Geometric Median (GM) Matching for Robust k-Subset Selection from Noisy Data

To Appear @ ICML 2025

Anish Acharya, Sujay Sanghavi, Alex Dimakis, Inderjit S Dhillon

k Subset Selection

- **Given:** a dataset of n samples:

$$x_1, x_2, \dots, x_n \sim p$$

- **Goal:** Select a **representative subset** of size $k \ll n$

$$D_S \subseteq \mathrm{D} = \{x_1, x_2, \dots, x_n\}, \ |D_S| = k$$

Let p_S denote the **empirical** measure induced by D_S

$$p_S \coloneqq \frac{1}{k} \sum_{x_i \in D_S} \delta_{x_i}$$

Then, one aims to solve:

$$\underset{D_S \subseteq D,\, |D_S| = k}{\arg\min} \ \Lambda(p_S, p)$$

for some appropriate **Divergence Measure** $\Lambda(\cdot\,, \ \cdot)$.

- D_S should yield similar performance when used for training .

p $\qquad$ p_S

Random Sampling

- **Given,** a dataset of n samples:

$$\mathrm{D} = \{x_1, x_2, \dots, x_n\}$$

- **Select**, $D_S \subseteq \mathrm{D}$, $|D_S| = k$, **uniformly at random,** i.e.

$$Pr(D_S = S) = \frac{1}{\binom{n}{k}}, \qquad \forall S \subseteq D, |S| = k$$

Random Sampling

Without additional structure, all samples are **exchangeable.**

No meaningful notion of:

- **Distance**: needs a metric space
- **Diversity**: needs either a feature space or kernel
- **Importance**: needs a label, loss, or task

Random Sampling is **Minimax Optimal** i.e., minimizes the worst-case risk under symmetric (permutation invariant) functionals.

Making it a strong baseline and the de-facto approach at scale.

Importance Scoring

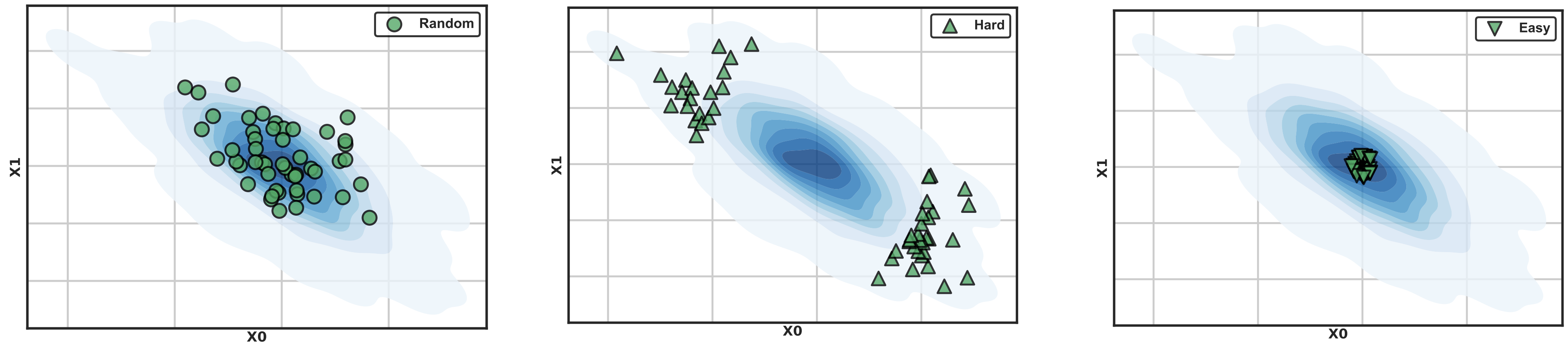
- Random sampling is minimax optimal under symmetric functionals.
- But, if **exchangeability is broken via structure**, we can expect to improve.
- **Assume access to an encoder** $\phi : R^d \rightarrow R^s$
- **Define a scoring function** quantifying **sample importance**.
  - ❖ **Geometric** approaches compute score based on $\phi(x)$**.**
  - ❖ **Task-aware** scoring uses prediction signals:

$$p(x) = \mathrm{softmax}\left(V^T \phi(x)\right) \in \Delta^C$$

    Loss, Entropy, Margin, Gradient Norm

- **Rank** samples from **hard to easy**, or from most **informative to** most **prototypical.**
- **Retain only a selected fraction,** those deemed most **representative, diverse**, or **informative** under the scoring criterion.

Importance Scoring



$$s_i = \mathrm{score}(x_i, D) = \left\| \phi(x_i) - \frac{1}{n}\sum_{x \in D} \phi(x) \right\|^2$$

🔷 **Low score ⇒ Easy sample:**
The sample lies close to the empirical **centroid in the embedding** space, likely most **prototypical** or **abundant**.

🔶 **High score ⇒ Hard sample:**
The sample lies far from the centroid — potentially **diverse, rare, or difficult**.

Noisy Sample Space - In the Wild

- In practice, we rarely have access to clean, perfectly representative data from the target distribution due to **imperfect semantic annotations**, **adversarial attacks**, or simply **measurement noise**.

- Instead, we **only have access to a noisy version** of the target distribution:

$$p'(\psi, x) = (1 - \psi)\, p(x) + \psi\, q(x)$$

$\boldsymbol{p}$ **:** clean distribution
$\boldsymbol{q}$ **:** adversarial distribution
$\boldsymbol{\psi} \in [\mathbf{0}, \mathbf{1/2})$ **: corruption rate**, denoting the fraction of corrupted samples

Noise Model : Gross Corruption

- **Given,** a dataset of n samples:

$$\{x_1, x_2, \dots, x_n\} \sim p$$

- **Adversary inspects** all the samples, and replace

$$0 \le \psi < 1/2$$

fraction of the samples with **arbitrary** points.

- The resulting noisy dataset

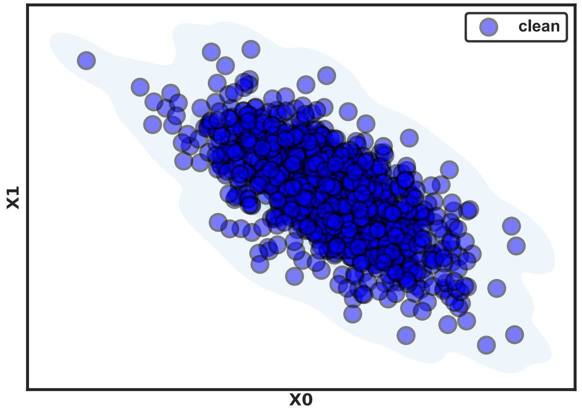
$$D = D_B \cup D_G$$

is referred as $\boldsymbol{\psi}$ **- grossly corrupted**.

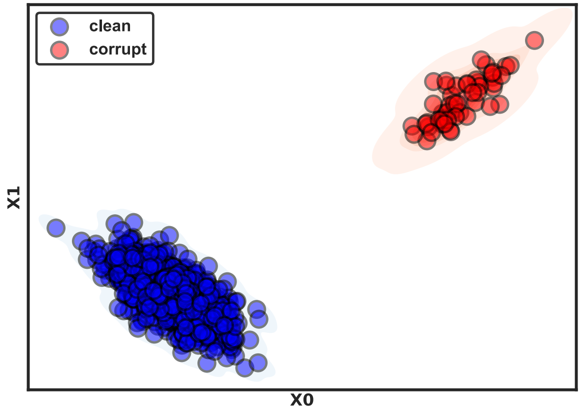
D_B, D_G denote the sets of corrupt and clean samples, respectively.

$$\frac{|D_B|}{|D_G|} = \frac{\psi}{\psi - 1} < 1$$

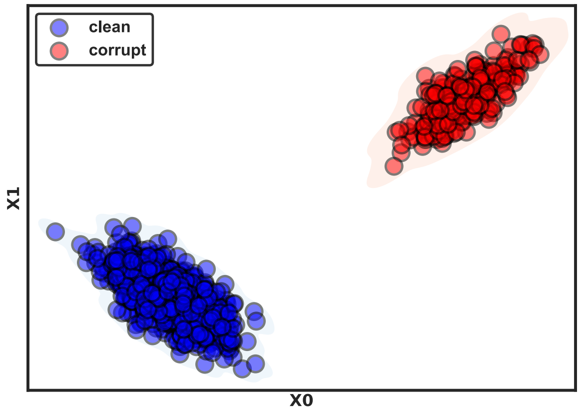
Noise Model



$\boldsymbol{p(x)}$



$\boldsymbol{p'(\psi = 0.05, x)}$



$\boldsymbol{p'(\psi = 0.2, x)}$

$$p'(\psi, x) = (1 - \psi)\, p(x) + \psi\, q(x)$$

$\boldsymbol{p}$ **:** clean distribution

$\boldsymbol{q}$ **:** adversary chosen **arbitrary** distribution

$\boldsymbol{\psi} \in [\mathbf{0}, \mathbf{1/2})$ **: corruption rate**, denoting the fraction of corrupted samples

Noise Model : Gross Corruption

- By allowing the corruption to be **arbitrary**, this noise model covers a wide variety (if not all) of corruption. e.g.,
  - ❑ **Feature Corruption** (e.g., sensor faults, occlusion)
  - ❑ **Label Noise**
  - ❑ **Adversarial Attacks**
- By further allowing the adversary to **inspect the samples**, it generalizes both –
  - ❑ **Huber Contamination** : oblivious, fixed corruption
  - ❑ **Byzantine Corruption**: worst-case, adaptive corruption.

Robust k Subset Selection

Given: a noisy dataset of n samples:

$$D = \{x_1, x_2, \dots, x_n\} = D_B \cup D_G$$

generated via ψ gross corruption, where the corruption rate

$$0 \leq \psi = \frac{|D_B|}{|D|} < \frac{1}{2}$$

and **no assumptions** on the distribution of corrupt samples D_B**.**

Goal: judiciously select a k subset

$$D_S \subseteq \mathrm{D},\ |D_S| = k$$

such that, the **empirical** measure induced by D_S is a close to the underlying clean distribution p, induced by D_G.

Robustness Measure

We can measure the robustness of subset selection algorithms via breakdown point analysis - a classic tool in robust optimization to assess the **resilience of an estimator**.

Breakdown Point :

The breakdown point ζ_T of an estimator $T(\cdot)$, is the smallest fraction ψ of corrupted samples that can cause it to diverge arbitrarily:

$$\zeta_T = \inf \left\{ 0 \leq \psi \leq 1 : \sup_{D_B} \| T(D_G \cup D_B) - T\,(D_G) \| = \infty \right\}$$

$T(\cdot)$ is said to achieve the **optimal breakdown point**

$$\boldsymbol{\zeta_T^* = \frac{1}{2}}$$

if it remains bounded $\forall 0 \leq \psi < \frac{1}{2}$.

Vulnerability of Importance Scoring

Given, a dataset of n samples:

$$\mathrm{D} = \{x_1, x_2, \dots, x_n\}$$

Consider a single grossly corrupt sample,

$$\tilde{x} = \left(n\mu_B - \sum_{x_i \in D \setminus \tilde{x}} \phi(x_i) \right)$$

This would result in estimating the centroid to any arbitrary target μ_B, chosen by the adversary causing the importance score to deviate arbitrarily :

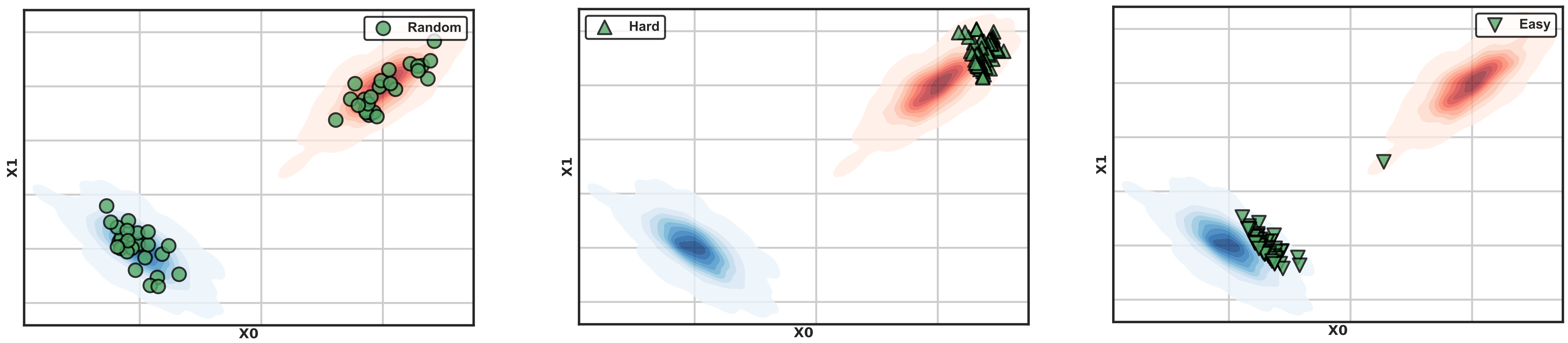
$$\Delta \mathrm{s_i} = \|\mu_B - \mu\|^2 - 2(\phi(x_i) - \mu)^T \|\mu_B - \mu\|$$

Thus, the asymptotic breakdown point is

$$\lim_{n \to \infty} \frac{1}{n} \to 0$$

Under gross corruption, **the notion of importance score is broken**.

Pitfalls of Importance Scoring in Noisy Setting



$$\mathbf{D} \sim \boldsymbol{p}'(\boldsymbol{\psi} = \mathbf{0.4}, \boldsymbol{x})$$

⚖️ Robustness vs Diversity

- **Noisy or corrupted samples** are often mistakenly scored as **hard** or **informative**
- In contrast, **easy samples** (far from decision boundary) are **more robust**, but typically, **prototypical** and **less diverse.**
- This leads to a **selection bias**:
  Discards rare but **clean and informative** examples.
- Introduces a **robustness vs. diversity trade-off:**
  Favoring robustness can **shrink coverage of the data manifold**, resulting in degraded generalization performance.

Is it possible to balance **robustness and diversity** in a single subset selection strategy?

Moment Matching

- Find a k subset such that Maximum Mean Discrepancy (MMD) between the empirical distribution induced by the subset and and the original dataset is minimized.

$$\underset{\substack{\mathcal{D}_{\mathcal{S}} \subseteq \mathcal{D} \\ |\mathcal{D}_{\mathcal{S}}| = k}}{\arg\min} \left[\Delta^2(\mathcal{D}_{\mathcal{S}}, \mathcal{D}) := \left\| \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x}_i \in \mathcal{D}} \phi(\mathbf{x}_i) - \frac{1}{k} \sum_{\mathbf{x}_j \in \mathcal{D}_{\mathcal{S}}} \phi(\mathbf{x}_j) \right\|^2 \right]$$

- This would ensure that the empirical distribution p_S induced by D_S is a close approximation of the original dataset.

- However, in the noisy setting , this no longer guarantees convergence to the true underlying (uncorrupted) moment. Instead, the subset selection can be hijacked by a single bad sample, warping the solution towards an adversarial target.

Robust Moment Matching

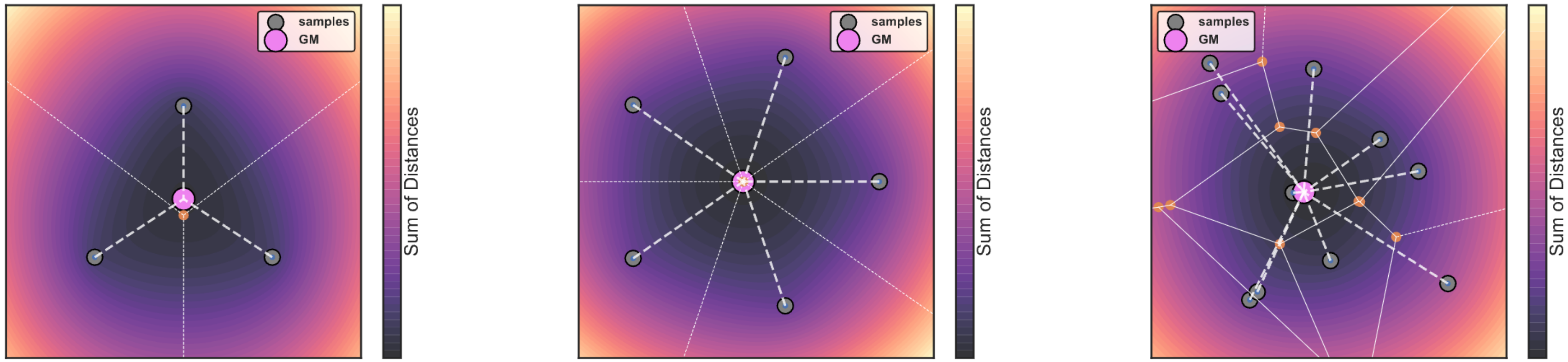
- **Given:** a noisy dataset of n samples:

$$D = \{x_1, x_2, \dots, x_n\} = D_B \cup D_G$$

generated via ψ gross corruption

- Our proposal is to solve a robust variant of the moment matching objective instead.
- The key idea is to **replace the empirical mean with a robust surrogate**, mitigating its susceptibility to corrupted samples.

Robust Mean Estimation



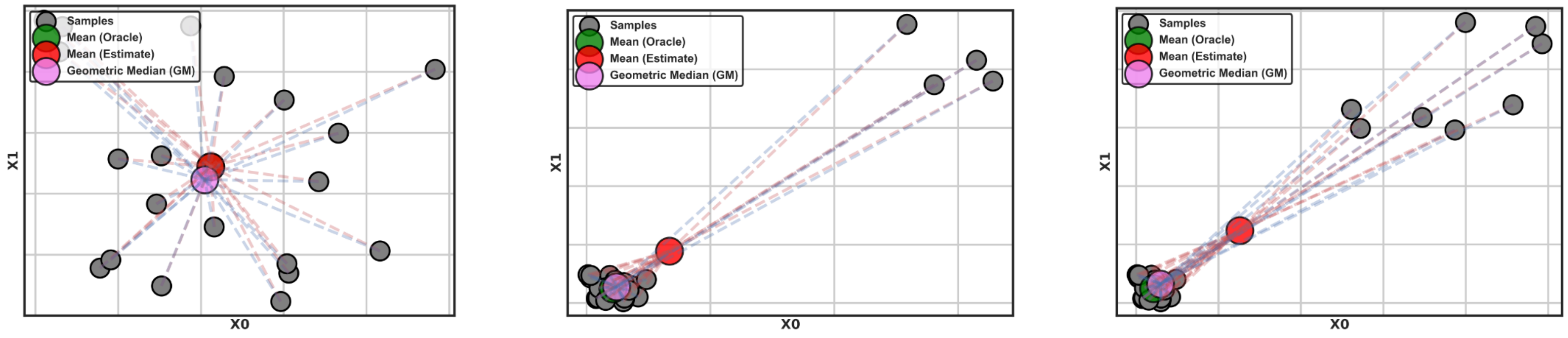
(a) **Triangle** (b) **Pentagon** (c) **Random**

Geometric Median.

Suppose, we are given a finite collection of observations $\{ \phi(x_1), \phi(x_2), \ldots, \phi(x_n) \}$ defined over Hilbert space $\mathcal{H} \in R^d$, equipped with norm $\|\cdot\|$ and inner product $\langle \cdot, \cdot \rangle$ operators. Then, the Geometric Median (Fermat-Weber point) is defined as:

$$\boldsymbol{\mu}^{\mathrm{GM}} = \underset{\mathbf{z} \in \mathcal{H}}{\arg\min} \left[\rho(\mathbf{z}) := \sum_{i=1}^{n} \left\| \mathbf{z} - \phi(\mathbf{x}_i) \right\| \right]$$

Robust Mean Estimation

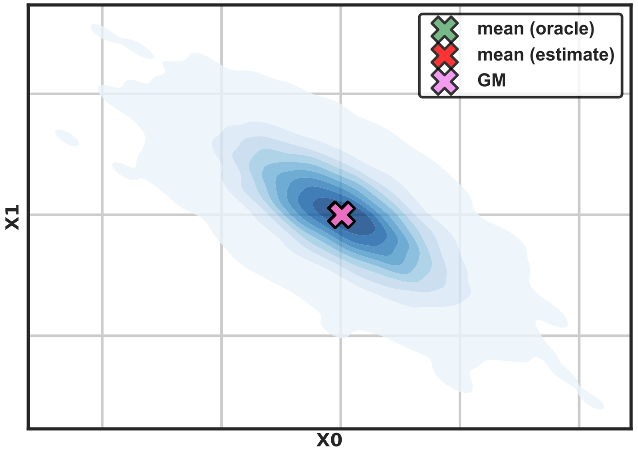


(a) **No Corruption** ($\psi = 0$) (b) **20% Corruption** ($\psi = 0.2$) (c) **40% Corruption** ($\psi = 0.4$)

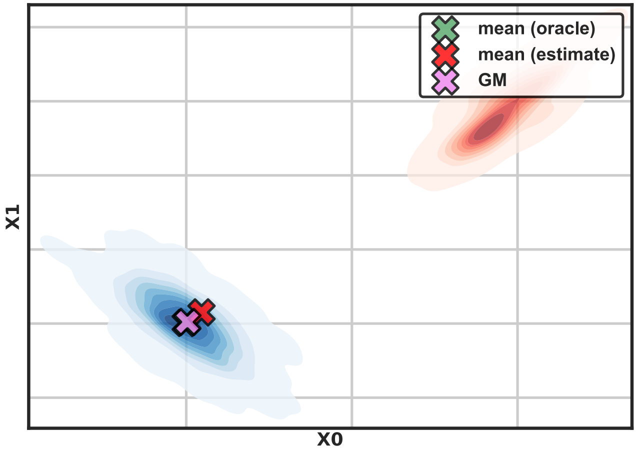
In contrast, the empirical mean is the minimizer of the squared Euclidean distances:

$$\hat{\boldsymbol{\mu}} = \arg\min_{\mathbf{z}\in\mathbb{R}^d} \rho(\mathbf{z}), \quad \text{where} \quad \rho(\mathbf{z}) = \sum_{i=1}^{n} \left\| \mathbf{x}_i - \mathbf{z} \right\|^2$$

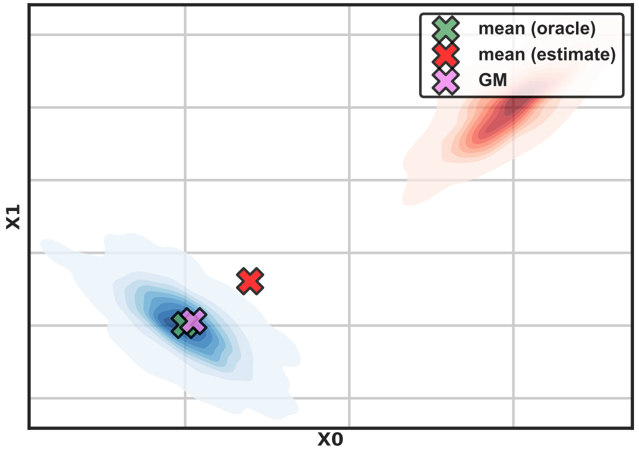
Robust Mean Estimation



No Corruption



5% Corruption



20% Corruption

- However, this also makes the empirical mean sensitive to outliers, as **extreme values have a disproportionately large effect on the sum of squared distances**.
- On the other hand, the linear penalty in the GM computation ensures that the objective is less influenced by outliers, as deviations are not amplified quadratically.

Approximate GM

- The GM optimization problem is inherently **non-smooth** due to the presence of the Euclidean norm $\|z - \phi(x_i)\|$, which leads to **non-differentiability at points where multiple distances are equal**, making gradient-based optimization difficult.

- Moreover, while a closed-form solution exists for $d = 1$, (Bajaj, 1988) showed that for dimensions $d \geq 2$, in general, the GM **does not admit a closed-form solution expressible in radicals**, rendering its exact computation **algebraically intractable**.

- However, since the problem is convex, iterative algorithms can be used to approximate the GM efficiently to arbitrary precision.

- **ϵ Approximate GM.**

$$\sum_{i=1}^{n} \left\| \boldsymbol{\mu}_{\epsilon}^{\mathrm{GM}} - \phi(\mathbf{x}_i) \right\| \leq (1+\epsilon) \sum_{i=1}^{n} \left\| \boldsymbol{\mu}^{\mathrm{GM}} - \phi(\mathbf{x}_i) \right\|$$

Bajaj et. al. The algebraic degree of geometric optimization problems, Discrete & Computational Geometry, 1988 .

Geometric Median Matching

Leveraging the breakdown and translation invariance properties of GM, we instead propose to solve for the following objective:

$$\underset{\substack{\mathcal{D}_{\mathcal{S}} \subseteq \mathcal{D} \\ |\mathcal{D}_{\mathcal{S}}| = k}}{\arg\min} \left(\Delta^2_{\mathrm{GM}}(\mathcal{D}_{\mathcal{S}}, \mathcal{D}) := \left\| \boldsymbol{\mu}^{\mathrm{GM}}_{\epsilon}(\mathcal{D}) - \frac{1}{k} \sum_{\mathbf{x}_i \in \mathcal{D}_{\mathcal{S}}} \phi(\mathbf{x}_i) \right\|^2 \right)$$

In essence, the idea is to find a k subset $D_S \subseteq D$, such that the empirical mean of the subset approximately matches the ϵ approximate GM $\mu_{\epsilon}^{\mathrm{GM}}(D)$ of the noisy dataset over a Reproducible Kernel Hilbert Space (RKHS).

Geometric Median Matching

- an instance of the famous subset sum problem – known to be **NP Hard** via a **reduction from k-set cover.**
- Remarkably, although the squared-distance function is not submodular in D_S, it can be transformed into a **submodular set cover instance**.
- This implies that even though the underlying problem is NP-hard, we can efficiently compute a subset D_S whose moment matching error is within a (1 + ε) multiplicative factor of the optimal error, while maintaining a polynomial runtime.

- Feige et. al., A threshold of ln n for approximating set cover, Journal of the ACM (JACM), 1998
- Mirzasoleiman et. al., Coresets for data-efficient training of machine learning models, ICML 2020
- Nemhauser et. al., An analysis of approximations for maximizing submodular set functions – I, Mathematical programming, 1978

Geometric Median Matching

- To solve the combinatorial GM Matching objective, we adopt a herding style greedy minimization procedure.

- Starting with a suitably chosen $\theta_0 \in \mathcal{H}$, we repeatedly perform the following updates, adding one sample at a time, k times:

$$\mathbf{x}_{t+1} := \arg\max_{\mathbf{x} \in \mathcal{D}} \langle \boldsymbol{\theta}_t, \phi(\mathbf{x}) \rangle$$

$$\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \left(\boldsymbol{\mu}_\epsilon^{\mathrm{GM}}(\mathcal{D}) - \phi(\mathbf{x}_{t+1}) \right)$$

- Note the resemblance to greedy matching pursuits and the Frank-Wolfe algorithm for convex optimization over the convex hull of $\{\phi(x) \mid x \in D\}$.

Geometric Median Matching

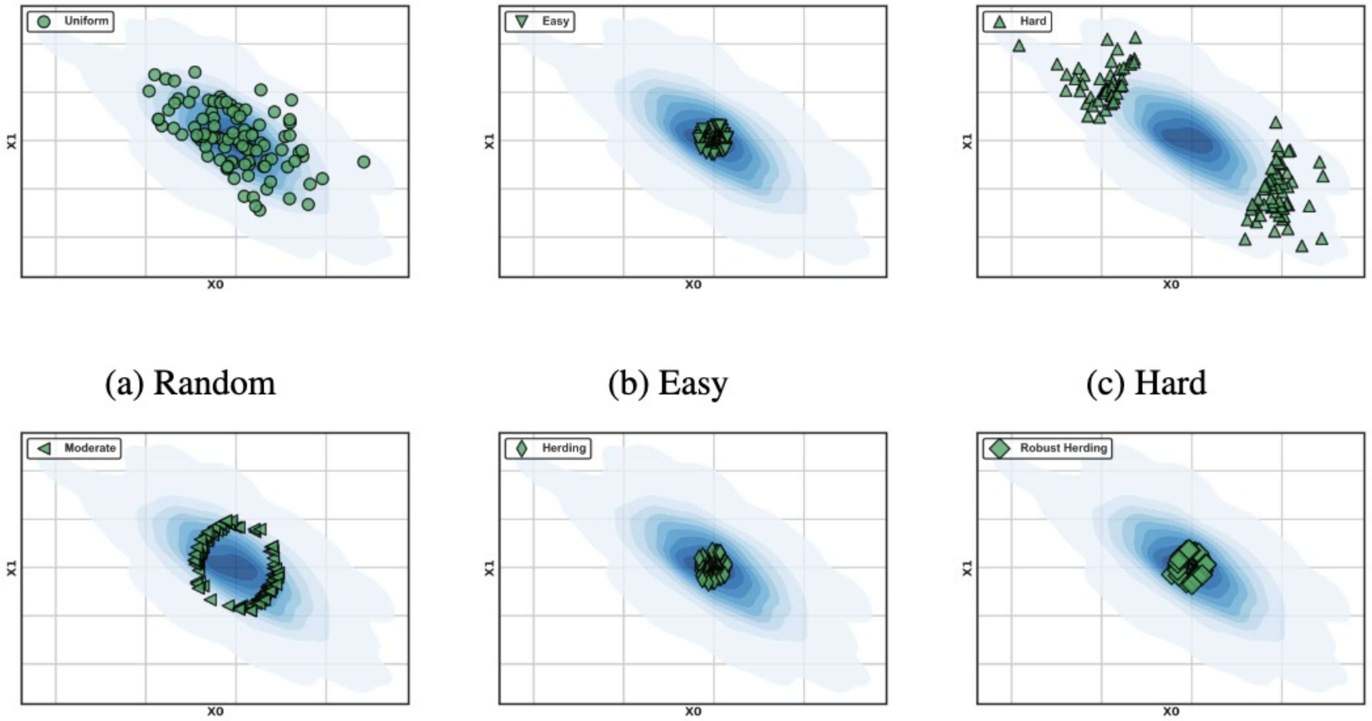
- Conceptually, θ_T represents the **vector pointing towards under sampled regions** of the target distribution induced by D at iteration T.

- Exploring underrepresented regions of the feature space, promotes diversity.

- by matching the GM rather than the empirical mean, the algorithm imposes larger penalties on outliers, which lie farther from the core distribution, **prioritizing samples near the convex hull of uncorrupted points**.

- Overall, GM Matching **promotes diversity in a balanced manner**, effectively exploring different regions of the distribution while avoiding distant, noisy points, thus mitigating the robustness vs. diversity trade-off.

Robust Data Pruning

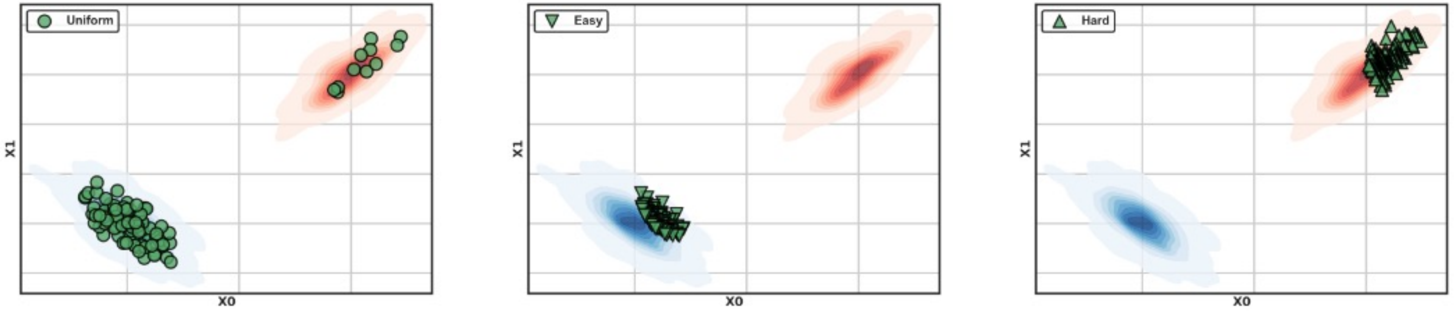


(a) Random (b) Easy (c) Hard

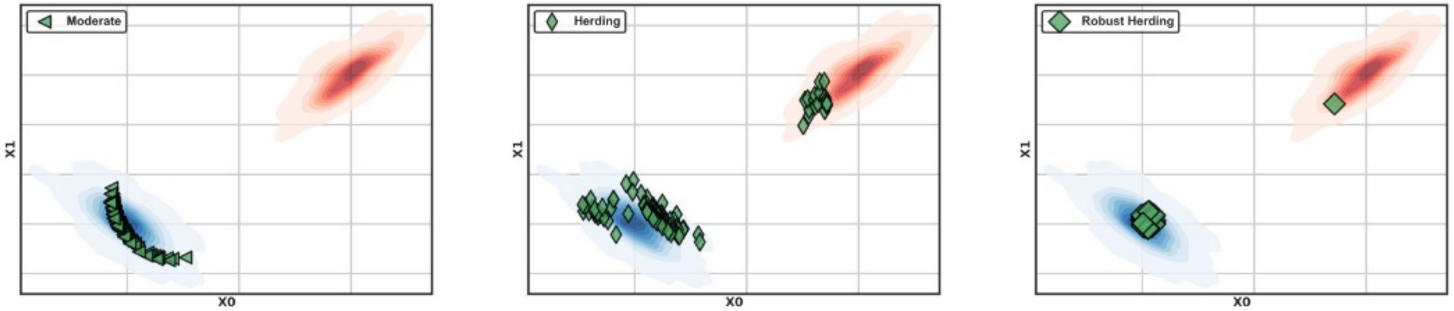
(d) Moderate (e) Herding (f) GM Matching

No Corruption

Robust Data Pruning



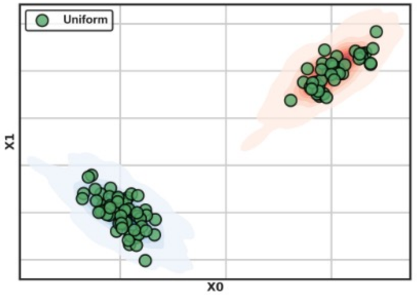
(a) Random (b) Easy (c) Hard



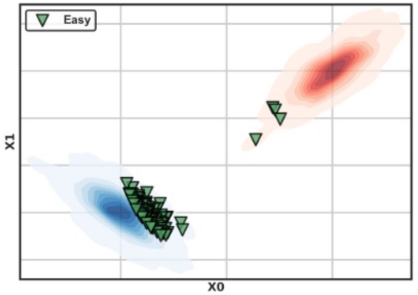
(d) Moderate (e) Kernel Herding (f) GM Matching

20% Corruption

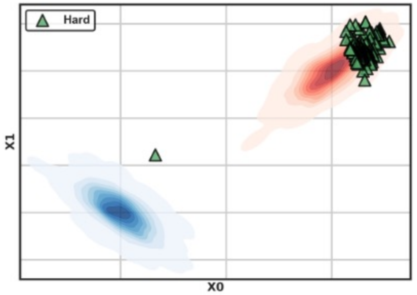
Robust Data Pruning



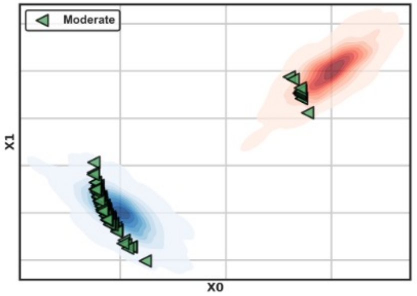
(a) **Random**



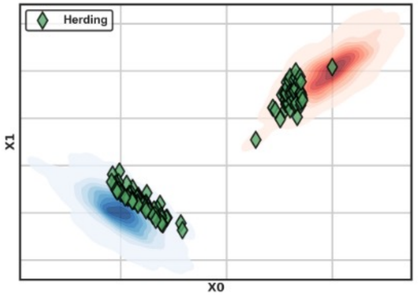
(b) **Easy**



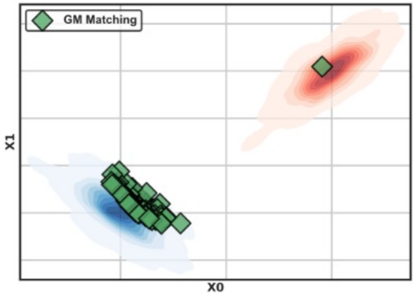
(c) **Hard**



(d) **Moderate**



(e) **Herding**



(f) GM **Matching**

40 % Corruption

Convergence Guarantee

Theorem.

Suppose that we are given a set of grossly corrupted samples $D = D_G \cup D_B$, ϵ-approx. GM oracle $\mu_\epsilon^{\mathrm{GM}}$, further assume that the characteristic feature map $\phi(\cdot)$ is bounded. Then GM Matching guarantees that the mean of the selected k subset converges to a δ neighborhood of the uncorrupted (true) mean $\mu(D_G)$ at **the rate $\mathcal{O}(\frac{1}{k})$ in RKHS** :

$$\delta^2 = \left\| \boldsymbol{\mu}_\epsilon^{\mathrm{GM}}(\mathcal{D}) - \boldsymbol{\mu}(\mathcal{D}_\mathcal{G}) \right\|^2 \leq \frac{8|\mathcal{D}_\mathcal{G}|^2}{(|\mathcal{D}_\mathcal{G}| - |\mathcal{D}_\mathcal{B}|)^2}\sigma^2(\mathcal{D}_\mathcal{G}) + \frac{2\epsilon^2}{(|\mathcal{D}_\mathcal{G}| - |\mathcal{D}_\mathcal{B}|)^2}$$

where we denoted $\sigma^2(D_G)$ denotes the variance of the uncorrupted samples.

Convergence Guarantee

Consequently, we can establish the following bound:

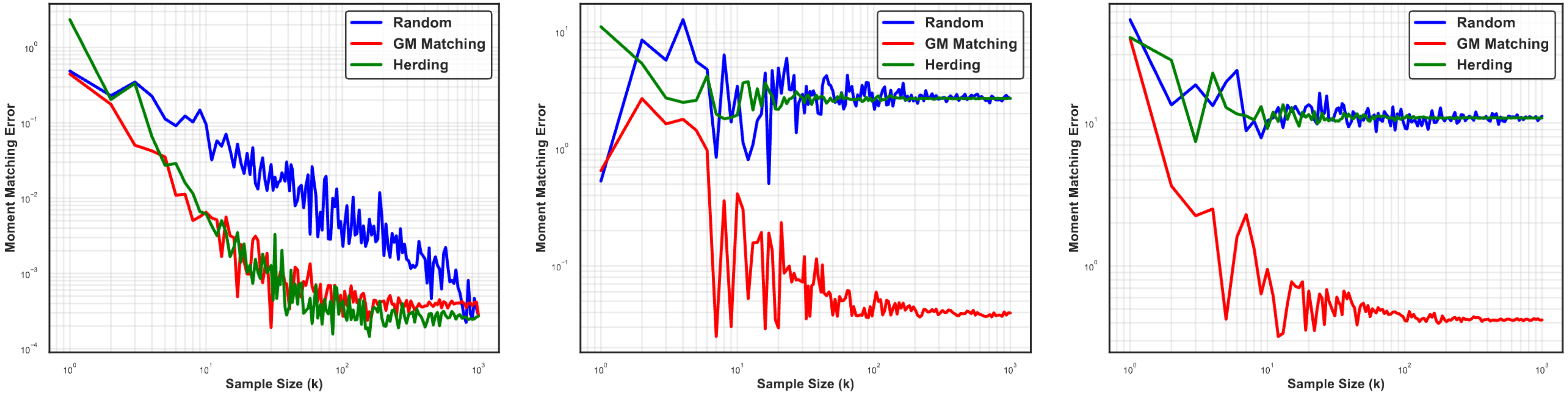
Lemma.

$$\Delta^2 = \left\| \boldsymbol{\mu}(\mathcal{D}_{\mathcal{S}}) - \boldsymbol{\mu}(\mathcal{D}_{\mathcal{G}}) \right\|^2 \leq \mathcal{O}\left(\frac{1}{k^2}\right) + \frac{16}{(1-\alpha)^2}\sigma_{\mathcal{G}}^2 + \frac{4\epsilon^2}{|\mathcal{D}_{\mathcal{G}}|^2(1-\alpha)^2}$$

- By matching the **uncorrupted mean**, D_S captures the uncorrupted distribution's first moment in the RKHS.
- since, $\phi(\cdot)$ is assumed to be a **characteristic feature map**, bounding $\|\mu(D_S) - \mu(D_G)\|$ immediately bounds **Maximum Mean Discrepancy**.

$$\Lambda^2_{\text{MMD}}\left(\hat{p}_{\mathcal{S}},\, p\right) \;=\; \left\| \mathbb{E}_{\hat{p}_{\mathcal{S}}}\left[\phi(\mathbf{x})\right] \;-\; \mathbb{E}_{p}\left[\phi(\mathbf{x})\right] \right\|^2_{\mathcal{H}}$$

Convergence Guarantee



(a) **No Corruption** (b) **20% Corruption** (c) **40% Corruption**

$$\Delta^2 = \|\mu(D_S) - \mu(D_G)\|^2$$ **as a function of subset size**

Experiments : No Corruption

Method / Ratio	20%	30%	40%	60%	80%	100%	Mean ↑
CIFAR-100							
Random	50.26±3.24	53.61±2.73	64.32±1.77	71.03±0.75	74.12±0.56	78.14±0.55	62.67
Herding	48.39±1.42	50.89±0.97	62.99±0.61	70.61±0.44	74.21±0.49	78.14±0.55	61.42
Forgetting	35.57±1.40	49.83±0.91	59.65±2.50	**73.34±0.39**	**77.50±0.53**	78.14±0.55	59.18
GraNd-score	42.65±1.39	53.14±1.28	60.52±0.79	69.70±0.68	74.67±0.79	78.14±0.55	60.14
EL2N-score	27.32±1.16	41.98±0.54	50.47±1.20	69.23±1.00	75.96±0.88	78.14±0.55	52.99
Optimization-based	42.16±3.30	53.19±2.14	58.93±0.98	68.93±0.70	75.62±0.33	78.14±0.55	59.77
Self-sup.-selection	44.45±2.51	54.63±2.10	62.91±1.20	70.70±0.82	75.29±0.45	78.14±0.55	61.60
Moderate-DS	51.83±0.52	57.79±1.61	64.92±0.93	71.87±0.91	75.44±0.40	78.14±0.55	64.37
GM Matching	**55.93± 0.48**	**63.08± 0.57**	**66.59± 1.18**	70.82± 0.59	74.63± 0.86	78.14± 0.55	**66.01**
Tiny ImageNet							
Random	24.02±0.41	29.79±0.27	34.41±0.46	40.96±0.47	45.74±0.61	49.36±0.25	34.98
Herding	24.09±0.45	29.39±0.53	34.13±0.37	40.86±0.61	45.45±0.33	49.36±0.25	34.78
Forgetting	22.37±0.71	28.67±0.54	33.64±0.32	41.14±0.43	**46.77±0.31**	49.36±0.25	34.52
GraNd-score	23.56±0.52	29.66±0.37	34.33±0.50	40.77±0.42	45.96±0.56	49.36±0.25	34.86
EL2N-score	19.74±0.26	26.58±0.40	31.93±0.28	39.12±0.46	45.32±0.27	49.36±0.25	32.54
Optimization-based	13.88±2.17	23.75±1.62	29.77±0.94	37.05±2.81	43.76±1.50	49.36±0.25	29.64
Self-sup.-selection	20.89±0.42	27.66±0.50	32.50±0.30	39.64±0.39	44.94±0.34	49.36±0.25	33.13
Moderate-DS	25.29±0.38	30.57±0.20	34.81±0.51	41.45±0.44	46.06±0.33	49.36±0.25	35.64
GM Matching	**27.88±0.19**	**33.15±0.26**	**36.92±0.40**	**42.48±0.12**	46.75±0.51	49.36±0.25	**37.44**

Proxy Teacher – In Domain, Shared Architecture.

ResNet-50 proxy teacher, pretrained on (clean) Tiny-ImageNet / CIFAR100, is used to find important samples from (clean) Tiny-ImageNet / CIFAR 100 , to train a ResNet-50 from scratch.

Experiments : No Corruption

ImageNet-1k						
Method / Ratio	**60%**	**70%**	**80%**	**90%**	**100%**	**Mean ↑**
Random	87.91 ± 0.37	88.63 ± 0.95	89.52 ± 0.73	89.57 ± 0.60	90.86 ± 0.71	89.30
Herding	88.25 ± 2.16	88.81 ± 1.06	89.60 ± 0.58	90.41 ± 0.33	90.86 ± 0.71	89.59
Forgetting	88.83 ± 0.92	89.81 ± 0.97	89.94 ± 0.26	90.41 ± 0.58	90.86 ± 0.71	89.97
GraNd-score	88.48 ± 1.73	89.82 ± 2.07	89.94 ± 0.81	90.41 ± 0.62	90.86 ± 0.71	89.90
EL2N-score	88.48 ± 2.81	89.82 ± 1.14	90.34 ± 0.87	90.57 ± 0.46	90.86 ± 0.71	90.01
Self-sup.-selection	87.59 ± 2.61	89.56 ± 1.97	**90.74 ± 0.27**	90.49 ± 0.98	90.86 ± 0.71	89.49
Moderate-DS	89.23 ± 0.96	89.94 ± 0.74	90.65 ± 0.51	90.75 ± 0.35	90.86 ± 0.71	90.29
GM Matching	**90.28 ± 0.38**	**90.54 ± 0.19**	90.72 ± 0.26	**90.84 ± 0.32**	90.86 ± 0.71	**90.65**

Proxy Teacher – In Domain, Shared Architecture.
ResNet-50 proxy teacher, pretrained on (clean) ImageNet-1k is used to find important samples from (clean) Tiny-ImageNet / CIFAR 100 , used to train a ResNet-50 from scratch.

Experiments : Feature Corruption

Proxy Teacher –
In Domain, Shared Architecture.

ResNet-50 proxy teacher, pretrained on (clean) Tiny-ImageNet, is used to find important samples from (noisy) Tiny-ImageNet.

The chosen subset is used to train a ResNet-50 from scratch.

Tiny ImageNet							
Method / Ratio	**20%**	**30%**	**40%**	**60%**	**80%**	**100%**	**Mean ↑**
5% Feature Corruption							
Random	23.51±0.22	28.82±0.72	32.61±0.68	39.77±0.35	44.37±0.34	49.02±0.35	33.82
Herding	23.09±0.53	28.67±0.37	33.09±0.32	39.71±0.31	45.04±0.15	49.02±0.35	33.92
Forgetting	21.36±0.28	27.72±0.43	33.45±0.21	40.92±0.45	45.99±0.51	49.02±0.35	33.89
GraNd-score	22.47±0.23	28.85±0.83	33.81±0.24	40.40±0.15	44.86±0.49	49.02±0.35	34.08
EL2N-score	18.98±0.72	25.96±0.28	31.07±0.63	38.65±0.36	44.21±0.68	49.02±0.35	31.77
Optimization-based	13.65±1.26	24.02±1.35	29.65±1.86	36.55±1.84	43.64±0.71	49.02±0.35	29.50
Self-sup.-selection	19.35±0.57	26.11±0.31	31.90±0.37	38.91±0.29	44.43±0.42	49.02±0.35	32.14
Moderate-DS	24.63±0.78	30.27±0.16	34.84±0.24	40.86±0.42	45.60±0.31	49.02±0.35	35.24
GM Matching	**27.46±1.22**	**33.14±0.61**	**35.76±1.14**	**41.62±0.71**	**46.83±0.56**	49.02±0.35	**36.96**
10% Feature Corruption							
Random	22.67±0.27	28.67±0.52	31.88±0.30	38.63±0.36	43.46±0.20	48.40±0.32	33.06
Herding	22.01±0.18	27.82±0.11	31.82±0.26	39.37±0.18	44.18±0.27	48.40±0.32	33.04
Forgetting	20.06±0.48	27.17±0.36	32.31±0.22	40.19±0.29	45.51±0.48	48.40±0.32	33.05
GraNd-score	21.52±0.48	26.98±0.43	32.70±0.19	40.03±0.26	44.87±0.35	48.40±0.32	33.22
EL2N-score	18.59±0.13	25.23±0.18	30.37±0.22	38.44±0.32	44.32±1.07	48.40±0.32	31.39
Optimization-based	14.05±1.74	29.18±1.77	29.12±0.61	36.28±1.88	43.52±0.31	48.40±0.32	29.03
Self-sup.-selection	19.47±0.26	26.51±0.55	31.78±0.14	38.87±0.54	44.69±0.29	48.40±0.32	32.26
Moderate-DS	23.79±0.16	29.56±0.16	34.60±0.12	40.36±0.27	45.10±0.23	48.40±0.32	34.68
GM Matching	**27.41±0.23**	**32.84±0.98**	**36.27±0.68**	**41.85±0.29**	**46.35±0.44**	48.40±0.32	**36.94**
20% Feature Corruption							
Random	19.99±0.42	25.93±0.53	30.83±0.44	37.98±0.31	42.96±0.62	46.68±0.43	31.54
Herding	19.46±0.14	24.47±0.33	29.72±0.39	37.50±0.59	42.28±0.30	46.68±0.43	30.86
Forgetting	18.47±0.46	25.53±0.23	31.17±0.24	39.35±0.44	44.55±0.67	46.68±0.43	31.81
GraNd-score	20.07±0.49	26.68±0.40	31.25±0.40	38.21±0.49	42.84±0.72	46.68±0.43	30.53
EL2N-score	18.57±0.30	24.42±0.44	30.04±0.15	37.62±0.44	42.43±0.61	46.68±0.43	30.53
Optimization-based	13.71±0.26	23.33±1.84	29.15±2.84	36.12±1.86	42.94±0.52	46.88±0.43	29.06
Self-sup.-selection	20.22±0.23	26.90±0.50	31.93±0.49	39.74±0.52	44.27±0.10	46.68±0.43	32.61
Moderate-DS	23.27±0.33	29.06±0.36	33.48±0.11	40.07±0.36	44.73±0.39	46.68±0.43	34.12
GM Matching	**27.19±0.92**	**31.70±0.78**	**35.14±0.19**	**42.04±0.31**	**45.12±0.28**	46.68±0.43	**36.24**

Experiments : Label Noise

Proxy Teacher –
In Domain, Shared Architecture.

ResNet-50 proxy teacher, pretrained on (clean) Tiny-ImageNet / CIFAR100, is used to find important samples from (noisy) Tiny-ImageNet / CIFAR 100.

The chosen subset is used to train a ResNet-50 from scratch.

	CIFAR-100 (Label noise)		Tiny ImageNet (Label noise)		
Method / Ratio	**20%**	**30%**	**20%**	**30%**	**Mean ↑**
		20% Label Noise			
Random	34.47±0.64	43.26±1.21	17.78±0.44	23.88±0.42	29.85
Herding	42.29±1.75	50.52±3.38	18.98±0.44	24.23±0.29	34.01
Forgetting	36.53±1.11	45.78±1.04	13.20±0.38	21.79±0.43	29.33
GraNd-score	31.72±0.67	42.80±0.30	18.28±0.32	23.72±0.18	28.05
EL2N-score	29.82±1.19	33.62±2.35	13.93±0.69	18.57±0.31	23.99
Optimization-based	32.79±0.62	41.80±1.14	14.77±0.95	22.52±0.77	27.57
Self-sup.-selection	31.08±0.78	41.87±0.63	15.10±0.73	21.01±0.36	27.27
Moderate-DS	40.25±0.12	48.53±1.60	19.64±0.40	24.96±0.30	31.33
GM Matching	**52.64±0.72**	**61.01±0.47**	**25.80±0.37**	**31.71±0.24**	**42.79**
		35% Label Noise			
Random	24.51±1.34	32.26±0.81	14.64±0.29	19.41±0.45	22.71
Herding	29.42±1.54	37.50±2.12	15.14±0.45	20.19±0.45	25.56
Forgetting	29.48±1.98	38.01±2.21	11.25±0.90	17.07±0.66	23.14
GraNd-score	23.03±1.05	34.83±2.01	13.68±0.46	19.51±0.45	22.76
EL2N-score	21.95±1.08	31.63±2.84	10.11±0.25	13.69±0.32	19.39
Optimization-based	26.77±0.15	35.63±0.92	12.37±0.68	18.52±0.90	23.32
Self-sup.-selection	23.12±1.47	34.85±0.68	11.23±0.32	17.76±0.69	22.64
Moderate-DS	28.45±0.53	36.55±1.26	15.27±0.31	20.33±0.28	25.15
GM Matching	**43.33± 1.02**	**58.41± 0.68**	**23.14± 0.92**	**27.76± 0.40**	**38.16**

Experiments : Label Noise

Tiny ImageNet (Label Noise)							
Method / Ratio	**20%**	**30%**	**40%**	**60%**	**80%**	**100%**	**Mean ↑**
Random	17.78±0.44	23.88±0.42	27.97±0.39	34.88±0.51	38.47±0.40	44.42±0.47	28.60
Herding	18.98±0.44	24.23±0.29	27.28±0.31	34.36±0.29	39.00±0.49	44.42±0.47	28.87
Forgetting	13.20±0.38	21.79±0.43	27.89±0.22	36.03±0.24	40.60±0.31	44.42±0.47	27.50
GraNd-score	18.28±0.32	23.72±0.18	27.34±0.33	34.91±0.19	39.45±0.45	44.42±0.47	28.34
EL2N-score	13.93±0.69	18.57±0.31	24.56±0.34	32.14±0.49	37.64±0.41	44.42±0.47	25.37
Optimization-based	14.77±0.95	22.52±0.77	25.62±0.90	34.18±0.79	38.49±0.69	44.42±0.47	27.12
Self-sup.-selection	15.10±0.73	21.01±0.36	26.62±0.22	33.93±0.36	39.22±0.12	44.42±0.47	27.18
Moderate-DS	19.64±0.40	24.96±0.30	29.56±0.21	35.79±0.36	39.93±0.23	44.42±0.47	30.18
GM Matching	**25.80±0.37**	**31.71±0.24**	**34.87±0.21**	**39.76±0.71**	**41.94±0.23**	44.42±0.47	**34.82**

Proxy Teacher –
In Domain, Shared Architecture.
ResNet-50 proxy teacher, pretrained on (clean) Tiny-ImageNet, is used to find important samples from (noisy) Tiny-ImageNet. The chosen subset is used to train a ResNet-50 from scratch.

Experiments : Adversarial Attack

Proxy Teacher –
In Domain, Shared Architecture.

ResNet-50 proxy teacher, pretrained on (clean) Tiny-ImageNet / CIFAR100, is used to find important samples from (noisy) Tiny-ImageNet / CIFAR 100.

The chosen subset is used to train a ResNet-50 from scratch.

	CIFAR-100 (PGD Attack)		CIFAR-100 (GS Attack)		
Method / Ratio	20%	30%	20%	30%	**Mean ↑**
Random	43.23±0.31	52.86±0.34	44.23±0.41	53.44±0.44	48.44
Herding	40.21±0.72	49.62±0.65	39.92±1.03	50.14±0.15	44.97
Forgetting	35.90±1.30	47.37±0.99	37.55±0.53	46.88±1.91	41.93
GraNd-score	40.87±0.84	50.13±0.30	40.77±1.11	49.88±0.83	45.41
EL2N-score	26.61±0.58	34.50±1.02	26.72±0.66	35.55±1.30	30.85
Optimization-based	38.29±1.77	46.25±1.82	41.36±0.92	49.10±0.81	43.75
Self-sup.-selection	40.53±1.15	49.95±0.50	40.74±1.66	51.23±0.25	45.61
Moderate-DS	43.60±0.97	51.66±0.39	44.69±0.68	53.71±0.37	48.42
GM Matching	**45.41 ±0.86**	**51.80 ±1.01**	**49.78 ±0.27**	**55.50 ±0.31**	**50.62**
	Tiny ImageNet (PGD Attack)		**Tiny ImageNet (GS Attack)**		
Method / Ratio	20%	30%	20%	30%	**Mean ↑**
Random	20.93±0.30	26.60±0.98	22.43±0.31	26.89±0.31	24.21
Herding	21.61±0.36	25.95±0.19	23.04±0.28	27.39±0.14	24.50
Forgetting	20.38±0.47	26.12±0.19	22.06±0.31	27.21±0.21	23.94
GraNd-score	20.76±0.21	26.34±0.32	22.56±0.30	27.52±0.40	24.30
EL2N-score	16.67±0.62	22.36±0.42	19.93±0.57	24.65±0.32	20.93
Optimization-based	19.26±0.77	24.55±0.92	21.26±0.24	25.88±0.37	22.74
Self-sup.-selection	19.23±0.46	23.92±0.51	19.70±0.20	24.73±0.39	21.90
Moderate-DS	21.81±0.37	27.11±0.20	23.20±0.13	28.89±0.27	25.25
GM Matching	**25.98 ±1.12**	**30.77 ±0.25**	**29.71 ±0.45**	**32.88 ±0.73**	**29.84**

Experiments : Vision Transformers

CIFAR-100 (ViT-S)					
Method	**No Corruption**	**Noisy Feature**	**Label Noise**	**Adv. Attack**	**Mean ↑**
Random	33.80±0.54	31.29±0.61	26.67±0.54	31.01±0.45	30.19
Herding	32.16±0.37	31.75±0.22	32.27±0.53	31.28±0.66	31.37
Forgetting	33.52±0.73	24.45±0.29	26.24±1.07	28.26±1.95	28.12
GraNd-score	22.49±0.47	18.40±0.11	22.13±0.90	19.27±1.27	20.07
EL2N-score	26.15±0.21	23.27±0.68	24.80±0.72	20.26±1.68	23.12
Optimization-based	31.84±0.63	30.12±0.73	30.12±0.70	29.36±0.75	30.36
Self-sup.-selection	33.35±0.31	30.72±0.90	29.16±0.27	28.49±0.56	30.93
Moderate-DS	34.43±0.32	32.73±0.35	31.86±0.49	32.61±0.40	32.91
GM Matching	**40.81±0.87**	**38.26±0.68**	**42.11±0.36**	**39.45±0.82**	**40.66**

Proxy Teacher – In Domain, Shared Architecture.

ViT-S proxy teacher, pretrained on CIFAR100, is used to find important samples from CIFAR 100. The chosen subset is used to train a ResNet-50 from scratch.

Experiments : Generalization to Unseen Network

Proxy Teacher –
In Domain, Different Architecture.

ResNet-50 proxy teacher, pretrained on (clean) Tiny-ImageNet, is used to find important samples from (clean) Tiny-ImageNet.

The chosen subset is used to train a VGGNet-16 and ShuffleNet from scratch.

	ResNet-50→SENet		**ResNet-50→EfficientNet-B0**		
Method / Ratio	**20%**	**30%**	**20%**	**30%**	**Mean ↑**
Random	34.13±0.71	39.57±0.53	32.88±1.52	39.11±0.94	36.42
Herding	34.86±0.55	38.60±0.68	32.21±1.54	37.53±0.22	35.80
Forgetting	33.40±0.64	39.79±0.78	31.12±0.21	38.38±0.65	35.67
GraNd-score	35.12±0.54	41.14±0.42	33.20±0.67	40.02±0.35	37.37
EL2N-score	31.08±1.11	38.26±0.45	31.34±0.49	36.88±0.32	34.39
Optimization-based	33.18±0.52	39.42±0.77	32.16±0.90	38.52±0.50	35.82
Self-sup.-selection	31.74±0.71	38.45±0.39	30.99±1.03	37.96±0.77	34.79
Moderate-DS	36.04±0.15	41.40±0.20	34.26±0.48	39.57±0.29	37.82
GM Matching	**37.93±0.23**	**42.59±0.29**	**36.31±0.67**	**41.03±0.41**	**39.47**
	ResNet-50→ VGG-16		**ResNet-50→ ShuffleNet**		
Method / Ratio	20%	30%	20%	30%	**Mean ↑**
Random	29.63±0.43	35.38±0.83	32.40±1.06	39.13±0.81	34.96
Herding	31.05±0.22	36.27±0.57	33.10±0.39	38.65±0.22	35.06
Forgetting	27.53±0.36	35.61±0.39	27.82±0.56	36.26±0.51	32.35
GraNd-score	29.93±0.95	35.61±0.39	29.56±0.46	37.40±0.38	33.34
EL2N-score	26.47±0.31	33.19±0.51	28.18±0.27	35.81±0.29	31.13
Optimization-based	25.92±0.64	34.82±1.29	31.37±1.14	38.22±0.78	32.55
Self-sup.-selection	25.16±1.10	33.30±0.94	29.47±0.56	36.68±0.36	31.45
Moderate-DS	31.45±0.32	37.89±0.36	33.32±0.41	39.68±0.34	35.62
GM Matching	**35.86±0.41**	**40.56±0.22**	**35.51±0.32**	**40.30±0.58**	**38.47**

Conclusion

- We introduced GM Matching, a robust data pruning algorithm that selects a k-subset such that the subset mean approximates the geometric median of a noisy dataset over a Reproducible Kernel Hibert Space.

- Unlike prior data pruning approaches that degrade under corruption, GM Matching is resilient to a wide array of corruption.

- **Limitations / Future Work:**
  - performance depends on accurate geometric median estimation, which can be computationally challenging or unstable in degenerate or high-dimensional settings.

  - Moreover, its effectiveness is influenced by the choice of embedding space and may deteriorate when encoders are biased or poorly calibrated.