Online Distributional Prediction via Latent Cluster Geometry Under Drift and Corruption

Anonymous authors
Paper under double-blind review

Abstract

Online learning in non-stationary streams is often formulated as tracking a point estimate, but many applications require predicting the full data-generating distribution. We study online distributional prediction under drift and adversarial corruption. Our approach represents each candidate law through a latent cluster geometry: a variable-size configuration of centers that organizes probability mass and induces a predictive distribution. A Gibbs quasi-posterior over these configurations yields an online predictor by posterior averaging, and the resulting variable-dimensional posterior can be sampled with reversible-jump MCMC. The method therefore avoids specifying a parametric streaming law while retaining a structured latent space for uncertainty, regularization, and comparison.

We evaluate performance by cumulative Wasserstein-1 regret against the time-varying true law. The analysis separates two effects: corruption perturbs the loss-based posterior update, whereas drift makes long-horizon posterior memory stale. We address the latter with a restarted variant that temporally localizes the same quasi-Bayesian update. The resulting high-probability bounds decompose into a PAC-Bayesian complexity term, a corruption-sensitive posterior perturbation term, and a dynamic optimal-transport term driven by $A_T^{\mathrm{OT}} = \sum_{t=2}^{T} W_2^2(p_{t-1}^*, p_t^*)$. Under bounded support, stable latent geometry, predictive-map regularity, oracle realizability, localized restart windows, sublinear transport action, and sublinear corruption budget, the restarted predictor achieves sublinear cumulative Wasserstein regret. These guarantees require no parametric model for the stream, drift mechanism, or corruption process.

1 Introduction

The ubiquitous presence of high-dimensional streaming data from sensors, networks, and online systems presents significant challenges for real-time statistical estimation. Applications such as network monitoring, anomaly detection, and online recommendation systems require algorithms that process data sequentially while adapting to changing environments (Aceto et al., 2013; Baraniuk, 2011; Dartmann et al., 2019; Mazhar & Shafiq, 2018; van Erven et al., 2021). Real streams also exhibit features that violate many standard assumptions: heavy tails, evolving distributions, and adversarial corruptions (Aghdai et al., 2013; Alizadeh et al., 2013; Herley, 2022; Ditzler & Polikar, 2013; Li et al., 2021; Mirsky et al., 2018; Ren et al., 2019; Sankararaman et al., 2022).

Most online estimation methods reduce this problem to estimating a point quantity such as a mean, a parameter vector, or a finite-dimensional statistic. This reduction is often too narrow. In settings such as risk control, anomaly detection, and uncertainty quantification, the object of interest is the predictive *distribution* itself. The question is therefore not only what the next observation is likely to be, but how probability mass is organized and how that organization changes over time.

The central idea of this paper is to model this organization through a latent clustering configuration space. Instead of placing a model directly on an unrestricted sequence of distributions, we represent each possible explanation of the current data law by a cluster-center configuration. This configuration acts as a latent geometry for the stream: it summarizes where probability mass is concentrated, induces a predictive law in observation space, and provides the object over which the learner can regularize, compare, and update.

The quasi-Bayesian component is the mechanism used to maintain uncertainty over this latent clustering space: a Gibbs quasi-posterior assigns weights to candidate configurations according to their accumulated clustering loss.

This latent-space viewpoint also identifies the main obstruction under non-stationarity. If the data law drifts, the relevant cluster geometry changes; a long-horizon posterior over configurations can therefore retain stale geometric information. Adversarial corruption perturbs the loss-based update, while persistent drift makes a single global comparator inadequate for a moving latent configuration. This motivates a restarted version of the same clustering predictor: the algorithm keeps the same latent configuration space and prior, but localizes memory in time so that the comparison is made against short-window cluster geometries. We analyze this temporal localization through dynamic optimal transport, which measures the cumulative movement of the underlying data law.

Contributions.

- We introduce a latent clustering-configuration view of online distributional prediction, where cluster-center configurations serve as structured latent geometries for the evolving data law.
- We derive high-probability regret bounds that explicitly expose how distributional prediction is affected by drift, corruption frequency, corruption magnitude, and data dimension through the corresponding transport, perturbation, and PAC-Bayesian complexity terms.
- We identify high-probability sublinear cumulative Wasserstein-regret regimes for the restarted predictor, showing that temporal localization of latent clustering memory can overcome the stale-geometry obstruction under controlled drift and corruption.

The quasi-Bayesian posterior update, reversible-jump sampling scheme, and dynamic optimal transport analysis are the technical mechanisms used to realize and analyze this latent-space formulation. Algorithmically, the predictor does not require a parametric likelihood for the streaming law, drift process, or corruption mechanism. The theoretical guarantees are proved under the structural latent-geometry assumptions stated in Section 3.

2 Related Work

Online robust estimation. Sankararaman & Narayanaswamy (2023) propose clipped SGD for online mean estimation under heavy tails, corruption, and non-stationarity, achieving $\mathcal{O}(\sqrt{T} + \Lambda_T T^{1/4})$ regret. Their bounds depend on explicit drift Φ_T and require strong convexity. We study a different distributional-prediction criterion and analyze it under latent-geometry assumptions rather than strong convexity of a mean-estimation objective. Heavy-tailed streaming estimation has been studied by Tsai et al. (2022), Liu & Zhou (2023), and Gorbunov et al. (2020), all within the parameter-estimation paradigm.

PAC-Bayesian and quasi-Bayesian methods. Our algorithmic foundation builds on the quasi-Bayesian online clustering of Li et al. (2016), which establishes PAC-Bayes bounds for Gibbs posteriors over cluster configurations. We extend their framework to distributional prediction with Wasserstein regret under adversarial corruption and non-stationarity. PAC-Bayesian bounds under hostile data are studied by van Erven et al. (2021).

Online convex optimization and dynamic regret. The connection to online mirror descent places our work in the broader OCO framework (Hazan, 2023; Zinkevich, 2003). Dynamic regret in non-stationary settings is studied by Besbes et al. (2015) and Baby & Wang (2021). Mirror descent with heavy-tailed noise is analyzed by Vural et al. (2022) and Nazin et al. (2019). Our contribution is to identify the KL-regularized OMD on the simplex (quasi-Bayesian) as a distributional analog of Euclidean OMD (clipped SGD), with complementary robustness properties.

Optimal transport and distribution drift. Optimal transport provides a natural geometry for comparing probability distributions and has become a central tool in modern distributional analysis. In particular, the dynamic formulation of transport due to Benamou & Brenier (2000) characterizes Wasserstein transport through an action minimization subject to the continuity equation, while the variational viewpoint of Jordan et al. (1998) has motivated a broad literature on evolution in Wasserstein space. Our use of dynamic optimal transport is different in emphasis: rather than employing transport as an optimization primitive, we use cumulative transport action as a complexity measure for non-stationary distribution drift in online

prediction. This allows us to express the cost of environment evolution directly at the level of probability laws, rather than only through parameter drift or comparator variation.

3 Problem Setup and Preliminaries

We study online prediction of a time-varying distributional stream under corruption. The main modeling decision is to learn in a structured *configuration space* while evaluating performance in *data space*. This section formalizes that setup and records the immediate consequences used later in the regret analysis.

3.1 Sequential Distributional Prediction

At each round $t \in [T]$, an uncorrupted sample $x_t \sim \mathbb{P}_t$ is drawn from the current data-generating law $\mathbb{P}_t$ on $\mathbb{R}^d$. The learner does not observe x_t directly. Instead it receives a corrupted observation

$$\tilde{x}_t = x_t + C_t,$$

where the corruption process is *causal*:

$$C_t = \mathcal{C}_t\big((x_s)_{s=1}^{t}, (C_s)_{s=1}^{t-1}\big).$$

Thus the adversary may depend on the entire revealed past and the current clean point, but not on future observations. We assume bounded corruption amplitude,

$$\|\tilde{x}_t - x_t\|_2 \le \delta,$$

and denote the cumulative corruption budget by

$$\Lambda_T := \sum_{t=1}^{T} \mathbf{1}\{C_t \neq 0\}.$$

Our global structural setting is that all true laws are supported on a common compact set of diameter D (equivalently, on a ball of radius $R = D/2$). This bounded-support assumption is the main regularity condition used throughout the paper; we do not assume a parametric family for $\mathbb{P}_t$, a prescribed drift model, or separate moment-growth conditions.

The estimator outputs a predictive law in data space. Given clean history $x_{1:t-1}$, let $p(x_t \mid x_{1:t-1})$ denote the learner's predictive distribution, and let $p_t^* := \mathbb{P}_t$ denote the true law at time t. We measure performance by cumulative Wasserstein-1 regret:

$$W_{\text{Regret}}^{\text{clean}}(T) := \sum_{t=1}^{T} W_1(p(x_t \mid x_{1:t-1}),\, p_t^*)\,. \tag{1}$$

Under corrupted history $\tilde{x}_{1:t-1}$, the learner instead forms $p(x_t \mid \tilde{x}_{1:t-1})$, and the total regret is

$$W_{\text{Regret}}^{\text{total}}(T) := \sum_{t=1}^{T} W_1(p(x_t \mid \tilde{x}_{1:t-1}),\, p_t^*)\,. \tag{2}$$

This criterion is strictly stronger than parameter-level control: for any two distributions P, Q with finite first moments,

$$\|\mathbb{E}_P[X] - \mathbb{E}_Q[X]\|_2 \le W_1(P, Q).$$

So controlling distributional error in W_1 automatically controls mean error, whereas the converse need not hold.

3.2 Latent Clustering Representation

The central modeling choice is to represent an evolving law through a lower dimensional latent geometry. A cluster-center configuration is denoted by $\boldsymbol{c} = (c_1, \ldots, c_k)$, and the clustering loss is

$$\ell(\boldsymbol{c}, x) := \min_j \|c_j - x\|_2^2. \tag{3}$$

The corresponding population risk under the true law at time t is

$$R_t(\boldsymbol{c}) := \mathbb{E}_{X \sim p_t^*}[\ell(\boldsymbol{c}, X)]. \tag{4}$$

We denote by

$$\boldsymbol{c}_t^* \in \arg\min_{\boldsymbol{c}} R_t(\boldsymbol{c})$$

the time-t oracle configuration. This oracle is an ideal benchmark: it is defined using the true law and is therefore available to the analysis but not to the learner.

The point of clustering here is not merely descriptive. Each configuration serves as a latent explanation of how probability mass is organized, and each such explanation induces a predictive law

$$\boldsymbol{c} \mapsto p(\cdot \mid \boldsymbol{c}),$$

so the method learns a distribution over latent geometries and pushes that uncertainty forward into data space.

This viewpoint is easiest to understand by comparison with ordinary Bayesian inference. In a fully specified Bayesian latent-variable model, one would place a prior $\pi(d\boldsymbol{c})$ on configurations and update it by multiplying with a likelihood term. Here we retain the prior over latent configurations, but do not insist on a fully correct parametric likelihood for the streaming law. Instead, following the quasi-Bayesian clustering framework of Li et al. (2016), we update configurations through a loss-based Gibbs weight:

$$\hat{\rho}_{t+1}(d\boldsymbol{c}) \propto \exp\big(-\lambda_t S_t(\boldsymbol{c})\big)\, \pi(d\boldsymbol{c}), \tag{5}$$

where $S_t(\boldsymbol{c})$ is the cumulative clustering score after observing the first t samples and λ_t is the learning-rate parameter. Thus $\hat{\rho}_{t+1}$ is used to predict the next observation, not the observation already included in S_t. The resulting object is a *quasi-posterior*: it retains the Bayesian architecture of prior, posterior, and predictive distribution, but the data-fit term is generated by the clustering loss rather than by a fully specified likelihood.

Prediction then takes place by averaging over latent configurations:

$$\hat{p}_{t+1}(\cdot) := \int p(\cdot \mid \boldsymbol{c})\, \hat{\rho}_{t+1}(d\boldsymbol{c}).$$

In particular, the prior, quasi-posterior, and KL regularization all live in configuration space, while the regret is measured in observation space. This separation is what makes the framework both statistically interpretable and mathematically tractable: learning is performed in a structured latent space, while accuracy is evaluated on the predictive law in data space.

3.3 Configuration Space and Prior Structure

The latent space itself is common to all algorithmic variants considered in the paper, so we record it here once. For each admissible cluster count $k \in \{1, \ldots, p_{\max}\}$, let

$$\mathcal{C}_k \subset (\mathbb{R}^d)^k$$

denote the set of k-center configurations allowed by the bounded-support model. The full latent configuration space is the union

$$\mathcal{C} := \bigcup_{k=1}^{p_{\max}} \big(\{k\} \times \mathcal{C}_k\big).$$

Thus a latent state consists of both a model order k and a center geometry $\boldsymbol{c} \in \mathcal{C}_k$.

This representation is shared by the base and restarted variants. What changes across those variants is the way the learner handles temporal memory, not the space itself.

We place a prior π on $\mathcal{C}$ that encodes uncertainty both about the number of active clusters and about their locations. Following Li et al. (2016), we write it as

$$\pi(\boldsymbol{c}) = \sum_{k=1}^{p_{\max}} q(k)\, \mathbf{1}_{\{\boldsymbol{c} \in \mathcal{C}_k\}}\, \pi_k(\boldsymbol{c}), \tag{6}$$

where q is a prior on the cluster count and π_k is a location prior on $\mathcal{C}_k$. A standard choice is

$$q(k) = \frac{\exp(-\eta k)}{\sum_{i=1}^{p_{\max}} \exp(-\eta i)},$$

with each π_k supported on the bounded configuration set.

All later PAC-Bayes comparisons, localized comparator constructions, and restart arguments are carried out on this common configuration space with respect to the prior architecture equation 6.

3.4 Drift, Lagged Risk, and Transport Complexity

The dynamic analysis compares the learner at time t not only to the current law p_t^*, but also to the previous law p_{t-1}^*. For this reason we introduce the lagged risk

$$\bar{R}_t(\boldsymbol{c}) := \mathbb{E}_{X \sim p_{t-1}^*}[\ell(\boldsymbol{c}, X)]. \tag{7}$$

This is simply the previous-step population risk, written inside the time-t argument. Its role is to separate two effects:

1. how well the learner tracks yesterday's law, and
2. how much the environment itself moves from $t-1$ to t.

The second effect is quantified directly at the distribution level through the cumulative transport action

$$A_T^{\mathrm{OT}} := \sum_{t=2}^{T} W_2^2(p_{t-1}^*, p_t^*). \tag{8}$$

This quantity plays the role of an environment complexity measure: small A_T^{OT} corresponds to gradual temporal evolution, whereas large A_T^{OT} corresponds to abrupt or persistent drift. The analogy with the dynamic formulation of optimal transport is useful here: $W_2^2(p_{t-1}^*, p_t^*)$ can be read as the least transport energy needed to move probability mass from yesterday's law to today's law, under a mass-conserving flow. We do not solve this continuous flow problem in the algorithm. We only use the resulting discrete-time action A_T^{OT} as a compact way to measure how much distributional drift the environment spends over the horizon.

3.5 Structural Assumptions

Beyond bounded support, the proof uses three structural assumptions that encode stability of the latent geometry and regularity of the predictive map. We state them in the form actually used later in the proofs, but it is helpful to note the more primitive conditions from which these working inequalities arise.

Stable local cluster geometry. Around the lagged oracle, the population risk should be locally well-shaped. At a more primitive level, one may think of this as a local identifiability condition: the oracle configuration is an isolated local minimizer of the lagged risk landscape. A sufficient smooth formulation is

$$\nabla \bar{R}_t(\boldsymbol{c}_{t-1}^*) = 0, \qquad \nabla^2 \bar{R}_t(\boldsymbol{c}_{t-1}^*) \succeq \mu I$$

for some $\mu > 0$ in a neighborhood of $\boldsymbol{c}_{t-1}^*$. By a local Taylor expansion, this implies quadratic growth. In the proofs we work directly with the resulting inequality

$$\bar{R}_t(\boldsymbol{c}) - \bar{R}_t(\boldsymbol{c}_{t-1}^*) \geq \mu \|\boldsymbol{c} - \boldsymbol{c}_{t-1}^*\|_2^2 \tag{9}$$

for some $\mu > 0$. Semantically, this means that each time slice admits a locally stable and non-ambiguous cluster geometry: perturbing the centers away from the oracle must increase risk in every direction. This is the mechanism that later converts excess risk into geometric error in center space.

Predictive-map regularity. The second ingredient is a bridge from latent geometry to observation-space prediction. We assume the predictive law varies Lipschitzly with the configuration:

$$W_1(p(\cdot \mid \boldsymbol{c}),\, p(\cdot \mid \boldsymbol{c}')) \le L_K \|\boldsymbol{c} - \boldsymbol{c}'\|_2. \tag{10}$$

This can itself be motivated from the translated-kernel picture below. For example, if

$$p(\cdot \mid \boldsymbol{c}) = \sum_{j=1}^{p_{\max}} \omega_j K(\cdot, c_j)$$

with common weights $(\omega_j)_j$ and $K(\cdot, c) = \mathrm{Law}(\boldsymbol{Z} + c)$, then coupling the same latent base variable $\boldsymbol{Z}$ across the two configurations gives

$$W_1(p(\cdot \mid \boldsymbol{c}),\, p(\cdot \mid \boldsymbol{c}')) \le \sum_{j=1}^{p_{\max}} \omega_j \|c_j - c_j'\|_2 \le L_K \|\boldsymbol{c} - \boldsymbol{c}'\|_2$$

for a suitable norm-dependent constant L_K. In the analysis we keep this as an abstract regularity assumption. Its meaning is that small perturbations of the latent geometry produce proportionally small perturbations of the predicted law.

To make this bridge target the true law, rather than only another model-induced predictive law, we assume that the time-varying data law is realized by the oracle latent path:

$$p_t^* = p(\cdot \mid \boldsymbol{c}_t^*), \qquad t = 1, \dots, T. \tag{11}$$

Here the same oracle configuration $\boldsymbol{c}_t^*$ is used in the clustering-risk comparison and in the predictive map. This is a well-specified latent-geometry assumption for the regret analysis; the learner does not observe $\boldsymbol{c}_t^*$. Under model misspecification, the same argument would acquire an additional approximation term of the form $\sum_t W_1(p(\cdot \mid \boldsymbol{c}_t^*), p_t^*)$.

Equations equation 10 and equation 11 provide the bridge from latent-space accuracy to distributional accuracy in data space: if the learner concentrates near the oracle configuration, then the induced predictive distribution must also be close to the true law.

Common translated-kernel representation. For the explicit dynamic-OT interpretation, we assume the true law can be written as a mixture of translated copies of a common kernel:

$$p_t^* = \sum_{j=1}^{p_{\max}} w_{t,j} K(\cdot, c_{t,j}^*), \qquad K(\cdot, c) = \mathrm{Law}(\boldsymbol{Z} + c). \tag{12}$$

This assumption is not a parametric drift model. Rather, it gives a geometric representation in which temporal evolution can be understood through center motion, weight reallocation, and birth or death of active components. Its main technical role is that it makes the OT action explicitly comparable to center motion. Indeed, if $\Pi_t \in \Gamma(w_{t-1}, w_t)$ couples the mixture weights at times $t-1$ and t, then transporting the common base variable $\boldsymbol{Z}$ along the coupled components yields

$$W_2^2(p_{t-1}^*, p_t^*) \le \sum_{i,j} \Pi_{t,ij} \|c_{t-1,i}^* - c_{t,j}^*\|_2^2,$$

and therefore

$$W_2^2(p_{t-1}^*, p_t^*) \le \min_{\Pi_t \in \Gamma(w_{t-1}, w_t)} \sum_{i,j} \Pi_{t,ij} \|c_{t-1,i}^* - c_{t,j}^*\|_2^2. \tag{13}$$

So the translated-kernel representation is what lets the dynamic OT quantity inherit a direct geometric interpretation in terms of evolving cluster configurations.

3.6 Immediate Consequences Used Later

The preceding setup immediately yields the facts repeatedly used in the proofs.

Loss stability in the sample variable. Under the bounded-support convention, for each fixed configuration $\boldsymbol{c}$ the map $x \mapsto \ell(\boldsymbol{c}, x)$ is $4D$-Lipschitz:

$$|\ell(\boldsymbol{c}, x) - \ell(\boldsymbol{c}, x')| \leq 4D \, \|x - x'\|_2.$$

Consequently, Kantorovich–Rubinstein duality implies

$$\Big| \mathbb{E}_{p^*_{t-1}}[\ell(\boldsymbol{c}, X)] - \mathbb{E}_{p^*_t}[\ell(\boldsymbol{c}, X)] \Big| \leq 4D \, W_1(p^*_{t-1}, p^*_t). \tag{14}$$

This is the basic way in which distribution drift enters the risk analysis: changes in the law induce controlled changes in the risk landscape.

Geometric control from excess risk. The quadratic-growth assumption yields

$$\|\boldsymbol{c} - \boldsymbol{c}^*_{t-1}\|_2 \leq \frac{1}{\sqrt{\mu}} \sqrt{\bar{R}_t(\boldsymbol{c}) - \bar{R}_t(\boldsymbol{c}^*_{t-1})}. \tag{15}$$

Thus small lagged excess risk implies small geometric error in configuration space. Combined with predictive-map regularity and oracle realizability, this gives a route from population risk control to Wasserstein prediction error.

Pathwise control of drift. Since $W_1 \leq W_2$, the cumulative transport action controls the total distributional motion:

$$\sum_{t=2}^{T} W_1(p^*_{t-1}, p^*_t) \leq \sum_{t=2}^{T} W_2(p^*_{t-1}, p^*_t) = \sum_{t=2}^{T} \sqrt{W_2^2(p^*_{t-1}, p^*_t)}.$$

Applying Cauchy–Schwarz to the sequence $\big(\sqrt{W_2^2(p^*_{t-1}, p^*_t)}\big)_{t=2}^{T}$ gives

$$\sum_{t=2}^{T} \sqrt{W_2^2(p^*_{t-1}, p^*_t)} \leq \sqrt{(T-1) \sum_{t=2}^{T} W_2^2(p^*_{t-1}, p^*_t)} = \sqrt{(T-1) A_T^{\mathrm{OT}}}. \tag{16}$$

This is the key reason dynamic OT enters the analysis: it converts the temporal evolution of the environment into a quantitative pathwise penalty that can be combined with the quasi-Bayesian risk bounds.

Why restart will appear later. The fixed-comparator quasi-Bayesian theorem is naturally aligned with a static benchmark, while our target oracle $\boldsymbol{c}^*_t$ moves with time. Restart-based segmentation resolves this mismatch by replacing one whole-horizon comparator with a sequence of local fixed comparators. The later restart analysis will therefore use segment-start oracles and localized comparator neighborhoods in configuration space rather than a single global point-mass benchmark.

4 Quasi-Bayesian Clustering Algorithms

4.1 Common Quasi-Bayesian Framework

We now turn from the common modeling setup to the learning rules built on top of it. The base and restarted procedures share the same latent space, prior architecture, and predictive target. In each case, the learner maintains a probability distribution on the common configuration space $\mathcal{C}$ and uses it to produce a predictive law in data space.

The shared quasi-Bayesian template is the following. Let $S_t(\boldsymbol{c})$ denote a cumulative score assigned to configuration $\boldsymbol{c}$ after observing the stream up to time t. The learner forms a Gibbs-type update for the next prediction,

$$\hat{\rho}_{t+1}(d\boldsymbol{c}) = \frac{\exp\big(-\lambda_t S_t(\boldsymbol{c})\big) \, \pi(d\boldsymbol{c})}{\int_{\mathcal{C}} \exp\big(-\lambda_t S_t(\boldsymbol{c}')\big) \, \pi(d\boldsymbol{c}')}, \tag{17}$$

where π is the prior from equation 6 and $\lambda_t > 0$ is the learning-rate or inverse-temperature parameter. The associated predictive distribution is then

$$\hat{p}_{t+1}(\cdot) = \int_{\mathcal{C}} p(\cdot \mid \boldsymbol{c})\, \hat{\rho}_{t+1}(d\boldsymbol{c}). \tag{18}$$

What distinguishes the algorithmic variants is not the latent representation itself, but the way historical memory is handled:

1. the *base* update uses the raw clustering loss accumulated over the full history;
2. the *restart* mechanism keeps the same latent model and prior but periodically resets the quasi-posterior so that stale history does not dominate under drift.

Thus the restarted variant should be understood as a temporal localization of the same quasi-Bayesian architecture: the latent configuration space, the prior, the score, and the predictive map remain fixed, while the memory window changes to address non-stationarity.

4.2 Vanilla Update

The baseline learner instantiates the common framework with a first-order Gibbs score over cluster configurations. Let z_t denote the observation available at time t. In the algorithmic description below, z_t may be read as the incoming stream seen by the learner. In the clean analysis it will later be instantiated as x_t, while in the corruption-aware version it will be instantiated as $\tilde{x}_t$.

Initialize $S_0(\boldsymbol{c}) = 0$. For $t \geq 1$, define the cumulative score recursively by

$$S_t(\boldsymbol{c}) := S_{t-1}(\boldsymbol{c}) + \ell(\boldsymbol{c}, z_t). \tag{19}$$

The corresponding Gibbs quasi-posterior at round $t+1$ is

$$d\hat{\rho}_{t+1}(\boldsymbol{c}) = \frac{\exp\big(-\lambda_t S_t(\boldsymbol{c})\big)}{\int_{\mathcal{C}} \exp\big(-\lambda_t S_t(\boldsymbol{c}')\big)\, d\pi(\boldsymbol{c}')}\, d\pi(\boldsymbol{c}), \tag{20}$$

or equivalently $d\hat{\rho}_{t+1}(\boldsymbol{c}) \propto \exp(-\lambda_t S_t(\boldsymbol{c}))\, d\pi(\boldsymbol{c})$. The predictive distribution used before observing the next sample is

$$p(z_{t+1} \mid z_{1:t}) = \int_{\mathcal{C}} p(z_{t+1} \mid \boldsymbol{c})\, \hat{\rho}_{t+1}(\boldsymbol{c})\, d\boldsymbol{c}. \tag{21}$$

This is a first-order version of the quasi-Bayesian clustering update. The second-order variance correction appearing in the original online clustering construction of Li et al. (2016) is not included in the score. In the present paper, the score is kept aligned with the cumulative data-fit loss, while temporal instability is handled through restart and the dynamic transport term.

Algorithm 1: Vanilla Quasi-Bayesian Clustering Update

Input: $p_{\max}$, prior $\pi \in \mathscr{P}(\mathcal{C})$, learning-rate schedule $\{\lambda_t\}_{t=0}^{T-1}$, initial score $S_0 = 0$
Output: Predictive distributions $\{p(z_{t+1} \mid z_{1:t})\}_{t=0}^{T-1}$

```
for t ← 0 to T − 1 do
    Form the quasi-posterior dρ̂_{t+1}(c) ∝ exp(−λ_t S_t(c)) dπ(c);
    Draw cluster configurations from ρ̂_{t+1} via RJMCMC;
    Output the predictive law p(z_{t+1} | z_{1:t}) = ∫ p(z_{t+1} | c) ρ̂_{t+1}(c) dc;
    Observe the next sample z_{t+1};
    Update the cumulative score using equation 19;
```

Because the latent space $\mathcal{C}$ includes configurations with different numbers of clusters, the target law in equation 20 is a variable-dimension distribution. Following Li et al. (2016), we approximate it by reversible-jump MCMC. At each round, the chain targets $\hat{\rho}_{t+1}$ and uses three types of moves: birth moves that increase the cluster count, death moves that decrease it, and within-model updates that modify the center locations while keeping k fixed. The normalizing constant in equation 20 need not be computed explicitly, since it cancels in the Metropolis–Hastings acceptance ratio.

If $\boldsymbol{c}^{(1)}, \ldots, \boldsymbol{c}^{(M)}$ are RJMCMC draws from $\hat{\rho}_{t+1}$ after burn-in, then the predictive integral in equation 21 is approximated by the empirical average

$$p(z_{t+1} \mid z_{1:t}) \approx \frac{1}{M} \sum_{m=1}^{M} p(z_{t+1} \mid \boldsymbol{c}^{(m)}). \tag{22}$$

In other words, the algorithm first samples plausible latent cluster configurations from the quasi-posterior and then averages the corresponding configuration-wise predictive laws. This is the practical mechanism through which uncertainty in configuration space is pushed forward into a predictive distribution in data space.

4.3 Restarted Update

The second modification concerns temporal memory rather than the loss itself. Fix restart times

$$1 = \tau_0 < \tau_1 < \cdots < \tau_m = T + 1$$

and let

$$I_r := \{\tau_{r-1}, \ldots, \tau_r - 1\}$$

denote segment r. On each segment, the quasi-posterior is reinitialized and updated only with the observations observed since the most recent restart.

For the restarted version, the same causal convention is applied locally within each segment. For $t \in I_r$, define the within-segment age

$$a_r(t) := t - \tau_{r-1},$$

and define the segmentwise past score, with the empty sum equal to zero, by

$$S_{t-1}^r(\boldsymbol{c}) := \sum_{s=\tau_{r-1}}^{t-1} \ell(\boldsymbol{c}, z_s), \qquad t \in I_r. \tag{23}$$

Before observing z_t, the restarted quasi-posterior on segment r is then

$$d\hat{\rho}_t^r(\boldsymbol{c}) = \frac{\exp\big(-\lambda_{a_r(t)} S_{t-1}^r(\boldsymbol{c})\big)}{\int_{\mathcal{C}} \exp\big(-\lambda_{a_r(t)} S_{t-1}^r(\boldsymbol{c}')\big)\, d\pi(\boldsymbol{c}')}\, d\pi(\boldsymbol{c}), \qquad t \in I_r. \tag{24}$$

After z_t is observed, the score is updated to $S_t^r(\boldsymbol{c}) = S_{t-1}^r(\boldsymbol{c}) + \ell(\boldsymbol{c}, z_t)$ and used only for later predictions in the same segment.

The predictive law is obtained by integrating the same map $p(\cdot \mid \boldsymbol{c})$ against the restarted quasi-posterior on the current segment:

$$p(z_t \mid z_{\tau_{r-1}:t-1}) = \int_{\mathcal{C}} p(z_t \mid \boldsymbol{c})\, \hat{\rho}_t^r(d\boldsymbol{c}), \qquad t \in I_r.$$

The role of restart is therefore purely temporal: it limits how much past data can influence the current update, without changing the latent representation, loss, prior, or configuration-wise predictive map.

5 Theoretical Guarantees

5.1 Total Regret

We now state the total-regret guarantees for the base and restarted predictors introduced in Section 4. To keep the theorem statements readable, we first isolate the corruption-side and clean-side terms that recur in both cases. The terminology follows the decomposition in Appendix A. At time t, the learner using the corrupted stream predicts from $p(x_t \mid \tilde{x}_{1:t-1})$, while the ideal clean-history learner would predict from $p(x_t \mid x_{1:t-1})$. Thus the total Wasserstein regret is controlled by two effects:

$$W_1(p(x_t \mid \tilde{x}_{1:t-1}), p_t^*) \leq W_1(p(x_t \mid \tilde{x}_{1:t-1}), p(x_t \mid x_{1:t-1})) + W_1(p(x_t \mid x_{1:t-1}), p_t^*)\,.$$

The first term is the corruption-side contribution: it measures how much the posterior predictive law changes because the history was corrupted. The second term is the clean-side contribution: it measures how well the clean quasi-Bayesian predictor tracks the moving true law p_t^* under drift. The clean dynamic analysis starts from $t = 2$, so we retain the initial prediction error $\Delta_1 := W_1(p(x_1 \mid \emptyset), p_1^*) \leq D$ explicitly below.

The bounds below are high-probability statements over the clean sample path and the idealized learner randomness. Corruption is treated as fixed, possibly adversarial, subject to the stated amplitude and budget constraints. The confidence level is denoted by $\delta_0 \in (0, 1)$.

For the no-restart update, define the global corruption contribution

$$\mathcal{E}_T^{\mathrm{nr}} := \frac{D}{2} \sum_{t=1}^{T} \left[\exp(2\lambda_{t-1} L_\ell \delta\, \Lambda_{t-1}) - 1\right], \tag{25}$$

where Λ_{t-1} is the number of corrupted observations in the global history up to time $t-1$. For the restarted update, let

$$\Lambda_{r,t-1} := \sum_{s=\tau_{r-1}}^{t-1} \mathbf{1}\{x_s \neq \tilde{x}_s\}, \qquad t \in I_r,$$

be the number of corruptions seen inside the current restart segment before time t. The segment-local corruption contribution is

$$\mathcal{E}_T^{\mathrm{rs}} := \frac{D}{2} \sum_{r=1}^{m} \sum_{t \in I_r} \left[\exp\left(2\lambda_{a_r(t)} L_\ell \delta\, \Lambda_{r,t-1}\right) - 1\right]. \tag{26}$$

The no-restart and restarted corruption comparisons that produce $\mathcal{E}_T^{\mathrm{nr}}$ and $\mathcal{E}_T^{\mathrm{rs}}$ are proved in Propositions A.1 and A.2, respectively.

For the clean dynamic terms, let $\{\lambda_t\}_{t=0}^{T-1}$ be a positive non-increasing learning-rate schedule, and define

$$\bar{\lambda}_T := \sum_{t=0}^{T-1} \lambda_t.$$

For the restarted scheme with equal segment length H, we reuse the same segment-local schedule $\{\lambda_s\}_{s=0}^{H-1}$ on each segment, and write

$$\bar{\lambda}_H := \sum_{s=0}^{H-1} \lambda_s.$$

For notational simplicity, the displayed restarted bounds write T/H for the number of restart segments; if H does not divide T, the same proof uses $m = \lceil T/H \rceil$ in place of T/H. The terms proportional to $\bar{\lambda}_T$ and $\bar{\lambda}_H$ are proof-side bounded-loss concentration terms. They are not part of the posterior score defined in Section 4.

For a posterior sequence $\rho = \{\rho_t\}_{t=2}^{T}$, define its current-risk excess by

$$E_T^{\mathrm{cur}}(\rho) := \sum_{t=2}^{T} \mathbb{E}_{\boldsymbol{c} \sim \rho_t}\left[R_t(\boldsymbol{c}) - R_t(\boldsymbol{c}_t^*)\right].$$

The following lemma is the latent-risk input used by the total-regret theorems. Its proof is deferred to Appendices A.5 and A.6, using the high-probability PAC-Bayes bound from Appendix A.4. More specifically, the no-restart and restarted current-risk excess terms are controlled in Lemmas A.10 and A.13.

Lemma 1 (High-probability latent excess-risk brackets)**.** *With probability at least* $1 - \delta_0$*, the no-restart posterior sequence* $\hat{\rho}^{\mathrm{nr}}$ *satisfies*

$$E_T^{\mathrm{cur}}(\hat{\rho}^{\mathrm{nr}}) + 8D\sqrt{(T-1)A_T^{\mathrm{OT}}} \leq \mathcal{B}_{T,\delta_0}^{\mathrm{nr}}(\{\lambda_t\}),$$

where

$$\begin{aligned}\mathcal{B}^{\mathrm{nr}}_{T,\delta_0}(\{\lambda_t\}) :=& L_c T\varepsilon + \frac{2C+\log(2/\delta_0)}{\lambda_{T-1}} + \frac{D^4}{2}\bar{\lambda}_T \\ &+ \frac{D^4 T\lambda_{T-1}}{8} + D^2\sqrt{\frac{T}{2}\log\frac{2}{\delta_0}} + 8DT\sqrt{(T-1)A_T^{\mathrm{OT}}} \\ &+ 8D\sqrt{(T-1)A_T^{\mathrm{OT}}}.\end{aligned} \tag{27}$$

For deterministic equal-length restart windows, the restarted posterior sequence $\hat{\rho}^{\mathrm{rs}}$ satisfies

$$E_T^{\mathrm{cur}}(\hat{\rho}^{\mathrm{rs}}) + 8D\sqrt{TA_T^{\mathrm{OT}}} \le \mathcal{B}^{\mathrm{rs}}_{T,\delta_0}(\{\lambda_s\}, H),$$

where

$$\begin{aligned}\mathcal{B}^{\mathrm{rs}}_{T,\delta_0}(\{\lambda_s\}, H) :=& L_c T\varepsilon + \frac{2(T/H)C+\log(2/\delta_0)}{\lambda_{H-1}} + \frac{D^4}{2}\frac{T}{H}\bar{\lambda}_H \\ &+ \frac{D^4 T\lambda_{H-1}}{8} + D^2\sqrt{\frac{T}{2}\log\frac{2}{\delta_0}} + 8DH\sqrt{TA_T^{\mathrm{OT}}} \\ &+ 8D\sqrt{TA_T^{\mathrm{OT}}}.\end{aligned} \tag{28}$$

The complexity C in Lemma 1 is the KL cost of a localized comparator. If $\boldsymbol{c}^\star \in \mathcal{C}_k$, define the localized comparator by truncating the prior equation 6 to a radius-r ball around $\boldsymbol{c}^\star$:

$$\nu_{\boldsymbol{c}^\star,r}(d\boldsymbol{c}) = \frac{\mathbf{1}\{\boldsymbol{c}\in\mathcal{C}_k,\ \|\boldsymbol{c}-\boldsymbol{c}^\star\|_2 \le r\}\,\pi(d\boldsymbol{c})}{\pi\{\boldsymbol{c}\in\mathcal{C}_k : \|\boldsymbol{c}-\boldsymbol{c}^\star\|_2 \le r\}}.$$

Then

$$\mathrm{KL}(\nu_{\boldsymbol{c}^\star,r}\|\pi) = \log\frac{1}{\pi\{\boldsymbol{c}\in\mathcal{C}_k : \|\boldsymbol{c}-\boldsymbol{c}^\star\|_2 \le r\}} \le C.$$

The prior-mass calculation in Lemma A.7 gives, up to constants, where R is the bounded-support radius introduced in Section 3,

$$C = C_{k,d}(r) \lesssim \log\frac{1}{q(k)} + kd\log\frac{2R}{r}.$$

Thus dimension enters the regret bounds through this PAC-Bayesian complexity factor.

The Wasserstein conversion in Corollary A.1 turns these latent brackets into clean dynamic regret terms. Define

$$\mathcal{G}^{\mathrm{nr}}_{T,\delta_0}(\{\lambda_t\}) := \frac{L_K}{\sqrt{\mu}}\sqrt{(T-1)\mathcal{B}^{\mathrm{nr}}_{T,\delta_0}(\{\lambda_t\})} + \sqrt{(T-1)A_T^{\mathrm{OT}}}, \tag{29}$$

and

$$\mathcal{G}^{\mathrm{rs}}_{T,\delta_0}(\{\lambda_s\}, H) := \frac{L_K}{\sqrt{\mu}}\sqrt{T\mathcal{B}^{\mathrm{rs}}_{T,\delta_0}(\{\lambda_s\}, H)} + \sqrt{TA_T^{\mathrm{OT}}}. \tag{30}$$

Theorem 1 (Quasi-Bayesian predictor)**.** *For the base update without restart, with probability at least $1-\delta_0$,*

$$W^{\mathrm{total}}_{Regret}(T) \le \Delta_1 + \mathcal{E}^{\mathrm{nr}}_T + \mathcal{G}^{\mathrm{nr}}_{T,\delta_0}(\{\lambda_t\}). \tag{31}$$

Theorem 2 (Restarted quasi-Bayesian predictor)**.** *For the restarted update with deterministic equal-length restart windows, with probability at least $1-\delta_0$,*

$$W^{\mathrm{total}}_{Regret}(T) \le \Delta_1 + \mathcal{E}^{\mathrm{rs}}_T + \mathcal{G}^{\mathrm{rs}}_{T,\delta_0}(\{\lambda_s\}, H). \tag{32}$$

The two theorems expose the role of temporal localization. The corruption term equation 25 uses the full corrupted history, whereas equation 26 uses only corruptions accumulated inside the current restart segment. Restart therefore localizes both the clean drift comparison and the corruption perturbation. On the clean side, it replaces the whole-horizon memory penalty equation 29 by the localized dynamic bound equation 30. All appearances of A_T^{OT} should be read as drift penalties: they measure the amount of transport energy required to move the true distribution path over time. When this action is small, the stream evolves gradually and the restart term can remain favorable. When it is large, the true law is spending substantial transport energy, and the regret bound correspondingly reflects a harder tracking problem.

5.2 Sublinear Regimes and Configuration Comparison

To compare the base and restarted configurations at the level of long-horizon growth, we use the polynomial ansatz

$$\lambda_t \asymp t^{-\beta}, \qquad \Lambda_T \asymp T^{\gamma}, \qquad \Lambda_{r,a} \lesssim a^{\gamma}, \qquad A_T^{\mathrm{OT}} \asymp T^{a}, \qquad H \asymp T^{h}. \tag{33}$$

Here β controls the learning-rate decay, γ the growth of the cumulative corruption budget, a the growth of the cumulative transport action, and h the growth of the restart window. For the restarted configuration, the corruption condition is interpreted segment-locally: $\Lambda_{r,a}$ counts corruptions inside segment r up to within-segment age a. This is the corruption quantity that appears in equation 26.

Proposition 1 (Sublinear regime comparison)**.** *Under the bounds of Theorems 1 and 2 and the scaling ansatz equation 33, suppose the localized-comparator approximation radius is chosen so that $\varepsilon = \varepsilon_T = o(1)$ and the confidence level is fixed, or more generally satisfies $\log(1/\delta_0) = T^{o(1)}$. Then the following hold.*

1. *The no-restart theorem, with the whole-horizon comparator used here, does not certify a sublinear high-probability total-regret bound under the current proof technique.*
2. *The restarted configuration admits a sublinear high-probability regret regime whenever one can choose exponents such that*

$$a < 1, \qquad \gamma < \beta < h < \frac{1-a}{2}. \tag{34}$$

Discussion of Proposition 1. The proof is given in Appendix A.8. The main point is that the corruption exponent scales as $\lambda_t \Lambda_t \asymp t^{\gamma-\beta}$, while the restarted drift term is sublinear when the restart window grows more slowly than $T^{(1-a)/2}$. □

The reason restart appears in the sublinear condition is the clean dynamic comparison. Without restart, the proof compares the learner to one fixed configuration over the whole horizon, so the moving oracle $\boldsymbol{c}_t^*$ is measured against an increasingly stale reference geometry. Restart partitions the horizon into segments $I_r = \{\tau_{r-1}, \ldots, \tau_r - 1\}$ and compares only to a local fixed oracle inside each segment. This is what replaces the whole-horizon clean term equation 29 by the localized term equation 30. The statement that no-restart is not certified sublinear is therefore a limitation of the present upper bound, not a lower bound against no-restart methods; the restarted predictor is the configuration for which the current analysis certifies sublinear regret under mild corruption and transport action.

6 Experiments

6.1 Experimental Goals

The experiments are designed to test the central methodological claim of the paper: restart improves tracking when the underlying distribution drifts over time. We compare the base quasi-Bayesian predictor with its restarted version under controlled changes in restart interval, corruption magnitude, corruption frequency, and drift scale.

The main synthetic experiment uses a smooth drifting Gaussian-mixture stream with persistent bounded corruption. This directly tests the theorem-level comparison between the raw quasi-Bayesian predictor and the same predictor with restart. We then include an abrupt-shift diagnostic designed to isolate stale posterior memory, together with supplementary sensitivity sweeps over restart interval, corruption magnitude, corruption frequency, and drift scale.

6.2 Experimental Setup

We use synthetic streaming data generated from temporally evolving Gaussian mixtures. Unless stated otherwise, the stream has dimension $d = 2$, two equally weighted mixture components, covariance $0.2I$, initial center separation 5, and horizon $T = 500$. The centers move at every step along fixed random directions with step size 0.03. The learner observes

$$z_t = x_t + \xi_t, \qquad \xi_t \sim \mathrm{Unif}([-\epsilon, \epsilon]^d),$$

with $\epsilon = 1$ in the main comparison. Thus the main synthetic experiment is a controlled stress test with persistent bounded corruption and smooth drift. The main figure uses the learning-rate schedule $\lambda_t = 0.1\sqrt{\log(t)/t}$ and restart interval $H = 10$. Supplementary figures repeat the same comparison for $\lambda_t = 0.1/t$ and $\lambda_t = 0.1/t^2$.

For posterior inference, we use the same RJMCMC sampler in all synthetic experiments. The prior over the number of clusters is $q(k) \propto \exp(-0.1k)$ on $k \in \{1, \ldots, p_{\max}\}$, with $p_{\max} = 4$ in the smooth-drift experiments. Conditional on k, centers are uniform on the Euclidean ball of radius $2R$ used by the sampler, with $R = 5$ for the smooth-drift synthetic experiments. The proposal first chooses $k' \in \{k-1, k, k+1\} \cap \{1, \ldots, p_{\max}\}$ uniformly and then proposes centers from a product Student-t distribution with three degrees of freedom, centered at the k'-means centers and scale $\tau = (p_{\max} t)^{-1/2}$. We use 200 RJMCMC iterations per time step, with the first quarter discarded as burn-in, and use 10 posterior configurations to form the predictive law. The sample-based sliced-W_1 estimates use 500 predictive samples, 300 Metropolis–Hastings burn-in steps, and 100 random projection directions.

The synthetic true law is known, so the reported regret is computed by comparing the learner's posterior predictive distribution with samples from the current true mixture. In dimension one this is the empirical Wasserstein-1 distance after sorting. In higher dimensions we use the standard sliced-W_1 Monte Carlo proxy, averaging one-dimensional Wasserstein distances over random projection directions.

Figure 1 gives the main synthetic comparison on the smooth drifting corrupted stream. It compares the base quasi-Bayesian predictor with the restarted predictor under $\lambda_t = 0.1\sqrt{\log(t)/t}$. The goal is to isolate the effect of localizing posterior memory under drift: restart limits how long corrupted and stale observations can influence the current posterior, which is the empirical counterpart of the localization mechanism in the theory. We additionally report online real-data prediction experiments on daily SPY return streams in Appendix B.4.

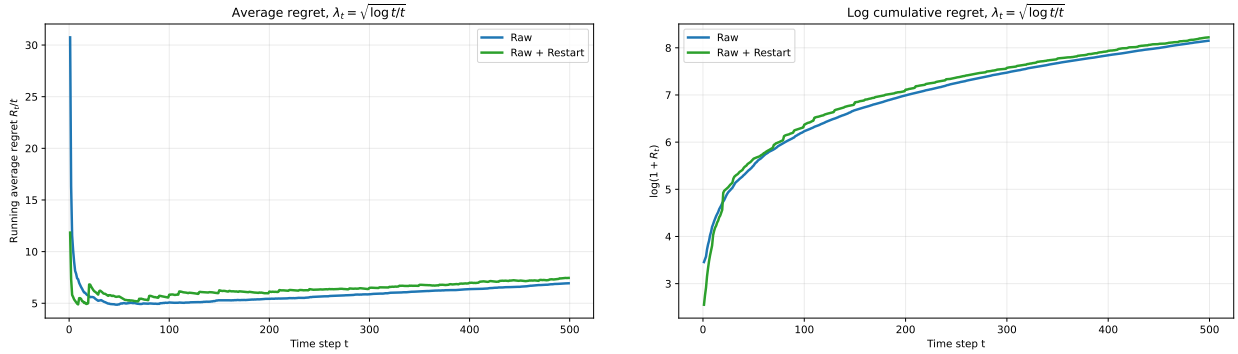

Figure 1: Main comparison between the raw quasi-Bayesian predictor and the restarted predictor under the learning-rate schedule $\lambda_t = 0.1\sqrt{\log(t)/t}$. Left: cumulative average regret R_t/t. Right: $\log(1 + R_t)$.

The smooth drifting streams above are useful for measuring tracking behavior, but they can be benign for the no-restart predictor because older observations may remain approximately informative. We therefore also include a stress test designed to isolate stale posterior memory. The stream is piecewise stationary: for block I_b,

$$p_t^* = \frac{1}{k} \sum_{j=1}^{k} \mathcal{N}(c_{j,b}, \sigma^2 I), \qquad t \in I_b.$$

At a block boundary, all centers undergo an abrupt translation,

$$c_{j,b+1} = c_{j,b} + \Delta u_b, \qquad \|u_b\|_2 = 1.$$

If the number of regime changes satisfies $M_T \asymp T^a$, then this construction has transport-action scale

$$A_T^{\mathrm{OT}} \asymp \sum_{b=1}^{M_T} \Delta^2 \asymp T^a.$$

Thus the experiment keeps the drift budget controlled while making old regimes actively misleading for a global-memory posterior. In the plotted diagnostic we use $p = d = 2$, $p_{\max} = 2$, covariance $0.05I$, initial separation 2.5, jump size $\Delta = 7$, no corruption, $\lambda_t = 2t^{-0.25}$, and restart interval $H = \lfloor T^{0.40} \rceil$. Figure 2 shows the $T = 4000$ trajectory. The no-restart predictor accumulates persistent error after regime changes, whereas the restarted predictor remains localized to the current regime. The corresponding multi-horizon slope diagnostics are shown in Appendix B.3.

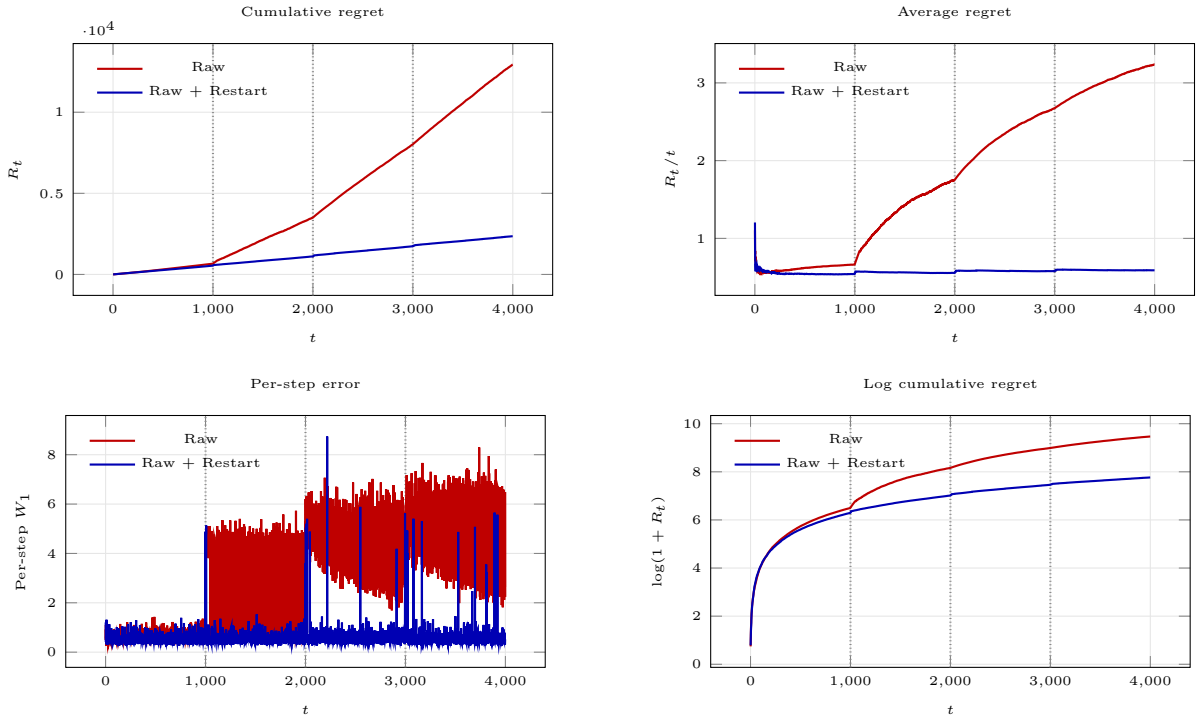

Figure 2: Abrupt-shift stale-memory experiment at $T = 4000$. Top left: cumulative Wasserstein regret R_t. Top right: average cumulative regret R_t/t. Bottom left: per-step Wasserstein error. Bottom right: $\log(1 + R_t)$. Vertical dashed lines indicate regime changes.

7 Discussion and Conclusion

The main conceptual point of the paper is that the online clustering problem is best understood through two spaces at once. Learning takes place in the latent configuration space of cluster-center arrangements, where the prior, quasi-posterior, and PAC-Bayesian comparison are defined. Prediction, however, is evaluated in data space through the induced law $p(\cdot \mid \boldsymbol{c})$. This separation is what makes the method both interpretable and analyzable: the latent space carries the clustering geometry, while the data space carries the predictive object that ultimately matters.

From this viewpoint, corruption and drift create two different obstacles. Corruption perturbs the loss-based posterior reweighting, while drift breaks the validity of a single global comparator over the whole horizon. The central mechanism developed here addresses the second obstruction: restart changes the temporal memory structure and localizes the comparison in time, which is what makes the dynamic optimal transport analysis effective. Under the current bounds, the no-restart predictor is not certified to recover the favorable long-horizon regime because the clean drift term remains global. Restart is what localizes that drift term in the guarantee.

The experiments support this interpretation at a qualitative level. They show that temporal localization improves tracking when the stream continues to move. At the same time, the experiments also suggest that the asymptotic regime distinctions in the theory are subtle in finite samples. This is not a contradiction,

but rather an indication that the current bounds capture the correct structural effect more clearly than they capture sharp finite-sample constants.

Several directions remain open. On the theoretical side, the corruption comparison could be sharpened further, and the restart mechanism could be made adaptive rather than fixed in advance. On the algorithmic side, the most natural next step is to replace the expensive online posterior tracking mechanism by an amortized or state-space-based inference module while preserving the latent clustering interpretation. More broadly, the framework suggests that quasi-Bayesian latent-geometry methods can be studied as sequential predictive models rather than only as static clustering procedures. This is the perspective from which the present paper should be read, and it is the main direction in which the methodology can be extended.

References

Giuseppe Aceto, Alessio Botta, Walter De Donato, and Antonio Pescapè. Cloud monitoring: A survey. *Computer Networks*, 57(9):2093–2115, 2013.

Ashkan Aghdai, Fan Zhang, Nadun Dasanayake, Kang Xi, and H. Jonathan Chao. Traffic measurement and analysis in an organic enterprise data center. In *2013 IEEE 14th International Conference on High Performance Switching and Routing (HPSR)*, pp. 49–55, 2013. doi: 10.1109/HPSR.2013.6602289.

Mohammad Alizadeh, Shuang Yang, Milad Sharif, Sachin Katti, Nick McKeown, Balaji Prabhakar, and Scott Shenker. pfabric: minimal near-optimal datacenter transport. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, pp. 435–446, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320566. doi: 10.1145/2486001.2486031. URL `https://doi.org/10.1145/2486001.2486031`.

Dheeraj Baby and Yu-Xiang Wang. Optimal dynamic regret in exp-concave online learning. In Mikhail Belkin and Samory Kpotufe (eds.), *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 359–409. PMLR, 15–19 Aug 2021. URL `https://proceedings.mlr.press/v134/baby21a.html`.

Richard Baraniuk. More is less: Signal processing and the data deluge. *Science (New York, N.Y.)*, 331: 717–9, 02 2011. doi: 10.1126/science.1197448.

Jean-David Benamou and Yann Brenier. A computational fluid mechanics solution to the monge–kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000. doi: 10.1007/s002110050002.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.

Guido Dartmann, Houbing Song, and Anke Schmeink. *Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things*. Elsevier Science Publishers B. V., NLD, 1st edition, 2019. ISBN 0128166371.

Gregory Ditzler and Robi Polikar. Incremental learning of concept drift from streaming imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 25(10):2283–2301, 2013. doi: 10.1109/TKDE.2012.136.

Eduard Gorbunov, Marina Danilova, and Alexander Gasnikov. Stochastic optimization with heavy-tailed noise via accelerated gradient clipping. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Elad Hazan. Introduction to online convex optimization, 2023. URL `https://arxiv.org/abs/1909.05207`.

Cormac Herley. Automated detection of automated traffic. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 1615–1632, Boston, MA, August 2022. USENIX Association. ISBN 978-1-939133-31-1. URL `https://www.usenix.org/conference/usenixsecurity22/presentation/herley`.

Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM Journal on Mathematical Analysis*, 29(1):1–17, 1998. doi: 10.1137/S0036141096303359.

Jia Li, Shimin Di, Yanyan Shen, and Lei Chen. Fluxev: A fast and effective unsupervised framework for time-series anomaly detection. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, WSDM '21, pp. 824–832, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450382977. doi: 10.1145/3437963.3441823. URL https://doi.org/10.1145/3437963.3441823.

Le Li, Benjamin Guedj, and Sébastien Loustau. A quasi-bayesian perspective to online clustering. *Electronic Journal of Statistics*, 12:3071–3113, 2016. URL https://api.semanticscholar.org/CorpusID:88521244.

Zijian Liu and Zhengyuan Zhou. Stochastic nonsmooth convex optimization with heavy-tailed noises: High-probability bound, in-expectation rate and initial distance adaptation. *arXiv preprint arXiv:2303.12277*, 2023.

M. Hammad Mazhar and Zubair Shafiq. Real-time video quality of experience monitoring for https and quic. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pp. 1331–1339, 2018. doi: 10.1109/INFOCOM.2018.8486321.

Yisroel Mirsky, Tomer Doitshman, Yuval Elovici, and Asaf Shabtai. Kitsune: An ensemble of autoencoders for online network intrusion detection, 2018. URL https://arxiv.org/abs/1802.09089.

Alexander V. Nazin, Arkadi S. Nemirovsky, A. Tsybakov, and Anatoli B. Juditsky. Algorithms of robust stochastic optimization based on mirror descent method. *Automation and Remote Control*, 80:1607 – 1627, 2019. URL https://api.semanticscholar.org/CorpusID:195820442.

Hansheng Ren, Bixiong Xu, Yujing Wang, Chao Yi, Congrui Huang, Xiaoyu Kou, Tony Xing, Mao Yang, Jie Tong, and Qi Zhang. Time-series anomaly detection service at microsoft. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pp. 3009–3017, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362016. doi: 10.1145/3292500.3330680. URL https://doi.org/10.1145/3292500.3330680.

Abishek Sankararaman and Balakrishnan Narayanaswamy. Online robust non-stationary estimation. *Advances in Neural Information Processing Systems*, 36:50506–50544, 2023.

Abishek Sankararaman, Balakrishnan Narayanaswamy, Vikramank Y Singh, and Zhao Song. FITNESS: (Fine tune on new and similar samples) to detect anomalies in streams with drift and outliers. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 19153–19177. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/sankararaman22a.html.

Che-Ping Tsai, Adarsh Prasad, Sivaraman Balakrishnan, and Pradeep Ravikumar. Heavy-tailed streaming statistical estimation. In *International Conference on Artificial Intelligence and Statistics*, pp. 1251–1282. PMLR, 2022.

Tim van Erven, Sarah Sachs, Wouter M Koolen, and Wojciech Kotlowski. Robust online convex optimization in the presence of outliers. In Mikhail Belkin and Samory Kpotufe (eds.), *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 4174–4194. PMLR, 15–19 Aug 2021. URL https://proceedings.mlr.press/v134/vanerven21a.html.

Nuri Mert Vural, Lu Yu, Krishna Balasubramanian, Stanislav Volgushev, and Murat A Erdogdu. Mirror descent strikes again: Optimal stochastic convex optimization under infinite noise variance. In Po-Ling Loh and Maxim Raginsky (eds.), *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 65–102. PMLR, 02–05 Jul 2022. URL https://proceedings.mlr.press/v178/vural22a.html.

Martin A. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, 2003. URL https://api.semanticscholar.org/CorpusID:553962.