Conditional Diffusion for Vendor-Agnostic Storage Performance Modeling

Aziz Temirkhanov [1] **Daniil Kopytov** [2] **Mikhail Hushchyn** [2]

[1]*HSE University, SB AI Lab, Moscow, Russia* [2]*HSE University, Moscow, Russia*. Correspondence to: Aziz Temirkhanov atemirkhanov@hse.ru.

1. Introduction

Storage performance is variable even under fixed settings: queueing effects, caching transitions, and background activity affect both throughput and latency. In practice, this variability is not well captured by a single mean–variance pair; the conditional response can be heteroscedastic and occasionally multimodal, especially when the system operates close to saturation or transitions between caching regimes. For planning and configuration studies, it is therefore natural to model *conditional distributions* rather than only predict point estimates.

We build on the vendor-agnostic, measurement-driven framework of [1], which formulates the task as learning $p(y \mid x)$ from measurements and evaluates generative surrogates using distributional metrics. The main change in this work is methodological: we replace the generative core with a conditional diffusion model, while keeping the dataset and evaluation protocol unchanged to enable a direct comparison.

2. Problem statement and dataset

Following [1], per-interval outcomes are treated as samples from

$$y \sim p(y \mid x), \qquad y = (\text{IOPS}, \text{ latency}), \tag{1}$$

where x encodes workload descriptors and, for pools, configuration parameters. The dataset and protocol are taken from [1] and span a cache component and SSD/HDD pools under random and sequential workloads. Inputs include access pattern and read/write mix, request size, and concurrency/intensity controls (e.g., jobs and queue depth), as well as pool descriptors such as RAID parameters and disk count. Targets are time-resolved measurements of IOPS and mean latency, yielding per-condition sample clouds for distributional learning.

3. Method

We model the joint target in log space, $z = (\log \text{IOPS}, \log \text{latency})$, and learn a conditional diffusion model that samples z given x. Using a standard DDPM setup [2], a denoiser takes a noisy target z_t together with timestep and condition embeddings and is trained to predict the injected noise. Sampling the reverse process yields joint draws $(\widehat{\text{IOPS}}, \widehat{\text{latency}})$, i.e., an explicit approximation of $\hat{p}(y \mid x)$. Compared to parametric output heads, diffusion offers a flexible way to represent non-Gaussian conditional structure without committing to a fixed family for $p(y \mid x)$.

4. Evaluation protocol and headline outcomes

4.1 Protocol

We follow the train/test split and per-condition evaluation of [1]. For each held-out condition x_i, we generate samples $\{\hat{y}_{ij}\}$ and compare them to observations $\{y_{ij}\}$. We report moment errors via PEM/PES (Eqs. 2–3) and distributional distances via FD/MMD (Eqs. 4–5). Baselines follow [1] and include CatBoost with a Gaussian head [3] and normalizing flows [4].

$$\text{PEM} = \left|\frac{\hat{\mu} - \mu}{\mu}\right| \times 100\%, \quad \mu = \frac{1}{k}\sum_{t=1}^{k} y_t, \tag{2}$$

$$\text{PES} = \left|\frac{\hat{\sigma} - \sigma}{\sigma}\right| \times 100\%, \quad \sigma = \sqrt{\frac{1}{k-1}\sum_{t=1}^{k}(y_t - \mu)^2}. \tag{3}$$

Fréchet distance (FD). Approximating the observed and generated clouds by Gaussians $\mathcal{N}(\mu, \Sigma)$ and $\mathcal{N}(\hat{\mu}, \hat{\Sigma})$, respectively, we compute

$$\text{FD} \;=\; \|\mu - \hat{\mu}\|_2^2 \;+\; \operatorname{tr}\left(\Sigma + \hat{\Sigma} - 2(\Sigma\hat{\Sigma})^{1/2}\right). \tag{4}$$

Maximum mean discrepancy (MMD). With a positive-definite kernel $K(\cdot,\cdot)$, the discrepancy between an observed cloud of size k and a generated cloud of size S is

$$\text{MMD} \;=\; \frac{1}{k^2}\sum_{i=1}^{k}\sum_{j=1}^{k} K(y_i, y_j) \;+\; \frac{1}{S^2}\sum_{i=1}^{S}\sum_{j=1}^{S} K(\hat{y}_i, \hat{y}_j) \;-\; \frac{2}{kS}\sum_{i=1}^{k}\sum_{j=1}^{S} K(y_i, \hat{y}_j). \tag{5}$$

4.2 Summary of observations

Across cache and SSD regimes, diffusion improves distributional fit (FD/MMD) and reduces dispersion error (PES) compared to the strongest baseline, while maintaining competitive mean errors. The cache regime shows the largest gain in FD, consistent with diffusion capturing non-Gaussian structure beyond what is induced by a Gaussian head. HDD remains the outlier: under the same protocol, CatBoost and flows are consistently better on FD/MMD. A plausible explanation is that the HDD regime exhibits sharp regime changes and stronger tails, where diffusion requires additional tuning (e.g., conditioning, schedules, or objective reweighting) to match the baselines.

5. Conclusion

Conditional diffusion is an effective surrogate for sampling (IOPS, latency) distributions in cache and

Table 1: Quality metrics for the storage cache under random data load

Metric	kNN	CatBoost	NF	Diffusion
PEM (IOPS), %	26 ± 2	6.3 ± 0.4	4.1 ± 0.4	**1.5** ± **2.11**
PEM (Lat), %	18 ± 1	4.9 ± 0.3	2.9 ± 0.2	**2.7** ± **0.7**
PES (IOPS), %	35 ± 2	23.9 ± 0.8	412 ± 23	**10.19** ± **14.2**
PES (Lat), %	30 ± 1	24.1 ± 0.9	299 ± 29	**11.76** ± **14.4**
FD	6655 ± 822	445 ± 57	365 ± 55	**9.44** ± **4.9**
MMD	1.61 ± 0.01	1.34 ± 0.02	0.59 ± 0.02	**0.49** ± **0.37**

Table 2: Quality metrics for the SSD pool under random data load

Metric	kNN	CatBoost	NF	Diffusion
PEM (IOPS), %	38 ± 3	8.9 ± 0.6	10.1 ± 0.8	**1.04** ± **1.62**
PEM (Lat), %	19.2 ± 0.9	7.4 ± 0.4	8.8 ± 0.6	**3.90** ± **0.52**
PES (IOPS), %	52 ± 3	22 ± 0.9	183 ± 14	**10.73** ± **15.19**
PES (Lat), %	40 ± 2	25 ± 1.1	135 ± 9	**23.09** ± **20.59**
FD	1647 ± 211	140 ± 19	161 ± 16	**5.60** ± **0.68**
MMD	1.38 ± 0.02	1.05 ± 0.02	0.83 ± 0.02	**0.42** ± **0.17**

Table 3: Quality metrics for the SSD pool under sequential data load

Metric	kNN	CatBoost	NF	Diffusion
PEM (IOPS), %	31 ± 3	10.2 ± 0.9	9.9 ± 1	**3.07** ± **6.09**
PEM (Lat), %	42 ± 3	10.7 ± 0.7	8.1 ± 0.7	**3.61** ± **0.79**
PES (IOPS), %	90 ± 12	37 ± 5	134 ± 16	**8.95** ± **54.13**
PES (Lat), %	101 ± 16	42 ± 5	131 ± 17	**7.11** ± **53.93**
FD	1692 ± 241	110 ± 18	89 ± 19	**3.67** ± **5.10**
MMD	1.23 ± 0.03	1.01 ± 0.03	0.69 ± 0.03	**0.33** ± **0.28**

Table 4: Quality metrics for HDD Pool.

Metric	kNN	CatBoost	NF	Diffusion
PEM (IOPS), %	27 ± 2	**10.6** ± **0.9**	11.4 ± 0.9	31.17 ± 3.23
PEM (Lat), %	49 ± 4	**16** ± **2**	18 ± 2	16.59 ± 11.11
PES (IOPS), %	33 ± 3	18 ± 1	43 ± 4	**15.32** ± **44.22**
PES (Lat), %	60 ± 8	**23** ± **2**	62 ± 7	35.09 ± 158.9
FD	96 ± 19	**5.0** ± **0.6**	6.7 ± 0.9	34.92 ± 5.88
MMD	0.81 ± 0.04	0.33 ± 0.02	**0.23** ± **0.02**	0.73 ± 0.043

SSD regimes under the dataset and protocol of [1]. The HDD results highlight clear failure modes and suggest that a regime-aware model choice, or HDD-specific adaptation of diffusion, is more realistic than a single global model across all storage types.

Acknowledgments

The article was prepared within the framework of the project "Mirror Laboratories" HSE University. The computation for this research was performed using the computational resources of HPC facilities at HSE University [5].

References

[1] A. Al-Maeeni, A. Temirkhanov, A. Ryzhikov, and M. Hushchyn. Performance modeling of data storage systems using generative models. *IEEE Access*, 2025.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[3] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[4] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. In *International Conference on Learning Representations (ICLR)*, 2017.

[5] PS Kostenetskiy, RA Chulkevich, and VI Kozyrev. Hpc resources of the higher school of economics. In *Journal of Physics: Conference Series*, volume 1740, page 012050. IOP Publishing, 2021.