GENERATION-RESISTANT DATA PUBLISHING WITH FORCE FIELD AND FINSLER GEOMETRY

Anonymous authors
Paper under double-blind review

ABSTRACT

In the era of generative AI, modern data publishing faces a new challenge: datasets released to preserve utility for downstream tasks can also be used to train or fine-tune generative AI models to learn the original data distribution. This paper proposes a unified framework for generation-resistant data publishing (GRDP) that simultaneously preserves classification utility while degrading generative capability, by leveraging force fields, Finsler geometry, and generative statistics. To preserve classification utility, we characterize boundary-sensitive local discriminative geometry in the embedding space through a physics-inspired bubble expansion process and construct a distance-decaying force field with intra-class attraction and inter-class repulsion. Under the induced Finsler geometry, anisotropic bubble shells evolve according to a Hamilton–Jacobi formulation, and force invariance is enforced along shell trajectories so that local decision-boundary geometry remains stable even when global distributional statistics are distorted. To degrade generative capability while preserving local geometry, we derive a Gaussian target prior whose covariance shift increases the FID-based feature-distribution mismatch under a quadratic approximation of the force-invariance budget. We further demonstrate that this covariance shift has a closed-form rank-one structure aligned with the minimum-eigenvalue direction of the induced force matrix, i.e., the least force-constrained direction. The published data are then matched to this prior through covariance shaping, distorting global distributional statistics while maintaining classification performance.

1 INTRODUCTION

Data publishing is the act of releasing a dataset in published form that is intended to be used by others Kratz & Strasser (2014); Poline (2018). A critical challenge is to protect sensitive information while preserving data utility for downstream tasks Fung et al. (2010); Li & Li (2009); Fung et al. (2007a). Prior work has addressed this challenge through differential privacy Dwork & Roth (2014b), anonymization-based data release Sweeney (2002b); Machanavajjhala et al. (2006), and synthetic or private synthetic data generation Park et al. (2018); Yoon et al. (2019); Li et al. (2024). However, the increasing popularity of Generative AI has introduced a new challenge in data sharing and publishing. A dataset released for benign purposes, such as training classifiers, can also be used to train or fine-tune generative AI models to learn the original data distribution. Therefore, an ideal data publishing solution should satisfy two conflicting goals: (1) preserving data utility, e.g., a classifier trained on the released data has similar performance to one trained on the original data; and (2) degrading generative capability, i.e., a generative model trained on the released data should no longer recover the original data distribution.

Existing techniques typically focus on only one of these objectives: (1) Dataset distillation and condensation methods aim to replace the original dataset with a small synthetic one that preserves classification performance by maintaining feature distributions Cazenavette et al. (2022); Wang et al. (2022); Deng & Russakovsky (2022); Zhou et al. (2022); Zhao & Bilen (2022); Nguyen et al. (2022); Zhao et al. (2023); Zhang et al. (2023); Chen et al. (2025); Zhao et al. (2025). However, these features can be utilized by generative models to reconstruct the original data distribution; (2) Traditional data publishing techniques protect sensitive information while maintaining downstream utility Dwork & Roth (2014b); Sweeney (2002b); Machanavajjhala et al. (2006); Park et al. (2018); Yoon et al. (2019); Li et al. (2024). However, the retained useful information can still be exploited

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

by generative models to learn the original data distribution; and (3) Data poisoning or perturbation-based approaches aim to degrade generative models by introducing perturbations or adversarial signals Shan et al. (2024); Pan et al. (2024); Wang et al. (2024); Jang et al. (2025); Chen et al. (2024); Salman et al. (2023); Fu et al. (2022); Huang et al. (2021); Liang & Wu (2023); Liang et al. (2023); Le et al. (2023); Shan et al. (2025). However, these methods often distort the underlying feature geometry, leading to degraded classification accuracy. As a result, current techniques fail to provide a unified solution that simultaneously preserves data utility and degrades generative capability.

We have made two insightful observations: (1) Classification mainly depends on **local discriminative geometry**, especially near decision boundaries (e.g., intra-class coherence and inter-class separation); and (2) Generative models rely heavily on **global distributional statistics**, such as distributional moments (e.g., mean and covariance captured by Fréchet Inception Distance (FID)). This suggests a possible solution: preserve local geometric structure while distorting global distributional statistics.

To our best knowledge, this work is the first generation-resistant data publishing (GRDP) framework that simultaneously preserves classification utility while degrading generative capability, by leveraging force field, Finsler geometry, and generative statistics theory.

In order to **maintain local discriminative geometry for preserving classification utility**, samples from the same class should remain within compact neighborhoods, samples from different classes should preserve clear separation, and the local geometric structure near decision boundaries should remain stable under the publishing conversion. We introduce a bubble expansion process in physics to characterize this boundary-sensitive local discriminative geometry in the embedding space. Specifically, we model each embedded sample as a bubble center, from which a shell evolves outward. The shell evolution is anisotropic and direction-dependent, governed by the resultant force induced by same-class attraction and different-class repulsion. To capture this phenomenon, we construct a distance-decaying force field through a closed-form potential, enabling the resultant force to be computed in the embedding space. Since shell propagation follows the local force direction under anisotropic dynamics, we formulate the shell evolution as a Hamilton-Jacobi equation under a Finsler geometry induced by the force field. By tracking shell trajectories, we obtain boundary-sensitive monitor points and constrain the publishing process by enforcing force invariance at these points. Consequently, although the publishing conversion may change the global data distribution, the local attraction raised by intra-class samples and the repulsion induced by inter-class samples remain approximately invariant within boundary-sensitive regions. As a result, the decision boundary is largely preserved, thereby maintaining downstream classification performance.

FID is widely used to evaluate generative quality by comparing the empirical means and covariances of real and generated samples in a fixed feature space, where a higher FID indicates a larger distributional discrepancy and lower generation quality Heusel et al. (2018); Kynkäänniemi et al. (2019); Dhariwal & Nichol (2021); Jayasumana et al. (2024). In order to **distort global distributional statistics for degrading generative capability**, under the constraints of local geometry preservation and force invariance enforcement derived above, we leverage the fact that the FID is determined by the empirical mean and covariance of feature distributions. Under the limited force budget, we theoretically derive a worst-case Gaussian target prior within the covariance-level relaxation, which maximizes the FID-based mismatch between the published and original distributions for degrading generation quality. We then match the empirical mean and covariance of the published centers to this prior, increasing distributional discrepancy while maintaining the geometric constraints. We further theoretically demonstrate that preserving classification utility can be promoted by enforcing local force invariance, and under the covariance-level relaxation of this constraint, the FID-based degradation of generative capability is achieved through a rank-one perturbation of the feature covariance aligned with the minimum-eigenvalue direction (i.e., the least force invariance-constrained direction) of the induced force matrix. The released dataset is then matched to this worst-case prior through covariance shaping. This operation changes the global second-order statistics while maintaining the local geometric constraints, reducing generative utility under classification performance-preserving constraints.

In comparison with existing techniques, our GRDP method exhibits three compelling advantages: (1) It preserves the classification utility by maintaining local discriminative geometry with invariant force induced by same-class attraction and different-class repulsion; (2) It degrades the generative capability by distorting global distributional statistics with a worst-case Gaussian target prior within

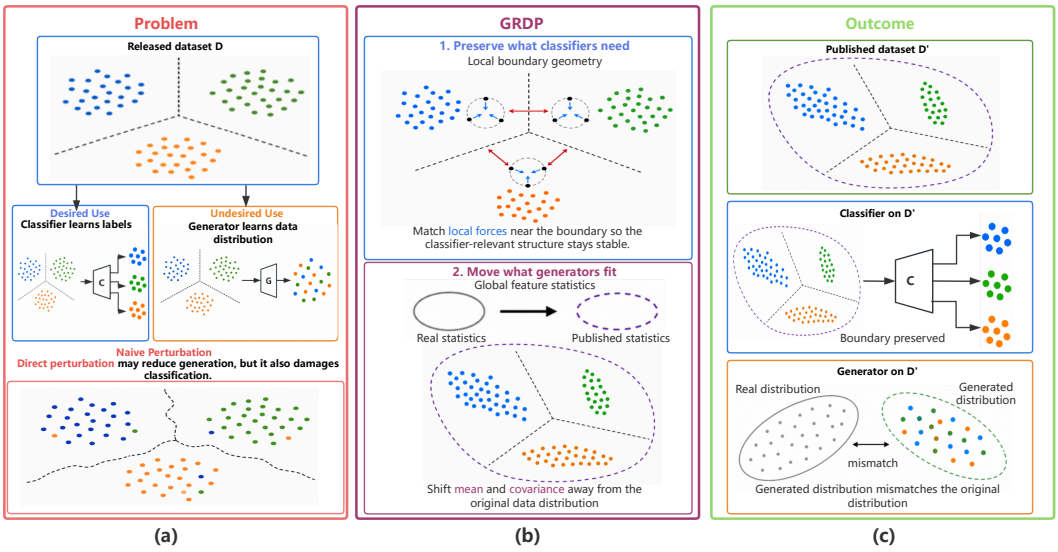


Figure 1: Overview of GRDP for task-specific data publishing. GRDP preserves classification structure while reducing the usefulness of the published data for generative training.

the covariance-level relaxation, which increases the FID-based distributional mismatch; and (3) It offers a unified optimization solution that preserves the classification utility while degrading the generative capability.

Empirical evaluation demonstrates the superior performance of our GRDP model on generation-resistant data publishing. More experiments, implementation details, and hyperparameter setting are presented in Appendices D.

Pipeline. Figure 1 summarizes the pipeline of GRDP. Figure 1(a) shows the data publishing setting. The publisher wants to release data that remains useful for classification, but does not want the same data to be useful for training a generative model. Figure 1(b) shows the main idea of GRDP. GRDP keeps the local forces near boundary-sensitive regions, so the class structure needed by classifiers remains stable. At the same time, it shifts the global feature statistics, so a generative model trained on the published data moves away from the original data distribution. Figure 1(c) shows the final outcome. Classifiers trained on the published data keep a stable decision boundary, while generators trained on the same data produce samples whose distribution does not match the original one.

2 Preliminary

2.1 Finsler metric

Let $\mathcal{F} : \mathbb{R}^d \times \mathbb{R}^d \to [0, \infty)$. For each fixed $z \in \mathbb{R}^d$, the map $v \mapsto \mathcal{F}(z, v)$ satisfies $\mathcal{F}(z, v) = 0$ if and only if $v = 0$, $\mathcal{F}(z, \lambda v) = \lambda \mathcal{F}(z, v)$ for all $\lambda > 0$, and $\mathcal{F}(z, v_1 + v_2) \leq \mathcal{F}(z, v_1) + \mathcal{F}(z, v_2)$ for all $v_1, v_2 \in \mathbb{R}^d$. For any piecewise smooth curve $\gamma : [0, 1] \to \mathbb{R}^d$, define its $\mathcal{F}$-length by $\int_0^1 \mathcal{F}(\gamma(t), \dot{\gamma}(t))dt$. For $z_0, z_1 \in \mathbb{R}^d$, define the induced distance $d_{\mathcal{F}}(z_0, z_1)$ as the infimum of this integral over all piecewise smooth curves with $\gamma(0) = z_0$ and $\gamma(1) = z_1$. The dual gauge is $\mathcal{F}^*(z, p) = \sup_{v \neq 0} \frac{\langle p, v \rangle}{\mathcal{F}(z, v)}$, where $\langle \cdot, \cdot \rangle$ is the Euclidean inner product on $\mathbb{R}^d$. In this paper, we use $\mathcal{F}_0$ as a location-dependent anisotropic metric in the embedding space and use $\mathcal{F}_0^*$ in the Hamilton–Jacobi form of shell evolution.

2.2 Fréchet Inception Distance (FID).

FID measures how far a generated dataset is from the real dataset in a fixed feature space. It compares the two feature distributions through their empirical mean and covariance, where a larger value indicates a larger mismatch.

Let $\{z_j\}_{j=1}^N \subset \mathbb{R}^d$ be a set of feature vectors and define the empirical mean and covariance

$$\mu = \frac{1}{N}\sum_{j=1}^{N} z_j, \quad \Sigma = \frac{1}{N}\sum_{j=1}^{N}(z_j-\mu)(z_j-\mu)^\top. \tag{1}$$

Under the standard Gaussian approximation used by FID, the real features are represented by $\mathcal{N}(\mu_0, \Sigma_0)$ and the generated features by $\mathcal{N}(\mu, \Sigma)$, which yields the closed form

$$\mathrm{FID}\big((\mu,\Sigma),(\mu_0,\Sigma_0)\big) = \|\mu-\mu_0\|_2^2 + \mathrm{Tr}\Big(\Sigma+\Sigma_0-2\big(\Sigma_0^{1/2}\Sigma\Sigma_0^{1/2}\big)^{1/2}\Big). \tag{2}$$

The first term compares means and the trace term compares covariances.

3 Force Invariant Shell Evolution for Boundary Preservation

In data publishing, we release a processed dataset D' instead of the original dataset D. To preserve classification performance, the publishing process should maintain the local discriminative geometry in the embedding space: same-class samples should remain compact, different-class samples should remain separated, and boundary-sensitive neighborhoods should remain stable. We describe this structure through force-invariant shell evolution. Each embedded sample acts as a bubble center, whose shell moves outward under same-class attraction and different-class repulsion. The resulting distance-decaying force field defines an anisotropic shell motion, which is formulated as a Hamilton–Jacobi equation under a force-induced Finsler geometry. We then sample monitor points along finite shell trajectories and enforce force invariance at these points, so the local decision geometry is preserved after publishing.

We write the original dataset as $\{(x_j, y_j)\}_{j=1}^N$ with $x_j \in \mathbb{R}^D$ and $y_j \in \{1,\dots,C\}$. An encoder $E : \mathbb{R}^D \to \mathbb{R}^d$ maps each input to an embedding center $z_j^0 = E(x_j) \in \mathbb{R}^d$, $\quad j = 1,\dots,N$. All forces and shell motion are defined in $\mathbb{R}^d$. Motivated by Theorem 3.1, we use a Rational Quadratic kernel to model distance-decaying center influence.

Theorem 3.1 (Point-source effect with Rational Quadratic decay)**.** *Let $f_\beta : \mathbb{R}^D \to \mathbb{R}^C$ be a differentiable multi-class network satisfying the local smoothness and radial averaging assumptions stated in Appendix C. Consider a training pair (x_0, y_0) with $x_0 \in \mathbb{R}^D$ and $y_0 \in \{1,\dots,C\}$. For a sufficiently small step size $\eta > 0$, one gradient-descent update $\beta^+ = \beta - \eta\nabla_\beta \ell\big(f_\beta(x_0), y_0\big)$ produces a first-order local effect in the input space. Under the local point-source approximation, its radially averaged strength at distance $r = \|x - x_0\|$ can be approximated by the Rational Quadratic decay*

$$K_{\mathrm{RQ}}(r) = \left(1+\frac{r^2}{2\alpha\ell_0^2}\right)^{-\alpha}, \quad r \ge 0, \tag{3}$$

for some $\alpha > 0$ and $\ell_0 > 0$.

Let $P > 0$, $\alpha > 0$, and $\ell_0 > 0$. Define the radial potential

$$\psi(r) = P\left(1+\frac{r^2}{2\alpha\ell_0^2}\right)^{-\alpha}, \quad r \ge 0. \tag{4}$$

For any $z \in \mathbb{R}^d$ and any center z_j^0, define the distance and the unit direction from z to the center z_j^0 as

$$r_j^0(z) = \|z - z_j^0\|, \qquad u_j^0(z) = \frac{z_j^0 - z}{\|z - z_j^0\|} \quad (z \ne z_j^0). \tag{5}$$

Let $\lambda_{\mathrm{diff}} > 0$ and define the class-dependent weight

$$w_{c,j} = \begin{cases} +1, & y_j = c, \\ -\lambda_{\mathrm{diff}}, & y_j \ne c, \end{cases} \quad c \in \{1,\dots,C\}. \tag{6}$$

The class-c resultant force under the original centers is

$$F_0^{(c)}(z) = \sum_{j=1}^{N} w_{c,j}\frac{P\|z-z_j^0\|}{\ell_0^2}\left(1+\frac{\|z-z_j^0\|^2}{2\alpha\ell_0^2}\right)^{-\alpha-1} u_j^0(z). \tag{7}$$

Since $u_j^0(z)$ points from the current location z to the center z_j^0, a positive weight pulls z toward that center, while a negative weight pushes z away from that center. Therefore, $w_{c,j} = +1$ gives same-class attraction and $w_{c,j} = -\lambda_{\text{diff}}$ gives different-class repulsion. The derivation of equation 7 is given in Appendix C.

To measure direction dependent shell motion, we weight each center by its distance decaying influence at the current location and normalize these weights

$$\widetilde{\omega}_j^0(z) = \psi\big(r_j^0(z)\big), \quad \omega_j^0(z) = \frac{\widetilde{\omega}_j^0(z)}{\sum_{k=1}^N \widetilde{\omega}_k^0(z)}. \tag{8}$$

Let $1 < q \leq 2$. Theorem 3.2 formalizes the local anisotropic geometry used by GRDP. It proves that F_0 is a valid Finsler gauge under a mild spanning condition on the local center directions, and derives the dual gauge F_0^* used in the Hamilton–Jacobi shell evolution. The range $1 < q \leq 2$ also allows the later force-mismatch term to be controlled by a quadratic covariance surrogate.

Theorem 3.2 (Finsler form and dual)**.** *Fix $1 < q \leq 2$ and let $q^* = \frac{q}{q-1}$. With $\omega_j^0(z)$ from equation 8 and $u_j^0(z)$ from equation 5, define*

$$\mathcal{F}_0(z, v) = \left(\sum_{j=1}^N \omega_j^0(z) |\langle v, u_j^0(z) \rangle|^q \right)^{1/q}. \tag{9}$$

Then $\mathcal{F}_0(z, v)$ is 1-homogeneous in v and satisfies the triangle inequality. If $\operatorname{span}\{u_j^0(z)\}_{j=1}^N = \mathbb{R}^d$, then $\mathcal{F}_0(z, \cdot)$ vanishes only at $v = 0$, and hence defines a location-dependent norm on the embedding space. The dual quantity $\mathcal{F}_0^*(z, p) = \sup_{v \neq 0} \frac{\langle p, v \rangle}{\mathcal{F}_0(z, v)}$ admits the equivalent coefficient form

$$\mathcal{F}_0^*(z, p) = \inf \left\{ \left(\sum_{j=1}^N \omega_j^0(z)^{-1/(q-1)} |a_j|^{q^*} \right)^{1/q^*} \middle| \, p = \sum_{j=1}^N a_j u_j^0(z) \right\}. \tag{10}$$

Please refer to Appendix C for detailed proof of Theorem 3.2. Let $R > 0$ and $(a)_+ = \max\{a, 0\}$. For each class c, define the local speed

$$V_c^0(z) = \sqrt{\Big(\mathcal{F}_0^*(z, F_0^{(c)}(z))^2 - R^2 \Big)_+}. \tag{11}$$

For each sample index i, let $\Phi_i(z, t)$ be a level-set function whose zero level set gives the bubble shell and whose negative region is inside. The shell evolution is

$$\frac{\partial \Phi_i}{\partial t}(z, t) + V_{y_i}^0(z) \mathcal{F}_0^*\big(z, \nabla_z \Phi_i(z, t)\big) = 0. \tag{12}$$

Let $r_0 > 0$. For each i, take unit directions $\{n_{i,k}\}_{k=1}^K \subset \mathbb{R}^d$ with $\|n_{i,k}\| = 1$, and initialize shell points $b_{i,k}(0) = z_i^0 + r_0 n_{i,k}$.

The Hamilton–Jacobi equation in equation 12 gives the continuous shell evolution that defines where the local force field should be preserved. Since our loss only requires a finite set of monitor points on this evolving shell, we sample shell trajectories by following the local force-induced motion from the initialized shell points. Specifically, for each initialized point, we compute a trajectory $\gamma_{i,k}(t)$ with $\gamma_{i,k}(0) = b_{i,k}(0)$, and advance it by explicit Euler steps

$$\gamma_{i,k}^{s+1} = \gamma_{i,k}^s + \Delta t V_{y_i}^0(\gamma_{i,k}^s) \frac{F_0^{(y_i)}(\gamma_{i,k}^s)}{\|F_0^{(y_i)}(\gamma_{i,k}^s)\|_2 + \varepsilon},$$

where $\Delta t > 0$ is the simulation step size and $\varepsilon > 0$ is a small constant for numerical stability. This gives a finite-direction approximation of the shell motion and provides the monitor points used in the force-invariance loss.

Let $T_{\max} > 0$ and choose sampling times $0 < \tau_{i,k,1} < \cdots < \tau_{i,k,L_{i,k}} \leq T_{\max}$. Define monitor points along the trajectory by $p_{i,k,\ell} = \gamma_{i,k}(\tau_{i,k,\ell})$. At each monitor point, record the original resultant force $G_{i,k,\ell} = F_0^{(y_i)}(p_{i,k,\ell})$.

In the following, we allow the embedding centers to move from $\{z_j^0\}_{j=1}^N$ to a new set $\{z_j^\star\}_{j=1}^N$. For the published centers, define

$$r_j^\star(z) = \|z - z_j^\star\|, \qquad u_j^\star(z) = \frac{z_j^\star - z}{\|z - z_j^\star\|} \quad (z \neq z_j^\star). \tag{13}$$

With $\{z_j^\star\}$, the class-c resultant force is

$$F_\star^{(c)}(z) = \sum_{j=1}^N w_{c,j} \frac{P\|z - z_j^\star\|}{\ell_0^2} \left(1 + \frac{\|z - z_j^\star\|^2}{2\alpha\ell_0^2}\right)^{-\alpha-1} u_j^\star(z). \tag{14}$$

At each monitor point, the mismatch is

$$\Delta F_{i,k,\ell} = F_\star^{(y_i)}(p_{i,k,\ell}) - G_{i,k,\ell}. \tag{15}$$

Let $M = \sum_{i=1}^N \sum_{k=1}^K L_{i,k}$. We measure the mismatch using $\mathcal{F}_0$ in equation 9 and define the force-invariance loss

$$\mathcal{L}_{\text{force}} = \frac{1}{M} \sum_{i=1}^N \sum_{k=1}^K \sum_{\ell=1}^{L_{i,k}} \frac{\mathcal{F}_0\big(p_{i,k,\ell}, \Delta F_{i,k,\ell}\big)^2}{\mathcal{F}_0\big(p_{i,k,\ell}, G_{i,k,\ell}\big)^2}. \tag{16}$$

4 Covariance Prior Matching for Generation Degradation

In Section 3, we preserve classification-relevant local geometry by matching the resultant force at monitor points sampled along bubble-shell trajectories. We now degrade generative capability by shifting the global feature statistics of the published data. Since FID compares real and generated distributions through empirical means and covariances in a fixed feature space, we use a covariance-level relaxation to derive a Gaussian target prior whose covariance shift increases the FID-based mismatch under the force budget. The published centers are then matched to this prior, so global statistics are distorted while local force geometry remains preserved.

We approximate generation-quality changes using the distribution of embedding centers. Let (μ_0, Σ_0) and $(\mu_\star, \Sigma_\star)$ denote the empirical mean and covariance of the original centers $\{z_j^0\}_{j=1}^N$ and published centers $\{z_j^\star\}_{j=1}^N$, respectively. For candidate centers $\{z_j\}$, define $\mu(\{z_j\}) = \frac{1}{N}\sum_{j=1}^N z_j$,$\Sigma(\{z_j\}) = \frac{1}{N}\sum_{j=1}^N (z_j - \mu(\{z_j\}))(z_j - \mu(\{z_j\}))^\top$. A natural objective is to increase the FID from the original center distribution while keeping the local force mismatch small:

$$\max_{\{z_j\}} \text{FID}\Big((\mu(\{z_j\}), \Sigma(\{z_j\})), (\mu_0, \Sigma_0)\Big) \quad \text{s.t.} \quad \Delta_{\text{force}}(\{z_j\}) \leq \kappa. \tag{17}$$

Here Δ_{force} is the normalized monitor-point force mismatch. In the covariance relaxation below, we use a weighted monitor-point surrogate $\widetilde{\Delta}_{\text{force}}$, defined in Appendix C, whose scaling leads to the trace constraint in Eq. equation 24. Since this problem is nonconvex and depends on all centers, we use a covariance-level relaxation with fixed mean $\mu_{\text{prior}} = \mu_0$, and allocate the force budget to a covariance shift.

We next construct a quadratic surrogate for the monitor-point force constraint. The force matching procedure fixes the monitor points $p_{i,k,\ell}$ and records the original resultant force at each monitor point as $G_{i,k,\ell} = F_0^{(y_i)}(p_{i,k,\ell})$. It also fixes the original geometry $\mathcal{F}_0$ through $\omega_m^0(p)$ and $u_m^0(p)$. For any monitor point p and any $v \in \mathbb{R}^d$, since $1 < q \leq 2$ and $\sum_{m=1}^N \omega_m^0(p) = 1$, the monotonicity of weighted L_p norms gives

$$\mathcal{F}_0(p, v)^2 = \Big(\sum_{m=1}^N \omega_m^0(p) |\langle v, u_m^0(p)\rangle|^q\Big)^{2/q} \leq \sum_{m=1}^N \omega_m^0(p) \langle v, u_m^0(p)\rangle^2 = v^\top W(p) v, \tag{18}$$

where

$$W(p) = \sum_{m=1}^N \omega_m^0(p) u_m^0(p) u_m^0(p)^\top, \quad v^\top W(p) v \geq 0, \forall v \in \mathbb{R}^d. \tag{19}$$

To connect this monitor-point surrogate with second-order statistics, we use a first-order expansion of the resultant force around the original centers. For each monitor point $p_{i,k,\ell}$ and each original center z_j^0, define

$$r_{i,k,\ell,j} = \|p_{i,k,\ell} - z_j^0\|, \qquad u_{i,k,\ell,j} = \frac{z_j^0 - p_{i,k,\ell}}{\|p_{i,k,\ell} - z_j^0\|}. \tag{20}$$

Define

$$a_{i,k,\ell,j} = \frac{P}{\ell_0^2}\left(1 + \frac{r_{i,k,\ell,j}^2}{2\alpha\ell_0^2}\right)^{-\alpha-1}, \quad b_{i,k,\ell,j} = -\frac{P(\alpha+1)}{\alpha\ell_0^4} r_{i,k,\ell,j}^2 \left(1 + \frac{r_{i,k,\ell,j}^2}{2\alpha\ell_0^2}\right)^{-\alpha-2}. \tag{21}$$

Let $c = y_i$. The local Jacobian of the class-c resultant force at $p_{i,k,\ell}$ is

$$J_{i,k,\ell}^0 = \sum_{j=1}^{N} w_{c,j}\left(a_{i,k,\ell,j} I + b_{i,k,\ell,j} u_{i,k,\ell,j} u_{i,k,\ell,j}^\top\right), \qquad c = y_i. \tag{22}$$

Combining the quadratic upper bound in equation 18 with this local linearization over all monitor points gives the matrix

$$A = \frac{N^2}{M} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\ell=1}^{L_{i,k}} \frac{L_{i,k}}{\mathcal{F}_0(p_{i,k,\ell}, G_{i,k,\ell})^2} \, (J_{i,k,\ell}^0)^\top W(p_{i,k,\ell}) J_{i,k,\ell}^0. \tag{23}$$

where $M = \sum_{i=1}^{N} \sum_{k=1}^{K} L_{i,k}$. We then impose the covariance-level condition

$$\mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big) \le \kappa, \quad \Sigma - \Sigma_0 \succeq 0, \tag{24}$$

as a sufficient quadratic surrogate for controlling the local force mismatch. The detailed derivation of this surrogate is provided in Appendix C.

We now maximize the covariance part of the FID objective under equation 24. For any candidate covariance Σ, define

$$\mathcal{J}(\Sigma) = \mathrm{Tr}\Big(\Sigma_0^{-1/2}\Sigma\Sigma_0^{-1/2} + I - 2\big(\Sigma_0^{-1/2}\Sigma\Sigma_0^{-1/2}\big)^{1/2}\Big). \tag{25}$$

We obtain the prior covariance from

$$\Sigma_{\mathrm{prior}} \in \arg\max_{\Sigma} \mathcal{J}(\Sigma) \quad \text{s.t.} \quad \mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big) \le \kappa, \quad \Sigma - \Sigma_0 \succeq 0. \tag{26}$$

Let $B = \Sigma_0^{1/2} A \Sigma_0^{1/2}$. Assume $B \succ 0$, and let $(\lambda_{\min}, u_{\min})$ be the minimum eigenpair of B:

$$B u_{\min} = \lambda_{\min} u_{\min}, \quad \|u_{\min}\|_2 = 1. \tag{27}$$

Theorem 4.1 (Prior covariance)**.** *Assume $\Sigma_0 \succ 0$, $\mu_{\mathrm{prior}} = \mu_0$, and $B = \Sigma_0^{1/2} A \Sigma_0^{1/2} \succ 0$. For any $\kappa > 0$, an optimizer of equation 26 is*

$$\Sigma_{\mathrm{prior}} = \Sigma_0^{1/2}\Big(I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top\Big)\Sigma_0^{1/2}. \tag{28}$$

We keep the prior mean fixed as $\mu_{\mathrm{prior}} = \mu_0$ and allocate the force budget to the covariance shift.

Please refer to the appendix for the proof of Theorem 4.1.

We then match the empirical mean and covariance of the published centers to this prior. The empirical mean and covariance of the published centers are

$$\mu_\star = \frac{1}{N}\sum_{j=1}^{N} z_j^\star, \quad \Sigma_\star = \frac{1}{N}\sum_{j=1}^{N} (z_j^\star - \mu_\star)(z_j^\star - \mu_\star)^\top. \tag{29}$$

We get

$$\mathcal{L}_{\mathrm{match}} = \|\mu_\star - \mu_{\mathrm{prior}}\|_2^2 + \left\| \Sigma_0^{-1/2}\Sigma_\star\Sigma_0^{-1/2} - \left(I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top\right)\right\|_F^2. \tag{30}$$

Finally, we optimize the published centers with a single objective that combines the force term from Section 3 and the prior matching term,

$$\mathcal{L}_{\mathrm{total}}(\{z_j^\star\}) = \mathcal{L}_{\mathrm{force}}(\{z_j^\star\}) + \mathcal{L}_{\mathrm{match}}(\{z_j^\star\}). \tag{31}$$

Here $\mathcal{L}_{\mathrm{force}}$ preserves local classification geometry, while $\mathcal{L}_{\mathrm{match}}$ shifts global feature statistics toward this Gaussian target prior to reduce generative utility.

Table 1: Comparison with poisoning-oriented baselines under a 5% modified-data budget. Acc-R18 and Acc-VGG denote top-1 accuracy (%) of ResNet-18 and VGG16 trained on the published data.

Method	CIFAR-100				Tiny ImageNet			
	Acc-R18	Acc-VGG	DDIM-FID	DDIM-LPIPS	Acc-R18	Acc-VGG	DDIM-FID	DDIM-LPIPS
BiBad	74.75	59.44	32.57	0.231	62.79	54.00	41.85	0.581
UIB	75.78	59.15	29.35	0.243	63.97	44.71	40.33	0.593
Model Collapse	74.72	60.14	39.88	0.251	62.58	44.36	52.73	0.592
VillanDiffusion	**76.61**	60.23	29.22	0.249	64.66	44.34	41.37	0.586
GRDP	76.47	**61.35**	**53.84**	**0.265**	**64.71**	**56.14**	**82.01**	**0.627**

5 Experimental Evaluation

In this section, we evaluate the effectiveness of our GRDP method compared to representative data publishing baselines. Through comprehensive experiments on two image datasets, we show that GRDP preserves the utility of the published data for downstream classification while reducing its usefulness for training generative models. In particular, classifiers trained on the published data achieve accuracy close to those trained on the original data, while diffusion models trained on the same published data produce lower-quality samples. Please refer to the appendix for detailed experimental settings and additional results.

Datasets and Models. We conduct experiments on two widely used image classification datasets: CIFAR-100 (Krizhevsky, 2009) and Tiny ImageNet (Le & Yang, 2015). The classifiers are trained on the published training sets and evaluated on the official test or validation sets. For both datasets, we use ResNet-18 (He et al., 2015) and VGG16 (Simonyan & Zisserman, 2015) as downstream classification models. For generative evaluation, we use unconditional DDIM-based diffusion models (Song et al., 2020). Specifically, we use the unconditional CIFAR-10 diffusion checkpoint for CIFAR-100 and the unconditional ImageNet-64 diffusion checkpoint for Tiny ImageNet, and fine-tune each model on the corresponding published dataset.

Baselines. We compare GRDP with eight related methods that are closest to our setting. We first consider poisoning-oriented methods, including BiBad Pan et al. (2024), UIB Han et al. (2025), Model Collapse Shi et al. (2025), and VillanDiffusion Chou et al. (2023). These methods modify part of the training data to reduce the quality of generative models trained on the released dataset. For these baselines, we evaluate poisoning ratios of 1%, 2%, 5%, and 10%. We also compare with dataset distillation methods, including SRe^{2}L Yin et al. (2024), RDED Sun et al. (2024), CDA Yin & Shen (2024), and D3HR Zhao et al. (2025). These methods release compressed training sets instead of the full dataset. For the distilled-data comparison, we evaluate the released data under the same downstream classification setting as GRDP. Detailed settings for each baseline are provided in the appendix.

Evaluation Metrics. We evaluate each published dataset from both classification and generation perspectives. For classification, we train downstream classifiers on the published data using a fixed training schedule and report the final accuracy on the official test or validation set. Higher accuracy means that the published data better preserves the information needed for classification. For generation, we train a DDIM-based diffusion model on the published or poisoned dataset and evaluate the generated samples using Fréchet Inception Distance (FID) (Jayasumana et al., 2024) and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018). FID measures the distribution gap between generated images and real images, while LPIPS measures their perceptual difference in a learned feature space. Higher FID and LPIPS indicate that the generated images are farther from the original data distribution, which means the released data is less useful for generative training.

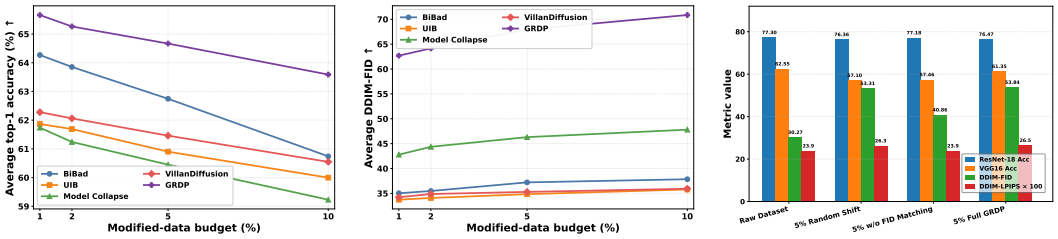
Generative degradation with preserved classification utility. Table 1 reports the results under a 5% modified-data budget. On CIFAR-100, GRDP achieves 76.47% accuracy on ResNet-18 and the best 61.35% accuracy on VGG16, while also reaching the highest 53.84 DDIM-FID and 0.265 DDIM-LPIPS. The best baseline only reaches 39.88 DDIM-FID and 0.251 DDIM-LPIPS. On Tiny ImageNet, GRDP gives the best classification results on both models, with 64.71% on ResNet-18 and 56.14% on VGG16. It also achieves the strongest degradation, with 82.01 DDIM-FID and 0.627 DDIM-LPIPS, improving over the best baseline by 29.28 in DDIM-FID and 0.034 in DDIM-LPIPS. These results show that GRDP does not degrade generation by simply damaging the data.

Table 2: Comparison with distilled-data baselines at IPC=50. Acc denotes the downstream classification accuracy (%) under the same evaluation protocol.

Method	CIFAR-100			Tiny ImageNet		
	Acc	FID	LPIPS	Acc	FID	LPIPS
SRe^{2}L	50.98	25.58	0.162	48.44	75.70	0.567
RDED	62.44	26.53	0.159	**57.98**	74.84	0.571
CDA	63.80	26.51	0.163	48.86	78.16	0.579
D3HR	**65.70**	92.33	0.201	56.17	97.00	0.601
GRDP	65.25	**94.77**	**0.212**	56.21	**97.89**	**0.622**

Its force-invariance constraint keeps the local discriminative structure for classification, while FID prior matching shifts the global distribution used by diffusion models.

Utility-protection trade-off against distilled-data baselines. Table 2 compares GRDP with distilled-data baselines at IPC=50. On CIFAR-100, GRDP reaches 65.25% accuracy, close to D3HR at 65.70%, while achieving higher degradation with 94.77 DDIM-FID and 0.212 DDIM-LPIPS. On Tiny ImageNet, GRDP obtains 56.21% accuracy, close to D3HR at 56.17% and below RDED at 57.98%. However, RDED gives much weaker degradation, with only 74.84 DDIM-FID and 0.571 DDIM-LPIPS, while GRDP achieves the highest 97.89 DDIM-FID and 0.622 DDIM-LPIPS. This shows that distillation mainly keeps data useful for classification, but it does not directly reduce its value for generative training. In contrast, GRDP preserves classification forces and matches a generation-degrading prior, leading to a better utility-protection trade-off.



(a) Classification utility under different modified-data budgets.

(b) Generative degradation under different modified-data budgets.

(c) Ablation study on CIFAR-100 under a 5% budget.

Figure 2: Main trend and ablation results.

Trend under different modified-data budgets. Figure 2a and Figure 2b show the results from 1% to 10% modified-data budgets. GRDP keeps the highest average classification accuracy at all budgets. Its accuracy changes from 65.66% at 1% to 63.59% at 10%, while BiBad drops from 64.27% to 60.74%. GRDP also gives the largest generation degradation. Its average DDIM-FID increases from 62.69 to 70.86, while Model Collapse only increases from 42.78 to 47.80. This means that GRDP does not reduce generative utility by simply corrupting the data. The force-invariance term preserves the local same-class attraction and different-class repulsion near the boundary, so classifiers can still learn stable decision boundaries. At the same time, the FID prior matching term shifts the global mean and covariance of the published features, so diffusion models trained on the published data generate samples that are farther from the original data distribution.

Ablation study. Figure 2c shows the ablation study on CIFAR-100 under the 5% budget. The raw dataset has high accuracy but low degradation, with 30.27 DDIM-FID and 0.239 DDIM-LPIPS. Random shift increases DDIM-FID to 53.31 and DDIM-LPIPS to 0.263, but it reduces VGG16 accuracy to 57.10%, which shows that direct perturbation can damage the discriminative structure. Without FID prior matching, ResNet-18 accuracy remains high at 77.18%, but DDIM-FID drops to 40.86, showing that force preservation alone is not enough to strongly degrade generation. Full GRDP keeps high accuracy, with 76.47% on ResNet-18 and 61.35% on VGG16, and achieves the strongest degradation, with 53.84 DDIM-FID and 0.265 DDIM-LPIPS. These results show the role of each part: force invariance preserves classification utility, while FID prior matching provides the main pressure that makes generative training less effective.

REFERENCES

George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A. Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories, 2022. URL https://arxiv.org/abs/2203.11932.

Mingyang Chen, Jiawei Du, Bo Huang, Yi Wang, Xiaobo Zhang, and Wei Wang. Influence-guided diffusion for dataset distillation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=0whx8MhysK.

Ruoxi Chen, Haibo Jin, Yixin Liu, Jinyin Chen, Haohan Wang, and Lichao Sun. Editshield: Protecting unauthorized image editing by instruction-guided diffusion models, 2024. URL https://arxiv.org/abs/2311.12066.

Weixin Chen, Dawn Song, and Bo Li. Trojdiff: Trojan attacks on diffusion models with diverse targets, 2023. URL https://arxiv.org/abs/2303.05762.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. Villandiffusion: A unified backdoor attack framework for diffusion models, 2023. URL https://arxiv.org/abs/2306.06874.

Zhiwei Deng and Olga Russakovsky. Remember the past: Distilling datasets into addressable memories for neural networks, 2022. URL https://arxiv.org/abs/2206.02916.

Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=AAWuCvzaVt.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407, August 2014a. ISSN 1551-305X. doi: 10.1561/0400000042. URL https://doi.org/10.1561/0400000042.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9:211–407, 2014b. URL https://api.semanticscholar.org/CorpusID:207178262.

Shaopeng Fu, Fengxiang He, Yang Liu, Li Shen, and Dacheng Tao. Robust unlearnable examples: Protecting data against adversarial learning, 2022. URL https://arxiv.org/abs/2203.14533.

Benjamin C. M. Fung, Ke Wang, and Philip S. Yu. Anonymizing classification data for privacy preservation. *IEEE Trans. on Knowl. and Data Eng.*, 19(5):711–725, May 2007a. ISSN 1041-4347. doi: 10.1109/TKDE.2007.1015. URL https://doi.org/10.1109/TKDE.2007.1015.

Benjamin C. M. Fung, Ke Wang, and Philip S. Yu. Anonymizing classification data for privacy preservation. *IEEE Transactions on Knowledge and Data Engineering*, 19, 2007b. URL https://api.semanticscholar.org/CorpusID:9869870.

Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Comput. Surv.*, 42(4), June 2010. ISSN 0360-0300. doi: 10.1145/1749603.1749605. URL https://doi.org/10.1145/1749603.1749605.

Yong-Feng Ge, Hua Wang, Jinli Cao, Yanchun Zhang, and Xiaohong Jiang. Privacy-preserving data publishing: an information-driven distributed genetic algorithm. *World Wide Web*, 27:1–21, 2024. URL https://api.semanticscholar.org/CorpusID:266985891.

Xiao Han, Yuncong Yang, and Junjie Wu. Hyobscure: Hybrid obscuring for privacy-preserving data publishing, 2021. URL https://arxiv.org/abs/2112.07850.

Yuning Han, Bingyin Zhao, Rui Chu, Feng Luo, Biplab Sikdar, and Yingjie Lao. Uibdiffusion: Universal imperceptible backdoor attack for diffusion models, 2025. URL https://arxiv.org/abs/2412.11441.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. URL https://arxiv.org/abs/1706.08500.

Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable, 2021. URL https://arxiv.org/abs/2101.04898.

Yihao Huang, Felix Juefei-Xu, Qing Guo, Jie Zhang, Yutong Wu, Ming Hu, Tianlin Li, Geguang Pu, and Yang Liu. Personalization as a shortcut for few-shot backdoor attack against text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2305.10701.

Sangwon Jang, June Suk Choi, Jaehyeong Jo, Kimin Lee, and Sung Ju Hwang. Silent branding attack: Trigger-free data poisoning attack on text-to-image diffusion models, 2025. URL https://arxiv.org/abs/2503.09669.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation, 2024. URL https://arxiv.org/abs/2401.09603.

Honglu Jiang, S M Sarwar, Haotian Yu, and Sheikh Ariful Islam. Differentially private data publication with multi-level data utility, 2021. URL https://arxiv.org/abs/2112.07061.

Honglu Jiang, Haotian Yu, Xiuzhen Cheng, Jian Pei, Robert Pless, and Jiguo Yu. Dp2-pub: Differentially private high-dimensional data publication with invariant post randomization, 2022. URL https://arxiv.org/abs/2208.11693.

John Ernest Kratz and Carly Strasser. Data publication consensus and controversies. *F1000Research*, 3, 2014. URL https://api.semanticscholar.org/CorpusID:54565942.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL https://api.semanticscholar.org/CorpusID:18268744.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models, 2019. URL https://arxiv.org/abs/1904.06991.

Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc Tran, and Anh Tran. Anti-dreambooth: Protecting users from personalized text-to-image synthesis, 2023. URL https://arxiv.org/abs/2303.15433.

Ya Le and Xuan S. Yang. Tiny imagenet visual recognition challenge. 2015. URL https://api.semanticscholar.org/CorpusID:16664790.

Hyukki Lee, Soohyung Kim, Jong Wook Kim, and Yon Dohn Chung. Utility-preserving anonymization for health data publishing. *BMC Medical Informatics and Decision Making*, 17, 2017. URL https://api.semanticscholar.org/CorpusID:4882283.

Kecen Li, Chen Gong, Zhixiang Li, Yuzhong Zhao, Xinwen Hou, and Tianhao Wang. Privimage: Differentially private synthetic image generation using diffusion models with semantic-aware pretraining, 2024. URL https://arxiv.org/abs/2311.12850.

Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115, 2007. doi: 10.1109/ICDE.2007.367856.

Tiancheng Li and Ninghui Li. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pp. 517–526, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605584959. doi: 10.1145/1557019.1557079. URL https://doi.org/10.1145/1557019.1557079.

Chumeng Liang and Xiaoyu Wu. Mist: Towards improved adversarial examples for diffusion models, 2023. URL https://arxiv.org/abs/2305.12683.

Chumeng Liang, Xiaoyu Wu, Yang Hua, Jiaru Zhang, Yiming Xue, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples, 2023. URL https://arxiv.org/abs/2302.04578.

Yixin Liu, Kaidi Xu, Xun Chen, and Lichao Sun. Stable unlearnable example: Enhancing the robustness of unlearnable examples via stable error-minimizing noise, 2023. URL https://arxiv.org/abs/2311.13091.

Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. L-diversity: privacy beyond k-anonymity. *22nd International Conference on Data Engineering (ICDE'06)*, pp. 24–24, 2006. URL https://api.semanticscholar.org/CorpusID:679934.

Timothy Nguyen, Roman Novak, Lechao Xiao, and Jaehoon Lee. Dataset distillation with infinitely wide convolutional networks, 2022. URL https://arxiv.org/abs/2107.13034.

Zhuoshi Pan, Yuguang Yao, Gaowen Liu, Bingquan Shen, H. Vicky Zhao, Ramana Rao Kompella, and Sijia Liu. From trojan horses to castle walls: Unveiling bilateral data poisoning effects in diffusion models, 2024. URL https://arxiv.org/abs/2311.02373.

Noseong Park, Mahmoud Mohammadi, Kshitij Gorde, Sushil Jajodia, Hongkyu Park, and Youngmin Kim. Data synthesis based on generative adversarial networks. *Proceedings of the VLDB Endowment*, 11(10):1071–1083, 2018. ISSN 2150-8097. doi: 10.14778/3231751.3231757. URL http://dx.doi.org/10.14778/3231751.3231757.

Jean-Baptiste Poline. From data sharing to data publishing [version 2; peer review: 2 approved, 1 approved with reservations]. *MNI open research*, 2, 2018. URL https://api.semanticscholar.org/CorpusID:168170342.

Aria Rezaei, Chaowei Xiao, Jie Gao, Bo Li, and Sirajum Munir. Application-driven privacy-preserving data publishing with correlated attributes, 2021. URL https://arxiv.org/abs/1812.10193.

Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious ai-powered image editing, 2023. URL https://arxiv.org/abs/2302.06588.

Sina Shaham, Ming Ding, Bo Liu, Shuping Dang, Zihuai Lin, and Jun Li. Privacy preserving location data publishing: A machine learning approach. *IEEE Transactions on Knowledge and Data Engineering*, 33(9):3270–3283, 2021. doi: 10.1109/TKDE.2020.2964658.

Shawn Shan, Wenxin Ding, Josephine Passananti, Stanley Wu, Haitao Zheng, and Ben Y. Zhao. Nightshade: Prompt-specific poisoning attacks on text-to-image generative models, 2024. URL https://arxiv.org/abs/2310.13828.

Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y. Zhao. Glaze: Protecting artists from style mimicry by text-to-image models, 2025. URL https://arxiv.org/abs/2302.04222.

Lianghe Shi, Meng Wu, Huijie Zhang, Zekai Zhang, Molei Tao, and Qing Qu. A closer look at model collapse: From a generalization-to-memorization perspective, 2025. URL https://arxiv.org/abs/2509.16499.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. URL https://arxiv.org/abs/1409.1556.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ArXiv*, abs/2010.02502, 2020. URL https://api.semanticscholar.org/CorpusID:222140788.

Peng Sun, Bei Shi, Daiwei Yu, and Tao Lin. On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm, 2024. URL https://arxiv.org/abs/2312.03526.

Latanya Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, October 2002a. ISSN 0218-4885. doi: 10.1142/S0218488502001648. URL https://doi.org/10.1142/S0218488502001648.

Latanya Sweeney. k-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl. Based Syst.*, 10:557–570, 2002b. URL https://api.semanticscholar.org/CorpusID:361794.

Haonan Wang, Qianli Shen, Yao Tong, Yang Zhang, and Kenji Kawaguchi. The stronger the diffusion model, the easier the backdoor: Data poisoning to induce copyright breaches without adjusting finetuning pipeline. In *NeurIPS 2023 Workshop on Backdoors in Deep Learning - The Good, the Bad, and the Ugly*, 2024. URL https://openreview.net/forum?id=20FxHX25aq.

Kai Wang, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. Cafe: Learning to condense dataset by aligning features, 2022. URL https://arxiv.org/abs/2203.01531.

Yi Yin, Guangquan Zhang, Hua Zuo, and Jie Lu. Privacy-utility trade-off in data publication: A bilevel optimization framework with curvature-guided perturbation, 2025. URL https://arxiv.org/abs/2509.02048.

Zeyuan Yin and Zhiqiang Shen. Dataset distillation via curriculum data synthesis in large data era, 2024. URL https://arxiv.org/abs/2311.18838.

Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective, 2024. URL https://arxiv.org/abs/2306.13092.

Jinsung Yoon, James Jordon, and Mihaela van der Schaar. PATE-GAN: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=S1zk9iRqF7.

Lei Zhang, Jie Zhang, Bowen Lei, Subhabrata Mukherjee, Xiang Pan, Bo Zhao, Caiwen Ding, Yao Li, and Dongkuan Xu. Accelerating dataset distillation via model augmentation, 2023. URL https://arxiv.org/abs/2212.06152.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018. URL https://api.semanticscholar.org/CorpusID:4766599.

Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching, 2022. URL https://arxiv.org/abs/2110.04181.

Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation, 2023. URL https://arxiv.org/abs/2307.09742.

Lin Zhao, Yushu Wu, Xinru Jiang, Jianyang Gu, Yanzhi Wang, Xiaolin Xu, Pu Zhao, and Xue Lin. Taming diffusion for dataset distillation with high representativeness, 2025. URL https://arxiv.org/abs/2505.18399.

Yongchao Zhou, Ehsan Nezhadarya, and Jimmy Ba. Dataset distillation using neural feature regression, 2022. URL https://arxiv.org/abs/2206.00719.

A Related Work

In our task specific data publishing setting, we do not release the original dataset D. We release a processed dataset D' that remains useful for classification but does not match the true data distribution of D for generative training. Around this goal, related work largely falls into three categories.

Publishing side protection for diffusion generation and editing, and data poisoning. This line of work aligns with our goal of weakening generative capability. A first line of work assumes the publisher can modify images before they are shared, and uses that control to make diffusion based personalization fail. One representative method adds subtle perturbations to user photos so DreamBooth style fine tuning on the released set does not reproduce the intended person even when the prompts are correct Le et al. (2023). A parallel direction focuses on style imitation, where Shan et al. (2025) applies small edits that keep the artwork visually acceptable but make diffusion adaptation reproduce a different style. Diffusion specific adversarial constructions push this idea further by building the perturbation through the diffusion process itself. In Liang et al. (2023), training a diffusion model on the protected images yields noticeably worse painting imitation, and Liang & Wu (2023) improves this kind of protection so it remains effective across common diffusion implementations and fine tuning setups. Other work keeps the images visually intact, but makes them a bad training set. Huang et al. (2021) does this by adding perturbations that cause standard training to miss the target concept even though the images still look normal. Fu et al. (2022) builds perturbations that remain effective even when the learner uses adversarial training, and Liu et al. (2023) improves stability so the protection does not disappear under typical changes in optimization and training details. Diffusion models make it easy to take a published image and obtain an edited version from a short instruction, so several works protect the image at release time so that this editing pipeline fails on the protected input Chen et al. (2024); Salman et al. (2023). Chen et al. (2024) adds a small perturbation to the published image so diffusion based editing produces incorrect or implausible edits instead of the requested modification. Salman et al. (2023) follows the same goal and measures success by how much extra effort an attacker needs before obtaining a usable edit. Other works show that a released dataset can also be used to train a diffusion model with a hidden trigger behavior, where a small poisoned subset makes a chosen prompt reliably produce an attacker selected output after training Shan et al. (2024); Pan et al. (2024); Wang et al. (2024); Jang et al. (2025); Han et al. (2025); Chou et al. (2023); Huang et al. (2023); Chen et al. (2023). Shan et al. (2024) poisons a small subset so that specific prompts reliably trigger attacker chosen outputs after training. Wang et al. (2024) studies bilateral poisoning and shows that interactions between poisoned and clean data can change both attack success and side effects. Wang et al. (2024) reports that stronger diffusion models can be easier to backdoor and that data poisoning can induce copyright related misuse without changing the fine tuning pipeline. Jang et al. (2025) presents trigger free poisoning that creates a consistent branding effect without an explicit trigger token. Han et al. (2025) constructs universal and imperceptible backdoors so the trigger remains hard to notice while still working across settings. Chou et al. (2023) unifies attack components and evaluation choices so diffusion backdoor results can be compared under a shared setup.

Dataset Distillation and Condensation for classification performance. These methods replace D with a released dataset D' and aim for the classifier trained on D' to behave almost the same as the classifier trained on D Cazenavette et al. (2022); Wang et al. (2022); Deng & Russakovsky (2022); Zhou et al. (2022); Zhao & Bilen (2022); Nguyen et al. (2022); Zhao et al. (2023); Zhang et al. (2023); Chen et al. (2025); Zhao et al. (2025). Cazenavette et al. (2022) learns a small synthetic dataset whose training trajectory matches the trajectory from the full dataset so the final classifier stays accurate. Wang et al. (2022) learns a condensed dataset by aligning features of real and synthetic samples so training on the condensed set is more stable and accurate. Deng & Russakovsky (2022) stores training relevant information as a small set of addressable memories that can be recalled to train or recover a classifier. Distribution matching is another widely used approach for scalability. Zhao & Bilen (2022) learns condensed data by matching feature distributions between real and synthetic sets under a set of feature extractors. Zhao et al. (2023) improves distribution matching so the condensed set remains effective under more encoders and training settings. Nguyen et al. (2022) analyzes and constructs distilled datasets in an infinitely wide ResNet regime where training behavior becomes easier to match. Recent papers also use diffusion models as tools for producing distilled datasets, especially when the target is high resolution imagery. Chen et al. (2025) uses diffusion sampling guided by training sensitivity so the generated distilled set supports down-

stream classification on high resolution data. Zhao et al. (2025) improves diffusion based distillation so the distilled set covers the data more broadly and trains stronger classifiers.

Traditional privacy-preserving data publishing. Traditional privacy-preserving data publishing aims to release transformed, anonymized, perturbed, or synthetic data that protects sensitive information while retaining utility for downstream analysis Fung et al. (2010); Li & Li (2009); Fung et al. (2007b). Classical approaches use anonymization criteria such as k-anonymity, ℓ-diversity, and t-closeness to reduce record-level disclosure risk Sweeney (2002a); Machanavajjhala et al. (2006); Li et al. (2007), while differential privacy provides formal privacy guarantees for statistical release and data publication Dwork & Roth (2014a). Recent studies extend this goal to different publication settings, including location data publishing Shaham et al. (2021), hybrid generalization and obfuscation Han et al. (2021), health data anonymization Lee et al. (2017), high-dimensional differentially private publication Jiang et al. (2022), multi-level utility under differential privacy Jiang et al. (2021), application-driven publishing with correlated attributes Rezaei et al. (2021), information-driven genetic optimization for privacy-preserving publishing Yin et al. (2025); Ge et al. (2024), and synthetic or private synthetic data generation Park et al. (2018); Yoon et al. (2019); Li et al. (2024). These methods focus on balancing privacy protection and data utility, but they do not explicitly address generative reuse: the released data may still preserve distributional and semantic information that allows generative models to learn the original data distribution.

We publish a processed dataset D' instead of original dataset D, and we require two outcomes at the same time. A classifier trained on D' should behave almost the same as a classifier trained on D, while a generative model trained on D' should fail to fit the data distribution of D. Existing directions usually guarantee only one of these outcomes. Release time protections against personalization, style imitation, and instruction based editing are designed to make diffusion training or editing break, and they mainly ensure that the protected images still look acceptable. They do not control how samples with the same label form neighborhoods in feature space or how different labels stay separated, so the same perturbation that disrupts diffusion training can also distort the geometry that classification training needs and reduce accuracy. Dataset distillation and condensation methods take the opposite route. They are built to keep classification training unchanged after replacing D with D', which often means preserving the image feature that a classifier depends on, such as local structure and within class variation. That feature is also what diffusion models learn from when they are trained on images, so a D' that works well for classification can still let a generator learn a close approximation of the original distribution. We therefore design the publishing transform to satisfy both requirements in one procedure. It keeps the representation components that maintain label separation for classification, and it shifts the global feature statistics that diffusion models tend to fit. As a result, D' remains effective for training classifiers, while training a generator on D' no longer recovers the distribution of D.

B ALGORITHM

Lines 1-5 map each sample to an embedding center $z_j^0 = E(x_j)$ and build the original class-wise interaction field in $\mathbb{R}^d$. Each center contributes a distance-decaying interaction through $\psi(r)$, with the sign determined by $w_{c,j}$. Since $u_j^0(z)$ points from the current location z to the center z_j^0, positive weights induce same-class attraction and negative weights induce different-class repulsion. Summing these interactions gives the resultant force $F_0^{(c)}(z)$. The same block defines $\mathcal{F}_0$ and its dual $\mathcal{F}_0^*$, which measure vectors and gradients at location z under the direction weights $\omega_j^0(z)$.

Lines 6-18 construct the monitor points where we later enforce force invariance. For each center z_i^0, we initialize K shell points on a small sphere of radius r_0. The Hamilton–Jacobi equation gives the continuous shell-evolution model, while the implementation uses the force-guided trajectory update in Algorithm 1. Starting from $z_i^0 + r_0 n_{i,k}$, each trajectory is advanced by explicit time stepping along the normalized resultant-force direction with speed $V_{y_i}^0$. This gives trajectories $\gamma_{i,k}(t)$, from which we sample monitor points $p_{i,k,\ell} = \gamma_{i,k}(\tau_{i,k,\ell})$. At every monitor point, we record the original resultant force $G_{i,k,\ell} = F_0^{(y_i)}(p_{i,k,\ell})$.

Lines 20-24 compute the relaxed target second-order statistics used for prior matching. We first compute (μ_0, Σ_0) from the original centers $\{z_j^0\}$, then form the matrix A from the monitor points, recorded forces, and original centers. The matrix A measures the covariance-level cost of changing

the local force field under the quadratic surrogate. The minimum-eigenvalue direction of $\Sigma_0^{1/2} A \Sigma_0^{1/2}$ gives the lowest-cost covariance direction under this surrogate. We keep the prior mean fixed as $\mu_{\text{prior}} = \mu_0$ and allocate the budget κ along this covariance direction to obtain Σ_{prior}.

Lines 25-34 optimize the published centers $\{z_j^\star\}$. Starting from $z_j^\star = z_j^0$, each iteration recomputes the monitor-point forces under the current centers $\{z_j^\star\}$. The force loss $\mathcal{L}_{\text{force}}$ keeps $F_\star^{(y_i)}(p_{i,k,\ell})$ close to the recorded force $G_{i,k,\ell}$, so the local attraction–repulsion structure is preserved. The matching loss $\mathcal{L}_{\text{match}}$ computes $(\mu_\star, \Sigma_\star)$ from the current centers and matches these statistics to $(\mu_{\text{prior}}, \Sigma_{\text{prior}})$. We minimize $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{force}} + \mathcal{L}_{\text{match}}$ by gradient descent.

Lines 35-36 produce the released dataset in the input space. The publishing module $\mathcal{P}$ outputs x'_j so that $E(x'_j)$ is close to the optimized center $z_j^\star$. The final released dataset is $D' = \{(x'_j, y_j)\}_{j=1}^N$.

Algorithm 1 Force-invariant publishing with relaxed FID prior matching

Require: Labeled dataset $D = \{(x_j, y_j)\}_{j=1}^{N}$ and encoder $E : \mathbb{R}^D \to \mathbb{R}^d$
Require: Potential parameters P, α, ℓ_0 that define $\psi(r)$ in Eq. equation 4
Require: Repulsion scale λ_{diff} in Eq. equation 6 and exponent $q \in (1, 2]$ in Eq. equation 9
Require: Speed threshold R in Eq. equation 11
Require: Initial shell radius r_0, number of directions K, maximum evolution time $T_{\max}$, trajectory step size Δt, sampling times $\{\tau_{i,k,\ell}\}$ used to form monitor points
Require: Force budget κ used in Theorem 4.1
Require: Optimization steps T and step size η for updating $\{z_j^\star\}$
Require: Numerical constant $\varepsilon > 0$
Require: Publishing module $\mathcal{P}$ that outputs x'_j in the input space with embedding close to a target center, $E(x'_j) \approx z_j^\star$
Ensure: Published dataset $D' = \{(x'_j, y_j)\}_{j=1}^{N}$

$z_j^0 \leftarrow E(x_j), \quad j = 1, \dots, N$
Define $F_0^{(c)}(z)$ from $\{z_j^0\}$ — Eq. equation 7
Define $\omega_j^0(z)$ and $\mathcal{F}_0(z, v)$ from $\{z_j^0\}$ — Eq. equation 8,Eq. equation 9
Define $\mathcal{F}_0^*(z, p)$ — Theorem 3.2
Define $V_c^0(z)$ — Eq. equation 11
for $i = 1, \dots, N$ **do**
  Sample $\{n_{i,k}\}_{k=1}^{K}$ with $\|n_{i,k}\| = 1$
  for $k = 1, \dots, K$ **do**
    $b_{i,k}(0) \leftarrow z_i^0 + r_0 n_{i,k}$
    $\gamma_{i,k}^0 \leftarrow b_{i,k}(0)$
    for $s = 0, \dots, \lceil T_{\max}/\Delta t \rceil - 1$ **do**
      $\gamma_{i,k}^{s+1} \leftarrow \gamma_{i,k}^s + \Delta t V_{y_i}^0(\gamma_{i,k}^s) \frac{F_0^{(y_i)}(\gamma_{i,k}^s)}{\|F_0^{(y_i)}(\gamma_{i,k}^s)\|_2 + \varepsilon}$
    end for
    for $\ell = 1, \dots, L_{i,k}$ **do**
      $p_{i,k,\ell} \leftarrow \gamma_{i,k}(\tau_{i,k,\ell})$
      $G_{i,k,\ell} \leftarrow F_0^{(y_i)}(p_{i,k,\ell})$
    end for
  end for
end for
$M \leftarrow \sum_{i=1}^{N} \sum_{k=1}^{K} L_{i,k}$
Compute (μ_0, Σ_0) from $\{z_j^0\}$
Construct A from $\{p_{i,k,\ell}\}$, $\{G_{i,k,\ell}\}$, and $\{z_j^0\}$ — Eq. equation 23
Compute $(\lambda_{\min}, u_{\min})$ of $\Sigma_0^{1/2} A \Sigma_0^{1/2}$ — Eq. equation 27
$\mu_{\text{prior}} \leftarrow \mu_0, \quad \Sigma_{\text{prior}} \leftarrow \Sigma_0^{1/2} \Big(I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top \Big) \Sigma_0^{1/2}$ — Theorem 4.1
$z_j^\star \leftarrow z_j^0, \quad j = 1, \dots, N$
for $t = 1, \dots, T$ **do**
  Define $F_\star^{(c)}(z)$ from $\{z_j^\star\}$ — Eq. equation 14
  $\Delta F_{i,k,\ell} \leftarrow F_\star^{(y_i)}(p_{i,k,\ell}) - G_{i,k,\ell}$ — Eq. equation 15
  $\mathcal{L}_{\text{force}} \leftarrow \frac{1}{M} \sum_{i,k,\ell} \frac{\mathcal{F}_0(p_{i,k,\ell}, \Delta F_{i,k,\ell})^2}{\mathcal{F}_0(p_{i,k,\ell}, G_{i,k,\ell})^2}$ — Eq. equation 16
  Compute $(\mu_\star, \Sigma_\star)$ from $\{z_j^\star\}$ — Eq. equation 29
  $\mathcal{L}_{\text{match}} \leftarrow \|\mu_\star - \mu_{\text{prior}}\|_2^2 + \Big\| \Sigma_0^{-1/2} \Sigma_\star \Sigma_0^{-1/2} - \Big(I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top \Big) \Big\|_F^2$ — Eq. equation 30
  $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{force}} + \mathcal{L}_{\text{match}}$
  $z_j^\star \leftarrow z_j^\star - \eta \nabla_{z_j^\star} \mathcal{L}_{\text{total}}, \quad j = 1, \dots, N$
end for
$x'_j \leftarrow \mathcal{P}(x_j; z_j^\star), \quad j = 1, \dots, N$
$D' \leftarrow \{(x'_j, y_j)\}_{j=1}^{N}$

C PROOF OF THEOREM

Theorem 3.1 (Point-source effect with Rational Quadratic decay)**.** *Let $f_\beta : \mathbb{R}^D \to \mathbb{R}^C$ be a differentiable multi-class network satisfying the local smoothness and radial averaging assumptions stated in Appendix C. Consider a training pair (x_0, y_0) with $x_0 \in \mathbb{R}^D$ and $y_0 \in \{1, \dots, C\}$. For a sufficiently small step size $\eta > 0$, one gradient-descent update $\beta^+ = \beta - \eta \nabla_\beta \ell\big(f_\beta(x_0), y_0\big)$ produces a first-order local effect in the input space. Under the local point-source approximation, its radially averaged strength at distance $r = \|x - x_0\|$ can be approximated by the Rational Quadratic decay*

$$K_{\mathrm{RQ}}(r) = \left(1 + \frac{r^2}{2\alpha \ell_0^2}\right)^{-\alpha}, \quad r \geq 0, \tag{32}$$

for some $\alpha > 0$ and $\ell_0 > 0$.

Proof. Write the single-sample objective

$$L(\beta) = \ell\big(f_\beta(x_0), y_0\big), \tag{33}$$

and consider one gradient step

$$\beta^+ = \beta - \eta \nabla_\beta L(\beta) = \beta - \eta \nabla_\beta \ell\big(f_\beta(x_0), y_0\big). \tag{34}$$

For any $x \in \mathbb{R}^D$, define the output change

$$\Delta f(x) = f_{\beta^+}(x) - f_\beta(x) \in \mathbb{R}^C. \tag{35}$$

Let $p = \dim(\beta)$ and define the parameter Jacobian

$$J_\beta(x) = \nabla_\beta f_\beta(x) \in \mathbb{R}^{C \times p}. \tag{36}$$

Let $\delta\beta = \beta^+ - \beta$. By the update rule, $\delta\beta = -\eta \nabla_\beta L(\beta)$. Since $\beta \mapsto f_\beta(x)$ is differentiable, for each fixed x we apply a first-order Taylor expansion in β along the segment $\beta + s\delta\beta$ with $s \in [0, 1]$. There exists a remainder vector $R_\beta(x)$ such that

$$f_{\beta + \delta\beta}(x) = f_\beta(x) + J_\beta(x)\delta\beta + R_\beta(x). \tag{37}$$

Assume $\beta \mapsto f_\beta(x)$ is twice continuously differentiable on a neighborhood containing the segment $\{\beta + s\delta\beta\}_{s \in [0,1]}$ and the second derivative is bounded there. Then there exists a finite constant $B(x)$ such that

$$\|R_\beta(x)\| \leq \frac{B(x)}{2} \|\delta\beta\|^2. \tag{38}$$

Substituting $\beta + \delta\beta = \beta^+$ yields

$$\Delta f(x) = J_\beta(x)\delta\beta + R_\beta(x). \tag{39}$$

Next we compute $\delta\beta$ in closed form. Define

$$g_0 = \nabla_s \ell(s, y_0)\big|_{s = f_\beta(x_0)} \in \mathbb{R}^C. \tag{40}$$

By the chain rule,

$$\nabla_\beta L(\beta) = \nabla_\beta \ell\big(f_\beta(x_0), y_0\big) = J_\beta(x_0)^\top g_0, \tag{41}$$

so

$$\delta\beta = -\eta J_\beta(x_0)^\top g_0. \tag{42}$$

Substituting into $\Delta f(x)$ gives

$$\Delta f(x) = -\eta J_\beta(x) J_\beta(x_0)^\top g_0 + R_\beta(x). \tag{43}$$

Define the matrix kernel

$$\Theta_\beta(x, x_0) = J_\beta(x) J_\beta(x_0)^\top \in \mathbb{R}^{C \times C}, \tag{44}$$

then

$$\Delta f(x) = -\eta \Theta_\beta(x, x_0) g_0 + R_\beta(x). \tag{45}$$

Moreover,

$$\|\delta\beta\|^2 = \eta^2 \|J_\beta(x_0)^\top g_0\|^2, \quad \|R_\beta(x)\| \le \frac{B(x)}{2}\eta^2 \|J_\beta(x_0)^\top g_0\|^2. \tag{46}$$

We now define the strength of the local effect induced by the point (x_0, y_0). Using the same supervision direction g_0, define the scalar effect

$$\Delta(x) = -\langle g_0, \Delta f(x)\rangle. \tag{47}$$

Substituting the expression of $\Delta f(x)$ yields

$$\Delta(x) = \eta\langle g_0, \Theta_\beta(x, x_0) g_0\rangle - \langle g_0, R_\beta(x)\rangle. \tag{48}$$

By Cauchy–Schwarz,

$$|\langle g_0, R_\beta(x)\rangle| \le \|g_0\|\|R_\beta(x)\| \le \frac{B(x)}{2}\eta^2 \|g_0\|\|J_\beta(x_0)^\top g_0\|^2. \tag{49}$$

The same bound holds at $x = x_0$ with $B(x_0)$. Assume $\langle g_0, \Theta_\beta(x_0, x_0) g_0\rangle > 0$. For sufficiently small η, the quantity $\Delta(x_0)$ is strictly positive. We define the normalized strength

$$S(x) = \frac{\Delta(x)}{\Delta(x_0)}. \tag{50}$$

Write

$$\Delta(x) = \eta A(x) - E(x), \quad A(x) = \langle g_0, \Theta_\beta(x, x_0) g_0\rangle, \quad E(x) = \langle g_0, R_\beta(x)\rangle, \tag{51}$$

and similarly $\Delta(x_0) = \eta A(x_0) - E(x_0)$. Then

$$S(x) = \frac{A(x)}{A(x_0)} + \varepsilon_\eta(x), \tag{52}$$

where the error term $\varepsilon_\eta(x)$ satisfies the explicit bound

$$|\varepsilon_\eta(x)| \le \frac{|E(x)|}{\eta A(x_0) - |E(x_0)|} + \frac{\eta A(x)|E(x_0)|}{A(x_0)(\eta A(x_0) - |E(x_0)|)}. \tag{53}$$

Using the remainder bounds, $|E(x)|$ and $|E(x_0)|$ are both $O(\eta^2)$ with constants given above, hence $\varepsilon_\eta(x) = O(\eta)$ uniformly on any neighborhood where $B(\cdot)$ is bounded.

The theorem statement concerns decay as a function of the distance $r = \|x - x_0\|$. Since $S(x)$ can depend on the direction of $x - x_0$, we define a radial strength by averaging over directions. Let n be uniformly distributed on the unit sphere $\mathbb{S}^{D-1}$ and define

$$\bar{S}(r) = \mathbb{E}_n\big[S(x_0 + rn)\big], \quad r \ge 0. \tag{54}$$

Define

$$\kappa(x, x_0) = \frac{A(x)}{A(x_0)} = \frac{\langle g_0, \Theta_\beta(x, x_0) g_0\rangle}{\langle g_0, \Theta_\beta(x_0, x_0) g_0\rangle}, \quad \bar{\kappa}(r) = \mathbb{E}_n\big[\kappa(x_0 + rn, x_0)\big]. \tag{55}$$

Then

$$\bar{S}(r) = \bar{\kappa}(r) + \bar{\varepsilon}_\eta(r), \quad \bar{\varepsilon}_\eta(r) = \mathbb{E}_n\big[\varepsilon_\eta(x_0 + rn)\big], \tag{56}$$

and $|\bar{\varepsilon}_\eta(r)| \le \mathbb{E}_n|\varepsilon_\eta(x_0 + rn)|$.

We now study the local dependence of $\bar{\kappa}(r)$ on r near $r = 0$ and show it admits a Rational Quadratic form with suitable parameters. Define the parameter-space feature

$$\varphi(x) = J_\beta(x)^\top g_0 \in \mathbb{R}^p. \tag{57}$$

Then

$$\langle g_0, \Theta_\beta(x, x_0) g_0\rangle = \langle J_\beta(x)^\top g_0,\ J_\beta(x_0)^\top g_0\rangle = \langle \varphi(x), \varphi(x_0)\rangle, \tag{58}$$

and therefore

$$\kappa(x, x_0) = \frac{\langle \varphi(x), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2}. \tag{59}$$

Assume φ has continuous derivatives up to order four in a neighborhood of x_0. Write $h = x - x_0$. A fourth-order Taylor expansion gives

$$\kappa(x_0+h, x_0) = 1+\sum_a A_a h_a+\frac{1}{2}\sum_{a,b} B_{ab}h_a h_b+\frac{1}{6}\sum_{a,b,c} C_{abc}h_a h_b h_c+\frac{1}{24}\sum_{a,b,c,d} D_{abcd}h_a h_b h_c h_d+O(\|h\|^6), \tag{60}$$

where

$$A_a = \frac{\langle \partial_a \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2}, \quad B_{ab} = \frac{\langle \partial_{ab} \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2}, \tag{61}$$

$$C_{abc} = \frac{\langle \partial_{abc} \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2}, \quad D_{abcd} = \frac{\langle \partial_{abcd} \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2}. \tag{62}$$

Set $h = rn$ with $\|n\| = 1$ and take expectation over $n \sim \mathrm{Unif}(\mathbb{S}^{D-1})$. Using the standard spherical moment identities

$$\mathbb{E}_n[n_a] = 0, \quad \mathbb{E}_n[n_a n_b] = \frac{\delta_{ab}}{D}, \quad \mathbb{E}_n[n_a n_b n_c] = 0, \tag{63}$$

$$\mathbb{E}_n[n_a n_b n_c n_d] = \frac{\delta_{ab}\delta_{cd} + \delta_{ac}\delta_{bd} + \delta_{ad}\delta_{bc}}{D(D+2)}, \tag{64}$$

all odd-order terms vanish after averaging and we obtain

$$\bar{\kappa}(r) = 1 + \frac{r^2}{2D}\sum_a B_{aa} + \frac{r^4}{8D(D+2)}\sum_{a,b} D_{aabb} + O(r^6). \tag{65}$$

Introduce the input Laplacian acting componentwise on φ

$$\Delta_x \varphi(x_0) = \sum_{a=1}^{D} \partial_{aa}\varphi(x_0), \quad \Delta_x^2 \varphi(x_0) = \sum_{a=1}^{D}\sum_{b=1}^{D} \partial_{aabb}\varphi(x_0), \tag{66}$$

so the expansion becomes

$$\bar{\kappa}(r) = 1 + \frac{r^2}{2D}\frac{\langle \Delta_x \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2} + \frac{r^4}{8D(D+2)}\frac{\langle \Delta_x^2 \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2} + O(r^6). \tag{67}$$

Assume $\langle \Delta_x \varphi(x_0), \varphi(x_0)\rangle < 0$ so the radial strength decreases at second order. Define

$$\ell_0^2 = -\frac{D\|\varphi(x_0)\|^2}{\langle \Delta_x \varphi(x_0), \varphi(x_0)\rangle}, \tag{68}$$

which ensures

$$\frac{r^2}{2D}\frac{\langle \Delta_x \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2} = -\frac{r^2}{2\ell_0^2}. \tag{69}$$

For any $\alpha > 0$, the Rational Quadratic function

$$K_{\mathrm{RQ}}(r) = \left(1 + \frac{r^2}{2\alpha\ell_0^2}\right)^{-\alpha} \tag{70}$$

satisfies the local expansion

$$K_{\mathrm{RQ}}(r) = 1 - \frac{r^2}{2\ell_0^2} + \frac{\alpha+1}{8\alpha\ell_0^4}r^4 + O(r^6). \tag{71}$$

Define the fourth-order coefficient of $\bar{\kappa}$ as

$$c_4 = \frac{1}{8D(D+2)}\frac{\langle \Delta_x^2 \varphi(x_0), \varphi(x_0)\rangle}{\|\varphi(x_0)\|^2}. \tag{72}$$

If $c_4 > 0$, choose $\alpha > 0$ that matches the fourth-order term by solving

$$c_4 = \frac{\alpha+1}{8\alpha\ell_0^4}. \tag{73}$$

This yields

$$\alpha = \frac{1}{8c_4\ell_0^4 - 1}, \tag{74}$$

which is positive whenever $8c_4\ell_0^4 > 1$. Under this choice, $\bar{\kappa}(r) = K_{\mathrm{RQ}}(r) + O(r^6)$ as $r \to 0$.

Finally, since $\bar{S}(r) = \bar{\kappa}(r) + \bar{\varepsilon}_\eta(r)$ and $\bar{\varepsilon}_\eta(r)$ is $O(\eta)$ with the explicit bound inherited from $\varepsilon_\eta(x)$, for sufficiently small step size η and for r in a local neighborhood where the Taylor remainder $O(r^6)$ is dominated by the leading terms, the first-order local effect strength at distance $r = \|x - x_0\|$ follows the Rational Quadratic decay $K_{\mathrm{RQ}}(r)$ in equation 32 for some $\alpha > 0$ and $\ell_0 > 0$ constructed above. □

Theorem 3.2 (Finsler form and dual)**.** *Fix $1 < q \leq 2$ and let $q^* = \frac{q}{q-1}$. With $\omega_j^0(z)$ from equation 8 and $u_j^0(z)$ from equation 5, define*

$$\mathcal{F}_0(z, v) = \left(\sum_{j=1}^{N} \omega_j^0(z) |\langle v, u_j^0(z) \rangle|^q \right)^{1/q}. \tag{75}$$

Then $\mathcal{F}_0(z, v)$ is 1-homogeneous in v and satisfies the triangle inequality. If $\mathrm{span}\{u_j^0(z)\}_{j=1}^N = \mathbb{R}^d$, then $\mathcal{F}_0(z, \cdot)$ vanishes only at $v = 0$, and hence defines a location-dependent norm on the embedding space. The dual quantity $\mathcal{F}_0^*(z, p) = \sup_{v \neq 0} \frac{\langle p, v \rangle}{\mathcal{F}_0(z, v)}$ admits the equivalent coefficient form

$$\mathcal{F}_0^*(z, p) = \inf \left\{ \left(\sum_{j=1}^{N} \omega_j^0(z)^{-1/(q-1)} |a_j|^{q^*} \right)^{1/q^*} \middle| \, p = \sum_{j=1}^{N} a_j u_j^0(z) \right\}. \tag{76}$$

Proof. Fix $z \in \mathbb{R}^d$ such that $z \neq z_j^0$ for all $j = 1, \ldots, N$. From equation 8, $\widetilde{\omega}_j^0(z) = \psi(r_j^0(z))$ and

$$\omega_j^0(z) = \frac{\widetilde{\omega}_j^0(z)}{\sum_{k=1}^{N} \widetilde{\omega}_k^0(z)}.$$

Since $\psi(r) > 0$ for all $r \geq 0$, we have $\omega_j^0(z) > 0$ for every j and $\sum_{j=1}^{N} \omega_j^0(z) = 1$. For $z \neq z_j^0$, equation 5 gives

$$u_j^0(z) = \frac{z_j^0 - z}{\|z - z_j^0\|}$$

and $\|u_j^0(z)\| = 1$. The proof below only needs $q > 1$; the additional restriction $q \leq 2$ is used later when $\mathcal{F}_0(p, v)^2$ is upper bounded by a quadratic form.

Define a linear map $A_z : \mathbb{R}^d \to \mathbb{R}^N$ by

$$(A_z v)_j = \omega_j^0(z)^{1/q} \langle v, u_j^0(z) \rangle, \quad j = 1, \ldots, N. \tag{77}$$

Then

$$\|A_z v\|_q^q = \sum_{j=1}^{N} |(A_z v)_j|^q = \sum_{j=1}^{N} \omega_j^0(z) |\langle v, u_j^0(z) \rangle|^q. \tag{78}$$

Hence $\mathcal{F}_0(z, v) = \|A_z v\|_q$. Therefore

$$\mathcal{F}_0(z, \lambda v) = |\lambda| \mathcal{F}_0(z, v)$$

for any $\lambda \in \mathbb{R}$, and

$$\mathcal{F}_0(z, v_1 + v_2) = \|A_z(v_1 + v_2)\|_q = \|A_z v_1 + A_z v_2\|_q \leq \|A_z v_1\|_q + \|A_z v_2\|_q = \mathcal{F}_0(z, v_1) + \mathcal{F}_0(z, v_2) \tag{79}$$

for any $v_1, v_2 \in \mathbb{R}^d$. In particular, for $\lambda > 0$, $\mathcal{F}_0(z, \lambda v) = \lambda \mathcal{F}_0(z, v)$, so $\mathcal{F}_0$ is 1-homogeneous in the Finsler sense.

Moreover, if $\mathcal{F}_0(z, v) = 0$, then $A_z v = 0$, which implies

$$\langle v, u_j^0(z) \rangle = 0, \qquad j = 1, \ldots, N,$$

because $\omega_j^0(z)^{1/q} > 0$. If $\{u_j^0(z)\}_{j=1}^N$ spans $\mathbb{R}^d$, then the only vector orthogonal to all $u_j^0(z)$ is $v = 0$. Under this spanning condition, $\mathcal{F}_0(z, \cdot)$ is a norm for each fixed z.

We now compute the dual quantity. By positive homogeneity,

$$\mathcal{F}_0^*(z, p) = \sup_{v \neq 0} \frac{\langle p, v \rangle}{\mathcal{F}_0(z, v)} = \sup_{\mathcal{F}_0(z,v) \leq 1} \langle p, v \rangle = \sup_{\|A_z v\|_q \leq 1} \langle p, v \rangle. \tag{80}$$

Introduce an auxiliary variable $t = A_z v \in \mathbb{R}^N$ to obtain the equivalent convex program

$$\sup_{v \in \mathbb{R}^d, t \in \mathbb{R}^N} \langle p, v \rangle \quad \text{s.t.} \quad t = A_z v, \quad \|t\|_q \leq 1. \tag{81}$$

The problem is strictly feasible with respect to the inequality constraint by taking $v = 0$ and $t = 0$, so strong duality holds. With a Lagrange multiplier $s \in \mathbb{R}^N$ for the equality constraint $t - A_z v = 0$, the Lagrangian is

$$\mathcal{L}(v, t, s) = \langle p, v \rangle + \langle s, t - A_z v \rangle = \langle p - A_z^\top s, v \rangle + \langle s, t \rangle. \tag{82}$$

If $p - A_z^\top s \neq 0$, then

$$\sup_v \langle p - A_z^\top s, v \rangle = +\infty,$$

hence finiteness requires $A_z^\top s = p$. Under this condition,

$$\sup_{\|t\|_q \leq 1} \langle s, t \rangle = \|s\|_{q^*}, \quad q^* = \frac{q}{q-1}, \tag{83}$$

by Hölder's inequality and the standard equality case. Therefore the dual problem is

$$\inf_{A_z^\top s = p} \|s\|_{q^*}, \tag{84}$$

and strong duality yields

$$\mathcal{F}_0^*(z, p) = \inf_{A_z^\top s = p} \|s\|_{q^*}. \tag{85}$$

Expanding $A_z^\top s$ gives

$$A_z^\top s = \sum_{j=1}^N \omega_j^0(z)^{1/q} s_j u_j^0(z). \tag{86}$$

Let

$$a_j = \omega_j^0(z)^{1/q} s_j.$$

Then the constraint $A_z^\top s = p$ becomes

$$p = \sum_{j=1}^N a_j u_j^0(z), \tag{87}$$

and the objective satisfies

$$\|s\|_{q^*}^{q^*} = \sum_{j=1}^N |s_j|^{q^*} = \sum_{j=1}^N \omega_j^0(z)^{-q^*/q} |a_j|^{q^*} = \sum_{j=1}^N \omega_j^0(z)^{-1/(q-1)} |a_j|^{q^*}. \tag{88}$$

Here we used

$$\frac{q^*}{q} = \frac{1}{q-1}.$$

Hence the dual can be written equivalently as

$$\mathcal{F}_0^*(z, p) = \inf \left\{ \left(\sum_{j=1}^N \omega_j^0(z)^{-1/(q-1)} |a_j|^{q^*} \right)^{1/q^*} \middle| p = \sum_{j=1}^N a_j u_j^0(z) \right\}. \tag{89}$$

This proves the claimed coefficient form of the dual gauge. □

From monitor-point force mismatch to a covariance surrogate.

Let the monitor points $p_{i,k,\ell}$ and the recorded original forces $G_{i,k,\ell} = F_0^{(y_i)}(p_{i,k,\ell})$ be fixed as in Section 3, and let the original geometry $\mathcal{F}_0$ be fixed through $\omega_m^0(p)$ and $u_m^0(p)$. In the main text, Δ_{force} denotes the monitor-point force mismatch used to state the ideal constrained problem. For the covariance relaxation, we use a weighted version of this mismatch:

$$\widetilde{\Delta}_{\mathrm{force}}(\{z_j\}) = \frac{N^2}{M} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\ell=1}^{L_{i,k}} \frac{L_{i,k}}{\mathcal{F}_0(p_{i,k,\ell}, G_{i,k,\ell})^2}\, \mathcal{F}_0(p_{i,k,\ell}, \Delta F_{i,k,\ell})^2, \quad M = \sum_{i=1}^{N} \sum_{k=1}^{K} L_{i,k}, \tag{90}$$

where

$$\Delta F_{i,k,\ell} = F^{(y_i)}(p_{i,k,\ell}; \{z_j\}) - F_0^{(y_i)}(p_{i,k,\ell}).$$

The quantity $\widetilde{\Delta}_{\mathrm{force}}$ is not an additional training objective. It is used only in the covariance-level relaxation. It is constructed from the same monitor-point force differences as $\mathcal{L}_{\mathrm{force}}$, but includes the scaling and weights needed to express the local linearized force change as a trace constraint on the covariance shift.

Let $W(p)$ be defined by equation 19, and let $J_{i,k,\ell}^0$ be defined by equation 22. Using the first-order force linearization and the quadratic upper bound in equation 18, we construct the matrix A in equation 23. We then use

$$\mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big) \le \kappa, \quad \Sigma - \Sigma_0 \succeq 0, \tag{91}$$

as the covariance-level quadratic surrogate in equation 24. Here Σ_0 is the empirical covariance of the original centers and Σ is the empirical covariance of the candidate centers.

Derivation of the surrogate. Start from the weighted monitor-point mismatch in equation 90:

$$\widetilde{\Delta}_{\mathrm{force}}(\{z_j\}) = \frac{N^2}{M} \sum_{i,k,\ell} \frac{L_{i,k}}{\mathcal{F}_0(p_{i,k,\ell}, G_{i,k,\ell})^2}\, \mathcal{F}_0(p_{i,k,\ell}, \Delta F_{i,k,\ell})^2. \tag{92}$$

For any monitor point p and any $v \in \mathbb{R}^d$, equation 18 gives

$$\mathcal{F}_0(p, v)^2 \le v^\top W(p) v, \quad W(p) = \sum_{m=1}^{N} \omega_m^0(p) u_m^0(p) u_m^0(p)^\top. \tag{93}$$

Applying this bound at each $p_{i,k,\ell}$ yields

$$\mathcal{F}_0(p_{i,k,\ell}, \Delta F_{i,k,\ell})^2 \le \Delta F_{i,k,\ell}^\top W(p_{i,k,\ell}) \Delta F_{i,k,\ell}. \tag{94}$$

Therefore,

$$\widetilde{\Delta}_{\mathrm{force}}(\{z_j\}) \le \frac{N^2}{M} \sum_{i,k,\ell} \frac{L_{i,k}}{\mathcal{F}_0(p_{i,k,\ell}, G_{i,k,\ell})^2}\, \Delta F_{i,k,\ell}^\top W(p_{i,k,\ell}) \Delta F_{i,k,\ell}. \tag{95}$$

We next connect the local force change to second-order statistics. Under the first-order linearization around the original centers, and under the fixed-mean covariance relaxation, we represent the effective force change at $p_{i,k,\ell}$ as

$$\Delta F_{i,k,\ell} \approx J_{i,k,\ell}^0 \varepsilon, \tag{96}$$

where $\varepsilon \in \mathbb{R}^d$ is a zero-mean effective center perturbation with covariance

$$\mathbb{E}[\varepsilon \varepsilon^\top] = \Sigma - \Sigma_0, \qquad \Sigma - \Sigma_0 \succeq 0.$$

Substituting this local approximation into the quadratic form gives

$$\Delta F_{i,k,\ell}^\top W(p_{i,k,\ell}) \Delta F_{i,k,\ell} \approx \varepsilon^\top (J_{i,k,\ell}^0)^\top W(p_{i,k,\ell}) J_{i,k,\ell}^0 \varepsilon. \tag{97}$$

For any symmetric matrix Q, the identity

$$\mathbb{E}\big[\varepsilon^\top Q \varepsilon\big] = \mathrm{Tr}\big(Q \mathbb{E}[\varepsilon \varepsilon^\top]\big) \tag{98}$$

follows from

$$\varepsilon^\top Q\varepsilon = \mathrm{Tr}(Q\varepsilon\varepsilon^\top)$$

and the linearity of trace and expectation. Applying this identity with

$$Q = (J^0_{i,k,\ell})^\top W(p_{i,k,\ell}) J^0_{i,k,\ell}$$

yields

$$\mathbb{E}\left[\Delta F^\top_{i,k,\ell} W(p_{i,k,\ell}) \Delta F_{i,k,\ell}\right] \approx \mathrm{Tr}\left((J^0_{i,k,\ell})^\top W(p_{i,k,\ell}) J^0_{i,k,\ell}(\Sigma - \Sigma_0)\right). \tag{99}$$

Combining the monitor-point weights and using the linearity of trace gives

$$\begin{aligned}\mathbb{E}\big[\widetilde{\Delta}_{\mathrm{force}}(\{z_j\})\big] &\lesssim \mathrm{Tr}\left(\frac{N^2}{M}\sum_{i,k,\ell}\frac{L_{i,k}}{\mathcal{F}_0(p_{i,k,\ell}, G_{i,k,\ell})^2}(J^0_{i,k,\ell})^\top W(p_{i,k,\ell}) J^0_{i,k,\ell}(\Sigma - \Sigma_0)\right)\\ &= \mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big),\end{aligned} \tag{100}$$

where the last equality uses the definition of A in equation 23. Thus,

$$\mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big)$$

is a covariance-level upper surrogate for the expected weighted monitor-point force mismatch under the local linearization. We therefore impose

$$\mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big) \le \kappa, \quad \Sigma - \Sigma_0 \succeq 0, \tag{101}$$

as a sufficient condition for controlling this surrogate. This gives the covariance-level condition in equation 24. □

Theorem 4.1 (Prior covariance under fixed mean)**.** *Assume $\Sigma_0 \succ 0$, $\mu_{\mathrm{prior}} = \mu_0$, and $B = \Sigma_0^{1/2} A \Sigma_0^{1/2} \succ 0$. For any $\kappa > 0$, an optimizer of equation 26 is*

$$\Sigma_{\mathrm{prior}} = \Sigma_0^{1/2}\Big(I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top\Big)\Sigma_0^{1/2}. \tag{102}$$

We keep the prior mean fixed as $\mu_{\mathrm{prior}} = \mu_0$ and allocate the surrogate force budget to the covariance shift.

Proof. Write the feasible set in equation 26 as

$$\mathcal{C} = \Big\{\Sigma \in \mathbb{S}^d_+ \;\Big|\; \Sigma - \Sigma_0 \succeq 0, \mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big) \le \kappa\Big\}. \tag{103}$$

Since $\Sigma_0 \succ 0$ and $\Sigma_0^{-1/2}$ appears in equation 25, define the whitened variable

$$X = \Sigma_0^{-1/2}\Sigma\Sigma_0^{-1/2}, \qquad \Sigma = \Sigma_0^{1/2} X \Sigma_0^{1/2}. \tag{104}$$

Then

$$\Sigma - \Sigma_0 = \Sigma_0^{1/2}(X - I)\Sigma_0^{1/2}, \tag{105}$$

so $\Sigma - \Sigma_0 \succeq 0$ is equivalent to $X - I \succeq 0$. The trace constraint becomes

$$\mathrm{Tr}\big(A(\Sigma - \Sigma_0)\big) = \mathrm{Tr}\big(\Sigma_0^{1/2} A \Sigma_0^{1/2}(X - I)\big) = \mathrm{Tr}\big(B(X - I)\big), \quad B = \Sigma_0^{1/2} A \Sigma_0^{1/2}. \tag{106}$$

The objective equation 25 depends on Σ only through X and equals

$$\mathcal{J}(\Sigma) = \mathrm{Tr}\big(X + I - 2X^{1/2}\big) = \mathrm{Tr}\big((X^{1/2} - I)^2\big). \tag{107}$$

Hence equation 26 is equivalent to

$$\max_X \mathrm{Tr}\big((X^{1/2} - I)^2\big) \quad \text{s.t.} \quad X - I \succeq 0, \mathrm{Tr}\big(B(X - I)\big) \le \kappa. \tag{108}$$

Let $\Delta = X - I \succeq 0$. Then the constraints become $\mathrm{Tr}(B\Delta) \le \kappa$, and the objective becomes $\mathrm{Tr}\big(((I + \Delta)^{1/2} - I)^2\big)$. Let the eigenvalues of X be $\lambda_m = 1 + t_m$ with $t_m \ge 0$. Then

$$\mathrm{Tr}\big((X^{1/2} - I)^2\big) = \sum_{m=1}^d \phi(t_m), \quad \phi(t) = (\sqrt{1 + t} - 1)^2. \tag{109}$$

Let B have minimum eigenpair $(\lambda_{\min}, u_{\min})$ as in equation 27. For any unit vector u, we have $u^\top B u \geq \lambda_{\min}$. Writing $\Delta = \sum_{m=1}^{d} t_m u_m u_m^\top$, we have

$$\mathrm{Tr}(B\Delta) = \sum_{m=1}^{d} t_m u_m^\top B u_m \geq \lambda_{\min} \sum_{m=1}^{d} t_m. \tag{110}$$

Thus any feasible Δ satisfies

$$\sum_{m=1}^{d} t_m \leq \frac{\kappa}{\lambda_{\min}}. \tag{111}$$

Next we show that ϕ is superadditive on $\mathbb{R}_+$, namely for all $a, b \geq 0$,

$$\phi(a) + \phi(b) \leq \phi(a+b). \tag{112}$$

Using $\phi(t) = t + 2 - 2\sqrt{1+t}$, this inequality is equivalent to

$$\sqrt{1+a} + \sqrt{1+b} \geq 1 + \sqrt{1+a+b}. \tag{113}$$

Let $u = \sqrt{1+a} \geq 1$ and $v = \sqrt{1+b} \geq 1$. Squaring the inequality $u + v - 1 \geq \sqrt{u^2 + v^2 - 1}$ yields

$$(u+v-1)^2 \geq u^2 + v^2 - 1 \quad \Longleftrightarrow \quad 2(u-1)(v-1) \geq 0, \tag{114}$$

which holds. Repeatedly applying superadditivity gives, for any nonnegative $\{t_m\}$,

$$\sum_{m=1}^{d} \phi(t_m) \leq \phi\Big(\sum_{m=1}^{d} t_m\Big) \leq \phi\Big(\frac{\kappa}{\lambda_{\min}}\Big). \tag{115}$$

Therefore $\mathcal{J}(\Sigma) \leq \phi(\kappa/\lambda_{\min})$ for every feasible Σ.

We now construct a feasible point attaining this upper bound. Set

$$X^\star = I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top. \tag{116}$$

Then $X^\star - I \succeq 0$, and

$$\mathrm{Tr}\big(B(X^\star - I)\big) = \frac{\kappa}{\lambda_{\min}} \mathrm{Tr}\big(B u_{\min} u_{\min}^\top\big) = \frac{\kappa}{\lambda_{\min}} u_{\min}^\top B u_{\min} = \kappa. \tag{117}$$

Hence $X^\star$ is feasible. Since $u_{\min} u_{\min}^\top$ is rank one, $X^\star$ has eigenvalues $1 + \kappa/\lambda_{\min}$ and 1 with multiplicity $d-1$, so

$$\mathrm{Tr}\big((X^{\star 1/2} - I)^2\big) = \phi\Big(\frac{\kappa}{\lambda_{\min}}\Big), \tag{118}$$

which matches the upper bound. Thus $X^\star$ is optimal. Mapping back to Σ gives

$$\Sigma_{\mathrm{prior}} = \Sigma_0^{1/2} X^\star \Sigma_0^{1/2} = \Sigma_0^{1/2}\Big(I + \frac{\kappa}{\lambda_{\min}} u_{\min} u_{\min}^\top\Big)\Sigma_0^{1/2}. \tag{119}$$

Finally, equation 26 optimizes only the covariance term $\mathcal{J}(\Sigma)$ under the covariance-level surrogate derived from the fixed monitor-point geometry in Section 3. The mean is not optimized in this problem. We fix it by design as $\mu_{\mathrm{prior}} = \mu_0$, so the surrogate budget is allocated to the covariance shift. □

D Experimental Setup

Environment. All experiments were conducted on a Linux server running Ubuntu 22.04.5 LTS. The server is equipped with two AMD EPYC 9555 64-Core processors, 2.2 TB of system memory, and 6 NVIDIA H200 NVL GPUs, each with approximately 143 GB of GPU memory. Our implementation is based on Python 3.11.14, PyTorch 2.9.1, and Torchvision 0.24.1, with CUDA 12.8 and cuDNN support enabled.

Datasets and Evaluation Protocol. We conduct experiments on two image datasets, CIFAR-100 and Tiny-ImageNet. CIFAR-100 contains 50,000 training images and 10,000 test images. Tiny-ImageNet contains 100,000 training images and 10,000 validation images. We evaluate our method

from both discriminative and generative perspectives. For classification evaluation, we train classifiers on each published dataset with a fixed training schedule and report the final accuracy on the official test set for CIFAR-100 and the official validation set for Tiny-ImageNet. For generative evaluation, we train a diffusion model with DDIM sampling on each target dataset and report FID and LPIPS. In all FID/LPIPS evaluations, we generate 5,000 synthetic samples as the fake set and use the full real training set of the corresponding dataset as the real reference set.

Poisoning Baselines. We compare our method against four poisoning-oriented baselines, including BiBad, UIB, Model Collapse, and VillanDiffusion. For poisoning experiments, we consider four poisoning ratios, namely 1%, 2%, 5%, and 10%, on both CIFAR-100 and Tiny-ImageNet. Classification performance is evaluated using ResNet-18 and VGG-16. Specifically, ResNet-18 is trained for 100 epochs with batch size 256 and learning rate 0.1 on both CIFAR-100 and Tiny-ImageNet. For VGG-16, we train for 100 epochs with batch size 256, using learning rate 0.01 on CIFAR-100 and 0.05 on Tiny-ImageNet.

Among these poisoning baselines, Model Collapse requires additional clarification. Its main idea is to recursively train a new generative model on samples synthesized by a previously trained generative model, so that the generated distribution gradually drifts away from the original data distribution after multiple rounds of self-training. In our experiments, we use samples generated by the model after five rounds of such recursive training, and replace 1%, 2%, 5%, or 10% of the original training data with these generated samples to construct the corresponding poisoning settings.

Distilled Baselines. We further compare our method with distilled-data baselines, including SRe^{2}L, RDED, CDA, and D3HR. Since dataset distillation itself is not the main focus of this work, we do not propose a new distillation algorithm from scratch. Instead, we adopt the compressed datasets generated by D3HR as the base distilled datasets, and then apply our publishing mechanism to 5% of the compressed samples. The resulting published compressed dataset is treated as our method in the distilled-data comparison. For CIFAR-100, we use compressed datasets with IPC 10 and IPC 50 generated with 1k distillation iterations. For Tiny-ImageNet, we use compressed datasets with IPC 10 and IPC 50 generated with 4k distillation iterations. Classification accuracy in the distilled-data comparison is evaluated using ResNet-18.

Generative Model Evaluation. To evaluate the distributional quality of the poisoned or published datasets, we train a diffusion model with DDIM sampling on each constructed dataset. Unless otherwise specified, the diffusion model is trained for 100 epochs with batch size 64 and learning rate 2×10^{-4}. We use 1,000 diffusion timesteps during training and 50 DDIM sampling steps during inference. After training, we sample 5,000 images and compute FID and LPIPS against the full real training set of the corresponding dataset.

Implementation Details. Our publishing method operates by selecting a subset of samples, modifying them in the latent space, and decoding them back into the image space. Unless otherwise specified, we use the default hyperparameter settings listed in Table 3. These settings are kept fixed across the main experiments unless otherwise noted.

Our method is implemented as a latent-space publishing framework built on top of a learned encoder-decoder pipeline. Given an input image x, we first map it into a latent representation $z = E(x)$, where E is a ResNet-18-based encoder. More specifically, the encoder is constructed from a classifier backbone with a trainable projection head, so that the backbone preserves semantic structure while the projection head maps images into a compact latent space of dimension 64. The decoder D reconstructs an image from the latent code, i.e., $\hat{x} = D(z)$. Therefore, our reconstruction module is a learned autoencoding-style model rather than an optimization-based reconstruction procedure.

To obtain a usable latent space for publishing, we train the encoder-decoder pair with a reconstruction objective that combines multiple terms. In particular, the reconstruction loss includes pixel-level mean squared error and L_1 loss between the reconstructed image and the original image. In addition, we include a perceptual term, a classification-consistency term based on cross-entropy with the class label, and a knowledge-distillation term that aligns the classifier predictions of reconstructed images with those of the original images. This design encourages the latent representation to preserve both low-level appearance and high-level semantic information.

Our publishing module P is implemented as a learnable multilayer perceptron that operates in the latent space. Given a latent code z, we first optionally apply a stochastic latent perturbation to

obtain an intermediate code $\tilde{z}$, and then feed it into the publishing module to produce a modified latent representation $z' = P(\tilde{z})$. The final published image is obtained by decoding z' through the decoder. More specifically, after obtaining the latent representation $z = E(x)$ for a selected sample, we treat z as its embedding center in the latent space. The publishing operation is then performed on this latent center to produce a modified representation z'. To map the modified latent center back to the input space, we pass z' through the decoder and obtain the corresponding published image $x' = D(z')$. Therefore, the transition from embedding centers back to input-space data is fully realized through the learned decoder, rather than by solving a separate optimization problem in pixel space. In the current implementation, P is a feed-forward network with three hidden layers of width 256.

A key design choice of our method is that modifications are applied only to a restricted subset of training samples. Under a given modified-data budget, we randomly sample a subset of indices from the training set according to the target ratio (e.g., 1%, 2%, 5%, or 10%). The selected samples are treated as controllable samples and are passed through the publishing pipeline, while all unselected samples remain unchanged and keep their original image values. After decoding, the resulting image is clamped to the valid image range and is used to replace the corresponding original sample in the training set. Hence, the final modified dataset is a mixed set composed of published samples and untouched original samples.

Our current implementation does not impose an explicit hard constraint on the image-space perturbation magnitude, such as an L_p-norm bound between the original and modified images. Instead, the modification process is controlled implicitly through latent-space regularization, reconstruction quality, and the matching-based objective used to train the publishing module. In practice, the decoder output is clamped to the valid image range, and the latent manipulation is regularized so that the published samples remain structurally meaningful while still inducing measurable changes in downstream classification and generation behavior.

For the ablation study, we further instantiate simplified variants of this framework. In the Random variant, the learnable publishing module is removed, and the selected latent codes are directly perturbed by random latent noise before decoding. In the NoMatch variant, the learnable publishing module is retained, but the matching term in the publishing objective is removed. The Full variant corresponds to the complete method with both learnable latent publishing and matching regularization.

Table 3: Default hyperparameter settings used in our method.

Parameter	**Value**
Latent embedding dimension	64
Latent randomization mode	Gaussian
Latent noise scale	0.25
Number of directional samples k_{dir}	16
Number of local samples l_{samples}	3
Maximum simulation time $t_{\max}$	0.2
Simulation step size τ	0.01
Publishing epochs g_{epochs}	10
Publishing learning rate g_{lr}	5×10^{-4}
Publishing batch size g_{batch}	128
Reconstruction epochs $recon_{\text{epochs}}$	80
Reconstruction learning rate $recon_{\text{lr}}$	2×10^{-4}
Reconstruction batch size $recon_{\text{batch}}$	128
Diffusion training epochs	100
Diffusion batch size	64
Diffusion learning rate	2×10^{-4}
Diffusion timesteps	1000
DDIM sampling steps	50
Number of generated samples for FID/LPIPS	5000

E Additional Experiments

Poisoning-budget classification results. Tables 4–7 report the downstream classification accuracy under different modified-data budgets. GRDP keeps stable accuracy across datasets and models. On CIFAR-100 with VGG16, GRDP is the best at all budgets, from 62.11% at 1% to 60.04% at 10%. On Tiny ImageNet with VGG16, GRDP also stays clearly higher than the baselines, with 56.14% at 5% and 55.60% at 10%. On ResNet-18, GRDP is close to the best baseline in all settings. For example, on CIFAR-100 at 5%, GRDP reaches 76.47%, only 0.14% lower than VillanDiffusion. These results show that GRDP does not rely on random corruption. The force-invariance constraint keeps the local class structure stable, so classifiers trained on the published data can still learn useful boundaries.

Table 4: Classification accuracy (%) on CIFAR-100 with ResNet-18 under different modified-data budgets.

Method	1%	2%	5%	10%
BiBad	76.25	75.86	74.75	72.76
UIB	76.24	76.14	75.78	74.66
Model Collapse	75.19	74.88	74.72	74.33
VillanDiffusion	77.19	77.18	**76.61**	**76.57**
GRDP	**77.28**	**77.27**	76.47	75.27

Table 5: Classification accuracy (%) on Tiny ImageNet with ResNet-18 under different modified-data budgets.

Method	1%	2%	5%	10%
BiBad	64.34	63.71	62.79	60.40
UIB	64.63	64.28	63.97	63.53
Model Collapse	63.77	63.15	62.58	61.03
VillanDiffusion	65.26	65.02	64.66	**64.28**
GRDP	**65.59**	**65.03**	**64.71**	63.44

Table 6: Classification accuracy (%) on CIFAR-100 with VGG16 under different modified-data budgets.

Method	1%	2%	5%	10%
BiBad	60.93	60.60	59.44	59.06
UIB	59.86	59.62	59.15	58.88
Model Collapse	61.08	60.62	60.14	59.41
VillanDiffusion	60.67	60.39	60.23	59.96
GRDP	**62.11**	**61.83**	**61.35**	**60.04**

Table 7: Classification accuracy (%) on Tiny ImageNet with VGG16 under different modified-data budgets.

Method	1%	2%	5%	10%
BiBad	55.56	55.24	54.00	50.74
UIB	46.74	46.72	44.71	42.92
Model Collapse	46.92	46.31	44.36	42.15
VillanDiffusion	45.99	45.66	44.34	41.38
GRDP	**57.66**	**56.92**	**56.14**	**55.60**

Poisoning-budget generation results. Tables 8–11 show the DDIM-FID and DDIM-LPIPS results under different modified-data budgets. GRDP gives the strongest generative degradation in all settings. On CIFAR-100, GRDP increases DDIM-FID from 50.28 at 1% to 56.95 at 10%, while the best baseline only reaches 41.73. On Tiny ImageNet, GRDP reaches 82.01 DDIM-FID at 5% and 84.76 at 10%, much higher than Model Collapse at 52.73 and 53.88. GRDP also gives the highest LPIPS values, such as 0.265 on CIFAR-100 and 0.638 on Tiny ImageNet. This shows that GRDP changes the global feature statistics used by diffusion models, so generated samples move farther from the original data distribution.

Table 8: DDIM-FID on CIFAR-100 under different modified-data budgets. Higher values indicate stronger degradation for generative training.

Method	1%	2%	5%	10%
BiBad	31.32	31.41	32.57	32.85
UIB	29.23	29.24	29.35	31.06
Model Collapse	37.43	39.55	39.88	41.73
VillanDiffusion	29.34	29.53	29.22	29.08
GRDP	**50.28**	**51.75**	**53.84**	**56.95**

Table 9: DDIM-LPIPS on CIFAR-100 under different modified-data budgets. Higher values indicate stronger perceptual mismatch.

Method	1%	2%	5%	10%
BiBad	0.221	0.230	0.231	0.234
UIB	0.242	0.242	0.243	0.246
Model Collapse	0.245	0.246	0.251	0.253
VillanDiffusion	0.244	0.247	0.249	0.251
GRDP	**0.249**	**0.252**	**0.265**	**0.265**

Table 10: DDIM-FID on Tiny ImageNet under different modified-data budgets. Higher values indicate stronger degradation for generative training.

Method	1%	2%	5%	10%
BiBad	38.73	39.55	41.85	42.84
UIB	38.24	38.91	40.33	40.51
Model Collapse	48.13	49.19	52.73	53.88
VillanDiffusion	39.11	40.21	41.37	42.78
GRDP	**75.09**	**76.56**	**82.01**	**84.76**

Table 11: DDIM-LPIPS on Tiny ImageNet under different modified-data budgets. Higher values indicate stronger perceptual mismatch.

Method	1%	2%	5%	10%
BiBad	0.577	0.577	0.581	0.584
UIB	0.582	0.581	0.593	0.594
Model Collapse	0.588	0.592	0.592	0.611
VillanDiffusion	0.579	0.584	0.586	0.591
GRDP	**0.620**	**0.629**	**0.627**	**0.638**

Distilled-data classification results. Tables 12 and 13 compare GRDP with distilled-data baselines under IPC=10 and IPC=50. On CIFAR-100, GRDP obtains 49.32% at IPC=10 and 65.25% at IPC=50, close to the best results of 49.85% and 65.70%. On Tiny ImageNet, GRDP reaches 42.88% at IPC=10 and 56.21% at IPC=50, close to D3HR at 43.24% and higher than D3HR at IPC=50. These results show that GRDP can keep classification utility comparable to strong distillation methods, although it is not designed only for dataset compression.

Table 12: Classification accuracy (%) on CIFAR-100 under distilled-data baselines with different IPC values.

Method	IPC=10	IPC=50
SRe^{2}L	27.05	50.98
RDED	42.62	62.44
CDA	**49.85**	63.80
D3HR	49.40	**65.70**
GRDP	49.32	65.25

Table 13: Classification accuracy (%) on Tiny ImageNet under distilled-data baselines with different IPC values.

Method	IPC=10	IPC=50
SRe^{2}L	27.79	48.44
RDED	41.35	**57.98**
CDA	21.33	48.86
D3HR	**43.24**	56.17
GRDP	42.88	56.21

Distilled-data generation results. Tables 14–17 report the generative evaluation for distilled-data baselines. GRDP achieves the highest DDIM-FID and DDIM-LPIPS on both datasets and both IPC values. On CIFAR-100, GRDP reaches 148.26 DDIM-FID at IPC=10 and 94.77 at IPC=50, both higher than D3HR. On Tiny ImageNet, GRDP reaches 176.02 DDIM-FID at IPC=10 and 97.89 at IPC=50. For LPIPS, GRDP also obtains the highest values, with 0.212 on CIFAR-100 and 0.622 on Tiny ImageNet at IPC=50. This shows the difference between GRDP and distillation methods: distillation mainly keeps compact data useful for classification, while GRDP also directly optimizes the published features to reduce their usefulness for generative training.

Table 14: DDIM-FID on CIFAR-100 under distilled-data baselines with different IPC values.

Method	IPC=10	IPC=50
SRe^{2}L	77.78	25.58
RDED	78.14	26.53
CDA	74.63	26.51
D3HR	147.75	92.33
GRDP	**148.26**	**94.77**

Table 15: DDIM-LPIPS on CIFAR-100 under distilled-data baselines with different IPC values.

Method	IPC=10	IPC=50
SRe^{2}L	0.146	0.162
RDED	0.141	0.159
CDA	0.143	0.163
D3HR	0.195	0.201
GRDP	**0.198**	**0.212**

Table 16: DDIM-FID on Tiny ImageNet under distilled-data baselines with different IPC values.

Method	IPC=10	IPC=50
SRe^{2}L	174.51	75.70
RDED	172.32	74.84
CDA	122.08	78.16
D3HR	153.01	97.00
GRDP	**176.02**	**97.89**

Table 17: DDIM-LPIPS on Tiny ImageNet under distilled-data baselines with different IPC values.

Method	IPC=10	IPC=50
SRe^{2}L	0.561	0.567
RDED	0.553	0.571
CDA	0.535	0.579
D3HR	0.569	0.601
GRDP	**0.602**	**0.622**

Ablation results. Table 18 reports the ablation study on CIFAR-100 under the 5% budget. The raw dataset has high accuracy, with 77.30% on ResNet-18 and 62.55% on VGG16, but its DDIM-FID is only 30.27. Random shift increases DDIM-FID to 53.31, but reduces VGG16 accuracy to 57.10%, showing that direct perturbation damages discriminative structure. Without FID prior matching, ResNet-18 accuracy remains high at 77.18%, but DDIM-FID drops to 40.86. Full GRDP keeps strong classification accuracy, with 76.47% on ResNet-18 and 61.35% on VGG16, and gives the best generation degradation, with 53.84 DDIM-FID and 0.265 DDIM-LPIPS. This confirms that force invariance preserves classification utility, while FID prior matching provides the main pressure for generative protection.

Table 18: Ablation study on CIFAR-100 under the 5% modified-data budget.

Variant	ResNet-18 Acc	VGG16 Acc	DDIM-FID	DDIM-LPIPS
Raw Dataset	**77.30**	**62.55**	30.27	0.239
5% Random Shift	76.36	57.10	53.31	0.263
5% w/o FID Matching	77.18	57.46	40.86	0.239
5% Full GRDP	76.47	61.35	**53.84**	**0.265**

E.1 Evaluation on an ImageNet-1K Subset

We further evaluate GRDP on a larger-scale ImageNet-1K subset. We randomly use 50 training images per class, resulting in 50K training images. We evaluate classification utility using a pretrained ResNet-18 and generative utility using a shared 64×64 unconditional DDIM. The DDIM is trained for 100 epochs using Adam with a learning rate of 2×10^{-4}, batch size 32, and 1,000 diffusion steps. We generate 10K images using 50-step deterministic DDIM sampling with $\eta = 0$. FID is computed against a fixed set of 10K clean ImageNet validation images, and LPIPS-Alex is evaluated over 2K pairs.

Table 19 shows that GRDP maintains the highest top-5 classification accuracy among all compared methods at both modification ratios. At the 5% ratio, GRDP achieves 52.12% top-5 accuracy, compared with 51.79% for villanDiffusion, 46.26% for BiBad, and 43.11% for UIB. At the 10% ratio,

Table 19: Results on the ImageNet-1K subset under 5% and 10% modification ratios. Higher Acc-top5 indicates better classification utility, while higher FID and LPIPS indicate a larger discrepancy in generative output.

Method	Ratio	Acc-top5 ↑	FID ↑	LPIPS ↑
UIB	5%	43.11%	100.761	0.535
villanDiffusion	5%	51.79%	103.038	0.529
BiBad	5%	46.26%	**106.447**	0.560
GRDP	5%	**52.12%**	104.943	**0.566**
UIB	10%	33.71%	117.726	0.518
villanDiffusion	10%	47.49%	**120.914**	0.572
BiBad	10%	42.04%	115.696	0.538
GRDP	10%	**51.37%**	120.124	**0.573**

Table 20: Image-space fidelity between modified published samples and their corresponding raw training samples. Results are averaged over all modified samples under a 5% modification ratio.

Dataset	Method	Raw-Pub LPIPS	PSNR	SSIM
CIFAR-100	GRDP	0.251	13.645	0.144
CIFAR-100	UIB	0.146	11.444	0.402
Tiny ImageNet	GRDP	0.663	12.355	0.076
Tiny ImageNet	UIB	0.591	8.629	0.051

the advantage becomes larger, with GRDP achieving 51.37% accuracy while the other methods range from 33.71% to 47.49%.

GRDP also produces a large generative distribution discrepancy. It achieves the highest LPIPS at both 5% and 10% modification ratios and the second-highest FID in both settings. These results show that the classification–generation trade-off observed in the main experiments also holds on the larger ImageNet-1K subset.

E.2 Fidelity of Published Images

We additionally measure the image-space difference between each modified training image and its exact raw counterpart. The evaluation is conducted only on modified samples. Under the 5% modification ratio, this corresponds to 2,500 image pairs for CIFAR-100 and 5,000 pairs for Tiny ImageNet. Images are evaluated in the [0,1] RGB range without augmentation. Raw-Pub LPIPS is computed using LPIPS-v0.1 with an AlexNet backbone, while PSNR and SSIM are computed with `data_range` $= 1.0$. SSIM is evaluated over RGB channels. GRDP and UIB use the same evaluation implementation and preprocessing. For GRDP, we use the Gaussian randomization configuration with $\sigma = 0.25$.

Table 20 reports the raw-to-published image differences under several complementary metrics. On CIFAR-100, GRDP has a larger perceptual difference measured by LPIPS than UIB (0.251 vs. 0.146), while obtaining a higher PSNR (13.645 vs. 11.444) and a lower SSIM (0.144 vs. 0.402). On Tiny ImageNet, GRDP also produces a higher LPIPS (0.663 vs. 0.591), together with higher PSNR and SSIM. The different behavior of LPIPS, PSNR, and SSIM shows that the publishing conversion cannot be characterized by a single image-space distortion measure, since these metrics capture different aspects of perceptual, pixel-level, and structural similarity.

E.3 Robustness Across Generative Models

We further evaluate whether the degradation of generative capability generalizes across different generator architectures. Experiments are conducted on CIFAR-100 using two representative generators, DDIM and StyleGAN-XL. For each generator, we compare models trained on the original raw dataset, a UIB-published dataset with a 5% modification ratio, and a GRDP-published dataset with the same ratio. Within each generator family, all methods use the same model architecture,

Table 21: Generator robustness on CIFAR-100. Higher FID and KID indicate a larger distributional discrepancy between generated and real samples.

Generator	Training Data	FID ↑	KID ↑
DDIM	Raw	49.708	0.0483
DDIM	UIB 5%	29.350	0.0419
DDIM	**GRDP 5%**	**53.836**	**0.0524**
StyleGAN-XL	Raw	77.335	0.008869
StyleGAN-XL	UIB 5%	70.118	0.006765
StyleGAN-XL	**GRDP 5%**	**83.016**	**0.010860**

Table 22: Hyperparameter sensitivity on CIFAR-100 with a 5% modification ratio. The default configuration is $\alpha = 2$, $K = 16$, and $\sigma = 0.25$.

Parameter	Value	Acc ↑	FID ↑
α	1	**76.81%**	52.251
	2 (default)	76.47%	53.836
	4	75.51%	**54.558**
K	8	**76.66%**	49.987
	16 (default)	76.47%	53.836
	32	75.12%	**54.179**
σ	0.15	75.13%	49.189
	0.25 (default)	**76.47%**	**53.836**
	0.35	74.68%	51.997

optimization settings, training budget, data augmentation, random seed, sampling procedure, and evaluation pipeline. Only the training data are changed.

For DDIM, we train at 32×32 resolution for 100 epochs using Adam with a learning rate of 2×10^{-4}, batch size 32, and 1,000 diffusion steps. Samples are generated using 50-step DDIM sampling with $\eta = 0$. For GRDP, 2,500 out of 50,000 CIFAR-100 training images are modified using the Gaussian publishing configuration with $\sigma = 0.25$.

For StyleGAN-XL, we use a class-conditional StyleGAN3-T generator with a projected discriminator. The discriminator uses DeiT-Base distilled and EfficientNet-Lite0 as frozen feature backbones. All models are trained for 2,000 kimg with a batch size of 64, horizontal flipping, and the same random seed. GRDP and UIB use the same 5% modification ratio.

As shown in Table 21, GRDP consistently increases both FID and KID across the two generator families. For DDIM, GRDP increases the FID from 49.708 on the raw dataset to 53.836 and the KID from 0.0483 to 0.0524. In contrast, UIB reduces both metrics to 29.350 and 0.0419. A similar trend is observed for StyleGAN-XL, where GRDP increases the FID from 77.335 to 83.016 and the KID from 0.008869 to 0.010860, while UIB again produces lower values than the raw dataset.

These results show that the reduction in generative utility is not specific to DDIM. GRDP produces a consistent distributional degradation for both diffusion-based and GAN-based generators under the same 5% modification ratio. The comparison with UIB also indicates that this effect cannot be explained by the modification ratio alone.

E.4 Hyperparameter Sensitivity

We conduct a hyperparameter sensitivity study on CIFAR-100 under the same 5% publishing ratio used in the main experiments. Unless otherwise specified, the default GRDP configuration is $\alpha = 2$, $K = 16$, and Gaussian noise scale $\sigma = 0.25$. We vary one hyperparameter at a time while keeping all other publishing, classifier, and generator settings unchanged. Specifically, we evaluate $\alpha \in 1, 2, 4$, $K \in 8, 16, 32$, and $\sigma \in 0.15, 0.25, 0.35$. Here, α controls the strength of the publishing objective, K denotes the number of monitored/optimization directions during publication, and σ controls the scale of Gaussian randomization.

Table 22 shows the effect of each hyperparameter on classification accuracy and generative degradation. Increasing α from 1 to 4 gradually increases FID from 52.251 to 54.558, while classification accuracy decreases from 76.81% to 75.51%, showing the expected trade-off between the two objectives. We therefore use $\alpha = 2$ as the default setting.

Increasing K from 8 to 16 substantially increases FID from 49.987 to 53.836 with only a small change in classification accuracy. Increasing it further to 32 provides only a small additional FID increase, from 53.836 to 54.179, while reducing accuracy to 75.12%. We therefore choose $K = 16$.

For the Gaussian randomization scale, $\sigma = 0.25$ provides both the highest classification accuracy and the highest FID among the tested values. A smaller scale of $\sigma = 0.15$ gives a lower FID, while increasing it to $\sigma = 0.35$ does not further improve generative degradation and also reduces classification accuracy. We therefore use $\sigma = 0.25$ throughout the main experiments.

Overall, classification accuracy remains between 74.68% and 76.81%, while FID ranges from 49.189 to 54.558 across all tested configurations. The default setting provides a favorable balance between preserving classification utility and degrading generative capability.

F Social Impact

This work studies task-specific data publishing. A positive impact is that it gives data publishers a way to release useful data for classification while reducing the risk that the same release can be used to train or fine tune generative models. This is relevant when the original data contains visual patterns, styles, or examples that the publisher wants to protect from generation-based reuse. Our method keeps the local class structure needed by classifiers, while shifting the global feature statistics used by diffusion models. This can help researchers design safer data release procedures when a dataset should support discriminative learning but should not support high-quality generative training.

The experiments use public image classification datasets and standard model architectures. No private data or human-subject study is involved. The main negative risk is dual use. A malicious publisher could use a similar method to release data that appears useful for classification but is intentionally less useful for other legitimate downstream tasks, especially generative modeling. This may make the released dataset less transparent and may harm reproducibility if the processing is not disclosed. We present the method in a controlled data publishing setting, with explicit evaluation on both classification accuracy and generative degradation. The goal is not to hide poisoned data in public releases, but to show that data utility is task-dependent and that data publishers should clearly state how a released dataset has been processed and which downstream uses it is intended to support.

G Limitations

This work focuses on task-specific data publishing, where the released data should preserve classification utility while reducing its usefulness for generative training. Our experiments study this setting on standard image classification benchmarks with commonly used classifier and diffusion architectures. This scope is enough to test the main claim of the paper, but it does not cover every possible data release scenario. Extending the same idea to more tasks and data modalities is an interesting direction for future work.

GRDP introduces an additional publishing step before the dataset is released. This step constructs local force information, samples shell monitor points, and optimizes the published centers. The cost is paid only once by the publisher and does not change the downstream training procedure for users of the released data.

H The use of large language models

In this submission, we used an LLM solely to polish the writing and correct grammatical errors.