a fixed s, by Lemma 3 and (A1)-(A2), a geometric neighbor falls in the success cone of half-angle φ and radius $< \tau_{\text{geo}}$ with probability $p_{\text{geom}}(s, q, \varphi)$; by (A3), it is available with probability at least α, independently across neighbors. Thus a single trial succeeds with probability $\geq \alpha\, p_{\text{geom}}$, and the no-success probability across $m_{\mathcal{S}}(s)$ trials is at most $\exp(-\alpha\, m_{\mathcal{S}}(s)\, p_{\text{geom}})$, proving the first bound. For the beam bound, any success from $s^\star$ yields a point strictly closer than $d^\star$, hence $p_{\text{prog}}(\mathcal{B}; q) \geq p_{\text{prog}}(s^\star; q, \varphi_{s^\star})$. □

Corollary 1. *Fix $\varphi \in (0, \pi/2]$. The exponent in Theorem 1 grows linearly in $m_{\mathcal{S}}(s)$. Moreover, by Lemma 2, $\mathbb{E}[\ell_s(\mathcal{S}) \mid m]$ decreases strictly with m for $m \geq 3$. Therefore, driving down $\text{Cost}^{\phi}_{\text{LID}}$* systematically tends *to increase local neighbor counts at the aligned scale (suppressing $m \in \{0, 1, 2\}$ events) and thus amplifies the layer-wise progress probability for both greedy ($b = 1$) and beam ($b \geq 1$) expansions, even when trajectories are non-monotone.*

Note. If $k(s) = \lceil D(s) \rceil < 2$, one may simply replace $k(s)$ by the ambient dimension d in $\mu_{k(s)}(\varphi)$ to obtain a looser but still valid bound; all conclusions above remain unchanged.

The general principle of fidelity. The preceding analysis, using graph traversal as a concrete testbed, reveals a fundamental principle. Lemma 1 quantifies how fidelity loss is a direct symptom of small local neighbor counts $m_{\mathcal{S}}(s)$. This is why our $\text{Cost}^{\phi}_{\text{LID}}$ assigns the maximum penalty when a representative s has no sampled neighbors ($m_{\mathcal{S}}(s) = 0$), as this signifies a complete breakdown of local navigability. Theorem 1 then demonstrates that these small counts lead to an exponentially higher probability of algorithmic failure (in this case, "no-progress" events) for processes that explore local neighborhoods. Consequently, minimizing $\text{Cost}^{\phi}_{\text{LID}}$ is not merely an optimization for search. This also preserves the essential local connectivity of the data, ensuring that the sample $\mathcal{S}$ remains a faithful representation of the full dataset $\mathcal{D}$. While our formal analysis uses graph traversal, the principle is broadly applicable.

3.2 Key Challenges

Solving Problem 1 efficiently faces three main challenges.

Challenge 1 (C1): Representative sampling problem is NP-hard, making exact solutions unrealistic. We formally establish this hardness result as follows:

Theorem 2. *Problem 1 (Representative Sampling) is NP-hard.*

Proof. We prove NP-hardness via a polynomial-time reduction from the decision version of the metric k-median problem, which is NP-complete [7]. The metric k-median decision problem is defined as follows: given a dataset $\mathcal{D}$ of n points, an integer k, and a total distance budget T, does there exist a subset $\mathcal{S} \subseteq \mathcal{D}$ with $|\mathcal{S}| = k$ such that the sum of distances from each point to its nearest center in $\mathcal{S}$ is at most T? That is, does a subset $\mathcal{S}$ exist satisfying

$$\sum_{x \in \mathcal{D}} \min_{s \in \mathcal{S}} \text{dist}(x, s) \leq T \ ?$$

Given an instance of the k-median decision problem $(\mathcal{D}, k, T)$, we construct an instance of Problem 1 by setting the sampling ratio and the coverage budget as:

$$\rho \leftarrow k/n, \qquad C_{\max} \leftarrow T/n.$$

With these parameters, the sample size constraint in our problem becomes $|\mathcal{S}| = \lfloor n\rho \rfloor = k$, and the coverage constraint becomes $\text{Cost}_{\text{cov}}(\mathcal{S}, \mathcal{D}) \leq T/n$.

Assume there exists a polynomial-time algorithm (an oracle) that solves Problem 1. We can use this oracle to decide the k-median instance. If the oracle returns a feasible solution $\mathcal{S}^\star$, then by

definition, $\mathcal{S}^\star$ must satisfy the coverage constraint:

$$\mathrm{Cost}_{\mathrm{cov}}(\mathcal{S}^\star, \mathcal{D}) = \frac{1}{n} \sum_{x \in \mathcal{D}} \min_{s \in \mathcal{S}^\star} \mathrm{dist}(x, s) \le C_{\max}.$$

Substituting $C_{\max} = T/n$ and multiplying by n, we obtain $\sum_{x \in \mathcal{D}} \min_{s \in \mathcal{S}^\star} \mathrm{dist}(x, s) \le T$. Since the size constraint ensures $|\mathcal{S}^\star| = k$, we have found a valid solution for the k-median problem. Thus, the answer to the decision instance is "yes".

Conversely, if the oracle reports that the problem is infeasible, it implies that no subset $\mathcal{S} \subseteq \mathcal{D}$ of size k satisfies the coverage constraint $\mathrm{Cost}_{\mathrm{cov}}(\mathcal{S}, \mathcal{D}) \le C_{\max}$. This is equivalent to stating that for all subsets $\mathcal{S}$ of size k, $\sum_{x \in \mathcal{D}} \min_{s \in \mathcal{S}} \mathrm{dist}(x, s) > T$. Therefore, no solution to the k-median problem exists, and the answer to the decision instance is "no".

Since we can solve the NP-complete metric k-median decision problem in polynomial time using a hypothetical polynomial-time solver for Problem 1, we conclude that Problem 1 is NP-hard. □

Challenge 2 (C2): There is an inherent trade-off between the two competing objectives. Global coverage and local fidelity inherently trade off against each other. Minimizing $\mathrm{Cost}_{\mathrm{cov}}$ improves global coverage by spreading samples broadly across the space and reducing the nearest-representative distance, especially in sparse regions. However, it reduces the sampling density within complex regions, thereby harming local fidelity. Conversely, minimizing $\mathrm{Cost}^{\phi}_{\mathrm{LID}}$ enhances local fidelity by allocating more samples to *high-LID* areas to stabilize local neighborhood structures. But it sacrifices global coverage by under-representing distant regions. Hence, improving one objective degrades the other, requiring efficient methods that effectively balance the trade-off and perform well on both metrics.

Challenge 3 (C3): Difficulty in selecting representative samples from local neighborhoods. To preserve local data distribution, samples are typically drawn from neighbors within $B(s, R_{\mathcal{D}}(s))$. However, when these neighbors are too close to each other, they effectively behave as a single point, failing to capture the true local structure. Moreover, selecting only one point while ignoring its immediate neighborhood often amplifies $\mathrm{Cost}^{\phi}_{\mathrm{LID}}$, as local sparsity further harms local fidelity. Therefore, an effective sampling algorithm must intelligently select representative points from diverse neighborhoods.

4 Methods

In this section, we present two algorithms for Problem 1: *LASS-Lite* and *LASS-NA*. LASS-Lite is a simple baseline: it constructs a well-spread seed set C via LID-stratified D^2 seeding and then attaches a small cohort of nearby neighbors to each seed. LASS-NA, our primary algorithmic contribution, builds on these seeds and replaces independent, per-seed neighbor attachment with a principled, neighbor-aware submodular optimization (Problem 2), enabling an efficient greedy algorithm with a $(1 - 1/e)$ approximation guarantee (Theorem 3). We also provide theoretical and complexity analyses of both algorithms. Empirically, LASS-NA mainly improves local fidelity by using the neighbor budget more effectively, while preserving global coverage comparable to LASS-Lite (Fig. 4; Exp. 5, Table 5).

4.1 LASS-Lite: a simple and robust baseline

We propose *LASS-Lite* (LID-aware stratified seeding), a simple yet robust baseline. The core idea is to first select *seeds* that are diverse in estimated LID over the dataset, and then attach a small cohort of nearest neighbors to each seed to raise local counts and stabilize LID. All neighbors can be efficiently obtained using approximate nearest neighbor search (ANNS) algorithms, since computing exact neighbors is prohibitively expensive in high-dimensional spaces [16, 34, 38]. To

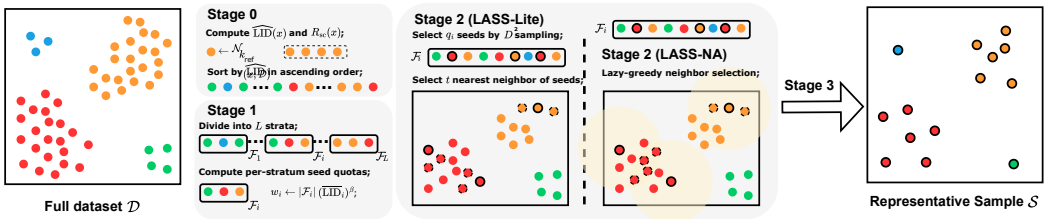


Fig. 3. Pipeline overview of LASS-Lite and LASS-NA. Panel labels match the stage headers in Algorithms 1-2.

keep the procedure internally consistent, we define the self-consistent scale as:

$$R_{\mathrm{sc}}(x) \;=\; \mathrm{dist}\big(x, \mathrm{NN}_{k_{\mathrm{sol}}}(x; \mathcal{D}\setminus\{x\})\big). \tag{2}$$

This aligned internal radius is reused for reinforcement and per-point LID estimation.

LASS-Lite algorithm proceeds in four stages (Algorithm 1).

- **Stage 0 (lines 1–6):** Set the sample budget $m = \lfloor n \cdot \rho \rfloor$, internal neighbor count k_{sol}, and per-seed reinforcement budget t. Cap seeds $N_{\mathrm{seed}} = \lfloor m/(t+1) \rfloor$. For each $x \in \mathcal{D}$, query an approximate k_{sol}-NN list from G to compute $\widehat{\mathrm{LID}}(x)$ and define the self-consistent radius $R_{\mathrm{sc}}(x)$ as in (2).
- **Stage 1 (lines 7-11):** Choose the number of strata L (default $L = \lfloor \sqrt{m} \rfloor$) and a bias parameter $\beta \geq 0$. We sort all points in $\mathcal{D}$ by their $\widehat{\mathrm{LID}}$ values and then partition this sorted list into L contiguous blocks of equal size to form the strata $\{\mathcal{F}_i\}_{i=1}^{L}$. Let $\overline{\mathrm{LID}}_i$ be the mean in stratum i and set $w_i = |\mathcal{F}_i|\,(\overline{\mathrm{LID}}_i)^{\beta}$. Compute fractional quotas $q_i' = N_{\mathrm{seed}} \cdot w_i / \sum_{j=1}^{L} w_j$; obtain integer q_i by rounding so that $\sum_{i=1}^{L} q_i = N_{\mathrm{seed}}$.
- **Stage 2 (per-stratum D^2-seeding; lines 12–19):** For each LID stratum $\mathcal{F}_i$ *independently*, we select q_i seeds within $\mathcal{F}_i$ by D^2-sampling *restricted to that stratum*. Concretely, initialize an empty per-stratum set $C_i \leftarrow \emptyset$. If $q_i > 0$, pick one initial seed uniformly at random from $\mathcal{F}_i$ and put it into C_i (per-stratum warm-start). Then, while $|C_i| < q_i$, sample $x \in \mathcal{F}_i \setminus C_i$ with probability

$$\Pr(x \mid C_i) \;=\; \frac{\mathrm{dist}(x, C_i)^2}{\sum_{y \in \mathcal{F}_i \setminus C_i} \mathrm{dist}(y, C_i)^2}, \;\; \mathrm{dist}(x, C_i) = \min_{c \in C_i} \mathrm{dist}(x, c),$$

  where distances are retrieved via ANNS on the proximity graph G. After all strata finish, we set the global seed set $C \leftarrow \bigcup_{i=1}^{L} C_i$. For each $c \in C$, attach up to t nearest neighbors (via ANNS on G) restricted to the ball $B\big(c, R_{\mathrm{sc}}(c)\big)$; then deduplicate to form $\mathcal{S}$. This stage maximizes intra-stratum diversity while decoupling strata from one another.
- **Stage 3 (lines 18-20):** If $|\mathcal{S}| < m$, repeatedly add farthest-first points $x \in \mathcal{D} \setminus \mathcal{S}$ maximizing $\mathrm{dist}(x, \mathcal{S})$ (distances retrieved via G) until $|\mathcal{S}| = m$.

Link to challenges. To circumvent the NP-hardness of problem 1 (C1), LASS-Lite employs a multi-stage heuristic that decomposes the task into a sequence of scalable, near-linear time operations. The algorithm explicitly navigates the conflict between objectives (C2) through its sequential "coverage-first, fidelity-second" structure: it first establishes broad spatial coverage by seeding diverse points across LID strata, and then directly improves local fidelity by reinforcing these seeds with nearby neighbors. This final reinforcement step is crucial for addressing the instability of local data distribution (C3). By ensuring each seed is accompanied by at least $t \geq 3$ neighbors, it pushes the local sample count $m_{\mathcal{S}}(s)$ into the finite-variance regime for the LID estimator (Lemma 1), thereby improving the local fidelity.

Algorithm 1: LASS-Lite

Input: $\mathcal{D}$, sampling ratio ρ, k_{sol}, LID weight $\beta \geq 0$, reinforcement $t \in \mathbb{N}$.
Output: $\mathcal{S}$ with $|\mathcal{S}| = m = \lfloor n \cdot \rho \rfloor$.

/* Stage 0: PG & LID on the self-consistent scale */

$m \leftarrow \lfloor n \cdot \rho \rfloor$; $N_{\text{seed}} \leftarrow \lfloor m/(t+1) \rfloor$; # strata $L \leftarrow \lfloor \sqrt{m} \rfloor$;
Build or reuse a proximity graph G on $\mathcal{D}$ (for ANNS);
foreach $x \in \mathcal{D}$ **do**
  obtain $\mathcal{N} \leftarrow$ approximate k_{sol}-NN of x from G;
  compute $\widehat{\text{LID}}(x)$ from $\mathcal{N}_{k_{\text{sol}}}(x; \mathcal{D})$;
  set $R_{\text{sc}}(x)$ by (2);

/* Stage 1: LID stratification & per-stratum seed quotas */

Partition $\mathcal{D}$ into L equal-frequency strata $\{\mathcal{F}_i\}$ by $\widehat{\text{LID}}$;
for $i = 1$ **to** L **do**
  $\overline{\text{LID}}_i \leftarrow \frac{1}{|\mathcal{F}_i|} \sum_{x \in \mathcal{F}_i} \widehat{\text{LID}}(x)$;
  $w_i \leftarrow |\mathcal{F}_i| \, (\overline{\text{LID}}_i)^{\beta}$;
Let $W \leftarrow \sum_{j=1}^{L} w_j$ and $q'_i \leftarrow N_{\text{seed}} \cdot \frac{w_i}{W}$ for $i = 1, \ldots, L$; round $\{q'_i\}$ to integers $\{q_i\}$ by largest-remainder so that $\sum_i q_i = N_{\text{seed}}$;

/* Stage 2: independent D^2-seeding within each stratum */

$C \leftarrow \emptyset$;
for $i = 1$ **to** L **do**
  if $q_i = 0$ ***or*** $|\mathcal{F}_i| = 0$ **then continue**
  $C_i \leftarrow \emptyset$;
  /* per-stratum warm-start: one uniform seed if quota > 0 */
  pick $c_0 \sim \text{Unif}(\mathcal{F}_i)$ and set $C_i \leftarrow \{c_0\}$;
  while $|C_i| < q_i$ **do**
    compute $\text{dist}(x, C_i) = \min_{c \in C_i} \text{dist}(x, c)$ for all $x \in \mathcal{F}_i \setminus C_i$ via ANNS on G;
    sample x from $\mathcal{F}_i \setminus C_i$ with probability $\Pr(x \mid C_i) \propto \text{dist}(x, C_i)^2$;
    $C_i \leftarrow C_i \cup \{x\}$;
  $C \leftarrow C \cup C_i$;
$\mathcal{S} \leftarrow \emptyset$; **foreach** $c \in C$ **do**
  add c to $\mathcal{S}$ and then add up to t nearest neighbors of c within $B(c, R_{\text{sc}}(c))$ (via ANNS on G);
Remove duplicates from $\mathcal{S}$.

/* Stage 3: Budget normalization (coverage-friendly) */

while $|\mathcal{S}| < m$ **do**
  add to $\mathcal{S}$ the point in $\mathcal{D} \setminus \mathcal{S}$ that is farthest from the current $\mathcal{S}$;
return $\mathcal{S}$.

4.2 LASS-NA: neighbor-aware cohort selection as budgeted concave coverage

In LASS-Lite, reinforcement neighbors are chosen independently for each seed. However, any selected neighbor u can be a shared neighbor, lying in multiple self-consistent balls $B(s, R_{\text{sc}}(s))$ and thus potentially stabilizing the LID for several seeds at once. To capitalize on this, we reformulate the neighbor selection task. Our goal is to directly improve local fidelity by stabilizing LID estimates. Since the Mean Squared Error (MSE) of the LID estimator decreases as the local neighbor count m_s grows (Lemma 1), we can frame this as a problem of maximizing the total MSE reduction across all seeds. We formalize this as a budgeted selection problem.

Algorithm 2: LASS-NA

Input: $\mathcal{D}$, ρ, k_{sol}, L, β, candidate cap ℓ.
Output: $\mathcal{S}$ with $|\mathcal{S}| = m = \lfloor n \cdot \rho \rfloor$.

/* **Stage 2: Seeds** */
Run Algorithm 1 through Stage 2 to get C; $m \leftarrow \lfloor n \cdot \rho \rfloor$; $B \leftarrow m - |C|$;
foreach $s \in C$ **do**
  build $\mathcal{B}(s) = \{u \in \mathcal{D} \setminus \{s\} : \text{dist}(u, s) \le R_{\text{sc}}(s)\}$;
  keep the ℓ closest elements in $\mathcal{B}(s)$; set $m_s \leftarrow 0$;
$\mathcal{U} \leftarrow \bigcup_s \mathcal{B}(s)$; $\mathcal{S} \leftarrow C$;
/* **Lazy-greedy on** F */
Initialize a max-heap keyed by an upper bound on $\text{Score}(u) = \sum_{s: u \in \mathcal{B}(s)} \Delta g_s(m_s)$;
for $b = 1$ **to** B **do**
  extract u with the largest key;
  recompute its exact $\text{Score}(u)$;
  if *it is still best* **then**
    add u to $\mathcal{S}$;
    for all s with $u \in \mathcal{B}(s)$ set $m_s \leftarrow m_s + 1$;
    delete u from $\mathcal{U}$;
  else
    push u back with the refined key;
/* **Stage 3: Budget normalization** */
while $|\mathcal{S}| < m$ **do**
  add to $\mathcal{S}$ the farthest point from $\mathcal{S}$ in $\mathcal{D} \setminus \mathcal{S}$;
return $\mathcal{S}$.

Problem 2 (Neighbor-Aware Cohort Selection). *Given the seed set C (from Algorithm 1 up to Stage 2), a residual cardinality budget $B = m - |C|$, and the per-seed candidate sets*

$$\mathcal{B}(s) \;=\; \big(\mathcal{D} \cap B(s, R_{\text{sc}}(s))\big) \setminus \{s\}, \qquad \mathcal{U} \;=\; \bigcup_{s \in C} \mathcal{B}(s),$$

select $\mathcal{A} \subseteq \mathcal{U}$ with $|\mathcal{A}| \le B$ to maximize

$$F(\mathcal{A}) \;=\; \sum_{s \in C} g_s\big(m_s(\mathcal{A})\big), \qquad m_s(\mathcal{A}) = |\mathcal{A} \cap \mathcal{B}(s)|, \tag{3}$$

where g_s is a nondecreasing, concave surrogate for the reduction in the LID-estimation MSE at seed s.

We instantiate g_s from Lemma 1. For $m \ge 3$,

$$\text{MSE}_s(m) = \frac{D(s)^2\,(m+2)}{(m-1)(m-2)}, \quad D(s) \approx \widehat{\text{LID}}(s), \tag{4}$$

$$\Delta g_s(m) = \text{MSE}_s(m) - \text{MSE}_s(m{+}1) = \frac{D(s)^2\,(m+6)}{m\,(m-1)\,(m-2)}. \tag{5}$$

For $m < 3$, define a concave extension by setting the discrete slopes $\Delta g_s(0) = \Delta g_s(1) = \Delta g_s(2) = \gamma_s\,\Delta g_s(3)$ with a fixed $\gamma_s \ge 1$ (default $\gamma_s = 2$), and then $g_s(m) = \sum_{r=0}^{m-1} \Delta g_s(r)$. This guarantees that $\Delta g_s(m)$ is nonincreasing in m on $\mathbb{Z}_{\ge 0}$, hence g_s is nondecreasing and concave on $\mathbb{Z}_{\ge 0}$.

LASS-NA algorithm augments LASS-Lite by jointly selecting reinforcement neighbors that benefit multiple seeds. With budget $B = m - |C|$, it maximizes $F(\mathcal{A})$ in Problem 2 via a lazy-greedy.

- **Stage 2 (lines 1–5): Seed reuse and candidate construction.** Reuse the seed set C from Algorithm 1 (up to Stage 2). For each $s \in C$, form $\mathcal{B}(s) = (\mathcal{D} \cap B(s, R_{\text{sc}}(s))) \setminus \{s\}$ using ANNS on G; cap each $\mathcal{B}(s)$ to its ℓ nearest elements. Initialize $m_s \leftarrow 0$ and set $\mathcal{U} \leftarrow \bigcup_{s \in C} \mathcal{B}(s)$.
- **(lines 6–15): Lazy-greedy neighbor selection.** Maintain a max-heap keyed by an upper bound on $\text{Score}(u) = \sum_{s:\, u \in \mathcal{B}(s)} \Delta g_s(m_s)$, where Δg_s is given in (5). For $b = 1, \ldots, B$: pop the top u, recompute its exact $\text{Score}(u)$; if it remains best, accept u into $\mathcal{S}$ and update all affected counts $m_s \leftarrow m_s + 1$; otherwise push u back with the refined key. Stop early if the heap becomes empty.

LASS-NA reformulates the task as a monotone submodular maximization problem (Problem 2). This shift not only makes the problem tractable via a near-linear time lazy-greedy algorithm but also provides a strong $(1 - 1/e)$ approximation guarantee (Theorem 3). Compared to LASS-Lite, the neighbor-aware selection improves local neighborhood stability (C3). The property of diminishing returns means the marginal gain, Δg_s, is largest for seeds with the fewest neighbors, compelling the algorithm to automatically allocate its budget toward repairing the most sparse or fragile local data distribution structures, thus preventing fidelity loss.

4.3 Algorithmic Analysis

We now analyze the theoretical guarantees, computational complexity, and practical parameterization of our proposed algorithms, connecting their design to the statistical motivations established in Section 3.

Theoretical guarantees for LASS-NA. The neighbor-aware selection stage of LASS-NA (Stage 2) is designed around a key insight: the problem of selecting neighbors to maximize the stability of LID estimates can be framed as maximizing a monotone submodular function. This structure allows us to provide a strong approximation guarantee. We prove these properties below.

Lemma 4. *For $m \geq 3$, $\text{MSE}_s(m)$ in (4) is strictly decreasing in m, and*

$$\Delta g_s(m) = \text{MSE}_s(m) - \text{MSE}_s(m{+}1) = \frac{D(s)^2\,(m+6)}{m\,(m-1)\,(m-2)}$$

is strictly positive and decreasing in m. Hence $g_s(m) = \sum_{r=0}^{m-1} \Delta g_s(r)$ is nondecreasing and concave on $\mathbb{Z}_{\geq 0}$.

Proof. A direct calculation shows $\text{MSE}_s(m{+}1) - \text{MSE}_s(m) = -\Delta g_s(m) < 0$ for $m \geq 3$, so MSE_s decreases. For $m \geq 3$, $\Delta g_s(m) > 0$ and $\Delta g_s(m{+}1) - \Delta g_s(m) < 0$ (monotone decrease in m), hence discrete concavity of g_s. □

Lemma 5. *Let F be given by (3) with each g_s nondecreasing and concave. Then $F : 2^{\mathcal{U}} \to \mathbb{R}_{\geq 0}$ is nondecreasing and submodular.*

Proof. For $\mathcal{A} \subseteq \mathcal{U}$ and $u \in \mathcal{U}$, the marginal gain is

$$\Delta_F(\mathcal{A}; u) = F(\mathcal{A} \cup \{u\}) - F(\mathcal{A}) = \sum_{s:\, u \in \mathcal{B}(s)} \big[g_s(m_s(\mathcal{A}){+}1) - g_s(m_s(\mathcal{A}))\big].$$

Since g_s is nondecreasing, each term is ≥ 0 and thus $\Delta_F(\mathcal{A}; u) \geq 0$ (monotonicity). If $\mathcal{A} \subseteq \mathcal{B}$ and $u \notin \mathcal{B}$ then $m_s(\mathcal{A}) \leq m_s(\mathcal{B})$ for all s, and discrete concavity implies $g_s(m{+}1) - g_s(m)$ is nonincreasing in m, so $\Delta_F(\mathcal{A}; u) \geq \Delta_F(\mathcal{B}; u)$ (diminishing returns), proving submodularity. □

Theorem 3. *Let B be the budget and let $\mathcal{A}_{\text{gr}}$ be the size-B output of (lazy-)greedy in Algorithm 2. Then*

$$F(\mathcal{A}_{\text{gr}}) \;\geq\; \big(1 - 1/e\big) \max_{|\mathcal{A}| \leq B} F(\mathcal{A}).$$

Proof. By Lemma 5, F is monotone submodular. Let $\mathcal{A}^\star$ be an optimal solution of size at most B, and let $\mathcal{A}_i$ denote the greedy set after i elements ($\mathcal{A}_0 = \emptyset$). For any set $\mathcal{S}$ and element u, write $\Delta_F(\mathcal{S}; u)$ for the marginal gain. Submodularity and monotonicity yield the *fundamental inequality*:

$$F(\mathcal{A}^\star) - F(\mathcal{A}_{i-1}) \le \sum_{u \in \mathcal{A}^\star \setminus \mathcal{A}_{i-1}} \Delta_F(\mathcal{A}_{i-1}; u) \le B \cdot \max_{u \in \mathcal{U} \setminus \mathcal{A}_{i-1}} \Delta_F(\mathcal{A}_{i-1}; u). \tag{6}$$

The last maximum is attained by the greedy choice u_i, hence

$$F(\mathcal{A}_i) - F(\mathcal{A}_{i-1}) = \Delta_F(\mathcal{A}_{i-1}; u_i) \;\ge\; \frac{1}{B}\Big(F(\mathcal{A}^\star) - F(\mathcal{A}_{i-1})\Big).$$

Rearranging gives a one-step contraction:

$$F(\mathcal{A}^\star) - F(\mathcal{A}_i) \;\le\; \Big(1 - \tfrac{1}{B}\Big)\Big(F(\mathcal{A}^\star) - F(\mathcal{A}_{i-1})\Big).$$

Unrolling for $i = 1, \ldots, B$,

$$F(\mathcal{A}^\star) - F(\mathcal{A}_B) \;\le\; \Big(1 - \tfrac{1}{B}\Big)^B F(\mathcal{A}^\star) \;\le\; e^{-1} F(\mathcal{A}^\star),$$

which implies $F(\mathcal{A}_B) \ge (1 - 1/e) F(\mathcal{A}^\star)$.

The lazy implementation preserves the selected sequence of elements, thus achieves the same guarantee. □

This guarantee ensures that LASS-NA's neighbor selection stage is provably effective at stabilizing local data distributions. By maximizing a proxy for MSE reduction, the algorithm directly addresses the fidelity objective and mitigates the local sparsity issues discussed in Section 3, thereby improving fidelity over LASS-Lite, especially when budgets are tight.

Connecting theory to practice: parameter rationale. Our theoretical analysis directly informs the choice of default hyperparameters, designed to be effective without dataset-specific tuning. The key parameters, the reinforcement budget (t or B) and the LID-stratification bias (β), are motivated by our fidelity objective. For LASS-Lite, we set a conservative local reinforcement budget of $t = 3$. This choice is not arbitrary; it is the minimum required to push the local neighbor count m_s into the finite-variance regime of the LID estimator ($m_s \ge 3$, per Lemma 1), thus ensuring a basic level of stability of local data with minimal impact on the number of initial seeds. For the LID-stratification bias, we set $\beta = 1$. A positive β intentionally allocates more seeds to strata with higher average LID, prioritizing the preservation of regions with complex local data distributions. A simple linear weighting ($\beta = 1$) provides a natural and robust baseline for this bias. These principled defaults ensure that our methods robustly navigate the coverage-fidelity trade-off based on the properties of the local data distribution established in Section 3.

Computational complexity. We analyze the complexity of the sampling process itself, assuming a reusable vector index G (e.g., HNSW) is available or has been pre-computed. The cost of building this index is a one-time, upfront expense that we exclude from this analysis. Let $c_{\text{dist}}(d)$ be the cost of a distance computation ($O(d)$) and $T_{\text{query}}(k; d, n)$ be the cost of an ANNS query, which is typically polylogarithmic in n.

LASS-Lite. Stage 0 computes $\widehat{\text{LID}}$ and R_{sc} by performing n approximate nearest neighbor queries, leading to a total cost of $O(n \cdot T_{\text{query}}(k_{\text{sol}}; d, n))$. Stratification and quota rounding are dimension-independent, costing a negligible $O(n)$. The subsequent D^2-seeding over L strata involves repeated distance calculations, costing $\tilde{O}(d \cdot N_{\text{seed}} \cdot n/L)$, while reinforcement adds $O(N_{\text{seed}} \cdot T_{\text{query}}(t; d, n))$ for the local ANNS calls. Finally, the coverage top-up using a max-heap costs $O((n - m) \log n)$

Table 2. Datasets summary.

Dataset	Base Size (n)	Query Size	Dim. (d)	LID Mean, Range
SIFT1M [3]	1,000,000	10,000	128	8.28, [1.31, 19.16]
GIST [3]	1,000,000	1,000	960	15.56, [1.16, 33.23]
MSONG [6]	992,272	200	420	7.97, [1.29, 79.02]
GLOVE [44]	1,183,514	10,000	100	16.28, [3.64, 45.49]
SIFT100M [3]	100,000,000	10,000	128	8.61, [2.05, 16.80]

for heap management, with an additional dimension-dependent cost for updating nearest-sample distances. The overall complexity is dominated by the initial query and seeding phases.

$$T_{\text{Lite}} = \tilde{O}\big(n \cdot T_{\text{query}}(k_{\text{sol}}; d, n) + d \cdot n \cdot N_{\text{seed}}/L\big).$$

LASS-NA. This method's complexity is also dominated by its initial data-processing stages. The cost of generating seeds, T_{seed}, is equivalent to that of LASS-Lite through Stage 2. Building the candidate pools for the $|C|$ seeds requires an additional radius or ANNS search per seed to find up to ℓ neighbors, adding a cost of $O(|C| \cdot T_{\text{query}}(\ell; d, n))$. The subsequent lazy-greedy selection performs B extractions with a cost of $O(|C|\ell + B\log(|C|\ell))$, which is independent of the dimension d as it operates on the pre-computed candidate sets. The total complexity is therefore the sum of these dimension-dependent data retrieval costs and the dimension-agnostic optimization cost.

$$T_{\text{NA}} = T_{\text{seed}} + O\big(|C| \cdot T_{\text{query}}(\ell; d, n) + B\log(|C|\ell)\big).$$

Memory overhead is $O(n)$ for per-point scalars plus $O(|C|\ell)$ for the candidate pools. In practice, as constants t, ℓ, k_{sol} and sample m are much smaller than n, algorithms remain near-linear in n.

5 Experiments

All experiments were conducted on a single server equipped with an Intel Xeon Gold 6330 (2.0 GHz) CPU, NVIDIA GeForce RTX 4090 GPU and 1 TB RAM.

Datasets. We evaluate four public million-scale datasets; basic statistics are summarized in Table 2. Our experiments span both large-scale vector benchmarks and downstream application workloads, and thus involve different datasets. Specifically, Exp. 1, Exp. 4, and Exp. 5 evaluate objective-side performance and system-level impact on the four million-scale datasets; Exp. 2–3 focus on GLOVE, whose semantic clusters are visually interpretable while its LID distribution is relatively challenging. Exp. 6 evaluates end-to-end scalability on SIFT100M, and Exp. 7 uses CIFAR-10 embeddings to study label-efficient model selection.

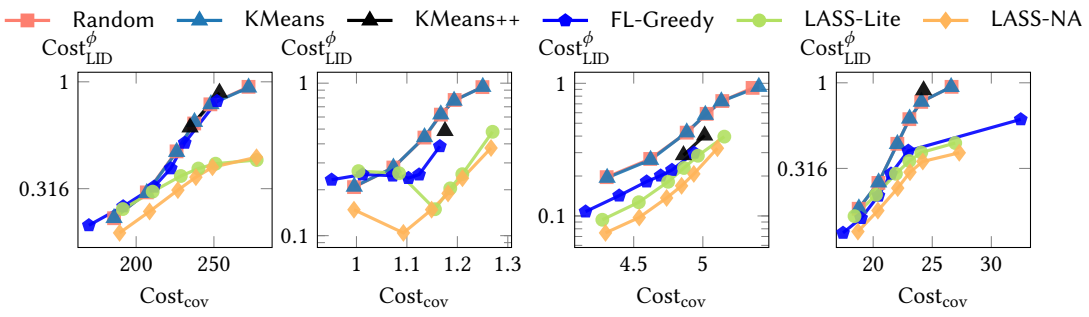
Compared methods. We implement all methods in `Python`.

• **Random [9, 51]**: A simple and widely used baseline that uniformly samples $m = \lfloor n \cdot \rho \rfloor$ points from the dataset.

• **KMeans++ [4]**: A classic method for generating spatially diverse points. We include it to establish a strong baseline optimized purely for coverage. It selects the centroids of m clusters using the FAISS [11, 24] KMeans++ implementation and then snaps each centroid to its nearest data point. While theoretically strong for coverage, its high complexity of $O(nmd \cdot \texttt{niter})$ makes it computationally expensive for large sample sizes.

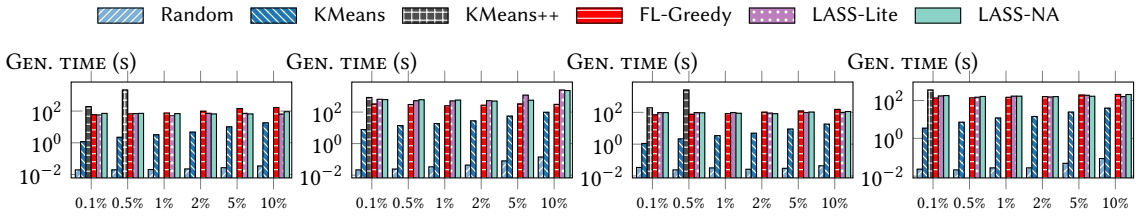
• **KMeans**: A lightweight variant of KMeans++. This method first partitions the dataset into $k = \lfloor \sqrt{m} \rfloor$ clusters, then samples points from each cluster, with the number of points proportional to the cluster size. For KMeans-based methods, we set $\texttt{niter} = 20$.

• **FL-Greedy [26, 60]**: A state-of-the-art greedy approach from the submodular optimization literature [45], widely used for data summarization. It models subset selection as a facility location problem, supporting global coverage. Our implementation operates on a sparse similarity graph



(a) *SIFT1M* (b) *GIST* (c) *GLOVE* (d) *MSONG*

Fig. 4. Objective-side trade-offs (Exp. 1): Cost_{cov} vs. $\text{Cost}^{\phi}_{\text{LID}}$ on four datasets. Each line denotes a method (corresponding to different budgets $\rho \in \{0.1\%, 0.5\%, 1\%, 2\%, 5\%, 10\%\}$). Lower-Left is better.



(a) *SIFT1M* (b) *GIST* (c) *GLOVE* (d) *MSONG*

Fig. 5. Sample-generation wall time by method and budget (log scale).

constructed using an approximate k-NN graph (HNSW, with $k = 128$) [38] and employs an efficient Lazier-than-Lazy algorithm [41] for selection.

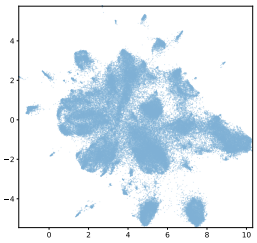
- **LASS-Lite (ours)**: LID-aware stratified seeding with local reinforcement of t neighbors within the self-consistent radius $R_{\text{sc}}(\cdot)$ (as defined in § 4.1). Defaults: k_{sol}=50, t=3, L=$\lfloor\sqrt{m}\rfloor$, and bias β=1.
- **LASS-NA (ours)**: Neighbor-aware cohort selection via lazy-greedy on a concave stabilization objective (as defined in § 4.2). Defaults: neighbor-selection budget t=8 and candidate cap ℓ=32.

We also considered SimSTV [5], but its cubic complexity, $O(dn^2 + n^3 \log n)$, makes it infeasible at scale. Similarly, we adapted the topology-preserving graph coarsening method GEC [40] by first constructing a k-NN graph (HNSW, with $k = 50$). This approach also proved intractable at scale due to its reliance on maximal clique enumeration, which suffers from a combinatorial explosion on the dense local subgraphs induced by k-NN construction.

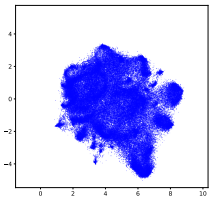
General setup. Unless otherwise stated, the graph-based ANNS index is HNSW with M=50, `efC`=400, and `efS`=200. When we report $\text{Cost}^{\phi}_{\text{LID}}$, we set $k_{\text{ref}} = 100$ and choose τ as the 5th percentile of the global LID distribution.

▷ **Exp.1: Objective-side performance and efficiency.** We evaluate the proposed methods on two critical dimensions: (i) the trade-off between global coverage (Cost_{cov}) and local fidelity ($\text{Cost}^{\phi}_{\text{LID}}$), and (ii) the computational efficiency (runtime) of sample generation. We sweep sampling ratios $\rho \in \{0.1\%, 0.5\%, 1\%, 2\%, 5\%, 10\%\}$ across all datasets to observe trends under varying budgets.

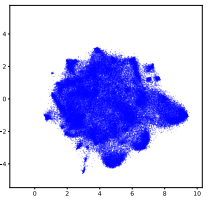
Figure 4 presents the trade-off between coverage and fidelity. LASS-NA consistently dominates the Pareto frontier, offering a superior balance compared to baselines. For instance, on GLOVE with a tight budget ($\rho = 0.5\%$), LASS-NA achieves a $\text{Cost}^{\phi}_{\text{LID}}$ of 0.207. This represents a 71.7% reduction in fidelity cost compared to KMeans (0.732) and also outperforms FL-Greedy (0.222) while maintaining comparable global coverage. Furthermore, on the larger GIST dataset with a larger $\rho = 5\%$ budget,



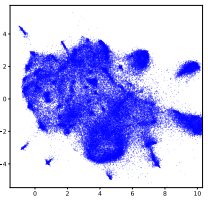
(a) Full GLOVE Dataset



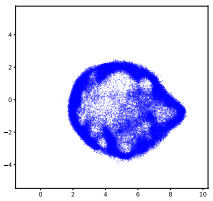
(b) Random, $\rho = 1\%$



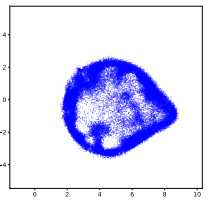
(c) KMeans, $\rho = 1\%$



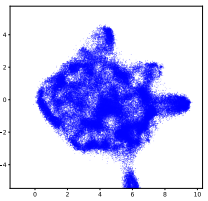
(d) LASS-NA, $\rho = 1\%$



(e) Random, $\rho = 0.1\%$



(f) KMeans, $\rho = 0.1\%$



(g) LASS-NA, $\rho = 0.1\%$

Table 3. Quantitative evaluation at $\rho = 0.1\%$.

Method	Overlap ↑	Trustworthiness ↑	Continuity ↑
Random	30.5%	83.2%	92.1%
KMeans	26.0%	81.9%	90.1%
LASS-NA	38.2%	86.5%	92.2%

Fig. 7. UMAP visualization of the GLOVE dataset. Each embedding is generated by a model trained on a sampled subset and then applied to the full data. (a) shows the ground-truth embedding from the full dataset. The subsequent plots compare visualizations from samples obtained via Random, KMeans, and LASS-NA at sampling ratios of (b-d) $\rho = 1.0\%$ and (e-g) $\rho = 0.1\%$.

Table 4. Performance on the self-supervised prototyping task (GLOVE). We report the mean ± std over 10 random seeds. Best results are highlighted in gray.

	$\rho = 0.1\%$			$\rho = 0.5\%$			$\rho = 1.0\%$		
Method	Steps-to-τ	Best Val	P@10	Steps-to-τ	Best Val	P@10	Steps-to-τ	Best Val	P@10
LASS-NA	63.2 ± 0.4	8.34 ± 0.03	0.600 ± 0.004	30.0 ± 0.0	7.24 ± 0.01	0.762 ± 0.003	20.8 ± 0.8	7.03 ± 0.00	0.793 ± 0.002
FL-Greedy	79.0 ± 5.1	8.55 ± 0.03	0.565 ± 0.004	43.8 ± 0.4	7.38 ± 0.01	0.706 ± 0.002	23.8 ± 0.4	7.02 ± 0.01	0.753 ± 0.002
Random	∞	9.09 ± 0.03	0.550 ± 0.002	41.0 ± 3.1	7.56 ± 0.01	0.703 ± 0.003	17.8 ± 1.1	7.05 ± 0.02	0.787 ± 0.004
KMeans	∞	9.32 ± 0.02	0.564 ± 0.004	34.0 ± 2.5	7.52 ± 0.01	0.703 ± 0.003	16.8 ± 0.8	7.02 ± 0.02	0.803 ± 0.002

LASS-NA reduces the local fidelity cost by 58.6% to 62.7% against three competitors at similar coverage levels.

Figure 5 shows the sample generation time as the sampling ratio ρ increases. Our LASS-Lite and LASS-NA exhibit a fixed-cost behavior dominated by the one-time index construction, resulting in a stable runtime that is effectively decoupled from the sampling ratio. In contrast, the clustering-based baselines are highly sensitive to the sample size, showing a linear growth profile. For instance, the runtime of KMeans on GLOVE surges by over 16× as ρ increases from 0.1% to 10%, and KMeans++ frequently fails to complete within the 3,600s cutoff.

▹ **Exp.2: Visualization of global structure preservation.** To evaluate how well a small sample preserves the global structure of the full dataset, we employ UMAP [39] for dimensionality reduction. We first train a UMAP model solely on the sampled subset $\mathcal{S}$, using parameters such as $n_{\text{neighbors}} = \min\{30, \lfloor|\mathcal{S}|/3\rfloor\}$, `min_dist`=0.1, and the cosine metric for 500 epochs. We then use this trained model to transform the entire dataset $\mathcal{D}$. The resulting embedding is aligned to the ground truth via Orthogonal Procrustes Analysis. We introduce three quantitative metrics widely used for evaluating neighborhood preservation in dimensionality reduction: *Overlap*, *Trustworthiness*, and *Continuity* [25, 32, 50].

The results are presented in Figure 7. At a moderate sampling ratio of $\rho = 1\%$, all methods capture the main cluster structures, though LASS-NA offers visibly sharper separation (Fig. 7(b-d)). The crucial distinction emerges at the tight budget of $\rho = 0.1\%$. Visually, models trained on the

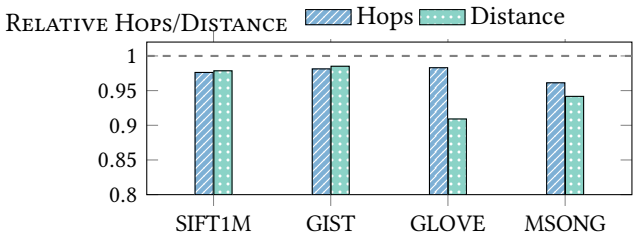


Fig. 8. Performance of the navigation graphs ($\rho = 0.1\%$) built by LASS-NA compared to Random, showing the relative reductions in search path length (hops) and initial seed-to-query distance. All metrics are normalized to Random (1.0). Lower values indicate better performance.

Random and KMeans samples exhibit sampling bias, failing to map distinct clusters. (Fig. 7(e, f)). In contrast, LASS-NA retains representatives from both dense and sparse regions, preserving the distinct local structures (Fig. 7(g)). This visual superiority is confirmed quantitatively in Table 3. LASS-NA achieves the highest scores across all three metrics.

▹ **Exp. 3: Downstream task performance via self-supervised prototyping.** To assess how sample quality affects downstream machine learning tasks, we conduct a self-supervised model prototyping experiment on GLOVE. The task involves pretraining a two-layer MLP ($f_\theta : \mathbb{R}^d \to \mathbb{R}^d$) using a graph-regularized objective. The loss combines a reconstruction error with a graph-smoothness term:

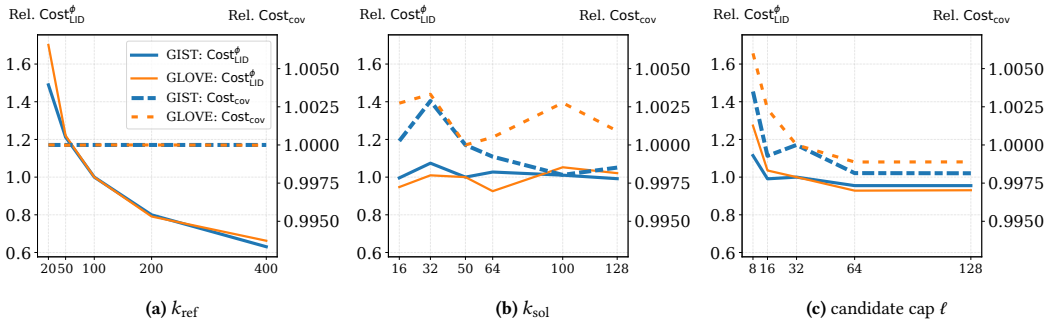
$$\mathcal{L} = \frac{1}{|B|} \sum_{i \in B} \|f_\theta(x_i) - x_i\|_2^2 + \lambda \frac{1}{|\mathcal{E}_B|} \sum_{(i,j) \in \mathcal{E}_B} w_{ij} \, \|f_\theta(x_i) - f_\theta(x_j)\|_2^2$$

The effectiveness of this regularizer hinges on the quality of the training graph G_S, which is constructed exclusively from the sample $\mathcal{S}$. We evaluate generalization on the full dataset $\mathcal{D}$ using three metrics: *Best Val* (lowest validation loss), *Steps-to-τ* (epochs to reach a validation threshold), and *P@10* (10-NN precision of the learned embedding). We hold all hyperparameters constant across methods, including the validation graph G_E built on $\mathcal{D}$. The results in Table 4 show that LASS-NA delivers the most robust performance, especially under tight data budgets (mean ± std over 10 random seeds). At the tightest budget ($\rho = 0.1\%$), our advantage is substantial: models trained on Random and KMeans samples fail to converge. While FL-Greedy produces a convergent model, LASS-NA converges faster (63.2 ± 0.4 vs. 79.0 ± 5.1 steps) and yields a better embedding (P@10 of 0.600 ± 0.004 vs. 0.565 ± 0.004). This highlights the importance of preserving local fidelity when data are sparse. At a moderate budget ($\rho = 0.5\%$), LASS-NA remains the top performer across all three metrics, achieving the fastest convergence, the best validation loss, and the highest neighborhood precision (P@10 of 0.762 ± 0.003). With a more generous budget ($\rho = 1.0\%$), the performance gap narrows; nevertheless, LASS-NA remains consistently strong in the most challenging low-budget settings, where sample quality is critical for successful model prototyping.

▹ **Exp. 4: Improving seed selection in graph-based ANNS.** We investigate the downstream impact of our sampling on a practical database application: seed selection in graph-based ANNS. Many systems employ a small in-memory navigation graph to guide searches into a much larger disk-resident graph index [53]. The quality of the sample used to build this navigation graph directly affects search efficiency. Our protocol is designed to isolate the effect of seed quality. The main search is always performed on a fixed Vamana graph [22] built over the full dataset. To obtain starting points (seeds), we construct a small auxiliary HNSW navigation graph [38] using a 0.1% sample, following the navigation-graph sampling ratio adopted in Starling [53]. The only variable is the sampling method used to build this navigation graph: LASS-NA versus Random. While final system throughput (QPS) depends on many factors (e.g., main-graph parameters and hardware),

Table 5. Ablation study on the neighbor-aware selection mechanism. For each sampling ratio, we highlight the method achieving lower (better) $\text{Cost}^{\phi}_{\text{LID}}$.

	GIST				GLOVE			
	LASS-Lite		LASS-NA		LASS-Lite		LASS-NA	
ρ	Cost_{cov}	$\text{Cost}^{\phi}_{\text{LID}}$	Cost_{cov}	$\text{Cost}^{\phi}_{\text{LID}}$	Cost_{cov}	$\text{Cost}^{\phi}_{\text{LID}}$	Cost_{cov}	$\text{Cost}^{\phi}_{\text{LID}}$
0.1%	1.225	0.464	1.224	0.477	5.129	0.339	5.119	0.329
0.5%	1.166	0.309	1.165	0.313	4.948	0.243	4.932	0.205
1.0%	1.141	0.263	1.140	0.244	4.856	0.196	4.848	0.172
2.0%	1.114	0.218	1.112	0.201	4.746	0.158	4.736	0.137
5.0%	1.061	0.162	1.059	0.146	4.541	0.115	4.541	0.099
10.0%	0.994	0.121	0.993	0.109	4.285	0.085	4.295	0.075



(a) k_{ref} **(b)** k_{sol} **(c)** candidate cap ℓ

(d) `efS` **(e)** $(M, \texttt{efC})$

Table 6. Comparison of the stability of LID estimates computed using exact k-NN versus HNSW approximate neighbors, reporting Recall@100, Spearman rank correlation, and mean relative error.

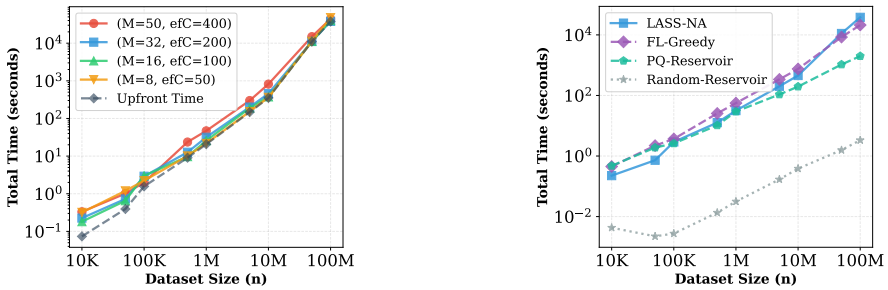
Dataset	Recall@100	Spearman	Mean Rel Err ↓
GIST1M	0.948	0.989	0.024
GLOVE1M	0.849	0.849	0.203

k_{ref}=100; defaults: M=50, `efC`=400, `efS`=200.

Fig. 9. Sensitivity analysis for LASS-NA. (a–e) vary one factor at a time and report the relative change of $\text{Cost}^{\phi}_{\text{LID}}$ and Cost_{cov} (normalized to the default setting, i.e., 1.0).

the initial seed-to-query distance and the subsequent search path length (hops) serve as more direct and sensitive proxies for seeding efficiency. We therefore evaluate seed quality using these two metrics. Figure 8 reports the results, normalized to the Random baseline (dashed line at 1.0), where lower is better. The figure shows that seeds obtained from the LASS-NA navigation graph are consistently better than those from Random. This improvement follows from our method's dual objectives. The superior coverage of the LASS-NA sample ensures that a navigation point is close to a given query, reducing the initial seed-to-query distance by over 9% on GLOVE and nearly 6% on MSONG. Meanwhile, its higher fidelity yields a more representative navigation graph, leading to shorter search paths, as reflected by the reduced hop count across all datasets, including a nearly 4% reduction on MSONG.

▹ **Exp. 5: Ablation study.** This experiment isolates the benefit of LASS-NA's core innovation: its neighbor-aware selection mechanism. We compare LASS-NA against LASS-Lite, configuring both methods with the same reinforcement budget ($t = 8, \beta = 1$) to ensure a fair comparison. Under this controlled setting, any performance difference is attributable to LASS-NA's joint-selection strategy rather than LASS-Lite's per-seed heuristic. Table 5 shows that LASS-NA consistently outperforms



(a) LASS-NA configurations (b) LASS-NA vs. baselines

Fig. 10. Scalability test on SIFT100M (Exp. 6).

LASS-Lite on both GIST and GLOVE, achieving a more favorable coverage–fidelity trade-off (higher local fidelity at comparable global coverage). The improvement is especially pronounced on GLOVE: at $\rho = 2\%$, LASS-NA reduces the fidelity cost by 13.3% (0.137 vs. 0.158) at nearly identical coverage. Overall, LASS-NA makes more effective use of the reinforcement budget than LASS-Lite.

To further assess the robustness of LASS-NA to its internal parameters and ANNS graph quality, we perform a study on GIST1M and GLOVE1M. Unless otherwise stated, we fix a tight sampling ratio of $\rho = 0.1\%$ and use the default configuration: k_{ref}=100, k_{sol}=50, candidate cap ℓ=32, and HNSW settings $(M, \texttt{efC}, \texttt{efS})$=$(50, 400, 200)$. We then vary one parameter at a time: k_{ref} (a), k_{sol} (b), ℓ (c), efS (d), and $(M, \texttt{efC})$ (e), and report the relative changes in $\text{Cost}^{\phi}_{\text{LID}}$ and Cost_{cov}, normalized to the default setting (Fig. 9(a–e)).

Figure 9 reveals four key patterns. First, the global coverage objective is highly stable: Cost_{cov} stays within a very narrow band (around 1.0 after normalization) across all sweeps on both datasets. This suggests that coverage performance is dominated by the stratified D^2 seeding stage and is not brittle to the details of the neighbor-aware refinement. Second, the local fidelity objective varies smoothly and exhibits clear saturation behavior. Varying k_{sol} over a wide range leads to only modest changes in $\text{Cost}^{\phi}_{\text{LID}}$, and increasing the candidate cap beyond $\ell \approx 32$ yields diminishing returns, whereas a very small cap can noticeably hurt fidelity due to insufficient neighbor candidates. Third, LASS-NA does not require a near-perfect ANNS graph. Even when search quality is reduced by lowering efS or using a lighter $(M, \texttt{efC})$ configuration, the objective metrics change moderately without catastrophic degradation. Fourth, to directly quantify the impact of approximate neighbors on LID estimation, we compare LID computed using exact k-NN with LID computed using HNSW approximate neighbors under k_{ref}=100 (Table 6). On GIST1M, HNSW yields nearly exact LID ranks (Spearman = 0.989) with a small mean relative error (2.4%). While GLOVE1M is more challenging (Spearman = 0.849, mean relative error = 20.3%), the end-to-end sensitivity trends in (a–e) remain smooth, indicating that our bounded fidelity loss is empirically robust to moderate kNN noise and ANNS approximation errors.

▹ **Exp. 6: Scalability on SIFT100M.** To validate scalability beyond million-scale datasets, we conduct an end-to-end scalability test on SIFT100M, varying the dataset size from 10K to 100M vectors. We fix the sampling budget at $\rho = 0.1\%$. For LASS-NA, we report the upfront time (including HNSW build and LID computation) and compare it with the total time under different HNSW configurations. Figure 10 summarizes the results. First, Figure 10(a) shows that the end-to-end runtime of LASS-NA scales smoothly up to 100M under all tested index configurations, and varies by only a small factor across settings due to the build/query trade-off. Second, the upfront cost accounts for a substantial fraction of the end-to-end runtime, suggesting that the overall runtime can be reduced by adopting lighter indices. Finally, Figure 10(b) compares LASS-NA with three scalable baselines: Random-Reservoir (streaming uniform reservoir sampling), PQ-Reservoir, and

Table 7. Model recommendation on CIFAR-10 embeddings. We report the mean ± std over 10 random seeds.

Budget	Norm. Acc ↑		Avg. Rank ↓		$\text{Cost}^{\phi}_{\text{LID}}$ ↓	
	LASS-NA	Random	LASS-NA	Random	LASS-NA	Random
100 (0.2%)	0.963 ± 0.012	0.902 ± 0.070	7.00 ± 4.16	7.00 ± 5.62	0.34	0.91
500 (1%)	0.986 ± 0.003	0.985 ± 0.006	3.70 ± 1.49	3.10 ± 2.77	0.28	0.62
1K (2%)	0.983 ± 0.006	0.973 ± 0.008	4.60 ± 2.67	5.50 ± 3.44	0.25	0.44
5K (10%)	0.992 ± 0.005	0.987 ± 0.006	1.70 ± 2.31	2.40 ± 2.88	0.15	0.19
10K (20%)	0.999 ± 0.000	0.997 ± 0.002	1.80 ± 0.42	2.20 ± 1.55	0.10	0.13

FL-Greedy. As expected, Random-Reservoir is faster because it avoids building an ANNS index, whereas PQ-Reservoir and FL-Greedy, which also rely on indices, exhibit similar scaling trends to LASS-NA. Overall, LASS-NA remains practically feasible at the 100M scale, and its dominant costs stem from a one-time index build and a single full pass of neighbor queries.

▹ **Exp. 7: Label-efficient model prototyping for classification.** We study a practical downstream workflow—model selection under a tight labeling budget on CIFAR-10 [29]. To isolate the effect of sampling on vector utility, we freeze a pre-trained ResNet-18 [18] and map each training image to an embedding $\mathbf{x} \in \mathbb{R}^{512}$, yielding a vector pool D with $n = 50{,}000$ instances. We consider a configuration set C of $|C| = 20$ classifiers with heterogeneous capacity and regularization, forming a non-trivial performance ladder.

For evaluation, we first train every $c \in C$ on the *fully labeled* training set to obtain its test accuracy $A(c)$ and the oracle ranking $\pi^\star$ (sorted by $A(c)$); let $A^\star = \max_{c \in C} A(c)$. Then, for a labeling budget B (i.e., $|S| = B$), a sampler produces a subset $S \subset D$ using either LASS-NA or Random, and we only use labels for points in S for prototyping. We split S into $S_{\text{tr}}/S_{\text{val}}$, train each configuration c on S_{tr}, and select $\hat{c} = \arg\max_{c \in C} \tilde{A}_{\text{val}}(c; S_{\text{val}})$, where $\tilde{A}_{\text{val}}$ is the validation accuracy measured on the sampled validation split. We evaluate the selected configuration using the metrics:

$$\text{Norm.Acc}(B) = \frac{A(\hat{c})}{A^\star}, \qquad \text{Avg.Rank}(B) = \mathbb{E}[\text{rank}_{\pi^\star}(\hat{c})],$$

where $\text{rank}_{\pi^\star}(\hat{c}) \in \{1, \ldots, |C|\}$ is the position of $\hat{c}$ in the oracle ranking (smaller is better).

Table 7 shows that LASS-NA yields consistently better model selection under limited labels. At the smallest budget (B=100, 0.2%), LASS-NA achieves substantially higher normalized accuracy (0.963 ± 0.012 vs. 0.902 ± 0.07) and much lower fidelity loss (0.34 vs. 0.91), indicating a markedly more faithful sample even when labels are extremely scarce. With B=1K (2%), LASS-NA improves both selection quality and stability, reducing the average oracle rank of the chosen configuration (4.60 ± 2.67 vs. 5.50 ± 3.44) while maintaining a higher normalized accuracy. At a larger budget (B=10K, 20%), both methods approach the optimum, but LASS-NA remains slightly better across all metrics. Overall, improving local fidelity translates into more reliable validation-based prototyping and thus better configuration choices under a strict labeling budget.

6 Related Work

Data sampling methods. Data sampling is a cornerstone of data systems, underpinning approximate query processing (AQP) [1, 19] and online aggregation [13, 51]. However, applying traditional methods to large-scale data reveals a fundamental trade-off. On the one hand, simple random sampling follows the data density, over-representing dense regions while missing sparse ones, leading to poor global coverage [9]. On the other hand, strategies designed to enforce uniformity, such as selecting k-means centers [4, 37], improve coverage but can fracture the local data distribution by creating artificial repulsion within dense areas (see Figure 1). A more sophisticated line of work frames selection as a multi-winner election to achieve proportional representation, ensuring the sample's composition quantitatively reflects the group proportions of the full dataset [5, 10, 21].

For applications like ANNS or manifold learning, preserving local neighborhood connectivity is paramount, as this structural property is not guaranteed by simply matching cluster population ratios. Our work formalizes this tension and proposes a framework that explicitly navigates the Pareto trade-off between global coverage and fidelity to the local data distribution.

Data sampling for machine learning. In machine learning, data sampling is widely employed to accelerate training by selecting a smaller, informative subset of data [64]. A significant body of work focuses on task-aware or model-aware selection, where the goal is to find a coreset [46] that maximizes performance for a specific model and task. Prominent examples include formulating subset selection as a submodular optimization problem [59] (e.g., CRAIG and GLISTER [27, 42]), as well as data valuation via influence functions [28] and dataset distillation [56]. While these methods are powerful, their reliance on a pre-defined model or task-specific feedback (e.g., gradients) makes them unsuitable for initial data exploration or model-agnostic prototyping. Similarly, in deep metric learning, sophisticated strategies have been developed to select informative training triplets within an already-sampled mini-batch to accelerate model convergence [36]. In contrast, our approach is fundamentally model-agnostic and unsupervised. By directly optimizing for intrinsic properties of the local data distribution, we produce a high-quality "mini-dataset" without reference to any downstream task. This makes our method ideal for applications like rapid model prototyping or providing an informative cold-start batch for active learning systems before any labels are available [47].

Data sampling in high-dimensional ANNS. Among various types of ANNS indexes, graph-based methods [15, 16, 38, 43, 49] have demonstrated state-of-the-art performance in terms of recall and throughput [34, 54]. During the construction of these graph indexes, automated tuning frameworks rely on representative samples to learn optimal configurations [12, 61]. To improve query efficiency, particularly for *disk-resident* graph indexes [22, 53], a navigation graph built over a small set of sampled vectors is often employed to identify better entry points into the full index. In *filtered* ANNS, random sampling is commonly used to estimate selectivity for planning [14, 35, 48, 52, 58]. Although data sampling is prevalent in this domain, simple random sampling remains the default choice in practice, often yielding suboptimal results. To the best of our knowledge, this work is among the first to introduce a *bi-objective* sampling framework for this area that *explicitly* preserves coverage and local data distribution, which are both crucial to index performance.

7 Conclusion

To address the challenge of simultaneously achieving global coverage and preserving local data distribution in large-scale, high-dimensional data sampling, this paper introduces a novel dual-objective optimization framework that balances global coverage and local fidelity. Based on this framework, we develop two efficient, near-linear-time samplers, LASS-Lite and LASS-NA. The latter frames the neighbor selection problem as a submodular optimization, providing a $(1 - 1/e)$ approximation guarantee for its objective of maximizing the reduction in Local Intrinsic Dimensionality (LID) estimation error. Experiments demonstrate that our methods perform exceptionally well on the proposed dual objectives. The resulting samples faithfully preserve local data distributions while maintaining broad coverage, leading to significant gains in downstream tasks such as data visualization, self-supervised prototyping, and approximate nearest neighbor search. We further validate scalability up to 100M vectors and show that improved local fidelity translates to better label-efficient model selection under strict labeling budgets.

Acknowledgments

This work was supported by Hong Kong RGC Projects Nos. 12201925, 12200424, 12202024 and C2003-23Y.

References

[1] Sameer Agarwal and et al. 2013. BlinkDB: Queries with Bounded Errors and Bounded Response Times on Very Large Data. In *ACM SIGMOD*.

[2] Laurent Amsaleg, Oussama Chelly, Teddy Furon, Stéphane Girard, Michael E Houle, Ken-ichi Kawarabayashi, and Michael Nett. 2015. Estimating local intrinsic dimensionality. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 29–38.

[3] Laurent Amsaleg and Hervé Jégou. [n.d.]. Datasets for Approximate Nearest Neighbor Search. http://corpus-texmex.irisa.fr/. Retrieved July 2025.

[4] David Arthur and Sergei Vassilvitskii. 2007. k-means++: The Advantages of Careful Seeding. In *Proceedings of SODA*.

[5] Rachel Behar and Sara Cohen. 2022. Representative query results by voting. In *ACM SIGMOD*. 1741–1754.

[6] Thierry Bertin-Mahieux, Daniel PW Ellis, Brian Whitman, and Paul Lamere. 2011. The million song dataset.. In *Ismir*. 10.

[7] Moses Charikar, Sudipto Guha, Éva Tardos, and David B Shmoys. 1999. A constant-factor approximation algorithm for the k-median problem. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*. 1–10.

[8] Lynn Chua, Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, and Chiyuan Zhang. 2024. Scalable DP-SGD: Shuffling vs. poisson subsampling. *Advances in Neural Information Processing Systems* 37 (2024), 70026–70047.

[9] William G. Cochran. 1977. *Sampling Techniques* (3rd ed.). Wiley.

[10] Sara Cohen and Helen Sternbach. 2025. Facility Location for Fair and Equitable Query Results. In *ICDE*. IEEE, 1153–1165.

[11] Matthijs Douze, Hervé Jégou, and Jeff Johnson. 2024. The Faiss Library. *CoRR* (2024).

[12] Hao Duan, Yitong Song, Bin Yao, and Anqi Liang. 2025. PGTuner: An Efficient Framework for Automatic and Transferable Configuration Tuning of Proximity Graphs. *ACM SIGMOD* 3, 4 (2025), 1–27.

[13] Pavlos S. Efraimidis and Paul G. Spirakis. 2006. Weighted Random Sampling with a Reservoir. *Inform. Process. Lett.* (2006).

[14] Joshua Engels, Benjamin Landrum, Shangdi Yu, Laxman Dhulipala, and Julian Shun. 2024. Approximate Nearest Neighbor Search with Window Filters. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024 (Proceedings of Machine Learning Research)*, Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (Eds.), Vol. 235. 12469–12490.

[15] Cong Fu, Changxu Wang, and Deng Cai. 2021. High dimensional similarity search with satellite system graph: Efficiency, scalability, and unindexed query compatibility. *IEEE TPAMI* 44, 8 (2021), 4139–4150.

[16] Cong Fu, Chao Xiang, Changxu Wang, and Deng Cai. 2019. Fast approximate nearest neighbor search with the navigating spreading-out graph. In *PVLDB*, Vol. 12. VLDB Endowment, 416–474.

[17] S Louis Hakimi. 1964. Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations research* 12, 3 (1964), 450–459.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[19] Joseph M. Hellerstein, Peter J. Haas, and Helen J. Wang. 1997. Online Aggregation. In *Proceedings of SIGMOD*.

[20] Michael E Houle. 2017. Local intrinsic dimensionality I: an extreme-value-theoretic foundation for similarity applications. In *International Conference on Similarity Search and Applications*. Springer, 64–79.

[21] Md Mouinul Islam, Soroush Vahidi, Baruch Schieber, and Senjuti Basu Roy. 2024. Promoting Fairness and Priority in k-Winners Selection Using IRV. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24)*. ACM, 1199–1210.

[22] Suhas Jayaram Subramanya, Fnu Devvrit, Harsha Vardhan Simhadri, Ravishankar Krishnawamy, and Rohan Kadekodi. 2019. Diskann: Fast accurate billion-point nearest neighbor search on a single node. *Advances in neural information processing Systems* 32 (2019).

[23] Herve Jegou, Matthijs Douze, and Cordelia Schmid. 2010. Product quantization for nearest neighbor search. *IEEE TPAMI* 33, 1 (2010), 117–128.

[24] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-Scale Similarity Search with GPUs. *IEEE Trans. Big Data* 7, 3 (2021), 535–547.

[25] Samuel Kaski, Janne Nikkilä, Merja Oja, Jarkko Venna, Petri Törönen, and Eero Castrén. 2003. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC bioinformatics* 4, 1 (2003), 48.

[26] Vishal Kaushal, Ganesh Ramakrishnan, and Rishabh Iyer. 2022. Submodlib: A Submodular Optimization Library. *CoRR* (2022).

[27] Krishnateja Killamsetty and et al. 2021. GLISTER: Generalization Based Data Subset Selection for Efficient and Robust Learning. In *AAAI*.

[28] Pang Wei Koh and Percy Liang. 2017. Understanding Black-box Predictions via Influence Functions. In *ICML*.

[29] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).
[30] Alfred A Kuehn and Michael J Hamburger. 1963. A heuristic program for locating warehouses. *Management science* 9, 4 (1963), 643–666.
[31] Alex Kulesza and Ben Taskar. 2012. *Determinantal Point Processes for Machine Learning*. Vol. 5. Emerald. 123–286 pages.
[32] John A Lee and Michel Verleysen. 2009. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing* 72, 7-9 (2009), 1431–1443.
[33] Shi Li. 2013. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Information and Computation* 222 (2013), 45–58.
[34] Wen Li, Ying Zhang, Yifang Sun, Wei Wang, Mingjie Li, Wenjie Zhang, and Xuemin Lin. 2019. Approximate nearest neighbor search on high dimensional data—experiments, analyses, and improvement. *TKDE* 32, 8 (2019), 1475–1488.
[35] Anqi Liang, Pengcheng Zhang, Bin Yao, Zhongpu Chen, Yitong Song, and Guangxu Cheng. 2025. UNIFY: Unified Index for Range Filtered Approximate Nearest Neighbors Search. *Proceedings of the VLDB Endowment* (2025).
[36] Yuyu Luo, Yihui Zhou, Nan Tang, Guoliang Li, Chengliang Chai, and Leixian Shen. 2023. Learned Data-Aware Image Representations of Line Charts for Similarity Search. *Proc. ACM Manag. Data* 1, 1, Article 88 (may 2023), 29 pages.
[37] J. B. MacQueen. 1967. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*. 281–297.
[38] Yu A Malkov and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE TPAMI* 42, 4 (2018), 824–836.
[39] Leland McInnes and John Healy. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *CoRR* (2018).
[40] Yuchen Meng, Rong-Hua Li, Longlong Lin, Xunkai Li, and Guoren Wang. 2024. Topology-Preserving Graph Coarsening: An Elementary Collapse-Based Approach. *Proc. VLDB Endow.* 17, 13 (Sept. 2024), 4760–4772.
[41] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. 2015. Lazier Than Lazy Greedy. In *AAAI*. 1812–1818.
[42] Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. 2020. Coresets for Data-Efficient Training of Deep Neural Networks. In *ICML*.
[43] Yun Peng, Byron Choi, Tsz Nam Chan, Jianye Yang, and Jianliang Xu. 2023. Efficient Approximate Nearest Neighbor Search in Multi-dimensional Databases. *ACM SIGMOD* 1, 1 (2023), 1–27.
[44] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. 1532–1543.
[45] Jacob Schreiber, Jeff Bilmes, and William Stafford Noble. 2020. apricot: Submodular Selection for Data Summarization in Python. *Journal of Machine Learning Research* 21, 48 (2020), 1–6.
[46] Ozan Sener and Silvio Savarese. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *International Conference on Learning Representations (ICLR)*.
[47] Burr Settles. 2009. *Active Learning Literature Survey*. Technical Report. UW–Madison.
[48] Yitong Song, Bin Yao, Zhida Chen, Xin Yang, Jiong Xie, Feifei Li, and Mengshi Chen. 2025. Efficient top-k spatial-range-constrained approximate nearest neighbor search on geo-tagged high-dimensional vectors. *The VLDB Journal* 34, 1 (2025), 14.
[49] Yitong Song, Pengcheng Zhang, Chao Gao, Bin Yao, Kai Wang, Zongyuan Wu, and Lin Qu. 2025. TRIM: Accelerating High-Dimensional Vector Similarity Search with Enhanced Triangle-Inequality-Based Pruning. *arXiv preprint arXiv:2508.17828* (2025).
[50] Jarkko Venna and Samuel Kaski. 2001. Neighborhood preservation in nonlinear projection methods: An experimental study. In *International conference on artificial neural networks*. Springer, 485–491.
[51] Jeffrey S. Vitter. 1985. Random Sampling with a Reservoir. *ACM TOMS* (1985).
[52] Jianguo Wang, Xiaomeng Yi, Rentong Guo, Hai Jin, Peng Xu, Shengjun Li, Xiangyu Wang, Xiangzhou Guo, Chengming Li, Xiaohai Xu, et al. 2021. Milvus: A purpose-built vector data management system. In *ACM SIGMOD*. 2614–2627.
[53] Mengzhao Wang, Weizhi Xu, Xiaomeng Yi, Songlin Wu, Zhangyang Peng, Xiangyu Ke, Yunjun Gao, Xiaoliang Xu, Rentong Guo, and Charles Xie. 2024. Starling: An i/o-efficient disk-resident graph index framework for high-dimensional vector similarity search on data segment. *ACM SIGMOD* 2, 1 (2024), 1–27.
[54] Mengzhao Wang, Xiaoliang Xu, Qiang Yue, and Yuxiang Wang. 2021. A Comprehensive Survey and Experimental Comparison of Graph-Based Approximate Nearest Neighbor Search. *Proc. VLDB Endow.* 14, 11 (2021), 1964–1978.
[55] Shu Wang, Yixiang Fang, Yingli Zhou, Xilin Liu, and Yuchi Ma. 2025. ArchRAG: Attributed Community-based Hierarchical Retrieval-Augmented Generation. *CoRR* (2025).
[56] Tongzhou Wang and et al. 2018. Dataset Distillation. In *NeurIPS*.
[57] Zhuang Wang, Zhen Jia, Shuai Zheng, Zhen Zhang, Xinwei Fu, TS Eugene Ng, and Yida Wang. 2023. Gemini: Fast failure recovery in distributed training with in-memory checkpoints. In *Proceedings of the 29th Symposium on Operating*

Systems Principles. 364–381.
[58] Chuangxian Wei, Bin Wu, Sheng Wang, Renjie Lou, Chaoqun Zhan, Feifei Li, and Yuanzhe Cai. 2020. Analyticdb-v: A hybrid analytical engine towards query fusion for structured and unstructured data. *PVLDB* 13, 12 (2020), 3152–3165.
[59] Kai Wei, Rishabh Iyer, and Jeff Bilmes. 2015. Submodular Subset Selection for Large-Scale Data. In *ICML*.
[60] Kai Wei, Rishabh Iyer, and Jeff Bilmes. 2015. Submodularity in Data Subset Selection and Active Learning. In *Proc. ICML*. 1954–1963.
[61] Tiannuo Yang, Wen Hu, Wangqi Peng, Yusen Li, Jianguo Li, Gang Wang, and Xiaoguang Liu. 2024. Vdtuner: Automated performance tuning for vector data management systems. In *ICDE*. IEEE, 4357–4369.
[62] Fangyuan Zhang, Zhengjun Huang, Yingli Zhou, Qintian Guo, Zhixun Li, Wensheng Luo, Di Jiang, Yixiang Fang, and Xiaofang Zhou. 2025. EraRAG: Efficient and Incremental Retrieval Augmented Generation for Growing Corpora. *CoRR* (2025).
[63] Yingli Zhou, Yaodong Su, Youran Sun, Shu Wang, Taotao Wang, Runyuan He, Yongwei Zhang, Sicong Liang, Xilin Liu, Yuchi Ma, and Yixiang Fang. 2025. In-depth Analysis of Graph-based RAG in a Unified Framework. *Proc. VLDB Endow.* 18, 13 (2025), 5623–5637.
[64] Xuliang Zhu, Xin Huang, Byron Choi, and Jianliang Xu. 2020. Top-k Graph Summarization on Hierarchical DAGs. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Virtual Event, Ireland) *(CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 1903–1912.

Received October 2025; revised January 2026; accepted February 2026