Always–On Probability Calibration With Vectorized Multiplicative–Weights*

Gaurav Sood†

July 15, 2025

Abstract

We propose a solver-free, streaming approach to post-hoc probability calibration based on Multiplicative-Weights Updates (MWU). Unlike standard Platt scaling or isotonic regression—which are trained in batch and periodically retrained offline—MWU performs a single exponential update per bucket or segment, requiring constant time per batch regardless of total traffic. Experiments on a synthetic ad-tech scenario with drift show that MWU matches the Brier score of classical calibrators while requiring **60–100**× less compute when recalibrating every mini-batch.

1 Introduction

Probability calibration is critical in ads, recommendations, and risk models (Niculescu-Mizil and Caruana, 2005; Guo et al., 2017). The dominant post-hoc techniques—Platt scaling (Platt, 1999) and isotonic regression (Zadrozny and Elkan, 2002)—are trained in batch and periodically refit. In high-velocity settings, this creates a *compute-drift trade-off*: infrequent retraining leads to miscalibration, whereas frequent retraining incurs heavy CPU costs.

We recast calibration as an online convex–concave game and apply the Multiplicative-Weights Update method (MWU) (Arora et al., 2012). The result is an *always-on* calibrator that adapts instantly to drift with constant per-batch cost.

2 Problem Setup

Given raw probabilities p_i^{raw} and binary outcomes $y\_i \in 0, 1$, let $b(i) \in 1, \dots, B$ denote the reliability bucket for event i. We seek bias factors $c\_b > 0$ such that calibrated probabilities:

*https://github.com/finite-sample/mw-calibration.

†Gaurav can be reached at gsood07@gmail.com

$$p_i^{\text{cal}} = \frac{c_{b(i)}, p_i^{\text{raw}}}{1 - p_i^{\text{raw}} + c_{b(i)}\, p_i^{\text{raw}}}$$

are (approximately) *self–calibrated*: $\hat{r}_b \approx \tilde{r}_b$ where $\hat{r}_b$ is the empirical click-through rate and $\tilde{r}_b$ the mean of p^{cal} in bucket b.

3 Multiplicative-Weights Calibrator

Let $\ell_b^{(t)} = \tilde{r}_b^{(t)} - \hat{r}_b^{(t)}$ be the calibration error for bucket b in batch t. MWU performs

$$c_b^{(t+1)} = c_b^{(t)} \exp\left(-\eta \ell_b^{(t)}\right), \tag{1}$$

followed by clipping $c_b \in [c_{\min}, c_{\max}]$. Under standard assumptions, MWU enjoys an $\mathcal{O}(\sqrt{T})$ regret bound (Arora et al., 2012).

4 Related Work

- **Batch calibration.** Platt (Platt, 1999) fits a logistic transform; isotonic regression uses the Pool-Adjacent-Violators (PAV) algorithm (Zadrozny and Elkan, 2002). More recent approaches include temperature scaling (Guo et al., 2017) and neural calibration heads (Kull et al., 2019).
- **Online calibration.** Blackwell approachability methods (Foster et al., 2018) guarantee online calibration under adversarial sequences but require projections onto calibrated sets. Multiplicative-Weights updates have been used in universal portfolios (Cover, 1991) and fairness-constrained classification (Agarwal et al., 2018), but— to our knowledge—have not been applied to streaming ad probability calibration.

5 Experiments

5.1 Synthetic Ad-Tech Stream

We simulate $200,\mathrm{k}$ impressions in 40 batches ($5,\mathrm{k}$ each) with drift $\mu_t = 0.7 \cdot t/T$. Calibration buckets $B = 100$. We compare:

1. Platt (logistic),
2. Isotonic regression (PAV),
3. **MWU** (Eq. 1).

All methods are recalibrated every batch.

5.2 Results

Metric	Platt	Isotonic	**MWU**
Mean per–batch Brier	**0.2051**	0.2045	0.2052
Std. Brier	0.0019	**0.0017**	0.0019
Mean CPU s/batch	0.0243	0.0181	**0.00039**

Table 1. Accuracy and compute over 40 batches. MWU matches Brier performance while requiring 60–100× less CPU.

6 Discussion

With per-batch refits, Platt/Isotonic deliver marginally lower Brier, but CPU load scales with cumulative traffic. In realistic deployments, they are often retrained hourly, introducing calibration drift between jobs. MWU removes this drift–compute trade-off: constant update cost and immediate correction.

7 Conclusion

MWU offers a lightweight, always-on alternative to batch calibration. Future work includes adaptive learning-rate schedules and large-scale deployment studies on production ad traffic.

References

Agarwal, A., Dudík, M., and Soudry, Z. (2018). Learning from conditional distributions via dual formulation. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 77–86.

Arora, S., Hazan, E., and Kale, S. (2012). The multiplicative weights update method: A meta-algorithm and applications. *Theory of Computing*, 8(1):121–164.

Cover, T. M. (1991). Universal portfolios. *Mathematical Finance*, 1(1):1–29.

Foster, D., Rakhlin, A., and Sridharan, K. (2018). Blackwell approachability and no-regret learning are equivalent. *Journal of Machine Learning Research*, 19(12):1–67.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 1321–1330.

Kull, M., Filho, T. M. S., and Flach, P. (2019). Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with Dirichlet calibration. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 12316–12326.

Niculescu-Mizil, A. and Caruana, R. (2005). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pages 625–632.

Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press.

Zadrozny, B. and Elkan, C. (2002). Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 694–699.