CATEGORICAL DRIFTING MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose CatDrift, a categorical extension of Drifting Models for one-step generation in discrete spaces. We reinterpret drifting as a local alignment problem: each generated sample is corrected in the direction of a data-dependent field while remaining close to its current value. In the original formulation, this constraint is Euclidean, yielding additive updates in sample space. This choice is natural for continuous data but ill-suited to categorical variables, whose states lie on the simplex. Replacing the Euclidean penalty with a Bregman divergence turns the drifting update into a mirror-descent step, a standard object in optimization with a natural information-geometric interpretation. For negative entropy, the categorical update has a closed form: build a kernel-weighted positive-minus-negative field from real samples and the generator's own outputs, add it to logits, project with softmax, and regress to the frozen target by cross-entropy. The result is a simple categorical drifting algorithm that obtains competitive-to-state-of-the-art one-step results on discrete image generation, molecular generation, and DNA sequence generation, without distilling diffusion models.

1 INTRODUCTION

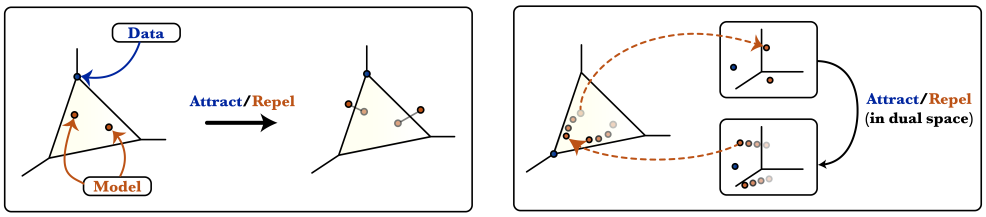
Modern high-quality generative models often rely on iterative sampling: diffusion/score-based models and flow-based models gradually transform noise into data by applying a learned denoising rule or vector field over many sequential steps (Ho et al., 2020; Song et al., 2021; Lipman et al., 2023). This sequential structure is costly at inference time, motivating a large body of work on distillation and few-step generation, including progressive distillation, consistency models, flow maps, mean flows, and related velocity-matching objectives (Salimans & Ho, 2022; Song et al., 2023; Song & Dhariwal, 2024; Boffi et al., 2025b;a; Geng et al., 2025a;b). However, these methods often replace expensive sampling with expensive training pipelines, pretrained teachers, or additional adaptation stages.

For discrete data, the problem is also geometric. Categorical objects such as discretized images, DNA sequences, language tokens, and molecular strings naturally live in constrained, non-Euclidean spaces. Their continuous relaxations lie on probability simplices, where naive additive updates can leave the state space, break normalization, or impose an artificial Euclidean ordering on categories. Recent work has addressed discrete generation through discrete diffusion, continuous relaxations, flow-map objectives, and Riemannian or statistical-manifold formulations (Austin et al., 2021; Sahoo et al., 2024; Lou et al., 2024; Shi et al., 2024; Campbell et al., 2022; Sun et al., 2023; Dieleman et al., 2022; Davis et al., 2024; Eijkelboom et al., 2024; Cheng et al., 2024; Park et al., 2026; Roos et al., 2026; Woo et al., 2026; Potaptchik et al., 2026; Lee et al., 2026), nevertheless remaining less developed than its continuous counterpart.

Drifting Models offer a simple alternative: move refinement from inference to training (Deng et al., 2026). A generator maps noise directly to a sample; during training, each generated sample is transported toward a locally improved target, and the network is trained to match this frozen target. Drifting is closely related to kernel moment-matching methods (Li et al., 2015; Dziugaite et al., 2015; Li et al., 2017), while recent analyses connect it to kernel-smoothed score differences, smoothed KL descent, identifiability, and the necessity of stop-gradient targets (Lai et al., 2026; Turan et al., 2026). Yet the original drifting update is Euclidean: it improves a generated sample by adding a displacement vector in ambient sample space. This is natural for continuous data, but poorly matched to categorical variables on the simplex.

We propose *CatDrift*, a categorical extension of Drifting Models for teacher-free one-step generation. We recast drifting as a local optimization problem: align with a positive-minus-negative improvement

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.



Standard Drifting **Categorical Drifting (CatDrift)**

Figure 1: Standard drifting can move categorical states off the simplex. *CatDrift* updates linearly in dual space while preserving the simplex in primal space.

signal while staying close to the generated state. Replacing the Euclidean penalty with a Bregman divergence yields a mirror-descent update. For categorical data, negative entropy gives KL geometry: construct local categorical evidence from real and generated samples, additively tilt the logits, apply softmax, and train with soft-target cross-entropy.

CatDrift preserves attraction–repulsion while changing the geometry of motion (Figure 1). The kernel defines *locality* (which samples provide evidence), and the Bregman divergence defines *motion* (how that evidence updates the state). This yields one-step generation for categorical grids, sequences, and molecular strings.

2 Background

2.1 One-step generation by drifting

Let p denote the data distribution on a space $\mathcal{X}$, and let $f_\theta : \mathcal{Z} \to \mathcal{X}$ be a generator that pushes a simple prior p_0 forward to the model distribution

$$q_\theta = [f_\theta]_{\#}\, p_0.$$

The goal of Drifting Models is to train f_θ so that q_θ approximates p, while keeping inference to a single forward pass. Drifting moves the refinement process from inference to training. Given a generated sample $x = f_\theta(\varepsilon)$, the method constructs a local improvement field around x. This field compares two local averages: attraction toward nearby real data samples $y^+ \sim p$ and repulsion from nearby generated samples $y^- \sim q_\theta$.

A positive kernel k determines which samples are considered nearby. The corresponding normalized local mean-shift fields are

$$V_p^+(x) := \frac{1}{Z_p(x)} \mathbb{E}_{y^+\sim p}\left[k(x,y^+)(y^+ - x)\right], \tag{1}$$

$$V_q^-(x) := \frac{1}{Z_q(x)} \mathbb{E}_{y^-\sim q_\theta}\left[k(x,y^-)(y^- - x)\right], \tag{2}$$

$$V(x) := V_p^+(x) - V_q^-(x). \tag{3}$$

Here

$$Z_p(x) = \mathbb{E}_{y^+\sim p}[k(x,y^+)], \qquad Z_q(x) = \mathbb{E}_{y^-\sim q_\theta}[k(x,y^-)]$$

are the local normalization constants.

The positive field points from x toward the local average of nearby real samples. The negative field points from x toward the local average of nearby generated samples; subtracting it therefore repels the model from its own local mass. When $q_\theta = p$, the two local mean-shift fields agree, and the drift field vanishes.

The generator is then trained to match a transported target state,

$$x^+ = x + \eta V(x), \tag{4}$$

where $\eta > 0$ is a step size. This target is a locally improved version of the current generated sample. The model is trained to regress to this target using a Euclidean penalty:

$$\mathcal{L}_{\text{drift}}(\theta) = \mathbb{E}_{\varepsilon}\left[\left\| f_\theta(\varepsilon) - \text{sg}(x^+)\right\|_2^2\right]. \tag{5}$$

The stop-gradient is essential: x^+ is treated as a fixed destination for the current update, rather than as an object through which the model can backpropagate.

This additive update is natural in Euclidean spaces, where $x + \eta V(x)$ remains a valid point in the same ambient space. For constrained spaces, however, this geometry is no longer appropriate. On the probability simplex, for example, an unconstrained additive step can leave the simplex, producing negative or unnormalized probabilities.

2.2 Bregman divergences and information geometry

To move beyond Euclidean geometry, we use Bregman divergences (Bregman, 1967). Let $\mathcal{X}$ be a convex subset of $\mathbb{R}^N$ and let $\psi : \mathcal{X} \to \mathbb{R}$ be a strictly convex differentiable potential. The Bregman divergence generated by ψ is

$$D_\psi(u, x) = \psi(u) - \psi(x) - \langle \nabla\psi(x), u - x\rangle. \tag{6}$$

Choosing ψ therefore chooses the geometry in which the update is regularized, as in mirror descent and related first-order optimization methods (Nemirovsky & Yudin, 1983; Beck & Teboulle, 2003). Thus, choosing ψ is not merely choosing a loss; it is choosing the geometry in which closeness is measured.

For example, the quadratic potential

$$\psi(x) = \frac{1}{2}\|x\|_2^2 \implies D_\psi(u, x) = \frac{1}{2}\|u - x\|_2^2, \tag{7}$$

recovering Euclidean geometry. On the probability simplex, for $r \in \Delta^{K-1}$, the natural choice is negative entropy,

$$\psi(r) = \sum_{k=1}^{K} r_k \log r_k \implies D_\psi(u, r) = \sum_{k=1}^{K} u_k \log \frac{u_k}{r_k} = \text{KL}(u \,\|\, r). \tag{8}$$

This also connects the update to information geometry (Amari, 2016). Although D_ψ is not generally a metric, its Hessian $\nabla^2\psi(x)$ defines the local metric obtained from the second-order expansion of the Bregman divergence. For negative entropy, this local metric is the Fisher–Rao geometry of categorical distributions. Mirror updates in this geometry therefore have a natural-gradient interpretation, which we make explicit in the CatDrift derivation below.

3 Categorical Drifting Models

CatDrift replaces Euclidean drifting's quadratic penalty with a Bregman divergence while retaining its local attraction–repulsion mechanism. For images, DNA sequences, and SMILES strings, the relaxed state space is a product of per-site probability simplices, hence convex; the set of valid discrete objects need not be convex. Hard one-hot states supply local evidence, while the update acts on soft probabilities and stays in the relative interior for finite softmax logits.

With negative entropy, the Bregman divergence is KL: CatDrift tilts logits toward local categorical evidence and trains against the frozen soft target. The kernel and feature map define *locality*, while the Bregman geometry defines *motion*.

3.1 Drifting as a Bregman update

In standard continuous drifting, the generator produces a state $x \in \mathbb{R}^d$. Given a local improvement field $V = V(x)$, the transported target is

$$x^+ = x + \eta V. \tag{9}$$

This additive step can be written equivalently as the solution of the local optimization problem

$$x^+ = \arg\max_{u \in \mathbb{R}^d} \left[\langle V, u - x \rangle - \frac{1}{2\eta} \|u - x\|_2^2 \right]. \tag{10}$$

Ordinary drifting aligns with the local field while a quadratic penalty keeps the target close to the current sample. Replacing this penalty by a Bregman divergence gives mirror drifting.

Definition 3.1 (Mirror drifting update)**.** *Let $\mathcal{X}$ be a convex state space, let ψ be a strictly convex, differentiable potential, and let D_ψ be its Bregman divergence. Given a current generated state $x \in \mathcal{X}$ and an improvement signal V, the mirror drifting target is*

$$x^+ = \arg\max_{u \in \mathcal{X}} \left[\langle V, u - x \rangle - \frac{1}{\eta} D_\psi(u, x) \right]. \tag{11}$$

The potential ψ therefore defines the geometry of the transported target. The usual Euclidean update is recovered by choosing the quadratic potential.

Proposition 3.1 (Euclidean drifting is quadratic mirror drifting)**.** *For $\psi(x) = \frac{1}{2}\|x\|_2^2$, the Bregman divergence is $D_\psi(u, x) = \frac{1}{2}\|u - x\|_2^2$, and the mirror drifting update becomes*

$$x^+ = x + \eta V.$$

Thus, standard drifting is exactly the quadratic instance of mirror drifting.

For categorical variables, the relaxed state space is the probability simplex; choosing negative entropy makes the update geometry respect this constraint.

3.2 Linear drifting in the dual space

The advantage of the Bregman view is that the update becomes additive after a change of coordinates. Ignoring active boundary constraints for the moment, the first-order condition of Eq. 11 gives

$$\nabla\psi(x^+) = \nabla\psi(x) + \eta V. \tag{12}$$

Thus, mirror drifting is linear in the dual coordinates $z = \nabla\psi(x)$ as shown in Figure 1, and the inverse mirror map returns the result to the original space:

$$z^+ = z + \eta V, \qquad x^+ = \nabla\psi^*(z^+). \tag{13}$$

Proposition 3.2 (Mirror drifting is linear in dual coordinates)**.** *When the Bregman subproblem has no active boundary constraints, the mirror drifting update satisfies*

$$z^+ = z + \eta V, \qquad x^+ = \nabla\psi^*(z^+), \qquad z = \nabla\psi(x).$$

That is, the update is additive in dual space and mapped back to the primal space by the inverse mirror map ψ^*, defined as usual by $\psi^*(z) = \sup_x \{\langle z, x\rangle - \psi(x)\}$.

For constrained spaces, Eq. 12 holds up to terms normal to the constraint set. On the simplex, this only adds a normalization constant to the log-probabilities. Since logits are defined up to additive constants, this does not change the resulting probability vector.

Euclidean updates act in primal coordinates and may leave the feasible set; mirror updates add in dual coordinates and map back to valid states. For the quadratic potential, $\nabla\psi(x) = x$, so Eq. 12 gives the standard step $x^+ = x + \eta V$.

To first order, the induced primal update is a natural-gradient step. Expanding Eq. 12 yields

$$x^+ - x = \eta \left(\nabla^2\psi(x)\right)^{-1} V + O(\eta^2).$$

Thus, in primal coordinates, mirror drifting follows the improvement signal preconditioned by the inverse local metric.

3.3 CATEGORICAL DRIFTING VIA NEGATIVE ENTROPY

For categorical data with K classes, we use the relaxed state space Δ^{K-1}. The generator outputs logits $z_\theta \in \mathbb{R}^K$ and probabilities

$$r_\theta = \mathrm{softmax}(z_\theta). \tag{14}$$

The hard categorical states used for neighborhood lookup are introduced in Section 3.4; for now, the important point is that logits are the natural dual coordinates for the simplex.

To see this, choose the negative entropy potential (8), such that its Bregman divergence is KL,

$$D_\psi(u, r) = \mathrm{KL}(u \,\|\, r) \implies \nabla\psi(r) = \log r + \mathbf{1}. \tag{15}$$

Thus the dual coordinates of a probability vector are log-probabilities, up to an additive constant. Since softmax is invariant to such constants, logits represent these dual coordinates.

Substituting negative entropy into the mirror-drifting update gives, for some normalizing constant c,

$$\log r^+ = \log r_\theta + \eta V + c\mathbf{1}. \tag{16}$$

Exponentiating and normalizing yields a logit tilt.

Proposition 3.3 (The softmax tilt is the entropy mirror update)**.** *Let $r_\theta = \mathrm{softmax}(z_\theta)$ and let $V \in \mathbb{R}^K$ be an improvement signal. For the negative entropy potential, the mirror drifting target*

$$r^+ = \arg\max_{u \in \Delta^{K-1}} \left[\langle V, u - r_\theta \rangle - \frac{1}{\eta} \mathrm{KL}(u \,\|\, r_\theta) \right]$$

has the closed-form solution

$$r_k^+ = \frac{r_{\theta,k} \exp(\eta V_k)}{\sum_{\ell=1}^K r_{\theta,\ell} \exp(\eta V_\ell)}.$$

Equivalently,

$$r^+ = \mathrm{softmax}(z_\theta + \eta V).$$

Thus, the softmax is not an extra projection step: it is the inverse mirror map for negative entropy. CatDrift therefore moves additively in logit space and returns to the simplex automatically by normalization. Only relative logits matter, so replacing V by $V + a\mathbf{1}$ leaves r^+ unchanged.

This gives the CatDrift update directly.

Definition 3.2 (CatDrift update)**.** *Given logits z_θ, probabilities $r_\theta = \mathrm{softmax}(z_\theta)$, and a categorical improvement signal V, define the tilted target*

$$r^+ = \mathrm{softmax}(z_\theta + \eta V). \tag{17}$$

The generator is trained with the frozen-target KL loss

$$\mathcal{L}_{\mathrm{CatDrift}}(\theta) = \mathbb{E}\left[\mathrm{KL}\left(\mathrm{sg}(r^+) \,\|\, r_\theta\right)\right], \tag{18}$$

where the expectation is over noise samples and the minibatches used to estimate the local improvement signal.

Since $\mathrm{sg}(r^+)$ is fixed during the generator update, this KL is simply soft-target cross-entropy up to an additive constant:

$$\mathrm{KL}\left(\mathrm{sg}(r^+) \,\|\, r_\theta\right) = -\sum_{k=1}^K \mathrm{sg}(r_k^+) \log r_{\theta,k} + \mathrm{const}. \tag{19}$$

In short, negative entropy turns categorical drifting into a standard classification-style objective: construct a logit-tilted soft target, stop its gradient, and train the generator to match it.

Remark: Fisher–Rao rather than Wasserstein. CatDrift uses the intrinsic information geometry of the simplex for motion. This is Fisher–Rao geometry, not Wasserstein geometry. Wasserstein updates require a ground metric between categories, which may be meaningful for ordinal or structured labels but is not canonical for nominal symbols. In CatDrift, the kernel k and feature map ϕ encode task-specific locality, while negative entropy defines the geometry of the update. Appendix A expands on this geometric view, including the local natural-gradient interpretation, the relation to f-divergences, the Fisher–Rao versus Wasserstein distinction, and a broader dynamical perspective connecting drifting updates to non-conservative and non-reciprocal systems.

3.4 Local categorical evidence in logit space

The categorical update in Proposition 3.3 has the form

$$r^+ = \mathrm{softmax}(z_\theta + \eta V).$$

The remaining question is how to construct the logit-space signal V. CatDrift uses the same positive-minus-negative principle as Euclidean drifting, but expresses it in the dual coordinates of the simplex. Since the entropy mirror map identifies dual coordinates with log-probabilities, we define V as a local log-evidence ratio between data and model neighborhoods.

For structured categorical objects such as images, sequences, or SMILES strings, write a sample as

$$x = (x_s)_{s\in\mathcal{S}}, \qquad x_s \in \{1, \dots, K\},$$

where s indexes a categorical site. The generator outputs sitewise logits $z_\theta = (z_{\theta s})_{s\in\mathcal{S}}$ and probabilities $r_{\theta s} = \mathrm{softmax}(z_{\theta s})$. To query local neighborhoods, we form the hard generated state

$$\hat{x}_s = \arg\max_c r_{\theta,s,c}.$$

This hard state is used only for lookup. Gradients flow through the soft probabilities in the KL loss, not through the $\arg\max$.

Given data particles $\{x_i^+\}_{i=1}^{N_+}$ and model particles $\{x_j^-\}_{j=1}^{N_-}$, define local weights around $\hat{x}$ by

$$a_i^+(\hat{x}) = k\big(\phi(\hat{x}), \phi(x_i^+)\big), \qquad a_j^-(\hat{x}) = k\big(\phi(\hat{x}), \phi(x_j^-)\big),$$

where ϕ is a task-specific feature map and k is a positive kernel. These weights determine which real and generated samples are considered local to the current output. At each site s, the weights define local categorical frequencies

$$\widehat{p}_{s,c}^+(\hat{x}) = \frac{\sum_{i=1}^{N_+} a_i^+(\hat{x})\,\mathbf{1}\{x_{i,s}^+ = c\}}{\sum_{i=1}^{N_+} a_i^+(\hat{x})}, \qquad \widehat{p}_{s,c}^-(\hat{x}) = \frac{\sum_{j=1}^{N_-} a_j^-(\hat{x})\,\mathbf{1}\{x_{j,s}^- = c\}}{\sum_{j=1}^{N_-} a_j^-(\hat{x})}.$$

Thus, $\widehat{p}_s^+$ records which categories appear in nearby data, while $\widehat{p}_s^-$ records which categories are already produced by nearby model samples. We smooth both distributions to avoid zero probabilities:

$$p_s^+(\hat{x}) = (1-\lambda)\widehat{p}_s^+(\hat{x}) + \lambda u_K, \qquad p_s^-(\hat{x}) = (1-\lambda)\widehat{p}_s^-(\hat{x}) + \lambda u_K, \qquad u_K = \frac{1}{K}\mathbf{1}.$$

Definition 3.3 (Categorical drifting field)**.** *Given smoothed local positive and negative categorical distributions $p_s^+(\hat{x})$ and $p_s^-(\hat{x})$, define*

$$V_{s,c}(\hat{x}) = \log p_{s,c}^+(\hat{x}) - \log p_{s,c}^-(\hat{x}).$$

This field is the categorical analogue of the positive-minus-negative drifting field. Categories that are locally more supported by the data than by the model receive a positive logit correction; categories that are locally overproduced by the model receive a negative correction. The choice $V = \log p^+ - \log p^-$ should be understood as a motivated dual-space estimator of local data/model discrepancy, not as the unique consequence of the mirror geometry itself.

Substituting this field into the entropy mirror update gives

$$r_{s,c}^+ \propto r_{\theta,s,c}\left(\frac{p_{s,c}^+(\hat{x})}{p_{s,c}^-(\hat{x})}\right)^\eta.$$

CatDrift multiplicatively reweights model probabilities by a local data/model evidence ratio, tilting logits toward underrepresented categories and away from overrepresented ones. The local distributions, field V, and target are frozen for each generator update. Section 4 describes the task-dependent kernels and feature maps for categorical grids, DNA sequences, and molecules.

Model particles represent the current model distribution and supply repulsion, not semantically negative examples; we observed modest batch-size sensitivity over the tested range. Using B data particles and B model particles per update requires $O(B^2)$ pairwise interactions at fixed object and feature dimensions, while inference uses one network evaluation without particle interactions or kernel evaluations.

4 EXPERIMENTS

Experimental scope. We evaluate CatDrift as a **teacher-free one-step generator** on discretized images, promoter DNA sequences, and SMILES-based QM9/ZINC250k molecules. These domains cover categorical grids, long sequences, and constrained symbolic strings; all use local evidence, logit tilting, and frozen-target regression.

Controlled comparison and tuning. Standard Drifting uses a quadratic potential, whereas CatDrift uses negative entropy. Our implementations share the codebase, generator architecture, frozen MAE features, particle and neighborhood construction, kernels, and training budget, with normalization/projection for Euclidean outputs.

Matched grid searches over kernel temperatures, learning rates, step sizes, smoothing, and related settings use equal search budgets and the same validation-based selection procedure; we report each method's best configuration. This comparison tests the geometric update with shared locality machinery.

PairFlow, Categorical Flow Maps, and Riemannian MeanFlow provide contextual domain-specific comparisons, rather than controlled geometry ablations. We exclude language-model distillation with pretrained autoregressive teachers. Setup details are in Appendix D.

4.1 DISCRETIZED IMAGE GENERATION

Following Park et al. (2026); Roos et al. (2026), we evaluate one-step generation on binarized MNIST and discretized CIFAR-10, spanning binary and multi-class image domains.

Setup and FID protocol. MNIST pixels are binary variables. For CIFAR-10, the generator directly predicts each color channel as a 256-way categorical variable, rather than generating continuous RGB values and discretizing afterward. Hard per-site $\arg\max$ states determine neighborhoods in MAE feature space.

FID uses 1000 generated and 1000 held-out real images. This sample-limited estimate is noisy and not directly comparable with the 50000-sample FID values typically reported for continuous CIFAR-10 models; neither a sharp sample grid nor FID alone establishes or rules out mode collapse.

Figure 2 shows a one-step MNIST FID of 8.02, improving over Categorical Flow Maps (10.1), quadratic Drifting (29.0), and the reported one-step PairFlow baselines, without a pretrained teacher.

On discretized CIFAR-10, quadratic Drifting obtains 210 ± 10.4 FID versus 58 ± 2.9 for negative-entropy CatDrift. Because the feature representation and locality machinery are shared, they alone cannot explain this gain. These results support logit tilting for one-step generation across binary and multi-class categorical image domains.

4.2 DNA SEQUENCE GENERATION

We next evaluate CatDrift on promoter DNA generation, a long categorical sequence task where each sample is a length-1024 sequence over $\{A, C, G, T\}$. This benchmark tests whether local logit tilting remains effective beyond image-like grids, in a setting with biological long-range structure. We report two complementary metrics: MSE between predicted regulatory signal profiles and 6-mer

Method	NFE	FID ↓
Binarized MNIST		
PairFlow, 1 step	1	40.58 ± 0.1
PairFlow + ReDi	1	12.90 ± 0.4
Categorical Flow Maps	1	10.1 ± 0.3
Drifting Model	1	29.0 ± 1.2
CatDrift (ours)	1	8.02 ± 0.1
Discretized CIFAR-10		
PairFlow, 1 step	1	269.87 ± 5.2
PairFlow + DCD	1	223.86 ± 2.5
Categorical Flow Maps	1	–
Drifting Model	1	210 ± 10.4
CatDrift (ours)	1	58 ± 2.9

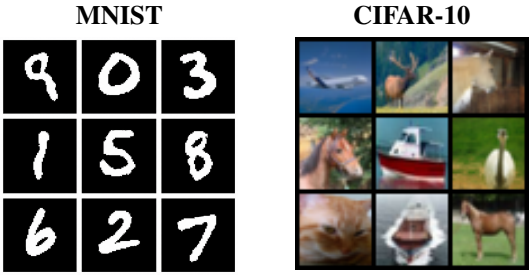


Figure 2: **Discretized image generation.** Left: one-step FID; quadratic Drifting and negative-entropy CatDrift share architecture, MAE features, locality, and matched training/tuning budgets. Other methods are contextual comparisons. Right: CatDrift samples.

Table 1: **Promoter DNA generation.** CatDrift is trained from scratch and uses one network evaluation at test time.

Method	NFE	MSE ↓	6-mer Corr. ↑
Many-step flow baselines			
Dirichlet FM	100	$0.034_{\pm 0.004}$	–
Fisher FM	100	$0.030_{\pm 0.001}$	$0.960_{\pm 0.010}$
One-step flow-map baselines			
E-RMF + x_1-pred	1	$0.030_{\pm 0.000}$	$0.960_{\pm 0.010}$
E-RMF + v-pred	1	$0.031_{\pm 0.001}$	$0.960_{\pm 0.000}$
L-RMF + x_1-pred	1	$0.027_{\pm 0.001}$	$0.880_{\pm 0.000}$
L-RMF + v-pred	1	$0.027_{\pm 0.001}$	$0.850_{\pm 0.010}$
S-RMF + x_1-pred	1	$0.030_{\pm 0.001}$	$0.840_{\pm 0.030}$
S-RMF + v-pred	1	$0.030_{\pm 0.001}$	$0.930_{\pm 0.020}$
Drifting			
CatDrift (ours)	1	$0.030_{\pm 0.002}$	$0.941_{\pm 0.040}$

Pearson correlation between generated and real sequence distributions. Full experimental details are provided in Appendix D.

Table 1 shows that CatDrift reaches the same performance regime as the strongest one-step flow-map baselines. With a single network evaluation, it obtains an MSE of 0.030 ± 0.002 and a 6-mer correlation of 0.941 ± 0.04. Its MSE matches Fisher FM, E-RMF, and S-RMF up to reported precision, while its 6-mer correlation is close to the best Fisher and E-RMF results.

Takeaway. CatDrift achieves **parity rather than a strict win**. RMF first learns an iterative manifold flow and then distills a one-step flow map; CatDrift trains directly from scratch using local categorical evidence, logit tilting, and frozen-target regression. It reaches this performance on long sequences without a pretrained teacher, distillation, or post-hoc adaptation.

4.3 Molecular generation

Finally, we evaluate CatDrift on SMILES-based molecular generation, following PairFlow (Park et al., 2026). Molecules form a challenging categorical domain: generated strings must satisfy both SMILES syntax and chemical validity. We use SMILES rather than graph-native representations, treating each molecule as a categorical sequence over the SMILES vocabulary.

This setting is deliberately harder in one respect than many graph-based QM9 benchmarks. Graph generators are often conditioned on an atom mask, or equivalently on the number of atoms, so the model

Method	NFE	Valid ↑	Unique ↑
QM9			
PairFlow, 1 step	1	$0.22_{\pm 0.01}$	$0.22_{\pm 0.01}$
PairFlow + DCD, best 1 step	1	$0.44_{\pm 0.02}$	$0.44_{\pm 0.02}$
Drifting Model	1	$0.08_{\pm 0.09}$	$0.23_{\pm 0.05}$
CatDrift (ours)	1	$0.62_{\pm 0.05}$	$0.68_{\pm 0.05}$
ZINC250k			
PairFlow, 1 step	1	$0.01_{\pm 0.00}$	$0.01_{\pm 0.00}$
PairFlow + DCD, best 1 step	1	$0.11_{\pm 0.01}$	$0.11_{\pm 0.01}$
Drifting Model	1	$0.00_{\pm 0.00}$	$0.22_{\pm 0.08}$
CatDrift (ours)	1	$0.23_{\pm 0.06}$	$0.36_{\pm 0.04}$

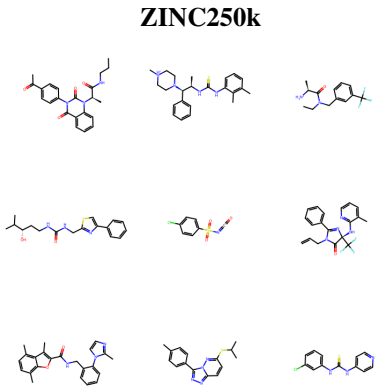


Figure 3: **Molecular generation.** Left: validity among 1024 generated SMILES and uniqueness among valid samples on QM9 and ZINC250k. Right: representative ZINC250k CatDrift samples, selected from generated molecules that are both **valid and unique**, and rendered as molecular structures.

is given the molecular size. In the SMILES setting, CatDrift must infer **both** the size/termination structure and the molecular content in a single forward pass.

Comparison protocol. We omit FCD because it is not reported consistently for the relevant one-step PairFlow settings; in our attempted reproduction, FCD was unstable in a way that suggested an unresolved evaluation or implementation mismatch. Full experimental details are provided in Appendix D.

Figure 3 reports validity and uniqueness over 1024 generated SMILES strings. On QM9, CatDrift obtains 62% validity and 68% uniqueness among valid samples with a single network evaluation. On ZINC250k, CatDrift obtains 23% valid and 36% unique samples. PairFlow reports one-step validity of $223.4/1024$ on QM9 and $9.9/1024$ on ZINC250k, while its 16-step results are substantially stronger. As far as we are aware, this puts CatDrift at the state-of-the-art on this style of one-step molecular generation.

Takeaway. CatDrift generates complete SMILES strings in one pass, without a graph mask or given atom count, and outperforms the reported one-step baselines. Extending its neighborhoods to permutation-aware graphs remains future work.

5 Conclusion and Future Work

CatDrift extends drifting to one-step categorical generation by retaining local attraction–repulsion and replacing the Euclidean penalty with Bregman geometry. Negative entropy yields an additive logit tilt, softmax as the inverse mirror map, and cross-entropy training against frozen soft targets. Thus, the kernel defines locality (which data and model samples provide evidence), while the geometry determines how the state moves.

We evaluated quadratic and negative-entropy potentials only. Mahalanobis, mixed, or learned potentials, and transport-aware geometries when categories have a meaningful ground metric, remain future work. CatDrift also depends on task-specific kernels, feature maps, and smoothing; learning these choices and extending the framework to structured graphs or hybrid discrete–continuous objects are open directions.

AI USE STATEMENT

We used large language models for brainstorming, coding assistance, and checking the final manuscript. The authors take responsibility for the final text, code, and scientific claims, including AI-assisted material.

REFERENCES

Shun-ichi Amari. *Information Geometry and Its Applications*. Springer, 2016.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021.

Yael Avni, Michel Fruchart, David Martin, Daniel Seara, and Vincenzo Vitelli. Nonreciprocal ising model. *Physical Review Letters*, 134(11):117103, 2025.

Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

Nicholas Boffi, Michael Albergo, and Eric Vanden-Eijnden. How to build a consistency model: Learning flow maps via self-distillation. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 33346–33382. Curran Associates, Inc., 2025a. doi: 10.52202/085713-1121.

Nicholas Matthew Boffi, Michael Samuel Albergo, and Eric Vanden-Eijnden. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *Transactions on Machine Learning Research*, 2025b.

Lev M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 28266–28279, 2022.

Jiarui Cao, Zixuan Wei, and Yuxin Liu. Gradient flow drifting: Generative modeling via wasserstein gradient flows of kde-approximated divergences. *arXiv preprint arXiv:2603.10592*, 2026.

Kathleen M Chen, Aaron K Wong, Olga G Troyanskaya, and Jian Zhou. A sequence-based global map of regulatory activity for deciphering human genetics. *Nature genetics*, 54(7):940–949, 2022.

Chaoran Cheng, Jiahan Li, Jian Peng, and Ge Liu. Categorical flow matching on statistical manifolds. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 54787–54819. Curran Associates, Inc., 2024. doi: 10.52202/079017-1738.

Oscar Davis, Samuel Kessler, Mircea Petrache, İsmail İlkan Ceylan, Michael Bronstein, and Avishek Joey Bose. Fisher flow matching for generative modeling over discrete data. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 139054–139084. Curran Associates, Inc., 2024. doi: 10.52202/079017-4413.

Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative modeling via drifting. *arXiv preprint arXiv:2602.04770*, 2026.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H. Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, Curtis Hawthorne, Rémi Leblond, Will Grathwohl, and Jonas Adler. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089*, 2022.

Gintare Karolina Dziugaite, Daniel M. Roy, and Zoubin Ghahramani. Training generative neural networks via maximum mean discrepancy optimization. In *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence*, UAI'15, pp. 258–267, 2015.

Floor Eijkelboom, Grigory Bartosh, Christian A. Naesseth, Max Welling, and Jan-Willem van de Meent. Variational flow matching for graph generation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Leonard T. Franz, Sebastian Hoffmann, Tim Weiland, Bernhard Schölkopf, and Georg Martius. Drifting fields are not conservative. *arXiv preprint arXiv:2604.06333*, 2026. URL https://arxiv.org/abs/2604.06333v3.

Michel Fruchart, Ryo Hanai, Peter B Littlewood, and Vincenzo Vitelli. Non-reciprocal phase transitions. *Nature*, 592(7854):363–369, 2021.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J. Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. *arXiv preprint arXiv:2512.02012*, 2025b.

Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

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. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Chung-Chau Hon, Jordan A Ramilowski, Jayson Harshbarger, Nicolas Bertin, Owen JL Rackham, Julian Gough, Elena Denisenko, Sebastian Schmeier, Thomas M Poulsen, Jessica Severin, et al. An atlas of human long non-coding rnas with accurate 5′ ends. *Nature*, 543(7644):199–204, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6980.

Alex Krizhevsky. Learning multiple layers of features from tiny images. M.Sc. thesis, University of Toronto, 2009. URL https://cave.cs.toronto.edu/kriz/.

Chieh-Hsin Lai, Bac Nguyen, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yuki Mitsufuji, Stefano Ermon, and Molei Tao. A unified view of score-based and drifting models. *arXiv preprint arXiv:2603.07514*, 2026. URL https://arxiv.org/abs/2603.07514v3.

Yann LeCun, Corinna Cortes, and Christopher J. C. Burges. The MNIST database of handwritten digits. Dataset, n.d. URL https://yann.lecun.org/exdb/mnist/index.html. Accessed 26 September 2026.

Chanhyuk Lee, Jaehoon Yoo, Manan Agarwal, Sheel Shah, Jerry Huang, Aditi Raghunathan, Seunghoon Hong, Nicholas M Boffi, and Jinwoo Kim. Flow map language models: One-step language modeling via continuous denoising. *arXiv preprint arXiv:2602.16813*, 2026.

Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. MMD GAN: Towards deeper understanding of moment matching network. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Yujia Li, Kevin Swersky, and Richard Zemel. Generative moment matching networks. In *Proceedings of the 32nd International Conference on Machine Learning*, ICML'15, pp. 1718–1727, 2015.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations*, 2019.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32819–32848. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/lou24a.html.

Arkadi S. Nemirovsky and David B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley, 1983.

Mingue Park, Jisung Hwang, Seungwoo Yoo, Kyeongmin Yeo, and Minhyuk Sung. Pairflow: Closed-form source-target coupling for few-step generation in discrete flow models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Peter Potaptchik, Jason Yim, Adhi Saravanan, Peter Holderrieth, Eric Vanden-Eijnden, and Michael S Albergo. Discrete flow maps. *arXiv preprint arXiv:2604.09784*, 2026.

Raghunathan Ramakrishnan, Pavlo O. Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1:140022, 2014. doi: 10.1038/sdata.2014.22.

Daan Roos, Oscar Davis, Floor Eijkelboom, Michael Bronstein, Max Welling, İsmail İlkan Ceylan, Luca Ambrogioni, and Jan-Willem van de Meent. Categorical flow maps. *arXiv preprint arXiv:2602.12233*, 2026.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 130136–130184. Curran Associates, Inc., 2024. doi: 10.52202/079017-4135.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 15078–15097, 2024. URL https://iclr.cc/virtual/2024/oral/19754.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/song23a.html.

Haoran Sun, Lijun Yu, Bo Dai, Dale Schuurmans, and Hanjun Dai. Score-based continuous-time discrete diffusion models. In *The Eleventh International Conference on Learning Representations*, 2023.

Erkan Turan, Nicolas Dufour, and Maks Ovsjanikov. Generative drifting is secretly score matching: a spectral and variational perspective. *arXiv preprint arXiv:2603.09936*, 2026. URL https://arxiv.org/abs/2603.09936v2.

Dongyeop Woo, Marta Skreta, Seonghyun Park, Kirill Neklyudov, and Sungsoo Ahn. Riemannian meanflow. *arXiv preprint arXiv:2602.07744*, 2026.

A GEOMETRIC BACKGROUND

This appendix gives a short geometric interpretation of the mirror-drifting update. The goal is not to develop the full theory of information geometry, but to clarify what is hidden inside the simple CatDrift rule: construct a local improvement signal, add it to the logits, and train with cross-entropy.

A.1 MIRROR DRIFTING AS MIRROR DESCENT

Recall the mirror-drifting target

$$x^+ = \arg\max_{u \in \mathcal{X}} \left[\langle V(x), u - x \rangle - \frac{1}{\eta} D_\psi(u, x) \right], \tag{20}$$

where

$$D_\psi(u, x) = \psi(u) - \psi(x) - \langle \nabla\psi(x), u - x \rangle \tag{21}$$

is the Bregman divergence generated by a strictly convex potential ψ.

This is a mirror-descent update written in the language of drifting. In standard optimization, mirror descent balances a linearized objective with a Bregman penalty that keeps the new point close to the current point. Here, the role of the linearized objective is played by the local improvement field $V(x)$. Thus, mirror drifting keeps the same intuition as ordinary drifting – move in the direction of local improvement – but lets the potential ψ define what "moving" and "staying close" mean.

The first-order optimality condition of Eq. 20 gives the central identity

$$\nabla\psi(x^+) = \nabla\psi(x) + \eta V(x). \tag{22}$$

Equivalently,

$$x^+ = \nabla\psi^* \left(\nabla\psi(x) + \eta V(x) \right), \tag{23}$$

where ψ^* is the convex conjugate of ψ. Therefore, mirror drifting is linear in the dual coordinates $\nabla\psi(x)$, even when the original state space $\mathcal{X}$ is constrained.

A.2 LOCAL NATURAL-GRADIENT FORM

The same equation also explains the connection to natural gradients. Suppose the step is small and write

$$x^+ = x + \delta x.$$

Taylor-expanding the left-hand side of Eq. 22 around x gives

$$\nabla\psi(x + \delta x) = \nabla\psi(x) + \nabla^2\psi(x)\delta x + O(\|\delta x\|^2).$$

Substituting into Eq. 22 and cancelling $\nabla\psi(x)$ yields

$$\nabla^2\psi(x)\delta x = \eta V(x) + O(\|\delta x\|^2).$$

Thus,

$$x^+ - x = \delta x = \eta \big[\nabla^2\psi(x)\big]^{-1} V(x) + O(\eta^2). \tag{24}$$

In primal coordinates, mirror drifting therefore moves in the direction of the improvement field preconditioned by the inverse local metric $\nabla^2\psi(x)$. This is exactly the form of a natural-gradient update: the ordinary direction is corrected by the geometry of the space.

The same metric appears by expanding the Bregman divergence itself. For a small displacement δx,

$$D_\psi(x + \delta x, x) = \frac{1}{2} \delta x^\top \nabla^2\psi(x)\delta x + O(\|\delta x\|^3). \tag{25}$$

Thus, a Bregman divergence is a finite-step object whose infinitesimal limit is a Riemannian metric.

A.3 NEGATIVE ENTROPY AND FISHER–RAO GEOMETRY

For categorical variables, the relaxed state space is the probability simplex Δ^{K-1}. CatDrift chooses the negative entropy potential

$$\psi(r) = \sum_{k=1}^{K} r_k \log r_k. \tag{26}$$

Its Bregman divergence is

$$D_\psi(u, r) = \mathrm{KL}(u \,\|\, r), \tag{27}$$

and its gradient and Hessian are

$$\nabla\psi(r) = \log r + \mathbf{1}, \qquad \nabla^2\psi(r) = \mathrm{diag}\left(\frac{1}{r_1}, \ldots, \frac{1}{r_K}\right). \tag{28}$$

On the tangent space of the simplex,

$$T_r\Delta^{K-1} = \left\{ \delta r \in \mathbb{R}^K : \sum_{k=1}^{K} \delta r_k = 0 \right\},$$

the local quadratic form becomes

$$D_\psi(r + \delta r, r) = \frac{1}{2} \sum_{k=1}^{K} \frac{(\delta r_k)^2}{r_k} + O(\|\delta r\|^3). \tag{29}$$

This is the Fisher–Rao metric for categorical distributions. Hence, although CatDrift is implemented as a finite logit-tilting update, its small-step limit is a natural-gradient-style update in the Fisher–Rao geometry of the simplex.

This also clarifies the role of logits. For negative entropy, the dual coordinates are

$$\nabla\psi(r) = \log r + \mathbf{1}.$$

The additive constant is irrelevant because probabilities are normalized. Thus, dual coordinates are log-probabilities, or equivalently logits up to a shift. The convex conjugate of negative entropy on the simplex is the log-partition function

$$\psi^*(z) = \log \sum_{k=1}^{K} \exp(z_k), \tag{30}$$

and its gradient is

$$\nabla\psi^*(z) = \mathrm{softmax}(z). \tag{31}$$

Therefore the mirror-drifting master equation becomes

$$r^+ = \nabla\psi^*(z + \eta V) = \mathrm{softmax}(z + \eta V),$$

which is precisely the CatDrift update.

A.4 RELATION TO f-DIVERGENCES

It is useful to distinguish two related but different facts. Bregman divergences and f-divergences are not the same class of divergences. A Bregman divergence is generated by a convex potential ψ through a tangent-plane gap. An f-divergence compares probability distributions through a convex function of their density ratio.

However, KL divergence sits at the intersection of these two families. On the simplex, KL is both the Bregman divergence generated by negative entropy and an f-divergence. This is why the CatDrift objective has both a mirror-descent interpretation and an information-geometric interpretation.

More generally, many f-divergences induce the same local geometry. For a small perturbation $r + \delta r$ of a categorical distribution r, one obtains

$$D_f(r + \delta r \,\|\, r) = \frac{f''(1)}{2} \sum_{k=1}^{K} \frac{(\delta r_k)^2}{r_k} + O(\|\delta r\|^3),$$

assuming the usual normalization $f(1) = 0$. Thus, up to a scalar factor, f-divergences locally induce the Fisher–Rao metric. CatDrift uses the KL case because it is also the Bregman divergence of negative entropy, which gives a finite-step mirror update with the closed-form softmax tilt.

A.5 Fisher–Rao versus Wasserstein geometry

The Fisher–Rao geometry used by CatDrift is different from Wasserstein geometry. Both are geometries on probability distributions, but they encode different assumptions.

Wasserstein geometry requires a ground metric between states. This is natural when the underlying space already has a meaningful notion of distance, such as Euclidean space. For example, many diffusion, Langevin, and Fokker–Planck dynamics can be interpreted as Wasserstein gradient flows of entropy-like functionals on $\mathbb{R}^d$.

For categorical variables, such a metric is usually not canonical. Tokens, classes, and symbols are often nominal: there is no intrinsic distance between category i and category j unless we impose one. A Wasserstein geometry on the simplex would therefore require an additional ground metric on the labels. This can be useful when such a metric is meaningful, for example for ordinal labels or structured discrete outputs, but it is not intrinsic to categorical data.

Fisher–Rao geometry, by contrast, is intrinsic to the simplex. It depends only on the fact that the state is a probability distribution. CatDrift therefore uses Fisher–Rao geometry for the update itself, while leaving the kernel k and feature map ϕ free to encode task-specific notions of locality.

A.6 A broader design space

Mirror drifting separates two choices that are entangled in Euclidean drifting. The field V specifies the local improvement signal. The potential ψ specifies the geometry of the update.

CatDrift studies the negative-entropy case because it is the natural choice for categorical variables and leads to a simple softmax update. But the framework is broader. The quadratic potential recovers standard Euclidean drifting. Mahalanobis-type potentials could encode correlated continuous outputs. Mixed potentials could be used for hybrid discrete–continuous variables. Learned potentials could define data-adaptive geometries. Wasserstein-style updates may be appropriate when the state space carries a meaningful ground metric.

The algorithmic template remains the same in each case: construct a local positive-minus-negative improvement signal, solve a geometry-regularized target problem, stop the target gradient, and train the generator to match the transported state.

A.7 Non-reciprocal thermodynamics

It is curious that the update rule defined for CatDrift in (17) (and indeed, for drifting models in general) does not obviously minimize an objective, i.e. the vector displace ments are not necessarily the gradient of a globally defined scalar energy (Cao et al., 2026; Franz et al., 2026). In this sense it is similar to systems governed by *non-reciprocal dynamics* (Fruchart et al., 2021), an area that is currently the subject of intense study in statistical physics. These systems – whose dynamics are *not* defined in terms of an energy function, and include systems such as active matter, swarms of robots, and flocks of birds – exhibit many phenomena (e.g. steady-state limit cycles) which are not found in conventional equilibrium statistical mechanics. They are sometimes defined by giving each individual degree of freedom a "selfish energy" which they individually try to minimize, and which generally do not assemble into a globally defined energy function (see e.g. (Avni et al., 2025)). This framing is conceptually similar to the local minimization step defined in (11), which also provides local dynamics without making reference to a global objective. It would be interesting to analyze dynamics of these drifting models in terms of the tools of non-reciprocal statistical physics, much as conventional diffusion models are understood using the tools of conventional thermodynamics (Sohl-Dickstein et al., 2015).

B PROOFS

B.1 EUCLIDEAN DRIFTING IS QUADRATIC MIRROR DRIFTING

Proof of Proposition 3.1. Let

$$\psi(x) = \frac{1}{2}\|x\|_2^2.$$

Then $\nabla\psi(x) = x$, and therefore

$$D_\psi(u, x) = \psi(u) - \psi(x) - \langle \nabla\psi(x), u - x\rangle \tag{32}$$

$$= \frac{1}{2}\|u\|_2^2 - \frac{1}{2}\|x\|_2^2 - \langle x, u - x\rangle \tag{33}$$

$$= \frac{1}{2}\|u - x\|_2^2. \tag{34}$$

Substituting this into the mirror drifting update gives

$$x^+ = \arg\max_{u\in\mathbb{R}^d} \left[\langle V, u - x_\theta\rangle - \frac{1}{2\eta}\|u - x_\theta\|_2^2\right].$$

The objective is strictly concave in u. Taking the gradient with respect to u and setting it to zero gives

$$V - \frac{1}{\eta}(u - x_\theta) = 0.$$

Hence the unique maximizer is

$$u = x_\theta + \eta V.$$

Therefore,

$$x^+ = x_\theta + \eta V,$$

which is exactly the standard Euclidean drifting step. □

B.2 MIRROR DRIFTING IS LINEAR IN DUAL COORDINATES

Proof of Proposition 3.2. The mirror drifting update is defined by

$$x^+ = \arg\max_{u\in\mathcal{X}} \left[\langle V, u - x_\theta\rangle - \frac{1}{\eta}D_\psi(u, x_\theta)\right].$$

Using the definition of the Bregman divergence,

$$D_\psi(u, x_\theta) = \psi(u) - \psi(x_\theta) - \langle \nabla\psi(x_\theta), u - x_\theta\rangle,$$

we differentiate the objective with respect to u. Since

$$\nabla_u D_\psi(u, x_\theta) = \nabla\psi(u) - \nabla\psi(x_\theta),$$

the first-order optimality condition at $u = x^+$ is

$$V - \frac{1}{\eta}\left(\nabla\psi(x^+) - \nabla\psi(x_\theta)\right) = 0.$$

Rearranging yields

$$\nabla\psi(x^+) = \nabla\psi(x_\theta) + \eta V.$$

Writing

$$z_\theta = \nabla\psi(x_\theta), \qquad z^+ = \nabla\psi(x^+),$$

we obtain

$$z^+ = z_\theta + \eta V.$$

Finally, since $\nabla\psi^*$ is the inverse of $\nabla\psi$, we have

$$x^+ = \nabla\psi^*(z^+).$$

Thus mirror drifting is additive in dual coordinates. □

B.3 THE SOFTMAX TILT IS THE ENTROPY MIRROR UPDATE

Proof of Proposition 3.3. For negative entropy on the simplex,

$$D_\psi(u, r_\theta) = \mathrm{KL}(u \,\|\, r_\theta).$$

The mirror drifting target is therefore

$$r^+ = \arg\max_{u \in \Delta^{K-1}} \left[\langle V, u - r_\theta \rangle - \frac{1}{\eta} \mathrm{KL}(u \,\|\, r_\theta) \right].$$

Terms independent of u do not affect the maximizer, so this is equivalent to maximizing

$$\sum_{k=1}^{K} u_k V_k - \frac{1}{\eta} \sum_{k=1}^{K} u_k \log \frac{u_k}{r_{\theta,k}}$$

subject to $\sum_{k=1}^{K} u_k = 1$ and $u_k \geq 0$.

Introduce a Lagrange multiplier λ for the normalization constraint. The Lagrangian is

$$\mathcal{J}(u, \lambda) = \sum_{k=1}^{K} u_k V_k - \frac{1}{\eta} \sum_{k=1}^{K} u_k \log \frac{u_k}{r_{\theta,k}} + \lambda \left(\sum_{k=1}^{K} u_k - 1 \right).$$

The first-order condition for each k is

$$V_k - \frac{1}{\eta} \left(\log \frac{u_k}{r_{\theta,k}} + 1 \right) + \lambda = 0.$$

Rearranging gives

$$\log \frac{u_k}{r_{\theta,k}} = \eta V_k + \eta\lambda - 1.$$

Therefore,

$$u_k = r_{\theta,k} \exp(\eta V_k) \exp(\eta\lambda - 1).$$

The final factor is constant in k, so normalization over the simplex gives

$$u_k = \frac{r_{\theta,k} \exp(\eta V_k)}{\sum_{\ell=1}^{K} r_{\theta,\ell} \exp(\eta V_\ell)}.$$

Thus

$$r_k^+ = \frac{r_{\theta,k} \exp(\eta V_k)}{\sum_{\ell=1}^{K} r_{\theta,\ell} \exp(\eta V_\ell)}.$$

If $r_\theta = \mathrm{softmax}(z_\theta)$, then

$$r_{\theta,k} = \frac{\exp(z_{\theta,k})}{\sum_{\ell=1}^{K} \exp(z_{\theta,\ell})}.$$

Hence

$$r_k^+ \propto \exp(z_{\theta,k}) \exp(\eta V_k) = \exp(z_{\theta,k} + \eta V_k),$$

and normalizing gives

$$r^+ = \mathrm{softmax}(z_\theta + \eta V).$$

□

C CatDrift Algorithm

Algorithm 1 Categorical Drifting

Require: generator g_θ, prior p_0, data distribution p, kernel k, feature map ϕ, step size η, smoothing constant λ, batch size B

1: Sample a noise batch ε of size B from p_0 and compute sitewise logits $z_\theta = g_\theta(\varepsilon)$

2: Compute model probabilities $r_{\theta\, s} = \mathrm{softmax}(z_{\theta\, s})$ for each site $s \in \mathcal{S}$

3: Form the hard generated state $\hat{x}_s = \arg\max_c r_{\theta\, s,c}$ ▷ used only for neighborhood lookup

4: Sample $N_+ = B$ data particles $\{x_i^+\}_{i=1}^{N_+}$ and use the current batch of $N_- = B$ model particles $\{x_j^-\}_{j=1}^{N_-}$

5: For each generated query, compute locality weights and mask its model self-neighbor

$$a_i^+(\hat{x}) = k\big(\phi(\hat{x}), \phi(x_i^+)\big), \qquad a_j^-(\hat{x}) = k\big(\phi(\hat{x}), \phi(x_j^-)\big)$$

6: Estimate local categorical frequencies, for each site s and category c,

$$\widehat{p}^+_{s,c}(\hat{x}) = \frac{\sum_{i=1}^{N_+} a_i^+(\hat{x})\, \mathbf{1}\{x_{i,s}^+ = c\}}{\sum_{i=1}^{N_+} a_i^+(\hat{x})}, \qquad \widehat{p}^-_{s,c}(\hat{x}) = \frac{\sum_{j=1}^{N_-} a_j^-(\hat{x})\, \mathbf{1}\{x_{j,s}^- = c\}}{\sum_{j=1}^{N_-} a_j^-(\hat{x})}$$

7: Smooth the local distributions

$$p_s^+(\hat{x}) = (1-\lambda)\widehat{p}_s^+(\hat{x}) + \lambda u_K, \qquad p_s^-(\hat{x}) = (1-\lambda)\widehat{p}_s^-(\hat{x}) + \lambda u_K, \qquad u_K = \frac{1}{K}\mathbf{1}$$

8: Compute the dual-space field

$$V_{s,c}(\hat{x}) = \log p^+_{s,c}(\hat{x}) - \log p^-_{s,c}(\hat{x})$$

9: Construct the frozen tilted target

$$r_s^+ = \mathrm{softmax}(z_{\theta\, s} + \eta V_s(\hat{x}))$$

10: Update θ by minimizing

$$\mathcal{L}_{\mathrm{CatDrift}}(\theta) = \sum_{s \in \mathcal{S}} \mathrm{KL}\big(\mathrm{sg}(r_s^+) \,\|\, r_{\theta\, s}\big)$$

D EXPERIMENTAL SETUP

Common CatDrift implementation. All CatDrift models generate a complete categorical object in a single forward pass. Given Gaussian noise ε, the generator outputs sitewise logits $z_\theta = g_\theta(\varepsilon)$ and probabilities $r_{\theta,s} = \mathrm{softmax}(z_{\theta,s})$. For neighborhood lookup we form the hard state

$$\hat{x}_s = \arg\max_c r_{\theta,s,c},$$

which is used only to compute local evidence. The locality weights, local categorical distributions, field V, and tilted target r^+ are all computed under a stop-gradient. Thus gradients flow only through the final cross-entropy/KL loss against the frozen target.

Across all tasks, the locality feature map ϕ is given by a separately trained masked autoencoder (MAE) (He et al., 2022). The MAE is trained on the corresponding training split and then frozen; CatDrift uses its encoder features only for neighborhood lookup. Unless otherwise stated, distances are computed in MAE feature space,

$$d(x, x') = \|\phi(x) - \phi(x')\|_2,$$

and are converted into local affinities before estimating the positive and negative categorical evidence.

Data/model particles and multi-temperature affinities. Each update uses $N_+ = N_- = B$: a minibatch of B training examples and the current batch of B generated hard samples. Data particles attract and model particles repel; diagonal model self-neighbors are masked out. Smaller batches yield noisier, less representative fields, while very large batches reduce useful stochasticity.

For robustness, we use a small set of locality temperatures

$$\mathcal{T} = \{0.02, 0.05, 0.2\}.$$

Let d_{bm} be the distance between generated query b and data or model particle m, normalized by the mean distance in the current update. For each $\tau \in \mathcal{T}$ we compute logits $\ell^{(\tau)}_{bm} = -d_{bm}/\tau$ and use the symmetrized double-softmax affinity

$$A^{(\tau)}_{bm} = \left[\mathrm{softmax}_m(\ell^{(\tau)}_{bm})\,\mathrm{softmax}_b(\ell^{(\tau)}_{bm})\right]^{1/2}.$$

The affinities are then renormalized separately over positive and negative neighbors to form the weights used in the local categorical frequencies. For each site s and category c,

$$\hat{p}^{+,\tau}_{s,c} = \sum_i \alpha^{(\tau)}_i \mathbf{1}\{x^+_{i,s} = c\}, \qquad \hat{p}^{-,\tau}_{s,c} = \sum_j \beta^{(\tau)}_j \mathbf{1}\{x^-_{j,s} = c\}.$$

We smooth both distributions with $\lambda = 10^{-3}$,

$$p^{+,\tau}_s = (1-\lambda)\hat{p}^{+,\tau}_s + \lambda u_K, \qquad p^{-,\tau}_s = (1-\lambda)\hat{p}^{-,\tau}_s + \lambda u_K, \qquad u_K = \frac{1}{K}\mathbf{1}.$$

The corresponding field is

$$V^\tau_{s,c} = \log p^{+,\tau}_{s,c} - \log p^{-,\tau}_{s,c}.$$

For numerical stability, each V^τ_s is centered across categories and RMS normalized. The final field is the sum over temperatures,

$$V_s = \sum_{\tau\in\mathcal{T}} V^\tau_s.$$

The frozen target is then

$$r^+_s = \mathrm{sg}[\mathrm{softmax}(\mathrm{sg}(z_{\theta,s}) + \eta\ \mathrm{sg}(V_s))],$$

and the generator is trained with

$$\mathcal{L}_{\mathrm{CatDrift}} = \sum_s \mathrm{KL}\big(r^+_s \,\|\, r_{\theta,s}\big).$$

Training cost. At fixed object and feature dimensions, field construction uses $O(B^2)$ pairwise interactions per update, in addition to generator and feature-extractor passes; inference requires neither interactions nor kernel evaluations. Experiments ran on a single NVIDIA RTX A6000 and completed within hours.

Optimization. For the image experiments, we train with AdamW (Kingma & Ba, 2015; Loshchilov & Hutter, 2019) using learning rate $2 \cdot 10^{-4}$, betas $(0.9, 0.95)$, weight decay 0.01, batch size 64, and global gradient clipping at 1.0. The learning rate is linearly warmed up from 10^{-6} to the base learning rate over 5000 steps and then kept constant. We maintain an exponential moving average of generator parameters with decay 0.999 and use the EMA weights for evaluation. Unless stated otherwise, reported image results use seed 42.

Image-generator architecture. For discretized images, g_θ is a patch-based bidirectional Transformer. The input noise has shape $H \times W \times d_z$ with $d_z = 4$. We patchify the noise with patch size 4, map patches to a hidden dimension d_{model}, prepend 8 learned register tokens, add fixed two-dimensional sinusoidal positional embeddings, and apply a stack of Transformer blocks with RMSNorm, QK-normalized self-attention, RoPE, and SwiGLU MLPs with expansion factor 4. The final head maps each patch back to sitewise logits. We use dropout 0.

For binarized MNIST, we use image size 28, vocabulary size $K = 2$, $d_{\text{model}} = 256$, 6 Transformer layers, and 4 attention heads. We train for 50000 steps. For CIFAR-10, we use image size 32, three color channels, $d_{\text{model}} = 384$, 8 Transformer layers, and 6 attention heads, and vocabulary size $K = 256$. We train for 100000 steps.

MAE feature extractors. The MNIST MAE uses patch size 4, mask ratio 0.5, an encoder with hidden size 384, depth 8, and 6 heads, and a decoder with hidden size 256, depth 4, and 4 heads. It is trained for 200 epochs with AdamW, learning rate $1.5 \cdot 10^{-4}$, betas $(0.9, 0.95)$, weight decay 0.05, batch size 256, gradient clipping at 1.0, EMA decay 0.9995, and cosine decay with a 5% warmup. At CatDrift training time, we encode unmasked inputs, mean-pool the final encoder sequence, and ℓ_2-normalize the resulting feature vector.

Evaluation and checkpointing. During training, we evaluate every 500 steps. For image experiments we generate 4096 samples for diagnostics and compute FID using 1000 generated samples and 1000 held-out real samples. Samples are generated both by argmax decoding and by stochastic categorical sampling; the main reported FID uses argmax decoding. We save the latest checkpoint and the best EMA checkpoint according to validation FID. Early stopping is enabled only after 5000 steps and triggers after 10 evaluations without improvement.

D.1 Discretized image generation

For binarized MNIST LeCun et al. (n.d.), each pixel is represented as a binary categorical variable. For discretized CIFAR-10 Krizhevsky (2009), each color channel is discretized into 256 bins, giving 256 categorical classes per channel. In both cases, the generator outputs per-site logits and generates the full image in a single forward pass. The hard state used for neighborhood lookup is obtained by taking the per-site $\arg\max$.

Local evidence uses $N_+ = N_- = B$ data and model particles in MAE feature space, where B is the CatDrift batch size. FID is computed from generated samples using the standard Fréchet Inception Distance protocol (Heusel et al., 2017).

D.2 DNA sequence generation

For promoter DNA generation, each sample is a length-1024 sequence over $\{A, C, G, T\}$. We use the FANTOM5 promoter dataset Hon et al. (2017) and follow the standard train/validation/test split used in prior promoter-design work. The generator outputs per-position logits over the four nucleotides and generates the entire sequence in one forward pass.

The locality feature map ϕ is the encoder of a masked autoencoder trained on promoter sequences. The kernel k is a Gaussian kernel in the resulting MAE feature space. Local evidence uses $N_+ = N_- = B$

data and model sequences per update, where B is the CatDrift batch size, with additive smoothing λ applied before taking log-ratios. CatDrift is trained from scratch and does not use a pretrained teacher, a distilled diffusion or flow model, or any post-hoc adaptation stage.

Following the promoter DNA setup of Riemannian MeanFlow, we evaluate generated sequences using two metrics: MSE between Sei-predicted regulatory signal profiles Chen et al. (2022) and Pearson correlation between generated and real 6-mer distributions.

D.3 MOLECULAR GENERATION

For molecular generation, we evaluate on QM9 (Ramakrishnan et al., 2014) and ZINC250k (Gómez-Bombarelli et al., 2018) following the SMILES-based setting used by PairFlow (Park et al., 2026). The generator outputs per-token logits and generates the full SMILES string in a single forward pass. The hard string used for neighborhood lookup is obtained by taking the per-token $\arg\max$.

The locality feature map ϕ is the encoder of a masked autoencoder trained on the corresponding SMILES training set. The kernel k is a Gaussian kernel in MAE feature space. Local evidence uses $N_+ = N_- = B$ data and model molecules per update, where B is the CatDrift batch size, with additive smoothing λ applied before taking log-ratios.

Following PairFlow, we report validity and uniqueness over 1024 generated SMILES strings. Uniqueness is counted among the valid samples.