Monitoring Risks in Test-Time Adaptation

Mona Schirmer [*1] **Metod Jazbec** [*1] **Christian A. Naesseth** [1] **Eric Nalisnick** [2]

Abstract

Encountering distribution shift at test time is a common challenge for deployed models. Test-time adaptation (TTA) addresses this by adapting models online using only unlabeled test data. While TTA can prolong a model's usefulness, it may eventually fail, requiring the model to be taken offline and retrained. We propose augmenting TTA with a risk monitoring framework that raises alarms when performance degrades beyond a predefined threshold. Our method extends sequential testing with confidence sequences to support model updates and operate without test labels. We validate our approach across diverse datasets, shift types, and TTA methods.

1. Introduction

When test data deviates from the training distribution, model performance can degrade, effectively rendering the model obsolete. Such degradation is particularly concerning in safety-critical applications. For instance, a medical model trained on one demographic may yield poor predictions when deployed on another.

Test-time adaptation (TTA) (Xiao and Snoek, 2024) offers a compelling solution by adapting model parameters online using only test features and no labels. Methods based on entropy minimization (Wang et al., 2021) or pseudo-labeling (Wang et al., 2022) can significantly improve robustness under distribution shift. However, TTA can also fail—especially under severe shifts or prolonged adaptation—sometimes collapsing into trivial predictions with near-zero accuracy (Press et al., 2024a). Alarmingly, these failures often occur silently, making detection without labels extremely difficult.

Timely detection of such degradations—whether due to harmful shifts or model collapse—is crucial for safe deployment. At the same time, false alarms that trigger unnecessary retraining can be costly. Sequential testing offers a principled framework for risk monitoring (Podkopaev and Ramdas, 2022), with time-uniform confidence sequences (Howard et al., 2021) enabling control over false alarm rates. However, existing methods either require test labels or do not account for model adaptation (Amoukou et al., 2024; Podkopaev and Ramdas, 2022).

In this work, we extend sequential testing to the setting of TTA, enabling risk monitoring of a continually adapting model without observing test labels. Our contributions are:

- In § 3, we propose a general monitoring framework applicable to arbitrary TTA methods and shifts. We notably extend unsupervised risk bounds (Amoukou et al., 2024) to classification error in the adaptive setting.
- In § 4, we extensively study our monitoring tool and demonstrate that (i) it reliably detects risk violations and (ii) does not raise false alarms on a range of TTA methods, datasets and shift types.

2. Preliminaries

Setting We consider a standard multi-class classification setting, where the input space is denoted by $\mathcal{X} \subseteq \mathbb{R}^D$ and the label space by $\mathcal{Y} = \{1, \dots, C\}$ for some finite number of classes C. Data points $(\boldsymbol{x}, y)$ are assumed to be realizations of random variables $(\mathbf{x}, \mathbf{y})$ drawn from an unknown joint distribution P over $\mathcal{X} \times \mathcal{Y}$. The samples in train $\mathcal{D}_{\text{train}}$ and calibration $\mathcal{D}_{\text{cal}}$ sets are drawn $i.i.d.$ from the *source* distribution $(\boldsymbol{x}_0, y_0) \sim P_0$. Test samples in $\mathcal{D}^k_{\text{test}}$ are assumed to arrive *sequentially* from a time-varying and possibly shifting *test* distribution $(\boldsymbol{x}_k, y_k) \sim P_k,\ k \geq 1$. We do not make any assumptions about the nature of the distribution shift. For the test stream, we distinguish between two settings. In the 'unsupervised' setting, only test features $\boldsymbol{x}_k \sim P_k(\mathbf{x})$ are observed, yielding a sequence of unlabeled test datasets $\mathcal{D}^k_{\boldsymbol{x}}, k \geq 1$. In the 'supervised' setting, the true labels $y_k \sim P_k(\mathbf{y} \mid \mathbf{x} = \boldsymbol{x}_k)$ are revealed after predictions are made on the observed features at each time step k, resulting in a sequence of labeled test datasets $\mathcal{D}^k_{\boldsymbol{x}y}, k \geq 1$. Lastly, with $p : \mathcal{X} \to \Delta^{C-1}$ we denote a probabilistic classifier, where Δ^{C-1} is the probability simplex over C classes.

[*]Equal contribution [1]UvA-Bosch Delta Lab, University of Amsterdam [2]Dept. of Computer Science, Johns Hopkins University. Correspondence to: Mona Schirmer <m.c.schirmer@uva.nl>.

2nd Workshop on Test-Time Adaptation: Putting Updates to the Test! (PUT)@ICML, Vancouver, Canada. PMLR 267, 2025.

Losses and Risks It is crucial to monitor the deployed model on test data to detect potential performance degradations early. To formally capture the concept of error, a problem-specific *loss* function, denoted as $\ell : \mathcal{O} \times \mathcal{Y} \to \mathbb{R}$ is first defined.[1] The *risk* of a model p on data drawn from distribution P_k is then given as the expected loss $R_k(p) := \mathbb{E}_{P_k}[\ell\big(p(\mathbf{x}), \mathbf{y}\big)]$. To ease notation, we denote the loss random variable on data from P_k with $\mathbf{z}_k := \ell(p(\mathbf{x}), \mathbf{y})$ henceforth. $R_0(p)$ represents the *source risk* on data coming from the source distribution P_0. We also define a *running test risk* as

$$\bar{R}_t(p) := \frac{1}{t} \sum_{k=1}^{t} R_k(p) \,, \tag{1}$$

which measures the model's running performance on data drawn from the (shifting) test distribution P_k. If, for some time index t^*, the running test risk starts to exceed the source risk, i.e., $\bar{R}_{t^*}(p) > R_0(p)$, this may indicate that the data distribution has shifted in a way that is harmful to the model's performance, suggesting that the model should potentially be taken offline and retrained. In practice, the 'true' risk is typically estimated using the *empirical risk*, defined as $\hat{R}_k(p; \mathcal{D}^k_{\boldsymbol{xy}}) = \frac{1}{N_k} \sum_{n=1}^{N_k} z_{k,n}$, where the loss realizations $z_{k,n}$ are based on *i.i.d.* samples from $\mathcal{D}^k_{\boldsymbol{xy}}$.

We consider two loss functions: the 0-1 loss, $\ell_{0\text{-}1}(p(\boldsymbol{x}), y) := \mathbb{1}[\hat{y}(\boldsymbol{x}) \neq y]$ where $\hat{y}(\boldsymbol{x}) := \arg\max_c p(\boldsymbol{x})_c$, and the brier loss $\ell_B(p(\boldsymbol{x}), y) := \frac{1}{2} \sum_{c=1}^{C} \big(p(\boldsymbol{x})_c - \mathbb{1}[y = c]\big)^2$.

Test-time Adaptation (TTA). In test-time adaptation (Wang et al., 2021), the model parameters θ are updated online as the model observes batches of unlabeled test features. Specifically, given a sequence of unlabeled test batches $\mathcal{D}^1_{\boldsymbol{x}}, \ldots, \mathcal{D}^t_{\boldsymbol{x}}$, the TTA method produces a sequence of classifiers $p_{\theta_1}, \ldots, p_{\theta_t}$, where each θ_k is adapted using $\mathcal{D}^k_{\boldsymbol{x}}$. This stands in contrast to the static source model p_{θ_0}, which is trained once on labeled training data $\mathcal{D}_{\text{train}}$ and remains fixed during deployment. For simplicity, we refer to the source model as p_0 and the adapted models as $p_1, \ldots, p_t$ henceforth.

3. Sequential Testing for TTA Monitoring

We now detail our approach to risk monitoring for test-time adaptation (TTA) methods using sequential testing. We begin by extending the risk monitoring framework of Podkopaev and Ramdas (2022) to a deployment setting in which the model is continuously updated (§ 3.1). Please refer to § A.1 for a summary of their risk monitoring framework. Next, inspired by Amoukou et al. (2024), we derive a sequential test for the running test risk that does not require access to labels on the test data stream (§ 3.2)—a key innovation that enables rigorous statistical testing in TTA settings where test labels are unavailable. We then propose a concrete instantiation of our unsupervised test based on model uncertainty (§ B.1) and online calibration of thresholds (§ B.2). Finally, we describe techniques to enhance the detection power of the proposed tests (§ B.3).

3.1. Risk Monitoring under Model Adaptation

Unlike in the static model setting considered by Podkopaev and Ramdas (2022), we are interested in scenarios where a classifier is being continuously updated using a TTA method. Hence, we are interested in monitoring the risk not of a static source model p_0, but rather of a sequence of models $p_{1:t}$. To this end we define the hypotheses tested by our TTA risk tracker as:

$$H^a_0 : \bar{R}_t(p_{1:t}) \leq R_0(p_0) + \epsilon_{\text{tol}}, \ \forall t \geq 1 \tag{2}$$

$$H^a_1 : \exists t^* \geq 1 : \bar{R}_{t^*}(p_{1:t}) > R_0(p_0) + \epsilon_{\text{tol}} \tag{3}$$

where $\bar{R}_t(p_{1:t}) = \frac{1}{t} \sum_{k=1}^{t} R_k(p_k)$ and $R_k(p_k) := \mathbb{E}_{P_k}[\ell\big(p_k(\mathbf{x}), \mathbf{y}\big) | \mathbf{x}_{1:k-1}]$. Note that conditioning on the (unlabeled) test stream $\mathbf{x}_{1:k-1}$ is included despite assuming an independent data stream, as it becomes necessary when the model p_k is updated using test data, such as in TTA. To reduce notational clutter, this conditioning is omitted hereafter unless explicitly required. We use $\mathbf{z}^{(j)}_k := \ell(p_j(\mathbf{x}), \mathbf{y})$ to denote the loss random variable of the model p_j on data from P_k.

To design the corresponding alarm function, we proceed similarly to (Podkopaev and Ramdas, 2022) (Eq. 7), checking if the lower bound on the test risk exceeds the upper bound on the source risk. However, rather than relying on a sequence of losses from the static source model, we instead use a sequence of losses from the continuously adapted models in the (lower) confidence sequence for test risk. This leads to the adapted alarm function:

$$\Phi^a_t := \mathbb{1}\left[L^a_t\big(\mathbf{z}^{(1)}_1, \ldots, \mathbf{z}^{(t)}_t\big) > U\big(\mathbf{z}^{(0)}_0\big) + \epsilon_{\text{tol}}\right] , \tag{4}$$

which also enjoys strong PFA control guarantees when using conjugate-mixture empirical Bernstein bounds (Howard et al., 2021). Throughout the rest of this section, we abbreviate $\mathbf{z}^{(k)}_k$ as $\mathbf{z}_k$ and we shorten the sequence notation from $\mathbf{z}_1, \ldots, \mathbf{z}_t$ to $\mathbf{z}_{1:t}$ to ease the notational burden.

3.2. Unsupervised Risk Monitoring

While the adapted alarm function Φ^a (Eq. 4) monitors the performance of adapted models—rather than a fixed static model—it still depends on access to a labeled test stream to compute the adapted lower bound L^a_t. Consequently, it

[1] The output space $\mathcal{O}$ may correspond either to the label space $\mathcal{Y}$ or to the space of probability distributions over $\mathcal{Y}$, depending on the loss type.

cannot be directly applied to track the performance of TTA methods where only an unlabeled test stream is available. To get around this, we propose to replace a sequence of supervised losses with a sequence of *loss proxies* that can be computed from unlabeled test streams. This allows us to derive an 'unsupervised' lower bound (Proposition 1) to the running test risk which we use to design an 'unsupervised' alarm function (Eq. 5).

As a first step, we introduce the notion of a loss proxy and specify its desirable properties. For a chosen proxy function g, a *loss proxy* of a model p is defined as $\mathbf{u} := g(\mathbf{x}, p)$. Besides being 'unsupervised' (i.e., it should depend only on features $\mathbf{x}$), the proxy should be (at least partially) informative of the corresponding loss variable $\mathbf{z}$. Before presenting our concrete choice of a proxy function based on model uncertainty (see § B.1), we formalize the notion of a proxy's informativeness with the following assumption.

Assumption 1. *Given a sequence of losses $\mathbf{z}_{0:t}$, let the corresponding sequence of loss proxies $\mathbf{u}_{0:t}$ and proxy thresholds $\lambda_0, \ldots, \lambda_t \in \mathbb{R}$, along with a loss threshold $\tau \in (0, M)$, be such that for all $t \geq 1$, the following inequality holds:*

$$\frac{1}{t}\sum_{k=1}^{t} \underbrace{\mathbb{P}_{P_k}\big(\mathbf{u}_k > \lambda_k, \mathbf{z}_k \leq \tau\big)}_{PFP_k} \leq \underbrace{\mathbb{P}_{P_0}\big(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 \leq \tau\big)}_{PFP_0} + \frac{1}{t}\sum_{k=1}^{t} \underbrace{\mathbb{P}_{P_k}\big(\mathbf{u}_k \leq \lambda_k, \mathbf{z}_k > \tau\big)}_{PFN_k}.$$

While Assumption 1 may initially appear rather complicated, it can be interpreted in terms of two intuitive desiderata for a valid (and effective) loss proxy. First, the proxy $\mathbf{u}$ should enable separation between low losses ($\mathbf{z} \leq \tau$) and high losses ($\mathbf{z} > \tau$) for a fixed loss threshold τ. This ensures that the probabilities of both false positives (PFP_k) and false negatives (PFN_k) are small. Second, this separability should be robust across the time-varying test distributions P_k, ensuring that the false positive rate on the test stream (PFP_k) remains comparable to that on the source distribution (PFP_0). Below we formalize how a sequence of loss proxies can be used to derive an unsupervised lower bound on the true running test risk.

Proposition 1. *Assume a non-negative, bounded loss $\ell \in [0, M], M > 0$. Further, assume that for a sequence of losses $\mathbf{z}_{0:t}$, a sequence of loss proxies $\mathbf{u}_{0:t}$ together with thresholds $\lambda_0, \ldots, \lambda_t \in \mathbb{R}, \tau \in (0, M)$ satisfying Assumption 1 are available. Then the running test risk can be lower bounded $\forall t \geq 1$ as*

$$\bar{R}_t(p_{1:t}) \geq \underbrace{\tau\left(\frac{1}{t}\sum_{k=1}^{t}\mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k) - \mathbb{P}_{P_0}(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 \leq \tau)\right)}_{:=B_t}.$$

We defer the full proof to § C.2. A similar bound was proposed by Amoukou et al. (2024), though with some key differences, which we discuss in detail in § B.3 and Appendix F. Importantly, the bound B_t from Proposition 1 depends only on the test loss proxies and the source loss, meaning its corresponding lower-bound confidence sequence L_t^b can be evaluated using a combination of unlabeled test data ($\mathcal{D}_{\boldsymbol{x}}^k$) and labeled source data ($\mathcal{D}_{\text{cal}}$). This makes it suitable for our proposed unsupervised alarm:

$$\Phi_t^b := \mathbb{1}\left[L_t^b\big(\mathbf{u}_{0:t}, \lambda_{0:t}, \mathbf{z}_0, \tau\big) > U\big(\mathbf{z}_0\big) + \epsilon_{\text{tol}}\right]. \quad (5)$$

In § C.3, we prove that such an alarm has a PFA control guarantee for the sequential test in Eq. 2. We choose uncertainty as our loss proxy $\mathbf{u}$. Specifically, we define the proxy function using the maximum class probability as $g(\mathbf{x}, p) := 1 - \max_c p(\mathbf{x})_c$. Our choice is motivated in § B.1. To obtain thresholds $\lambda_{0:t}$ such that 1 remains valid and L_t^b is sufficiently tight, we propose an online calibration procedure in § B.2.

4. Experiments

We empirically validate the effectiveness of our monitoring tool for a range of TTA methods under different distribution shifts. In § 4.1, we demonstrate the wide applicability of our monitoring tool across different TTA methods and datasets. In § 4.2, we show that the tool can be employed to detect risk increase arising from failed model adaptation. We provide comparison to baseline alarm functions in § B.4 as well as ablation on the loss proxy (§ B.6).

Risk Control Design Our goal is to approximate, $\hat{\bar{R}}_t := \frac{1}{t}\sum_{k=1}^{t}\hat{R}_k(p_k)$, the empirical estimate of the true, unobservable, running test risk $\bar{R}_t(p_{1:t})$ as closely as possible. Once $\hat{\bar{R}}_t$ exceeds a pre-defined risk threshold, we wish to raise an alarm as early as possible. We also verify the validity of Assumption 1 throughout adaptation by tracking $\Delta_t^b = PFP_0 + \frac{1}{t}\sum_{k=1}^{t} PFN_k - PFP_k$. The assumption is met in practice when $\Delta_t^b \geq 0$ and violated when $\Delta_t^b < 0$. Δ_t^b also reflects the tightness of L_t^b, so, ideally, it is also not too much above 0. We monitor test risk focusing on 0-1 loss. If not specified otherwise, we use a tolerance threshold of $\epsilon_{\text{tol}} = 0.05$. We set $\alpha = \alpha_{\text{source}} + \alpha_{\text{test}}$ to 0.2 using most budget for controlling the test risk, i.e. $\alpha_{\text{test}} = 0.175$ and $\alpha_{\text{source}} = 0.025$. For threshold selection (§ B.2) we use $N_{\text{cal}} = 1000$ labeled samples from P_0.

4.1. Generalization across Datasets, Shifts and TTA Methods

We evaluate the robustness of our monitoring tool by testing different TTA methods: TENT (Wang et al., 2021), CoTTA (Wang et al., 2022), SAR (Niu et al., 2023) and SHOT (Liang et al., 2020). Please see § E.3 for details. We study four test streams: In-distribution of ImageNet (no shift,

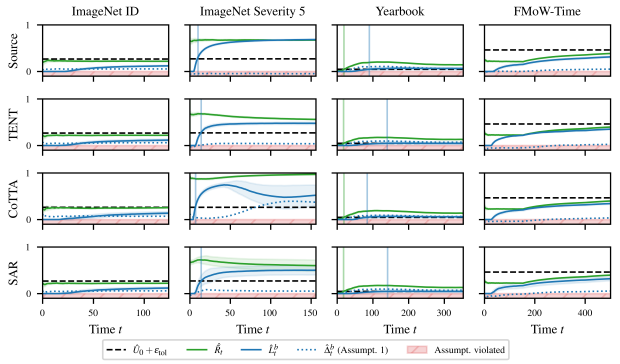

Figure 1: Estimated test risk for different datasets and TTA methods: Our lower bound $\hat{L}_t^b$ consistently exceeds the risk threshold $\hat{U}_0 + \epsilon_{\text{tol}}$ when a true risk violation occurs (ImageNet severity 5, Yearbook), while remaining below it on benign shifts (ImageNet ID, FMoW-Time), across all TTA methods.

alarm should remain silent), ImageNet-C Gaussian noise (GN) severity level 5 (strong shift), Yearbook (moderate shift) and FMoW (gradual shift). Since classification error is the most commonly used metric in TTA, we track a risk increase for 0-1 loss.

Risk violation is detected reliably Fig. 6 shows that the empirical running test risk $\hat{\bar{R}}_t$ (—) is closely mimicked by our $\hat{L}_t^b$ (—) across TTA methods, datasets and shifts. Our alarm function correctly remains silent on the ID stream (*first column*) of ImageNet, where test risk remains below the threshold (- -). For FMoW, the risk increases steadily but also remains below the alarm threshold; this is accurately reflected in our monitoring, as $\hat{L}_t^b$ tightly tracks $\hat{\bar{R}}_t$ without triggering false alarms. For the immediate risk violation on ImageNet-C severity level 5 (*second column*), our test triggers an alarm after < 25 steps for all TTA methods. Similar results are observed for Yearbook. SHOT results, consistent with Fig. 6, are shown in § B.5.

Assumption 1 holds after warm-up Importantly, we find that 1 is generally satisfied in practice, with $\hat{\Delta}_t^b$ (···) remaining above zero for most time steps, when using model uncertainty (§ B.1) with online adaptation of proxy thresholds (§ B.2). For some datasets, such as FMoW, we observe slight violations during the warm-up phase, i.e., for small t. Fortunately, the finite-sample penalty term in the confidence sequence is largest for small t, which may offset these minor violations and help prevent false alarms. The only instance where violations persist throughout the entire test stream is with the source model on ImageNet under a severity 5 shift. This is because our proposed threshold calibration procedure (§ B.2) keeps the proxy threshold fixed if the model is not updated on the test stream, making 1 more difficult to satisfy.

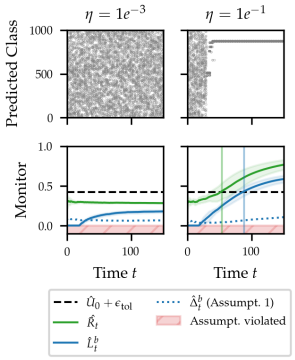

Figure 2: Collapsed vs. non-collapsed model on ImageNet-C (GN): When collapsed (*right*), the model always predicts the same class, which our monitor flags.

4.2. Detecting TTA Collapse

Unlike static models, the risk of a TTA model can increase not only due to distribution shift but also because the model deteriorates during adaptation. An extreme, yet well documented case of model failure in TTA is model collapse, where finally only a small subset or a single class is predicted (Nguyen et al., 2023; Su et al., 2023; Niu et al., 2023; Marsden et al., 2024). Alarmingly, these harmful collapses often occur silently (Niu et al., 2023). We next ask whether our statistical framework can detect risk increases caused by model failure. This is not a given, as the monitor relies on the model's own outputs (e.g., predictive uncertainty), which may become unreliable when the model itself fails. To induce model collapse, we follow (Boudiaf et al., 2022) and apply TENT with a high learning rate of $\eta = 1e^{-1}$ on the ImageNet-C (GN) corruption at severity level 1. We set a high $\epsilon_{\text{tol}} = 0.2$ to disregard risk increase caused by distribution shift.

Fig. 2 (*left*) displays predicted classes (*first row*) and estimated test risk (*second row*) for an adaptation with a suitable learning rate. The predicted classes remain diverse, and both the estimated test risk and our lower bound stay below the pre-defined risk threshold. This is in stark contrast to adaptation with a high learning (*right*): after few adaptation steps, the model assigns all input samples to the same class. This leads to a large increase in empirical test risk. Encouragingly, our bound $\hat{L}_t^b$ tracks this rise and detects a violation shortly after, demonstrating that our monitoring remains effective even when the underlying model collapses.

5. Conclusion

We introduced an unsupervised risk monitoring tool for test-time adaptation (TTA) based on sequential testing. Our method supports continuous adaptation and reliably detects performance drops due to harmful shifts or adaptation collapse, demonstrating broad applicability across diverse TTA methods.

References

Zehao Xiao and Cees GM Snoek. Beyond model adaptation at test time: A survey. *arXiv preprint arXiv:2411.03687*, 2024. 1, 23

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *International Conference on Learning Representations*, 2021. 1, 2, 3, 10, 22, 23

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211, 2022. 1, 3, 16, 22, 23

Ori Press, Ravid Shwartz-Ziv, Yann LeCun, and Matthias Bethge. The entropy enigma: success and failure of entropy minimization. In *Proceedings of the 41st International Conference on Machine Learning*, pages 41064–41085, 2024a. 1, 23

Aleksandr Podkopaev and Aaditya Ramdas. Tracking the risk of a deployed model and detecting harmful distribution shifts. *ICLR*, 2022. 1, 2, 9, 12, 16, 18, 23, 24

Steven R Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49 (2):1055–1080, 2021. 1, 2, 9, 18, 22

Salim I Amoukou, Tom Bewley, Saumitra Mishra, Freddy Lecue, Daniele Magazzeni, and Manuela Veloso. Sequential harmful shift detection without labels. *Advances in Neural Information Processing Systems*, 37:129279–129302, 2024. 1, 2, 3, 10, 11, 12, 13, 16, 18, 22, 23

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *International Conference on Learning Representations*, 2023. 3, 4, 22, 23

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020. 3, 22, 23

A Tuan Nguyen, Thanh Nguyen-Tang, Ser-Nam Lim, and Philip HS Torr. Tipi: Test time adaptation with transformation invariance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24162–24171, 2023. 4

Yi Su, Yixin Ji, Juntao Li, Hai Ye, and Min Zhang. Beware of model collapse! fast and stable test-time adaptation for robust question answering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12998–13011, 2023. 4

Robert A Marsden, Mario Döbler, and Bin Yang. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2555–2565, 2024. 4, 23

Malik Boudiaf, Romain Mueller, Ismail Ben Ayed, and Luca Bertinetto. Parameter-free online test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8344–8353, 2022. 4, 23

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023. 9, 23

Donald A Darling and Herbert Robbins. Confidence sequences for mean, variance, and median. *Proceedings of the National Academy of Sciences*, 58(1):66–68, 1967. 9

Elan Rosenfeld and Saurabh Garg. (almost) provable error bounds under distribution shift via disagreement discrepancy. *Advances in Neural Information Processing Systems*, 36:28761–28784, 2023. 10, 23

Charles Corbière, Nicolas Thome, Avner Bar-Hen, Matthieu Cord, and Patrick Pérez. Addressing failure prediction by learning model confidence. *Advances in neural information processing systems*, 32, 2019. 10, 23

Qingyang Zhang, Yatao Bian, Xinke Kong, Peilin Zhao, and Changqing Zhang. Come: Test-time adaption by conservatively minimizing entropy. *arXiv preprint arXiv:2410.10894*, 2024. 10

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 2019. 10

Andreas Kirsch and Yarin Gal. A note on "assessing generalization of SGD via disagreement"'. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. 10, 23

Fei Zhu, Zhen Cheng, Xu-Yao Zhang, and Cheng-Lin Liu. Rethinking confidence calibration for failure prediction. In *European conference on computer vision*, pages 518–536. Springer, 2022. 10

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017. 10

Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 2021. 15, 22, 23

Mona Schirmer, Dan Zhang, and Eric Nalisnick. Test-time adaptation with state-space models. In *ICML 2024 Workshop on Structured Probabilistic Inference and Generative Modeling*, 2024. 15, 22, 23

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International conference on machine learning*, pages 9690–9700. PMLR, 2020. 15

Yifei Ming, Yiyou Sun, Ousmane Dia, and Yixuan Li. How to exploit hyperspherical embeddings for out-of-distribution detection? *arXiv preprint arXiv:2203.04450*, 2022. 15

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020. 15

Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11920–11929, 2023. 16

Hyesu Lim, Byeonggeun Kim, Jaegul Choo, and Sungha Choi. Ttn: A domain-shift aware batch normalization in test-time adaptation. *arXiv preprint arXiv:2302.05155*, 2023. 16, 23

Sungha Choi, Seunghan Yang, Seokeon Choi, and Sungrack Yun. Improving test-time adaptation via shift-agnostic weight regularization and nearest source prototypes. In *European Conference on Computer Vision*, pages 440–458. Springer, 2022. 16

Kazuki Adachi, Shin'Ya Yamaguchi, and Atsutoshi Kumagai. Covariance-aware feature alignment with pre-computed source statistics for test-time adaptation to multiple image corruptions. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 800–804. IEEE, 2023. 16

Yarin Bar, Shalev Shaer, and Yaniv Romano. Protected test-time adaptation via online entropy matching: A betting approach. *Advances in Neural Information Processing Systems*, 37:85467–85499, 2024. 16, 23

Sanghun Jung, Jungsoo Lee, Nanhee Kim, Amirreza Shaban, Byron Boots, and Jaegul Choo. Cafa: Class-aware feature alignment for test-time adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19060–19071, 2023. 16

Taeyeop Lee, Jonathan Tremblay, Valts Blukis, Bowen Wen, Byeong-Uk Lee, Inkyu Shin, Stan Birchfield, In So Kweon, and Kuk-Jin Yoon. Tta-cope: Test-time adaptation for category-level object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21285–21295, 2023a. 16

Daeun Lee, Jaehong Yoon, and Sung Ju Hwang. Becotta: Input-dependent online blending of experts for continual test-time adaptation. *International Conference on Machine Learning*, 2024a. 16

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ICLR*, 2019. 21

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 21

Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1–7, 2015. 21

Huaxiu Yao, Caroline Choi, Bochuan Cao, Yoonho Lee, Pang Wei W Koh, and Chelsea Finn. Wild-time: A benchmark of in-the-wild distribution shift over time. *Advances in Neural Information Processing Systems*, 2022. 22

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 22

Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018. 22

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 22

Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019. 22

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 22

Steven R. Howard, Ian Waudby-Smith, and Aaditya Ramdas. ConfSeq: software for confidence sequences and uniform boundaries, 2021–. URL https://github.com/gostevehoward/confseq. [Online; accessed <today>]. 22

Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, pages 1–34, 2024. 23

Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems*, 2020. 23

Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint (arXiv:2006.10963)*, 2020. 23

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning*, pages 16888–16905. PMLR, 2022. 23

Yige Yuan, Bingbing Xu, Liang Hou, Fei Sun, Huawei Shen, and Xueqi Cheng. Tea: Test-time energy adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23901–23911, 2024. 23

Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013. 23

Hao Zhao, Yuejiang Liu, Alexandre Alahi, and Tao Lin. On pitfalls of test-time adaptation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 42058–42080. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/zhao23d.html. 23

Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation. *Advances in Neural Information Processing Systems*, 2022. 23

Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15922–15932, 2023. 23

Mario Döbler, Robert A Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7704–7714, 2023. 23

Tong Wu, Feiran Jia, Xiangyu Qi, Jiachen T Wang, Vikash Sehwag, Saeed Mahloujifar, and Prateek Mittal. Uncovering adversarial risks of test-time adaptation. *arXiv preprint arXiv:2301.12576*, 2023. 23

Hyejin Park, Jeongyeon Hwang, Sunung Mun, Sangdon Park, and Jungseul Ok. Medbn: Robust test-time adaptation against malicious test samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5997–6007, 2024. 23

Taeckyung Lee, Sorn Chottananurak, Taesik Gong, and Sung-Ju Lee. Aetta: Label-free accuracy estimation for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28643–28652, 2024b. 23, 24

Trung Hieu Hoang, MinhDuc Vo, and Minh Do. Persistent test-time adaptation in recurring testing scenarios. *Advances in Neural Information Processing Systems*, 37: 123402–123442, 2024. 23

Yongyi Su, Xun Xu, and Kui Jia. Towards real-world test-time adaptation: Tri-net self-training with balanced normalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 15126–15135, 2024. 23

Chaoqun Du, Yulin Wang, Jiayi Guo, Yizeng Han, Jie Zhou, and Gao Huang. Unitta: Unified benchmark and versatile framework towards realistic test-time adaptation. *arXiv preprint arXiv:2407.20080*, 2024. 23

Yuejiang Liu, Parth Kothari, Bastien Van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. Ttt++: When does self-supervised test-time training fail or thrive? *Advances in Neural Information Processing Systems*, 34:21808–21820, 2021. 23

Ori Press, Steffen Schneider, Matthias Kümmerer, and Matthias Bethge. Rdumb: A simple approach that questions our progress in continual test-time adaptation. *Advances in Neural Information Processing Systems*, 2024b. 23

Kentaro Hoffman, Stephen Salerno, Jeff Leek, and Tyler McCormick. Some models are useful, but for how long?: A decision theoretic approach to choosing when to refit large-scale prediction models. *arXiv preprint arXiv:2405.13926*, 2024. 23

Amartya Sanyal, Yaxi Hu, Yaodong Yu, Yian Ma, Yixin Wang, and Bernhard Schölkopf. Accuracy on the wrong line: On the pitfalls of noisy data for out-of-distribution generalisation. *arXiv preprint arXiv:2406.19049*, 2024. 23

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International conference on machine learning*, pages 7721–7735. PMLR, 2021. 23

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *International Conference on Learning Representations*, 2017. 23

Saurabh Garg, Sivaraman Balakrishnan, Zachary C Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. *International Conference on Machine Learning*, 2022. 23

Yuzhe Lu, Zhenlin Wang, Runtian Zhai, Soheil Kolouri, Joseph Campbell, and Katia Sycara. Predicting out-of-distribution error with confidence optimal transport. *arXiv preprint arXiv:2302.05018*, 2023. 23

Devin Guillory, Vaishaal Shankar, Sayna Ebrahimi, Trevor Darrell, and Ludwig Schmidt. Predicting with confidence on unseen distributions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1134–1144, 2021. 23

Yiding Jiang, Vaishnavh Nagarajan, Christina Baek, and J Zico Kolter. Assessing generalization of sgd via disagreement. *International Conference on Learning Representations*, 2022. 23, 24

Christina Baek, Yiding Jiang, Aditi Raghunathan, and J Zico Kolter. Agreement-on-the-line: Predicting the performance of neural networks under distribution shift. *Advances in Neural Information Processing Systems*, 35: 19274–19289, 2022. 23

Preetum Nakkiran and Yamini Bansal. Distributional generalization: A new kind of generalization. *arXiv preprint arXiv:2009.08092*, 2020. 23

Donghwan Lee, Behrad Moniri, Xinmeng Huang, Edgar Dobriban, and Hamed Hassani. Demystifying disagreement-on-the-line in high dimensions. In *International Conference on Machine Learning*, pages 19053–19093. PMLR, 2023b. 23

Eungyeup Kim, Mingjie Sun, Christina Baek, Aditi Raghunathan, and J Zico Kolter. Test-time adaptation induces stronger accuracy and agreement-on-the-line. *Advances in Neural Information Processing Systems*, 37:120184–120220, 2024. 23

Shubhanshu Shekhar and Aaditya Ramdas. Sequential changepoint detection via backward confidence sequences. In *International Conference on Machine Learning*, pages 30908–30930. PMLR, 2023a. 24

Shubhanshu Shekhar and Aaditya Ramdas. Reducing sequential change detection to sequential estimation. *arXiv preprint arXiv:2309.09111*, 2023b. 24

Appendix

A. Background

A.1. Supervised Risk Monitoring via Sequential Testing

To track how well a deployed model is performing, Podkopaev and Ramdas (2022) propose a *risk monitoring* framework based on *sequential testing* (Ramdas et al., 2023). The performance of a model p, in the presence of a *labeled* test stream, is tracked using the following sequential test:

$$H_0 : \bar{R}_t(p) \leq R_0(p) + \epsilon_{\text{tol}},\ \forall t \geq 1 \qquad H_1 : \exists t^* \geq 1 : \bar{R}_{t^*}(p) > R_0(p) + \epsilon_{\text{tol}} \tag{6}$$

where $\epsilon_{\text{tol}} > 0$ is a tolerance level that quantifies the acceptable drop in a model's test performance relative to its source performance.

To give the test anytime-valid properties (e.g. arbitrary stopping and restarting), Podkopaev and Ramdas (2022) rely on *confidence sequences*, which extend traditional confidence intervals to the sequential setting and offer time-uniform coverage guarantees (Darling and Robbins, 1967; Howard et al., 2021). A sequence of model losses on test data is used to construct an anytime-valid lower bound L_t on the true running test risk $\bar{R}_t$:

$$\mathbb{P}\big(\bar{R}_t(p) \geq L_t\big(\mathbf{z}_1, \ldots, \mathbf{z}_t\big),\ \forall t \geq 1\big) \geq 1 - \alpha_{\text{test}}$$

for a miscoverage level $\alpha_{\text{test}} \in (0, 1)$. To get an upper bound U on the source risk, a regular (static) confidence interval is computed using the loss on the source data:

$$\mathbb{P}\big(\ R_0(p) \leq U\big(\mathbf{z}_0\big)\ \big) \geq 1 - \alpha_{\text{source}}$$

for another miscoverage level $\alpha_{\text{source}} \in (0, 1)$. Combining the two bounds, the following *alarm function* is proposed

$$\Phi_t = \mathbb{1}\left[L_t\big(\mathbf{z}_1, \ldots, \mathbf{z}_t\big) > U\big(\mathbf{z}_0\big) + \epsilon_{\text{tol}}\right] \tag{7}$$

and used to reject the null hypothesis (Eq. 6) at $t_{\text{min}} := \inf\{t \geq 1 \mid \Phi_t = 1\}$. See Fig. 3 for an illustration. Note that in practice, the empirical bounds are computed using empirical risks:

$$\hat{U} = \hat{R}_0(p; \mathcal{D}_{\text{cal}}) + w_0\ , \quad \hat{L}_t = \frac{1}{t}\sum_{k=1}^{t} \hat{R}_k(p; \mathcal{D}^k_{\boldsymbol{xy}}) - w_t\ ,$$

where w_0, w_t are finite-sample correction terms (see § C.1 for concrete formulas). Owing to the power of confidence sequences, the alarm function Φ_t enjoys a time-uniform guarantee on the control of the probability of the false alarm (PFA)

$$\mathbb{P}_{H_0}(\exists t \geq 1, \Phi_t = 1) \leq \alpha_{\text{test}} + \alpha_{\text{source}}$$

which ensures that performance degradations are not incorrectly detected, thereby avoiding unnecessary (and potentially costly) interventions on the model. Notably, this guarantee requires only that the loss function is bounded $\ell \in [a, b]$. This minimal assumption makes the method broadly applicable, as it imposes no constraints on the data distributions, the predictive model, or the nature of distribution shift (beyond independence). To maintain minimal assumptions, it is necessary to rely on a conjugate-mixture empirical Bernstein bound (Howard et al., 2021) when constructing a lower confidence sequence for the test risk L_t (see § C.1 for more details).

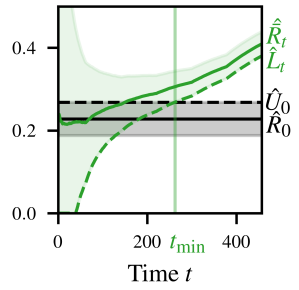

Figure 3: Alarm Φ_t is raised at t_{min} as the lower bound $\hat{L}_t$ on the running test risk $\hat{\bar{R}}_t$ exceeds the upper bound $\hat{U}_0$ on the source risk $\hat{R}_0$.

B. Additional Results

B.1. Uncertainty as Loss Proxy

After introducing a general loss proxy $\mathbf{u}$ in § 3.1, we here present a concrete instantiation based on model uncertainty. Specifically, we define the proxy function using the maximum class probability as $g(\mathbf{x}, p) := 1 - \max_c p(\mathbf{x})_c$. We choose uncertainty, firstly, due to it being easy to implement: it requires no modifications to the underlying model and avoids the need for additional components, unlike alternative proxies based on model disagreement (Rosenfeld and Garg, 2023) or separate error estimators (Amoukou et al., 2024; Corbière et al., 2019). Secondly, for 0-1 loss this score approximates the conditional risk, up to calibration error:

$$R_{\text{0-1}}\left(p; \mathbf{x}\right) \;=\; \sum_{c=1}^{C} P(\mathbf{y} = c|\mathbf{x}) \cdot \mathbb{1}[\hat{y}(\mathbf{x}) \neq c] \;\approx\; \sum_{c=1}^{C} p(\mathbf{x})_c \cdot \mathbb{1}[\hat{y}(\mathbf{x}) \neq c] \;=\; 1 - \max_c p(\mathbf{x})_c \,.$$

Although the conditional risk approximation improves when p is well-calibrated, we *do not* need to assume the model's uncertainty is well-calibrated under model adaptation (Zhang et al., 2024) nor under distribution shift (Ovadia et al., 2019), as some previous work has required (Kirsch and Gal, 2022). Returning to Assumption 1, uncertainty is a useful loss proxy when it separates high-loss and low-loss instances for a carefully chosen threshold λ_k—a task known as *failure prediction* (Corbière et al., 2019; Zhu et al., 2022). Failure prediction boils down to the ability to rank the test instances according to their loss values, which is a much weaker requirement in comparison to calibration (Guo et al., 2017). Before demonstrating empirically in § 4 that using model uncertainty in Proposition 1 yields valid and tight lower bounds when monitoring TTA performance, we describe our threshold selection mechanism below.

B.2. Online Threshold Calibration

We here describe our procedure for selecting the loss and proxy thresholds used in the lower bound from Proposition 1. This step is critical for the effectiveness of our risk monitoring tool: poorly chosen thresholds can yield bounds that are either invalid (i.e., violate Assumption 1) or vacuous (i.e., excessively loose). Since our goal is to simultaneously minimize false positives and false negatives (cf. Assumption 1), we determine the loss threshold $\tau \in (0, C)$ and the source proxy threshold $\lambda_0 \in (0, 1)$ by maximizing the F1 score based on the source model's proxy:

$$\hat{\lambda}_0, \hat{\tau} := \arg\max_{\lambda,\tau} \text{F1}(\lambda, \tau;\, \mathcal{D}_{\text{cal}}, p_0)\,, \qquad \text{F1}(\lambda, \tau) = \frac{2\text{TP}}{2\text{TP} + \text{FN} + \text{FP}}$$

where $\text{TP} = \sum_{i=1}^{N_{\text{cal}}} \mathbb{1}[u_{0,i} > \lambda, z_{0,i} > \tau]$, $\text{FN} = \sum_{i=1}^{N_{\text{cal}}} \mathbb{1}[u_{0,i} \leq \lambda, z_{0,i} > \tau]$, $\text{FP} = \sum_{i=1}^{N_{\text{cal}}} \mathbb{1}[u_{0,i} > \lambda, z_{0,i} \leq \tau]$ and $u_{0,i} \sim \mathbf{u}_0$ and $z_{0,i} \sim \mathbf{z}_0$ are proxy and loss realizations of the source model p_0 on samples in $\mathcal{D}_{\text{cal}}$, respectively. Similarly, to select test proxy thresholds $\lambda_{1:t}$ we maximize F1 score while keeping the loss threshold $\hat{\tau}$ fixed: $\hat{\lambda}_k := \arg\max_\lambda \text{F1}(\lambda, \hat{\tau}; \mathcal{D}_{\text{cal}}, p_k)$, where F1 is computed from proxy $u_0^{(k)} \sim \mathbf{u}_0^{(k)}$ and loss $z_0^{(k)} \sim \mathbf{z}_0^{(k)}$ realizations of the *adapted* model p_k on the (same) calibration dataset $\mathcal{D}_{\text{cal}}$ (since no test labels are available). We emphasize that continuously adapting the proxy threshold is essential for preserving an effective bound B_t under model adaptation. Using a static threshold throughout the test stream is insufficient, as many TTA methods can affect the scale of the observed uncertainties. For example, TENT (Wang et al., 2021) tends to reduce uncertainty over time due to its entropy minimization objective.

B.3. Improving Test Power

We have previously shown that our proposed unsupervised alarm (Eq. 5) provides strong false alarm control guarantees under H_0 (§ C.3)—that is, the alarm is guaranteed not to trigger when no performance degradation occurs in practice, preventing taking the model 'offline' prematurely. However, for a monitoring tool to be truly useful, it must also be 'reactive' under H_1—that is, it should raise an alarm when the model's performance degrades beyond an acceptable tolerance level (ϵ_{tol}), and ideally, it should do so with minimal detection delay. Since our unsupervised alarm is based on lower-bounding the true running test risk twice —a lower-bound confidence sequence L_t^b to a lower bound B_t is used—it is not too surprising that the procedure can sometimes exhibit overly conservative behavior under H_1. We next discuss our strategies for addressing this issue by improving the power of our proposed unsupervised sequential test.

0-1 loss We first note that for the 0-1 loss, the loss threshold τ can be omitted from the lower bound B_t in Proposition 1. This is a direct consequence of the binary nature of the 0-1 loss (see Corollary 1 in § C.4 for the full derivation). Omitting

this scaling for 0-1 loss yields a tighter lower bound B_t, which directly translates into a more reactive alarm function—while still maintaining false alarm guarantees. This is especially important when monitoring performance in TTA, where 0-1 loss is the one most widely used (as its risk corresponds to the classifier error).

Continuous Losses For continuous losses such as Brier, which can take on any value in $[0, 1]$, the lower bound must be scaled by a threshold $\tau \in (0, 1)$, resulting in looser bounds.[2] To recover some of the lost test power, we propose also lower bounding the source risk R_0 using the same threshold τ as in the test lower bound B_t (Proposition 1):

$$R_0 = \mathbb{E}[\mathbf{z}_0] \geq \tau \mathbb{P}(\mathbf{z}_0 > \tau) =: B_0$$

which follows directly from Markov's inequality. Denoting the corresponding upper-bound of the confidence interval for B_0 with U^b, we define the alarm function:

$$\Phi_t^\tau := \mathbb{1}\left[\frac{1}{\tau} L_t^b\big(\mathbf{u}_{0:t}, \lambda_{0:t}, \mathbf{z}_0, \tau\big) > \frac{1}{\tau} U^b\big(\mathbf{z}_0, \tau\big) + \tilde{\epsilon}_{\text{tol}} \right] \tag{8}$$

where $\tilde{\epsilon}_{\text{tol}} := \frac{\epsilon_{\text{tol}}}{\tau}$ and show its PFA guarantee for the following sequential test

$$H_0^\tau : \frac{1}{t} \sum_{k=1}^{t} \mathbb{P}_{P_k}(\mathbf{z}_k > \tau) \leq \mathbb{P}_{P_0}(\mathbf{z}_0 > \tau) + \tilde{\epsilon}_{\text{tol}}, \ \forall t \geq 1 \tag{9}$$

$$H_1^\tau : \exists t^* \geq 1 : \frac{1}{t^*} \sum_{k=1}^{t^*} \mathbb{P}_{P_k}(\mathbf{z}_k > \tau) > \mathbb{P}_{P_0}(\mathbf{z}_0 > \tau) + \tilde{\epsilon}_{\text{tol}} \, .$$

The proof is provided in § C.5. Comparing the two sequential tests, we note that Eq. 9 tracks the probability of high loss, whereas Eq. 2 makes a statement about the *expected* loss (i.e., risk). While the test in Eq. 2 is arguably more interpretable—especially considering that the loss threshold τ is not specified by the user but determined empirically through a threshold calibration procedure (see Algo. 2)—the advantage of the high-loss probability test in Eq. 9 lies in its greater reactivity. Specifically, the lower bound L_t^b in the alarm function (Eq. 8) is no longer scaled by τ (due to the multiplication by $\frac{1}{\tau}$), resulting in a tighter bound that can recover some of the statistical power lost in the continuous loss setting, albeit at the cost of reduced interpretability.

We also note that the scaled alarm function (Eq. 8) is closely related to the *quantile detector* proposed in Amoukou et al. (2024). Our work extends their approach in (at least) three main ways: first, by allowing for continuously evolving models, unlike (Amoukou et al., 2024) where a static model is assumed; second, by relaxing the assumption required for the loss proxy (see our Assumption 1 versus their Assumption 4.1); and third, by providing a theoretical justification for the increased reactivity of the high-probability test relative to the expected-loss test for continuous losses (via the cancellation of the loss threshold τ). We elaborate further on these differences in Appendix F.

[2] For a loss bounded in $[0, M]$ with $M > 1$, it is theoretically possible that threshold calibration yields $\hat{\tau} > 1$, in which case scaling by $\hat{\tau}$ could produce a tighter lower bound. However, since all losses considered in this work are bounded above by 1, we leave this case for future work.

B.4. Comparison to Baselines

Baselines We compare our unsupervised alarm Φ^b (Eq. 5 and Eq. 8), to several baseline monitors. While all monitors use the same upper bound on the source risk U_0, they differ in their choice of the test risk lower bound. We next present the alternatives to our proposed test risk lower bound L_t^b:

- $\hat{L}_t^a$: the estimated confidence lower bound on the running test risk under model adaptation (see Eq. 4). This direct extension of Podkopaev and Ramdas (2022) preserves false-alarm guarantees but observes labels at each time point. While inapplicable in the unsupervised TTA setting, it serves as an oracle baseline. Since $L_t^a \geq L_t^b$ (under Assumption 1) our alarm Φ_t^b can never trigger before this alarm, Φ_t^a, and consequently we inherit its detection delay. The closer $\hat{L}_t^b$ is to $\hat{L}_t^a$, the smaller is the price we pay for not observing test labels.
- $\hat{L}_t^c$: a naive estimate of the running test risk, formed by substituting the supervised losses $\mathbf{z}_{0:t}$ with unsupervised proxies $\mathbf{u}_{0:t}$ in the alarm from Eq. 7 (Podkopaev and Ramdas, 2022). While it avoids using test labels, it lacks false alarm guarantees due to omitting the lower bounding step in Prop. 1.
- $\hat{L}_t^d$: the estimated unsupervised lower bound on the running test risk of the static source model p_0 as presented in (Amoukou et al., 2024). While providing false alarm guarantees without access to labels, it uses a different calibration procedure and is not applicable to a time-varying predictive model p_k.

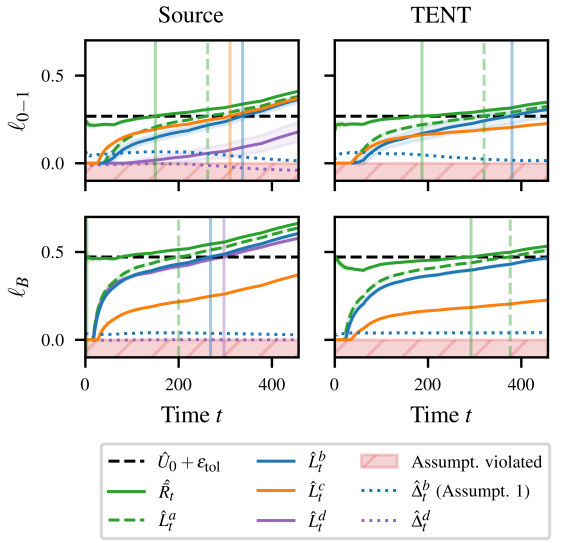

Figure 4: Test risk of increasing severity on ImageNet-C (GN): Our unsupervised lower bound $\hat{L}_t^b$ on the empirical test risk $\hat{\hat{R}}_t$ closely follows the supervised lower bound $\hat{L}_t^a$.

Illustrative Example In the first experiment, we study the behavior of our alarm in comparison to described baselines. We are notably interested how closely our unsupervised monitoring tool mimics the two oracle quantities having access to the ground truth test labels: empirical running test risk $\hat{\hat{R}}_t$ and $\hat{L}_t^a$ (Podkopaev and Ramdas, 2022). To simulate an increasing test risk, we construct a test stream from ImageNet-C by gradually increasing the severity level of Gaussian noise corruption from in-distribution (no shift) up to severity level 5. We track both the unadapted source model and TENT using 0-1 loss $\ell_{0\text{-}1}$ and Brier loss ℓ_B. We also verify the validity of Assumption 1 throughout adaptation by tracking $\Delta_t^b = PFP_0 + \frac{1}{t}\sum_{k=1}^{t} PFN_k - PFP_k$. The assumption is met in practice when $\Delta_t^b \geq 0$ and violated when $\Delta_t^b < 0$. Δ_t^b also reflects the tightness of L_t^b, so, ideally, it is also not too much above 0. We proceed analogously for Assumption 4.1 in Amoukou et al. (2024) and denote it with Δ_t^d.

The results are shown in Fig. 4. As the severity of the distribution shift increases, the empirical running test risk $\hat{\hat{R}}_t$ (—) increases as well, for both 0-1 loss $\ell_{0\text{-}1}$ (*top row*) and Brier loss ℓ_B (*bottom row*). As expected, the unadapted source model (*left*) exhibits a higher risk, while adaptation with TENT (*right*) postpones the point where the empirical risk crosses the specified performance requirement. However, as the distribution shift becomes increasingly severe, $\hat{\hat{R}}_t$ eventually exceeds the upper bound on the source risk, $\hat{U}_t$, plus the tolerance margin ϵ_{tol} (- -), despite model adaptation. From this time point (|), we wish to trigger an alarm. As expected, the lower confidence sequence on the empirical test risk, $\hat{L}_t^a$ (- -), which leverages test labels, detects the risk violation first. Encouragingly, our proposed unsupervised lower bound $\hat{L}_t^b$ (—) closely follows the supervised bound $\hat{L}_t^a$. This indicates that our bound is tight and the price for not seeing labels is relatively small. The naive plugin bound, $\hat{L}_t^c$ (—), is not only void of theoretical guarantees but also exhibits low power empirically by not detecting the risk violation in all but one case. The unsupervised bound by Amoukou et al. (2024), $\hat{L}_t^d$ (—), detects slightly later then $\hat{L}_t^b$ for Brier loss, but is extremely loose for 0-1 loss. Fig. 4 shows that our Assumption 1 (···) is met throughout the distribution shift in all cases, while the assumption of Amoukou et al. (2024) (···) is violated for 0-1 loss, making their bound invalid for large t.

Comparison across different datasets and TTA methods We next provide an extended baseline comparison by evaluating the baselines from § B.4 on the TTA methods and datasets studied in § 4.1.

Fig. 9 displays the estimated test risk across datasets and TTA methods. As expected, the oracle, supervised lower bound on the test risk, $\hat{L}_t^a$ (- -), reliably flags risk violations without causing false alarms across all datasets and TTA methods. In contrast, the naive plug-in bound $\hat{L}_t^c$ (—) triggers a false alarm on the in-distribution ImageNet test stream for CoTTA, despite the test risk remaining below the alarm threshold. This is unsurprising, as $\hat{L}_t^c$ lacks formal guarantees on the false alarm rate. While it yields reasonable risk estimates for the other TTA methods on ImageNet ID, as well as on ImageNet-C severity 5 and FMoW-Time, it fails to detect risk violations on Yearbook across all TTA methods. Even though not originally proposed for TTA, we extend the unsupervised test risk lower bound by Amoukou et al. (2024), $\hat{L}_t^d$ to the TTA setting to enable comparison on this plot. We note that it behaves poorly with TTA methods. $\hat{L}_t^d$ (—), also triggers a false alarm on ImageNet ID for CoTTA. For other TTA methods, it is largely unresponsive resulting in a consistently loose lower bound on the estimated true test risk $\hat{\bar{R}}_t$ (—). This looseness leads to missed alarms on the severe shift of ImageNet-C severity 5 for 3 out of 5 TTA methods. Furthermore, we observe that the required assumption of their method (···) is violated in nearly every practical setting.

In contrast, as shown in § 4.1, our unsupervised test risk lower bound $\hat{L}_t^b$ (—) detects risk violations promptly (ImageNet-C severity 5, Yearbook), while remaining inactive when the risk threshold is not breached (ImageNet ID, FMoW-Time).

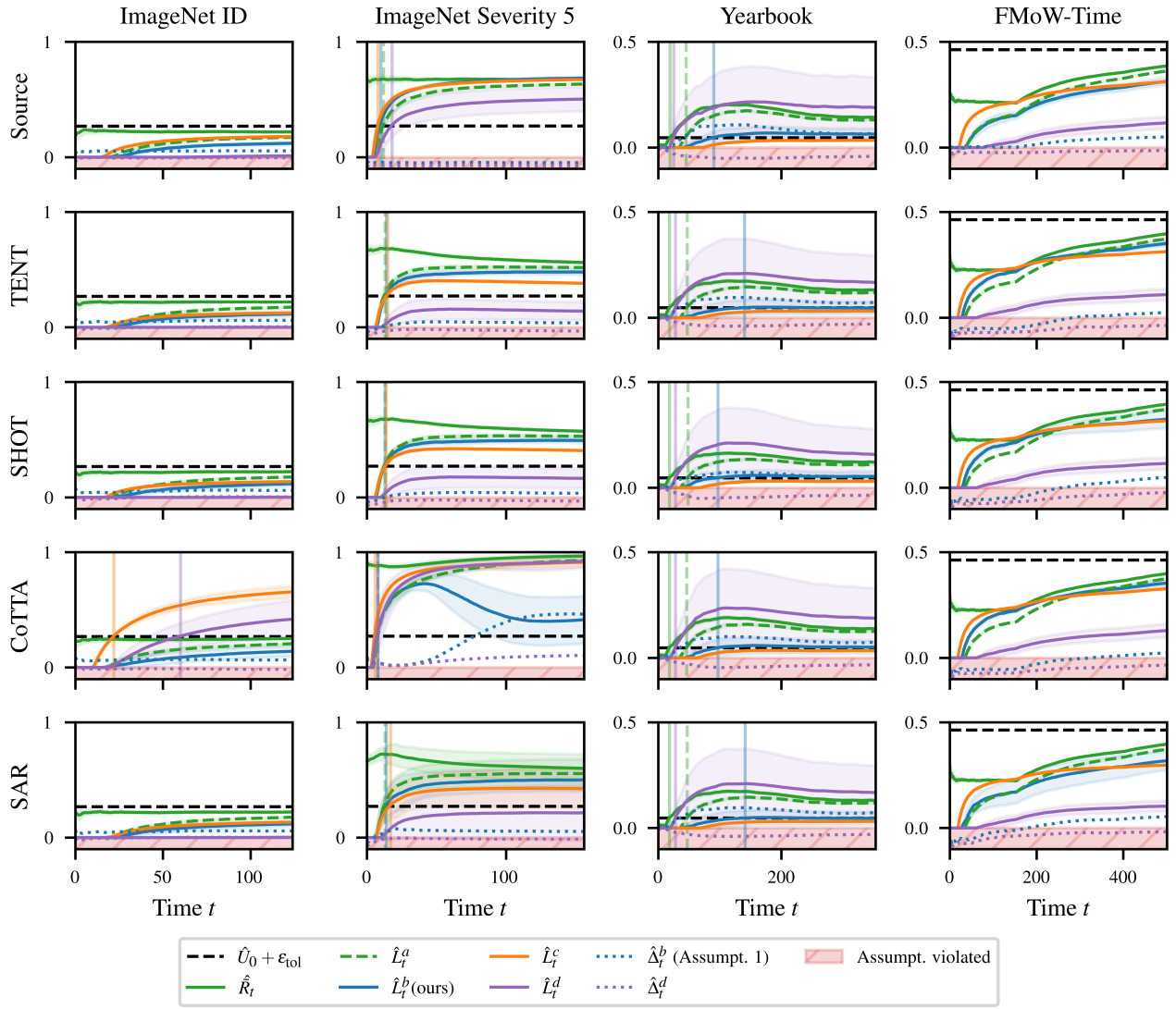

Figure 5: Estimated test risk for different baselines, datasets and TTA methods.

B.5. Results for SHOT

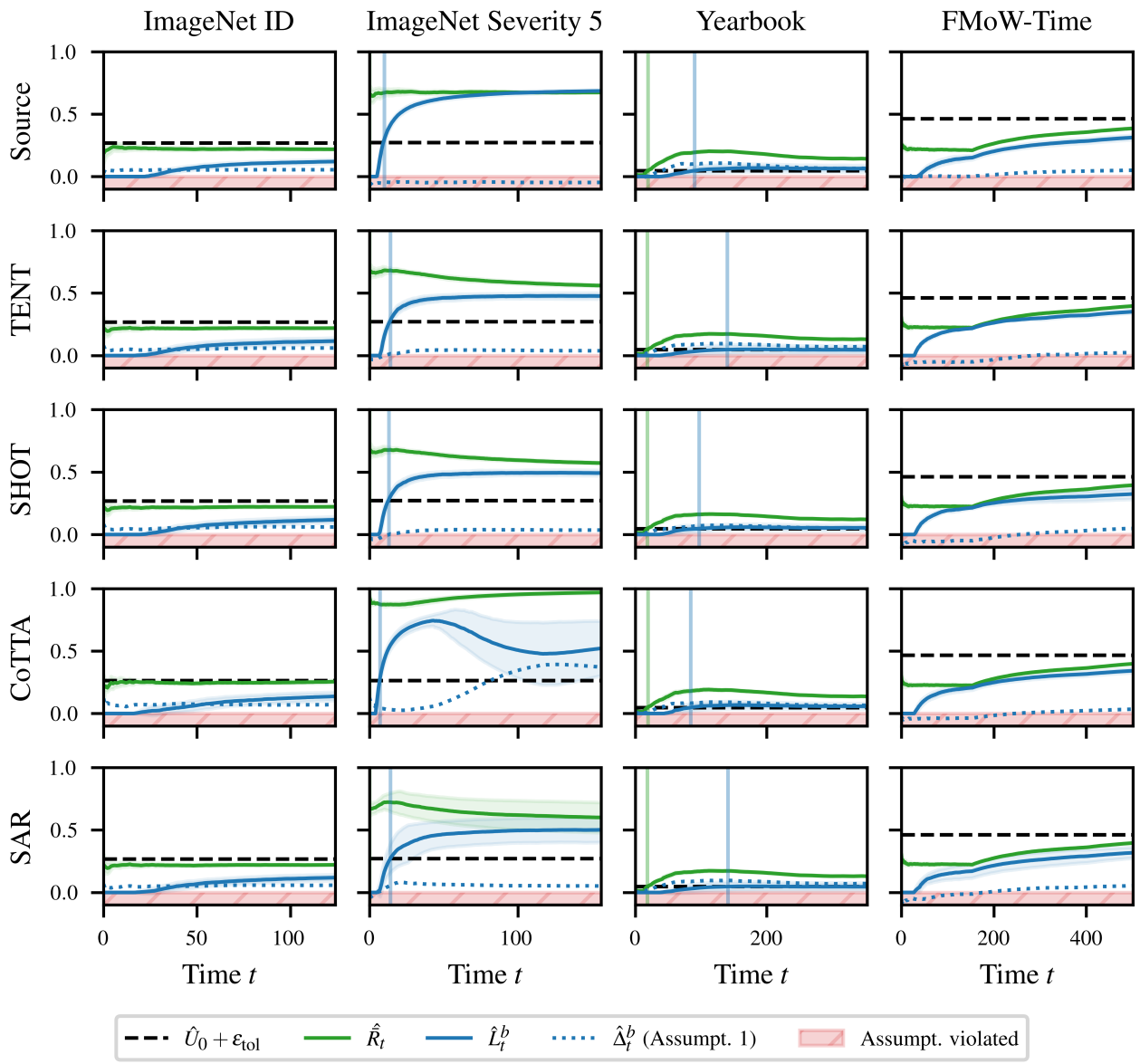

Figure 6: Estimated test risk for different datasets and TTA methods: We now also include results for SHOT to Fig. 6, which were omitted previously due to space constraints. The trend remains consistent, and risk violations are reliably detected for SHOT as well.

B.6. Alternative Loss Proxies

In § 4.1, we demonstrated that using model uncertainty as a loss proxy yields valid (according to Assumption 1) and, importantly, tight unsupervised lower bounds across a representative set of TTA methods and data shifts. Here, we supplement these results with a case where relying solely on model uncertainty proves insufficient for effective detection. Specifically, when monitoring "last-layer" TTA methods (Iwasawa and Matsuo, 2021; Schirmer et al., 2024)—which adapt only the classification head $W = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_C] \in \mathbb{R}^{H \times C}$—we observed that our unsupervised bound becomes overly loose, causing the alarm to fail under severe distribution shifts. We attribute this behavior to the normalization of each class prototype, i.e. $\boldsymbol{w}_c \,/\, ||\boldsymbol{w}_c||_2^2$, at every adaptation step. This normalization leads to (much) reduced variability in uncertainty across samples at the start of adaptation, making the separation of high and low losses harder. To overcome this, we propose using the distance to the closest prototype (Van Amersfoort et al., 2020; Ming et al., 2022), $g(\mathbf{x}, p) = \min_c ||f(\mathbf{x}) - \boldsymbol{w}_c||_2^2$, where f denotes the feature extractor of model p, as an alternative loss proxy. Unlike uncertainty, this measure is less affected by the normalization of class prototypes. In Fig. 7, we show that this yields tighter lower bounds for T3A (Iwasawa and Matsuo, 2021) and STAD (Schirmer et al., 2024)—both last-layer TTA methods—underscoring the importance of aligning the proxy choice with the specifics of the given TTA approach.

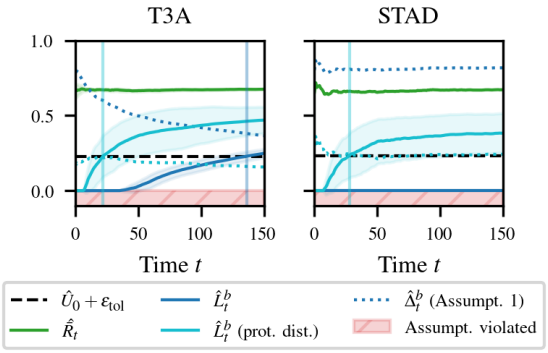

Figure 7: Comparison of loss proxies for last-layer TTA methods on ImageNet-C (GN) severity 5: Distance to class prototype is more effective than uncertainty for this TTA class.

We additionally compare our choice of using model uncertainty as a loss proxy (§ B.1) with the alternative of using the energy score (Liu et al., 2020), one of the most popular measures for detecting out-of-distribution (OOD) samples: $g(\mathbf{x}, p) = -\log \sum_{c=1}^{C} e^{m(\mathbf{x})_c}$, where $m(\mathbf{x}) \in \mathbb{R}^C$ is a vector of logits for model p.

We present results in Fig. 8. While using energy as a loss proxy also yields valid lower bounds (as indicated by $\hat{\Delta}_t^b$ being positive), the resulting bounds are looser compared to those obtained using uncertainty as a proxy across all considered TTA methods and distribution shifts. We attribute the underperformance of the energy score to its primary focus on distinguishing between in-distribution and OOD inputs. In contrast, a useful loss proxy (see Assumption 1) should be able to differentiate between correctly predicted samples (i.e., low loss) and incorrectly predicted ones (i.e., high loss).

Although these findings further support our choice of model uncertainty as a loss proxy, we believe that exploring alternative proxies that would lead to (even) tighter bounds remains an important direction for future work.

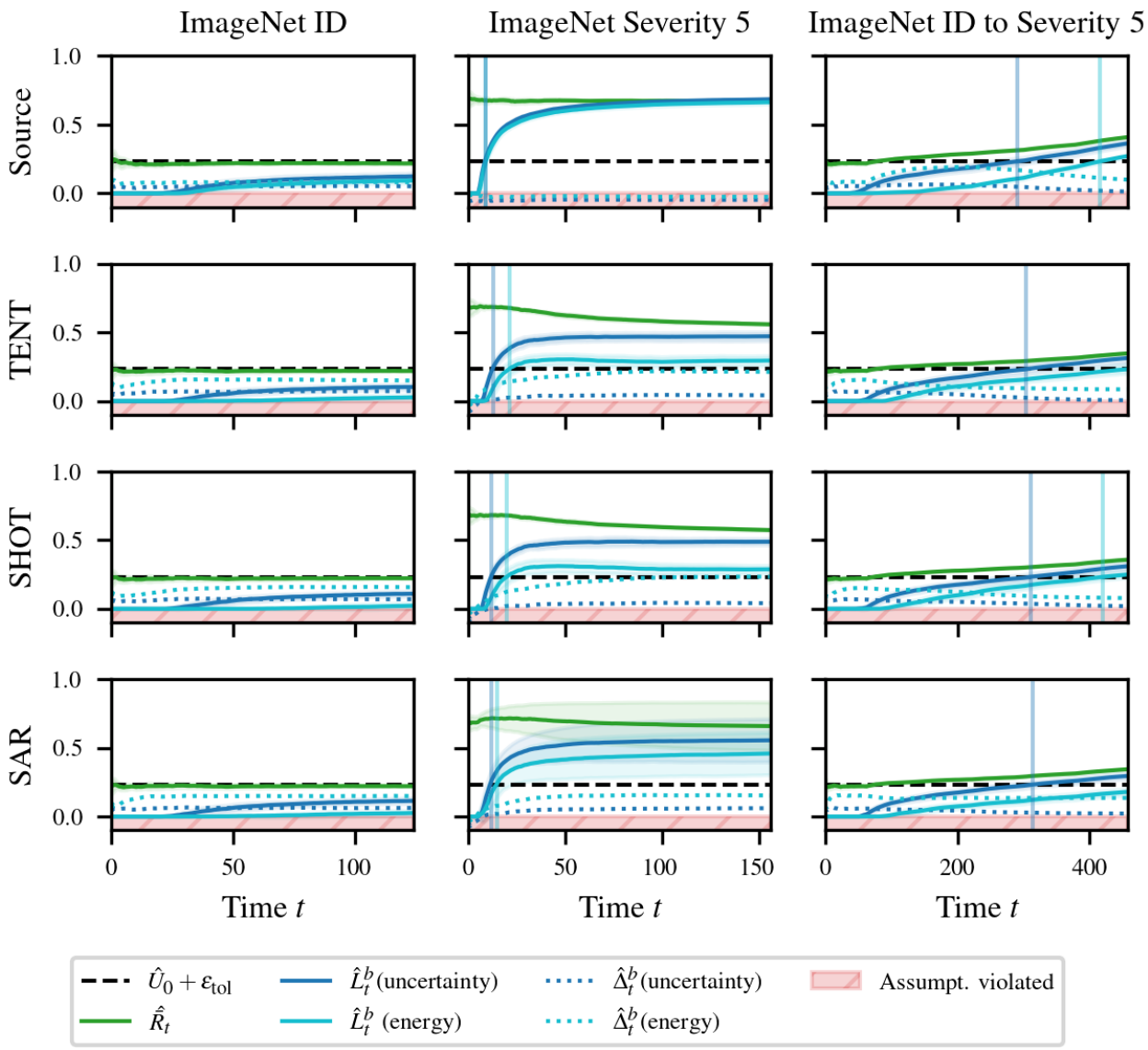

Figure 8: Estimated test risk for ImageNet test streams: We compare uncertainty and energy-score as loss proxies. Uncertainty yields consistently tighter lower bounds on the test risk than the energy score.

B.7. Ablation on Calibration Set Size

Our method requires a small labeled calibration set from the source distribution, which introduces additional computational overhead due to repeated evaluation of the adapted model on this set during adaptation. A small set of (labeled) source samples is commonly used for initializing TTA methods (Song et al., 2023; Lim et al., 2023; Wang et al., 2022; Choi et al., 2022; Adachi et al., 2023; Bar et al., 2024; Jung et al., 2023; Lee et al., 2023a; 2024a), and is indispensable for risk control (Podkopaev and Ramdas, 2022; Amoukou et al., 2024). We next investigate whether the calibration set size can be reduced. In addition to addressing the reliance on labeled data, a smaller calibration set also reduces the runtime overhead of the online calibration procedure.

Fig. 9 shows the estimated upper bound on the source risk $\hat{U}_0$ and our estimated lower bound on the test risk $\hat{L}_t^b$ for the default calibration set size of $N_{\text{cal}} = 1000$ (—) and a reduced size of $N_{\text{cal}} = 100$ (—). We find that reducing the calibration set to $N_{\text{cal}} = 100$ has minimal impact on both $\hat{U}_0$ and $\hat{L}_t^b$. Only in the setting with a gradually increasing distribution shift

(*right column*) do we observe a slight delay in risk detection compared to the default setup.

Future work may explore reducing computational overhead by performing calibration less frequently—evaluating only at fixed intervals or adaptively triggering calibration after an observed increase in risk, rather than at every time point

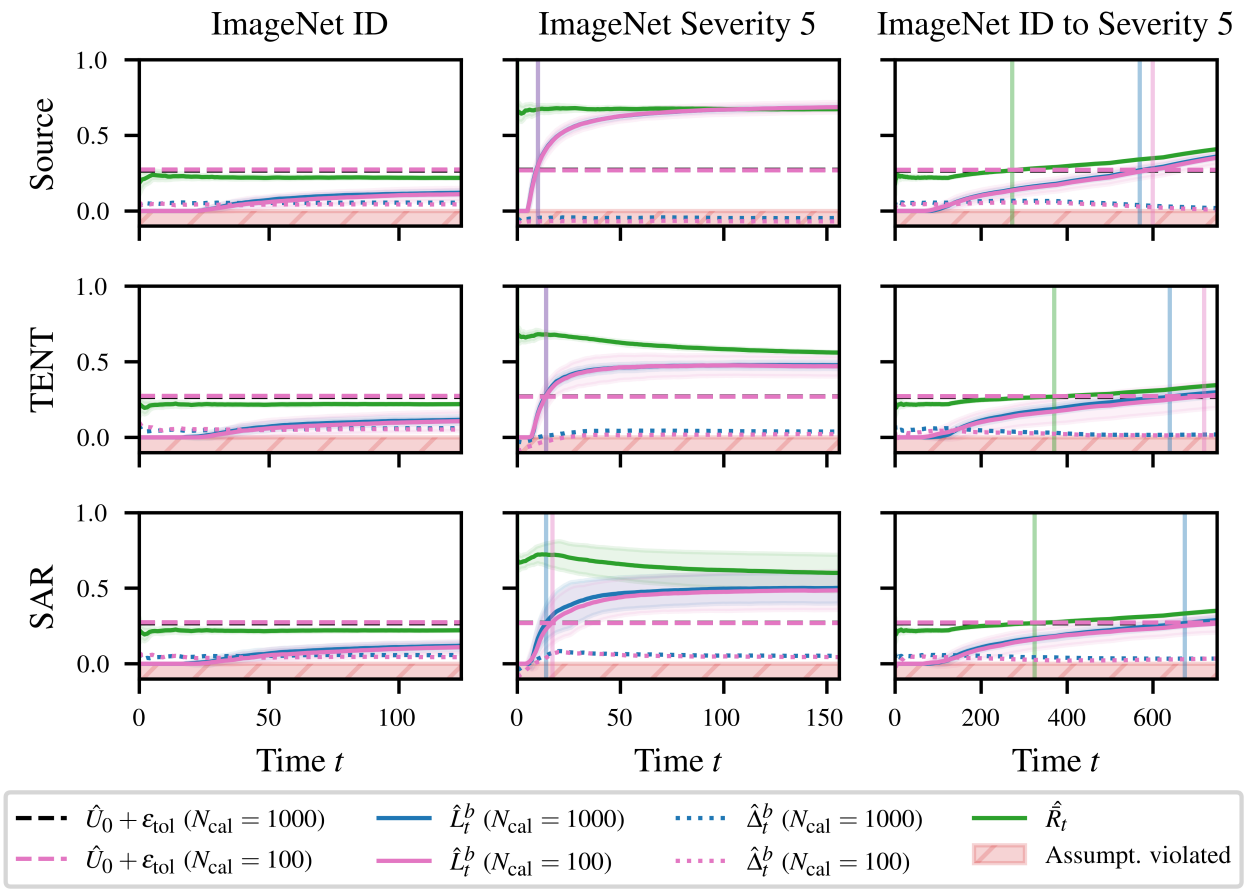

Figure 9: Estimated test risk for ImageNet test streams. We compare the default calibration set size of $N_{\text{cal}} = 1000$ to a smaller set with $N_{\text{cal}} = 100$. Reducing the calibration size leads to no or only small detection delays relative to the larger calibration set.

C. Theoretical Results

C.1. Choice of Confidence Sequences

For an introduction to confidence sequences, we refer the interested reader to Howard et al. (2021) and Appendix E of Podkopaev and Ramdas (2022). Throughout this paper, we use a Hoeffding confidence interval to estimate the upper bound on the source risk. Accordingly, the finite-sample penalty term is given by $w_0 = \sqrt{\log(1/\alpha_{\text{source}})/N_{\text{cal}}}$. For the lower bound on the test risk, we use the conjugate-mixture empirical Bernstein (CM-EB) confidence sequence proposed in Theorem 4 of Howard et al. (2021), chosen for its minimal assumptions. To obtain the finite-sample terms w_t for $t \geq 1$—which depend on α_{test} and the empirical variance of the observed sequence—we use the gamma-exponential mixture bound from Proposition 9 in Howard et al. (2021).

For the confidence sequence L_t^b used in our proposed unsupervised alarms Φ_t^b (Eq. 5) and Φ_t^τ (Eq. 8), we apply a CM-EB lower confidence sequence for $\frac{1}{t}\sum_{k=1}^t \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k)$ with α_{test_1}, and an upper Hoeffding confidence interval for $\mathbb{P}_{P_0}(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 \leq \tau)$ with α_{test_2}, such that $\alpha_{\text{test}_1} + \alpha_{\text{test}_2} = \alpha_{\text{test}}$.

C.2. Unsupervised Lower Bound Derivation (Propositon 1)

Proposition 1. *Assume a non-negative, bounded loss $\ell \in [0, M], M > 0$. Further, assume that for a sequence of losses $\mathbf{z}_{0:t}$, a sequence of loss proxies $\mathbf{u}_{0:t}$ together with thresholds $\lambda_0, \ldots, \lambda_t \in \mathbb{R}, \tau \in (0, M)$ satisfying Assumption 1 are available. Then the running test risk can be lower bounded as*

$$\bar{R}_t(p_{1:t}) \geq \underbrace{\tau\left(\frac{1}{t}\sum_{k=1}^t \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k) - \mathbb{P}_{P_0}(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 \leq \tau)\right)}_{:=B_t}, \forall t \geq 1.$$

Proof. The proof technique is inspired by the derivation presented in Amoukou et al. (2024); see Eqs. (12)–(15) in their paper. To derive a lower bound on the true running test risk, we first apply Markov's inequality and then invoke Assumption 1:

$$\bar{R}_t(p_{1:t}) = \frac{1}{t}\sum_{k=1}^t \mathbb{E}_{P_k}[\mathbf{z}_k] \overset{\text{Markov's}}{\geq} \frac{1}{t}\sum_{k=1}^t \mathbb{P}_{P_k}(\mathbf{z}_k > \tau)\cdot\tau =$$

$$\tau\left(\frac{1}{t}\sum_{k=1}^t \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k, \mathbf{z}_k > \tau) + \mathbb{P}_{P_k}(\mathbf{u}_k \leq \lambda_k, \mathbf{z}_k > \tau)\right) =$$

$$\tau\left(\frac{1}{t}\sum_{k=1}^t \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k) - \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k, \mathbf{z}_k \leq \tau) + \mathbb{P}_{P_k}(\mathbf{u}_k \leq \lambda_k, \mathbf{z}_k > \tau)\right) \overset{\text{Ass.1}}{\geq}$$

$$\tau\left(\frac{1}{t}\sum_{k=1}^t \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k) - \mathbb{P}_{P_0}(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 \leq \tau)\right)$$

□

If the risk definition includes conditioning on $\mathbf{x}_{1:k-1}$, i.e., $R_k(p_k) := \mathbb{E}_{P_k}[\mathbf{z}_k|\mathbf{x}_{1:k-1}]$, the proof proceeds analogously, with the only change being the use of conditional Markov's inequality. In this case, the resulting bound holds *almost surely*.

C.3. PFA Control Guarantee

Proposition 2. *The unsupervised alarm Φ_t^b (Eq. 5) for the TTA sequential test (Eq. 2) satisfies a probability of false alarm (PFA) control guarantee:*

$$\mathbb{P}_{H_0}(\exists t \geq 1, \Phi_t^b = 1) \leq \alpha_{test} + \alpha_{source}\,.$$

Proof. The proof closely follows the PFA proof for the supervised alarm from Podkopaev and Ramdas (2022), see Appendix

D there. To show the PFA guarantee we proceed as:[3]

$$\mathbb{P}_{H_0}(\exists t \geq 1, \Phi_t^b = 1) = \mathbb{P}_{H_0}(\exists t \geq 1, L_t^b - U > \epsilon_{\text{tol}}) = \mathbb{P}_{H_0}(\exists t \geq 1, (L_t^b - \bar{R}_t) - (U - R_0) > \epsilon_{\text{tol}} - (\bar{R}_t - R_0)) \leq \mathbb{P}_{H_0}(\exists t \geq 1, (L_t^b - \bar{R}_t) - (U - R_0) > 0)\,,$$

where the inequality follows from the fact that under H_0 (Eq. 2), we have that $\epsilon_{\text{tol}} \geq \bar{R}_t - R_0$. Since $\exists t \geq 1, (L_t^b - \bar{R}_t) - (U - R_0) > 0$ implies that either $\exists t \geq 1, L_t^b - \bar{R}_t > 0$ or $U - R_0 < 0$, we can use union bound to continue as:

$$\mathbb{P}_{H_0}(\exists t \geq 1, (L_t^b - \bar{R}_t) - (U - R_0) > 0) \leq \mathbb{P}(\exists t \geq 1, L_t^b - \bar{R}_t > 0) + \mathbb{P}(U - R_0 < 0) \leq \alpha_{\text{test}} + \alpha_{\text{source}}\,,$$

where the last inequality follows from the fact that L_t^b is a lower bound confidence sequence for the lower bound B_t, i.e., $\mathbb{P}(B_t \geq L_t^b, \forall t \geq 1) \geq 1 - \alpha_{\text{test}}$, together with Proposition 1, which ensures $\bar{R}_t \geq B_t, \forall t \geq 1$, and the fact that U is the upper bound of the confidence interval for R_0.

□

C.4. Tighter Bound for 0-1 Loss

Corollary 1. *For a* 0-1 *loss function, assume that for a sequence of losses* $\mathbf{z}_{0:t}$*, a sequence of loss proxies* $\mathbf{u}_{0:t}$ *together with thresholds* $\lambda_0, \ldots, \lambda_t \in \mathbb{R}$ *satisfying Assumption 1 are available. Then the running test risk can be lower bounded as*

$$\bar{R}_t(p_{1:t}) \geq \frac{1}{t}\sum_{k=1}^{t} \mathbb{P}_{P_k}(\mathbf{u}_k > \lambda_k) - \mathbb{P}_{P_0}(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 = 0)\,, \forall t \geq 1.$$

Proof. This (tighter) bound follows from the fact that for 0-1 loss, Markov's inequality is unnecessary due to the binary nature of the loss:

$$\bar{R}_t(p_{1:t}) = \frac{1}{t}\sum_{k=1}^{t} \mathbb{E}_{P_k}[\mathbf{z}_k] \overset{\text{(0-1)}}{=} \frac{1}{t}\sum_{k=1}^{t} \mathbb{P}_{P_k}[\mathbf{z}_k = 1]\,.$$

The remainder of the proof then proceeds identically to that of Proposition 1. □

Observe how the lower bound for 0-1 loss is the same as the lower bound for a general (bounded, non-negative) loss in Proposition 1 up to the loss threshold τ. Additionally, we leave out the loss threshold τ from Assumption 1, i.e., we assume that the proxy sequence $\mathbf{u}_{0:t}$ is such that:

$$\frac{1}{t}\sum_{k=1}^{t} \mathbb{P}_{P_k}\big(\mathbf{u}_k > \lambda_k, \mathbf{z}_k = 0\big) \leq \mathbb{P}_{P_0}\big(\mathbf{u}_0 > \lambda_0, \mathbf{z}_0 = 0\big) + \frac{1}{t}\sum_{k=1}^{t} \mathbb{P}_{P_k}\big(\mathbf{u}_k \leq \lambda_k, \mathbf{z}_k = 1\big).$$

C.5. PFA for "Probability of High Loss" Test

Proposition 3. *The unsupervised alarm* Φ_t^τ *(Eq. 8) for the 'probability of high loss' TTA sequential test (Eq. 9) satisfies a PFA control guarantee:*

$$\mathbb{P}_{H_0}(\exists t \geq 1, \Phi_t^\tau = 1) \leq \alpha_{\textit{test}} + \alpha_{\textit{source}}\,.$$

Proof. To simplify notation, denote with $\bar{R}_t^{\mathbb{P}} := \frac{1}{t}\sum_{k=1}^{t} \mathbb{P}_{P_k}(\mathbf{z}_k > \tau)$ and $R_0^{\mathbb{P}} := \mathbb{P}_{P_0}(\mathbf{z}_0 > \tau)$. From the proof of Proposition 1, it follows that $\bar{R}_t^{\mathbb{P}} \geq \frac{1}{\tau} B_t$, which, combined with the fact that L_t^b is a lower bound confidence sequence for

[3] To simplify notation, we omit all arguments of the relevant risks and confidence sequences in the proof (e.g., we abbreviate $U(\mathbf{z}_0)$ as U).

B_t, implies that $\mathbb{P}(\bar{R}_t^{\mathbb{P}} \geq \frac{1}{\tau} L_t^b, \forall t \geq 1) \geq 1 - \alpha_{\text{test}}$. Similarly, since U^b is an upper bound of the confidence interval for $\tau R_0^{\mathbb{P}}$, it follows that $\mathbb{P}(\frac{1}{\tau} U^b \geq R_0^{\mathbb{P}}) \geq 1 - \alpha_{\text{source}}$. The rest of the proof is then indentical to the proof of Proposition 2:

$$\mathbb{P}_{H_0}(\exists t \geq 1, \Phi_t^\tau = 1) = \mathbb{P}_{H_0}\big(\exists t \geq 1, \frac{1}{\tau} L_t^b - \frac{1}{\tau} U^b > \tilde{\epsilon}_{\text{tol}}\big) =$$

$$\mathbb{P}_{H_0}\big(\exists t \geq 1, (\frac{1}{\tau} L_t^b - \bar{R}_t^{\mathbb{P}}) - (\frac{1}{\tau} U - R_0^{\mathbb{P}}) > \tilde{\epsilon}_{\text{tol}} - (\bar{R}_t^{\mathbb{P}} - R_0^{\mathbb{P}})\big) \overset{H_0}{\leq}$$

$$\mathbb{P}_{H_0}\big(\exists t \geq 1, (\frac{1}{\tau} L_t^b - \bar{R}_t^{\mathbb{P}}) - (\frac{1}{\tau} U - R_0^{\mathbb{P}}) > 0\big) \leq$$

$$\mathbb{P}\big(\exists t \geq 1, \frac{1}{\tau} L_t^b - \bar{R}_t^{\mathbb{P}} > 0\big) + \mathbb{P}\big(\frac{1}{\tau} U - R_0^{\mathbb{P}} < 0\big) \leq \alpha_{\text{test}} + \alpha_{\text{source}} \,.$$

□

D. Algorithms

Algorithm 1: TTA with Risk Monitoring

Input : Calibration data $\mathcal{D}_{\text{cal}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_{\text{cal}}}$ with $(\boldsymbol{x}_i, y_i) \sim P_0$, test data $\mathcal{D}_{\boldsymbol{x}}^k = \{\boldsymbol{x}_i\}_{i=1}^{N_k}$ with $\boldsymbol{x}_i \sim P_k$, loss function ℓ, proxy function g, source model p_0, tolerance level ϵ_{tol}, significance levels α_{source} and α_{test}, TTA method $h : (p_{k-1} \times \mathcal{D}_{\boldsymbol{x}}^k) \mapsto p_k$

Compute source losses $z_{0,i} = \ell(p_0(\boldsymbol{x}_i), y_i)$
Compute source loss proxies $u_{0,i} = g(\boldsymbol{x}_i, p_0)$
Find source thresholds $\hat{\lambda}_0, \hat{\tau} := \arg\max_{\lambda,\tau} \text{F1}(\lambda, \tau; \{(z_{0,i}, u_{0,i})\}_{i=1}^{N_{\text{cal}}})$
Compute upper bound $\hat{U}$ using $\{z_{0,i}\}_{i=1}^{N_{\text{cal}}}$ and α_{source}
for $k \geq 1$ **do**
    Perform TTA update $p_k = h(p_{k-1}, \mathcal{D}_{\boldsymbol{x}}^k)$
    Compute losses of model p_k on $\mathcal{D}_{\text{cal}}$: $z_{0,i}^{(k)} = \ell(p_k(\boldsymbol{x}_i), y_i)$
    Compute loss proxies of model p_k on $\mathcal{D}_{\text{cal}}$: $u_{0,i}^{(k)} = g(\boldsymbol{x}_i, p_k)$
    Update proxy threshold $\hat{\lambda}_k := \arg\max_{\lambda} \text{F1}(\lambda, \hat{\tau}; \{(z_{0,i}^{(k)}, u_{0,i}^{(k)})\}_{i=1}^{N_{\text{cal}}})$
    Compute loss proxies of model p_k on $\mathcal{D}_{\boldsymbol{x}}^k$: $u_{k,i} = g(\boldsymbol{x}_i, p_k)$
    Compute lower bound $\hat{L}_k^b$ using $\{u_{1,i}\}_{i=1}^{N_1}, \ldots, \{u_{k,i}\}_{i=1}^{N_k}, \{z_{0,i}\}_{i=1}^{N_{\text{cal}}}, \hat{\lambda}_{0:k}, \hat{\tau}, \alpha_{\text{test}}$
    Compute alarm $\hat{\Phi}_k^b = \mathbb{1}\left[\hat{L}_k^b > \hat{U} + \epsilon_{\text{tol}}\right]$
    if $\hat{\Phi}_k^b = 1$ **then**
        Terminate TTA
        break
    else
        Predict using p_k on $\mathcal{D}_{\boldsymbol{x}}^k$: $\hat{y}_i = \arg\max_c p_k(\boldsymbol{x}_i)_c$
        continue

Algorithm 2: Online Threshold Calibration

Input : Calibration data $\mathcal{D}_{\text{cal}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_{\text{cal}}}$ with $(\boldsymbol{x}_i, y_i) \sim P_0$, loss function ℓ, proxy function g, source model p_0, TTA models $p_1, \ldots, p_t$

Output : loss threshold $\hat{\tau}$, proxy thresholds $\hat{\lambda}_{0:t}$

Compute source losses $z_{0,i} = \ell(p_0(\boldsymbol{x}_i), y_i)$
Compute source loss proxies $u_{0,i} = g(\boldsymbol{x}_i, p_0)$
Find source thresholds $\hat{\lambda}_0, \hat{\tau} := \arg\max_{\lambda,\tau} \text{F1}(\lambda, \tau; \{(z_{0,i}, u_{0,i})\}_{i=1}^{N_{\text{cal}}})$
for $k = 1 \to t$ **do**
    Compute losses of model p_k on $\mathcal{D}_{\text{cal}}$: $z_{0,i}^{(k)} = \ell(p_k(\boldsymbol{x}_i), y_i)$
    Compute loss proxies of model p_k on $\mathcal{D}_{\text{cal}}$: $u_{0,i}^{(k)} = g(\boldsymbol{x}_i, p_k)$
    Update proxy threshold $\hat{\lambda}_k := \arg\max_{\lambda} \text{F1}(\lambda, \hat{\tau}; \{(z_{0,i}^{(k)}, u_{0,i}^{(k)})\}_{i=1}^{N_{\text{cal}}})$
return $\hat{\tau}, \hat{\lambda}_{0:t}$

E. Experimental Details

E.1. Datasets

- **ImageNet-C** (Hendrycks and Dietterich, 2019): This dataset applies 15 types of algorithmic corruptions (e.g., Gaussian noise, blur, weather effects, digital distortions) at five severity levels to the original ImageNet (Deng et al., 2009) validation set. The dataset preserves the original 1,000-class classification task, using the same labels and image resolutions. In our setup, we focus on Gaussian noise corruption.

- **Yearbook** (Ginosar et al., 2015): This dataset contains portraits of American high school students taken over eight

decades, capturing changes in visual appearance due to evolving beauty standards, cultural norms, and demographics. We follow the Wild-Time preprocessing and evaluation protocol (Yao et al., 2022), resulting in 33,431 grayscale images (32×32 pixels) labeled with binary gender. Images from 1930–1969 are used for training, and those from 1970–2013 for testing.

- **FMoW-Time**: The Functional Map of the World (FMoW) dataset (Koh et al., 2021) consists of 224×224 RGB satellite images categorized into 62 land-use classes. Distribution shift arises from technological and economic changes that alter land usage over time. FMoW-Time (Yao et al., 2022) is a temporal split of FMoW-WILDS (Koh et al., 2021; Christie et al., 2018), dividing 141,696 images into a training period (2002–2014) and a testing period (2015–2017).

E.2. Models

For ImageNet-C, we use the pretrained ViT-Base model (Dosovitskiy et al., 2021) from the Timm library (Wightman, 2019), focusing on Gaussian noise (GN) corruptions. For Yearbook and FMoW, we follow the protocol of Yao et al. (2022), using their provided model weights: a small CNN for Yearbook and DenseNet121 (Huang et al., 2017) for FMoW.

E.3. TTA Methods

We evaluate our monitoring tool across several TTA methods, which differ in the set of adapted parameters (e.g., normalization layers, full model, classification head) and in their objective functions (e.g., entropy minimization, information maximization, log-likelihood maximization):

- **TENT**(Wang et al., 2021) updates normalization layers by minimizing test entropy.
- **SHOT** (Liang et al., 2020) adapts normalization layers using information maximization and self-supervised pseudo-labeling to align target representations with a frozen source classifier.
- **SAR** (Niu et al., 2023) updates normalization layers via an entropy minimization objective. It filters out high-entropy samples and guides adaptation toward flatter minima.
- **CoTTA** (Wang et al., 2022) updates all model parameters using a student-teacher approach on augmentation averaged predictions. It also employs stochastic weight restoration to mitigate forgetting.
- **T3A** (Iwasawa and Matsuo, 2021) adjusts only the final linear classifier by computing class-wise pseudo-prototypes from confident, normalized representations.
- **STAD** (Schirmer et al., 2024) updates only the last linear layer by tracking the evolution of feature representations with a probabilistic state-space model.

E.4. Implementation Details

All experiments are performed on NVIDIA RTX 6000 Ada with 48GB memory. We plot the mean and standard deviations over 20 runs. The variability across runs stems from calibration set sampling and test sample shuffling with different random seeds. For Fig. 7 and Fig. 9, we use 10 random seeds.

For each TTA method, we use the default hyperparameters proposed in the respective paper. We use a test batch size of 32 for ImageNet and 64 for Yearbook and FMoW-Time.

We use the `confseq` package (Howard et al., 2021–) by (Howard et al., 2021) to compute the conjugate-mixture empirical Bernstein confidence lower bound on the target risk. This confidence sequence framework supports tuning for an intrinsic time t_{opt}, which we set by default to the first 25% of the sequence length for all experiments. To implement the baseline $\hat{L}_t^d$ from Amoukou et al. (2024), we use the same loss proxy—uncertainty—as in our method.

F. Related Work

TTA (Xiao and Snoek, 2024; Liang et al., 2024) aims to improve model performance under distribution shift by updating the model using unlabeled test data. Classic approaches include recomputing normalization statistics (Schneider et al., 2020; Nado et al., 2020), optimizing unsupervised objectives such as test entropy (Wang et al., 2021; Niu et al., 2022; 2023), energy (Yuan et al., 2024), or pseudo-labels (Lee et al., 2013; Liang et al., 2020), or adapting the last layer (Iwasawa and Matsuo, 2021; Boudiaf et al., 2022; Schirmer et al., 2024). However, recent work has identified scenarios where TTA methods are ineffective (Zhao et al., 2023; Schirmer et al., 2024), and even harmful, degrading performance below that of the unadapted source model (Boudiaf et al., 2022; Gong et al., 2022; Yuan et al., 2023; Niu et al., 2023; Döbler et al., 2023; Press et al., 2024a; Wu et al., 2023; Park et al., 2024). While some studies propose heuristic indicators of TTA failure, such as high gradient norms (Niu et al., 2023), or estimate test accuracy directly (Lee et al., 2024b; Press et al., 2024a), there remains no principled framework for detecting risk violations of TTA methods with theoretical guarantees.

TTA robustness Recent work has identified several scenarios where TTA methods tend to degrade. These include adaptation under non-stationary test distributions (Wang et al., 2022; Hoang et al., 2024; Yuan et al., 2023; Lim et al., 2023; Döbler et al., 2023; Marsden et al., 2024; Su et al., 2024; Du et al., 2024; Schirmer et al., 2024), label shift (Gong et al., 2022; Niu et al., 2022; Boudiaf et al., 2022; Yuan et al., 2023), mixed domains within a test batch (Niu et al., 2023; Lim et al., 2023; Marsden et al., 2024; Du et al., 2024), small test batch sizes (Niu et al., 2023; Lim et al., 2023; Marsden et al., 2024; Döbler et al., 2023; Schirmer et al., 2024), and adaptation in the presence of malicious samples (Park et al., 2024; Wu et al., 2023). Most such work on TTA robustness has focused on proposing more robust TTA methods and developing evaluation benchmarks (Gong et al., 2022; Marsden et al., 2024; Du et al., 2024; Yuan et al., 2023; Su et al., 2024). Liu et al. (2021) analyze failure cases of the related test-time training paradigm, which requires a self-supervised objective during training. They derive an upper bound on test risk dependent on the effectiveness of the self-supervised loss. In contrast, our approach does not require any modification to the training procedure and provides guarantees that hold regardless of the model's original training objective. Also related to our work is research on TTA model collapse—where models degenerate to trivial solutions during adaptation (Press et al., 2024a; Niu et al., 2023; Lee et al., 2024b; Press et al., 2024b)—and efforts to identify optimal reset mechanisms that revert the model back to its source parameters during deployment (Niu et al., 2022; Lee et al., 2024b; Press et al., 2024b). In contrast, we propose a general-purpose monitoring tool that provides statistical guarantees on risk control for arbitrary TTA methods. Rather than focusing on a specific mitigation strategy, our tool can inform a range of interventions—such as resetting the model to its source for continued adaptation or taking it offline entirely for retraining (Hoffman et al., 2024).

Risk monitoring via sequential testing has been proposed by Podkopaev and Ramdas (2022), though in a setting where test stream labels are available and the model remains static. Most relevant to our work is that of Amoukou et al. (2024), who extend (Podkopaev and Ramdas, 2022) to the test scenario without labels. We further build upon their framework by: (i) incorporating model adaptation (§ 3.1); (ii) deriving an unsupervised bound on the expected loss, rather than only a bound on the probability of high loss (§ 3.2); (iii) using model uncertainty as a proxy for loss instead of a separate error estimator (§ B.1); (iv) proposing a simpler calibration method (§ B.2) and showing it's effectiveness for 0-1 loss (§ B.4); and (v) providing theoretical insights into why effective monitoring of continuous losses necessitates a change in the tested hypothesis (§ B.3). Also related is work by Bar et al. (2024), where a sequential test for TTA methods based on betting martingales (Ramdas et al., 2023) is proposed. However, their test is designed to detect changes in predictive entropy, which may or may not lead to a degradation in performance—unlike our method, which directly tests for performance drops.

Error and accuracy estimation aims to assess model performance on unlabeled test data, which is often subject to distribution shift (Sanyal et al., 2024; Miller et al., 2021). This is typically achieved via model uncertainty (Hendrycks and Gimpel, 2017; Corbière et al., 2019; Garg et al., 2022; Lu et al., 2023; Guillory et al., 2021) or model disagreement (Jiang et al., 2022; Baek et al., 2022; Kirsch and Gal, 2022; Rosenfeld and Garg, 2023; Lee et al., 2024b; Nakkiran and Bansal, 2020; Lee et al., 2023b). Uncertainty-based methods exploit the predictive distribution of the model— for example through the maximum class probability (Hendrycks and Gimpel, 2017) or the true class probability (Corbière et al., 2019)—and learn a threshold to distinguish correctly from incorrectly predicted samples (Garg et al., 2022; Lu et al., 2023; Guillory et al., 2021). Disagreement-based error prediction methods leverage the theoretical equivalence between model disagreement and test error under calibration (Jiang et al., 2022; Kirsch and Gal, 2022). However, these methods often require training multiple models—sometimes even from different architectures (Baek et al., 2022; Jiang et al., 2022). Closest to our work (Lee et al., 2024b; Press et al., 2024a; Kim et al., 2024), estimate the accuracy of TTA methods based on disagreement.

Notably, by exploiting theoretical results from (Jiang et al., 2022), Lee et al. (2024b) proposes an accuracy estimation method based on dropout disagreement. They differ from our work by (i) providing an estimator of test risk directly while we are interested in signaling a significant increase in test risk compared to the source risk; as such (ii) their method does not come with guarantees on the false alarm rate; and (iii) they require calibration (their Definition 3.3) to preserve theoretical validity of their risk estimator while we rely on separability of high and low error samples (Assumption 1).

G. Limitations and Future Work

While we have shown that our unsupervised alarm has detection delays not too much larger compared to its supervised counterpart (Podkopaev and Ramdas, 2022) (see Fig. 4), the observed delays might still be too big for applications where detecting late is (significantly) more costly compared to raising false alarms. For such settings, it would be worth weakening the requirement on the probability of false alarm control under H_0 in order to gain more power under H_1. Perhaps this could be done by aiming for a weaker average run length control as is commonly done in the literature on change-point detection using confidence sequences (Shekhar and Ramdas, 2023a;b). Moreover, although our proposed lower bound from Proposition 1 can be computed without access to test labels, verifying Assumption 1 for a given loss proxy still requires a labeled test stream. While we found empirically that this assumption holds in nearly all evaluated cases (Fig. 6), developing unsupervised diagnostics to flag potential violations of the assumption remains an important direction for future work.

H. Impact Statement

This work introduces a statistically grounded framework for detecting risk violations during TTA, a key challenge for deploying machine learning models in dynamic, real-world environments. By enabling risk monitoring without access to labels, our approach promotes safer and more trustworthy use of TTA methods—particularly in high-stakes domains such as healthcare, autonomous systems, and finance, where undetected model failure can have serious consequences. Our method complements existing adaptation techniques by offering a safeguard against silent performance degradation and model collapse, helping practitioners determine when adaptation is no longer effective. In doing so, it supports more responsible and robust deployment of adaptive models. While the framework provides high-probability guarantees, misuse or overreliance could lead to overconfidence in model reliability. We therefore emphasize the importance of understanding its assumptions and limitations. Overall, this work contributes to the safe deployment of adaptive machine learning models under distribution shift.