FORTA: FEEDBACK-GUIDED PREDICTIVE DISTRIBUTION CORRECTION FOR HETEROGENEOUS TEST-TIME ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time adaptation is difficult because the appropriate correction can change as target conditions evolve. Methods built around a particular adaptation mechanism restrict how the predictive distribution can be corrected. This work studies test-time adaptation with sparse delayed feedback: predictions are made before labels are available, and only a small fraction of labels are released later. Adaptation is formulated as learning and allocating a time-varying set of predictive distribution corrections from a shared frozen representation. FORTA maintains complementary source-anchored corrections. Every supervised member continues learning from the same released feedback regardless of its current weight, while saved pre-feedback predictions determine future influence. This separates how the correction set evolves from which corrections shape the next prediction. The analysis characterizes an information boundary for label-free conditional adaptation and, in a source-anchored construction, establishes an allocation advantage over separately tuned anchors and fixed mixtures of optimized candidates. Across seven image and signal benchmarks, FORTA improves macro-F1 over the source predictor. We verify how feedback availability and regime duration affect adaptation, and when adaptive allocation improves predictive performance, through synthetic and matched-portfolio experiments.

1 INTRODUCTION

A source classifier defines a predictive conditional distribution over labels. Distribution shift changes the conditions under which this distribution is used. Measurement shift can distort the representation of an input, while conditional change can alter the class probabilities required for the same representation (Quinonero-Candela et al., 2009; Gama et al., 2014). These failures do not call for one universal parameter update. They call for different corrections of the source predictive distribution.

Most TTA methods encode one form of correction, such as entropy minimization, normalization, or restoration (Wang et al., 2021; Niu et al., 2022; Wang et al., 2022; 2023). Recent expert-based methods broaden the available updates for heterogeneous streams (Vray et al., 2025; Fan et al., 2026). A remaining problem is deciding which correction improves target predictions at a given point in the stream. Input statistics provide evidence about changes in $P_t(X)$, but they do not identify unrestricted changes in $P_t(Y \mid X)$ (Ben-David et al., 2010). A correction that helps one target condition can therefore become ineffective or harmful under another.

We study this problem with sparse delayed feedback. The model predicts before query selection, and labels for only a small fraction of examples arrive later (Joulani et al., 2013; Csaba et al., 2024). A released label cannot change the prediction it evaluates. It instead supplies evidence about which predictive correction was useful when that example appeared. This timing makes saved pre-feedback outputs essential. Coverage also matters because feedback must sample the part of the input space where the labeling rule has changed.

FORTA treats a time-varying set of predictive distribution corrections as the common interface between adaptation mechanisms. Starting from a frozen representation and source classifier, each mechanism updates one source-anchored conditional by correcting source logits or source features. Released labels score saved candidate distributions, update their allocation, and train every supervised

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

correction. Learning a correction is independent of its current mixture weight. A member with little current influence therefore continues to learn and can regain influence when target conditions change. The unchanged source remains in the set.

FORTA's central novelty is learning a time-varying set of predictive distribution corrections and allocating its members from the same sparse delayed feedback. The set changes as heterogeneous mechanisms update the source conditional. Its allocation changes as saved pre-feedback predictions are scored. This separates two decisions that a single correction rule couples: how to revise the predictive distribution, and which revision should shape the next prediction. The analysis then addresses both parts of this design. A two-world lower bound quantifies the error before feedback covers a changed conditional. A source-anchored construction proves that feedback-guided allocation can outperform per-condition anchor tuning and every fixed pool of the same learned candidates, while accounting for candidate learning, routing memory, feedback delay, and segment duration. Experiments on seven image and signal benchmarks evaluate the complete correction framework. Controlled switching and matched portfolios isolate allocation from candidate learning, and synthetic experiments verify the predicted effects of feedback coverage and segment duration.

2 Related Work

Mechanisms for heterogeneous adaptation. Entropy, normalization, restoration, and target structure provide different TTA update rules (Wang et al., 2021; Niu et al., 2022; 2023; Wang et al., 2022; Yuan et al., 2023; Boudiaf et al., 2022; Wang et al., 2023). NOTE and SoTTA address temporal correlation and noisy streams (Gong et al., 2022; 2023). FreDA forms locally homogeneous clusters using frequency information (Su et al., 2024). ReservoirTTA routes inputs to domain-specialized models using style-based clustering (Vray et al., 2025). MoETTA diversifies entropy-based updates through decoupled experts (Fan et al., 2026). FORTA instead maintains a time-varying output-level correction set and allocates its members using delayed feedback.

Sparse and delayed supervision. Label-efficient prediction studies the cost of limited queries (Cesa-Bianchi et al., 2005). OSAMD combines active label queries and self-training under changing distributions (Zhou et al., 2022). Label Delay in Online Continual Learning evaluates delayed supervision and proposes importance-weighted memory sampling (Csaba et al., 2024). FORTA fixes uniform queries across comparators and uses each released label for two operations: evaluating saved predictions and updating all supervised candidates. Rapid Network Adaptation uses task feedback (Yeo et al., 2023), while TeST uses teacher–student self-training (Sinha et al., 2023). Replay and DER++ supply candidate-learning mechanisms (Rolnick et al., 2019; Buzzega et al., 2020).

Anchoring, aggregation, and feedback costs. Hypothesis-transfer bounds analyze source-biased classification (Aghbalou & Staerman, 2023). Aggregation of ordered linear smoothers achieves oracle guarantees for regularization selection (Bellec & Yang, 2020). Stacking and expert tracking learn combinations of predictions (Wolpert, 1992; Herbster & Warmuth, 1998). Delayed-learning theory studies the additional cost of late observations (Joulani et al., 2013; Zierahn et al., 2025), while prequential evaluation scores outputs before training on their labels (Dawid, 1984). FORTA builds on these ideas. Its positive theorem compares the specified discounted F1 router with individually tuned anchors and fixed pools after candidate optimization, then extends the construction to shared-label learning.

3 Problem: Test-Time Adaptation with Sparse Delayed Feedback

The objective is to reduce target classification error when $P_t(X)$, $P_t(Y \mid X)$, or both change, under sparse delayed feedback. Predictions precede label availability; only a small fraction of examples receive labels later, and those labels can affect only subsequent predictions. Classes are $\{0, \dots, K-1\}$. The target conditional is $K_t(y \mid x) = P_t(Y = y \mid X = x)$. Hats denote model predictions. FORTA outputs $\widehat{K}_t^{\mathrm{F}}$, with online classification risk

$$\mathcal{R}_T = \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[\frac{1}{n_t} \sum_{j=1}^{n_t} \mathbf{1}\left\{ \arg\max_c \widehat{K}_t^{\mathrm{F}}(c \mid X_{tj}) \neq Y_{tj} \right\} \right].$$

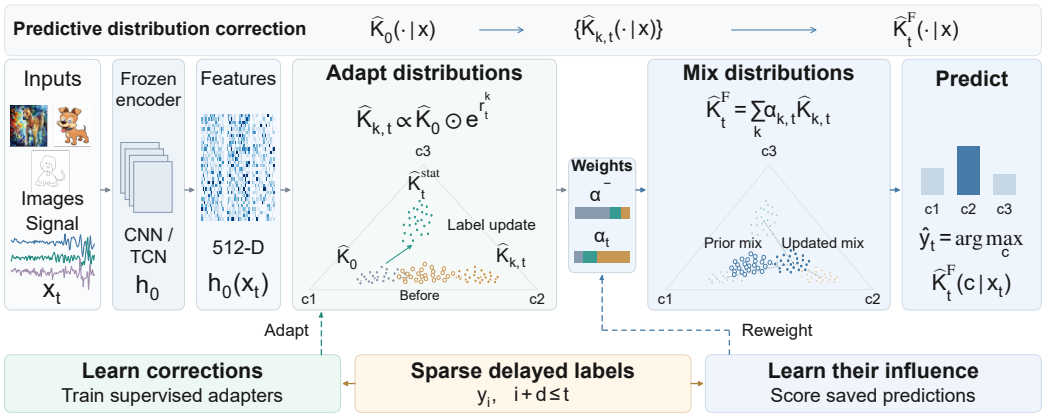


Figure 1: A time-varying set of predictive distribution corrections from a shared frozen embedding. The upper path updates candidate distributions, allocates their influence, and predicts. Hollow-to-filled clouds illustrate adaptation and reallocation on matched inputs. The lower path uses delayed feedback to score saved outputs and train candidates for future predictions. Inputs and features are real. Probability clouds are schematic, not uncertainty samples. Bars show one input's final distribution. The diagram uses global pooling. Images use class-specific normalization.

The expectation includes stream and feedback randomness. Predictions depend only on information available when they are made.

We train $f_0 = g_0 \circ h_0$ on source data D_0, then freeze the encoder h_0 and classifier, with logits $z^0(x) = W_0 h_0(x) + b_0$, and source kernel $\widehat{K}_0(c \mid x) = \mathrm{softmax}(z^0(x))_c = p_c^0(x)$. Batches $B_t = \{x_{tj}\}_{j=1}^{n_t}$ arrive in order; predictions precede all labels from the current batch.

We query examples uniformly after prediction under a label budget b, independently of confidence and class. The implementation calls a query an audit. For $d \geq 1$, labels queried in batch s arrive before prediction in batch $s + d$. At $d = 0$, they arrive after the same batch is predicted. Prediction uses source data, current and past inputs, and previously released labels. Current labels and query selections are unavailable. Appendix C specifies sampling and the information constraint.

Covariate shift over an interval means, on the common support, $P_t(Y \mid X) = P_0(Y \mid X)$ while $P_t(X) \neq P_0(X)$ for some t. Concept drift means $P_t(Y \mid X) \neq P_0(Y \mid X)$ on a set of positive target probability. FORTA observes neither shift identities nor change points and allows repeated or gradual changes in the joint distribution. Experiments report prequential macro-F1 on all emitted predictions, including queried examples. The analysis uses classification risk.

4 FORTA: Learning a Time-Varying Set of Predictive Corrections

The source model already provides a useful conditional distribution $\widehat{K}_0(Y \mid X)$. Relearning a full predictor from sparse delayed feedback would discard source knowledge and increase the estimation burden. A target shift also need not invalidate every source prediction. FORTA therefore keeps the source fixed and learns *corrections*: changes to its class probabilities from the frozen embedding. This retains the source as an exact candidate and concentrates the limited feedback on target prediction errors. Under heterogeneous shift, one correction need not remain useful. FORTA maintains a time-varying set of candidate corrections, then allocates their influence using evidence from earlier predictions (Figure 1).

4.1 Why residual correction?

The correction set $\mathcal{K}_t = \{\widehat{K}_0, \widehat{K}_{1,t}, \ldots, \widehat{K}_{M-1,t}\}$ contains M candidate class distributions, including the unchanged source at $k = 0$. Candidate k adds a residual score vector to the source logits:

$$z_t^k(x) = z^0(x) + r_t^k(h_0(x)), \qquad \widehat{K}_{k,t}(c \mid x) = p_t^k(c \mid x) = \mathrm{softmax}(z_t^k(x))_c.$$

Equivalently, each residual reweights the source conditional distribution:

$$\widehat{K}_{k,t}(c \mid x) = \frac{\widehat{K}_0(c \mid x) \exp\{r^k_{t,c}(h_0(x))\}}{\sum_v \widehat{K}_0(v \mid x) \exp\{r^k_{t,v}(h_0(x))\}}.$$

A residual difference $r^k_{t,c} - r^k_{t,v}$ shifts the source log odds between classes c and v. This form is an anchor, not a small-change assumption. For any positive conditional distribution $Q(\cdot \mid x)$ expressible from $h_0(x)$, the choice $r_c = \log Q(c \mid x) - \log \widehat{K}_0(c \mid x)$ recovers Q exactly. Meanwhile, $r^0_t = 0$ recovers the source. Residuals therefore give different adaptation mechanisms one distribution-level interface while preserving a stable fallback.

The set contains six image and five signal candidates, including the source. Replay, covariance-aware, nonlinear, and pseudo-label-assisted mechanisms produce distinct residual corrections. Feature-statistics alignment instead corrects the input to the source head. All mechanisms share the frozen representation and feedback schedule while keeping separate learning states.

All supervised candidates learn from every released label, independently of allocation. A small mixture weight therefore reduces a candidate's current influence without reducing its access to training feedback. This matters when the useful correction changes across target conditions. Appendix B gives the candidate objectives and replay states.

4.2 Learning the correction set and its allocation

FORTA's distinct object is a time-varying set of distribution corrections with feedback-guided allocation. Let $\mathcal{H}_t$ contain the inputs and feedback available before prediction at time t. In the global form, FORTA learns the increments $\delta_{k,t} = \widehat{K}_{k,t} - \widehat{K}_0$ and their allocation:

$$\begin{aligned} \widehat{K}^{\mathrm{F}}_t(\cdot \mid x) &= \sum_{k=0}^{M-1} \alpha_{k,t}(\mathcal{H}_t) \widehat{K}_{k,t}(\cdot \mid x), \\ \underbrace{\widehat{K}^{\mathrm{F}}_t - \widehat{K}_0}_{\text{deployed correction}} &= \sum_{k=1}^{M-1} \underbrace{\alpha_{k,t}(\mathcal{H}_t)}_{\text{feedback allocation}} \underbrace{\delta_{k,t}}_{\text{candidate correction}}, \qquad \alpha_t \in \Delta_M. \end{aligned} \tag{1}$$

Here $\Delta_M = \{a \in [0,1]^M : \sum_k a_k = 1\}$. The increments form a moving correction basis. A single adapter learns one increment. A fixed pool holds α_t constant. FORTA updates both the correction set and its allocation as target conditions change.

The delayed-feedback coupling is equally important. Let $\mathcal{L}_t$ index examples whose labels are released at time t, let $\mathcal{S}_{\mathrm{sup}}$ index supervised corrections, and let $\widehat{K}^{\mathrm{pre}}_{k,s}$ be candidate k's saved distribution before label y_s was available. With router $\mathcal{R}$ and candidate learner $\mathcal{A}_k$, one release performs

$$\begin{aligned} e_{k,s} &= \ell(\widehat{K}^{\mathrm{pre}}_{k,s}, y_s), && s \in \mathcal{L}_t, \\ \alpha^+_t &= \mathcal{R}(\alpha_t, \{e_{k,s}\}_{k,s}), \quad \theta^+_{k,t} = \mathcal{A}_k(\theta_{k,t}, \{(x_s, y_s) : s \in \mathcal{L}_t\}), && k \in \mathcal{S}_{\mathrm{sup}}. \end{aligned} \tag{2}$$

The candidate update has no $\alpha_{k,t}$ term. Every supervised candidate therefore learns from the same released labels, including candidates with little current influence. Routing uses pre-feedback errors, so a released label changes only future predictions. This separates learning members of the correction set from deciding when to deploy them. Appendix C.1 gives the complete timing and state updates.

At each input x, the global pool lies in $\mathrm{conv}\{\widehat{K}_{k,t}(\cdot \mid x)\}_k$. The class-specific image pool uses a normalized extension, defined below, that need not lie in this global convex hull. Neither formulation requires changing the source encoder. The next step is to estimate these weights from prediction performance.

4.3 Image allocation: global evidence and class-specific refinement

The image router first scores each candidate globally, then refines the weights by class. A correction useful for one class need not be useful for another. The class-specific refinement represents this distinction.

Let $p_{ki}(c)$ be strategy k's probability for class c, saved before audit i is selected. When y_i is released, FORTA evaluates the bounded Brier loss

$$\ell_{ki} = \frac{1}{2}\sum_{c=0}^{K-1}\big(p_{ki}(c) - \mathbf{1}\{y_i = c\}\big)^2 \in [0,1].$$

Let $\bar{\ell}_{k,t}$ denote its discounted mean loss, first averaged within each observed class and then equally across those classes. This balances represented classes. With score scale $\eta > 0$, prior share $\gamma \in (0,1)$, and normalized positive priors π, the global weights are

$$q_{k,t} \propto \pi_k \exp(-\eta\bar{\ell}_{k,t}), \qquad \widetilde{q}_{k,t} = (1-\gamma)q_{k,t} + \gamma\pi_k. \tag{3}$$

The prior share preserves $\widetilde{q}_{k,t} \geq \gamma\pi_k > 0$, including for the source. Before the first release, weights equal π. Appendix B.1 gives the discounted accumulators and numerical settings.

Class-specific offsets $u_{kc,t}$ refine the global weights:

$$a_{kc,t} = \frac{\exp(\log \widetilde{q}_{k,t} + u_{kc,t})}{\sum_{r=0}^{M-1}\exp(\log \widetilde{q}_{r,t} + u_{rc,t})},$$

Zero offsets recover global pooling. Class-specific weights generally produce class scores whose sum differs from one, so the output normalizes them:

$$\widehat{K}_t^{\mathrm{F}}(c \mid x) = \bar{p}_t(c \mid x) = \frac{\sum_k a_{kc,t}p_t^k(c \mid x)}{\sum_{v=0}^{K-1}\sum_k a_{kv,t}p_t^k(v \mid x)}.$$

For a released-label buffer $\mathcal{B}_t$, let n_c count labels of class c, and let $\mathcal{C}_t^{\mathcal{B}} = \{c : n_c > 0\}$. Write $\bar{p}_i^u$ for the normalized pool above, formed from the saved p_{ki} and current global weights. The offset objective is

$$\mathcal{J}_t(u) = -\frac{1}{|\mathcal{C}_t^{\mathcal{B}}|}\sum_{c\in\mathcal{C}_t^{\mathcal{B}}}\frac{1}{n_c}\sum_{i\in\mathcal{B}_t:y_i=c}\log \bar{p}_i^u(c) + \frac{\lambda_u}{MK}\|u\|_F^2. \tag{4}$$

Brier evidence scores individual candidates. This objective fits their combination, giving equal weight to represented classes. Both are surrogates for classification performance. At each release, saved probabilities and current global weights stay fixed during offset optimization. The penalty favors global pooling. Candidate training follows the allocation update.

4.4 Signal allocation: global performance weights

For time series, we use global weights to estimate fewer allocation parameters from sparse feedback. Let $F_{k,t}$ be candidate k's macro-F1 from discounted confusion counts of its saved predictions. The counts accumulate from the first release. After eight labels, allocation is

$$q_{k,t}^{\mathrm{ts}} \propto \pi_k \exp(32F_{k,t}), \qquad \widetilde{q}_{k,t}^{\mathrm{ts}} = .98q_{k,t}^{\mathrm{ts}} + .02\pi_k.$$

Higher measured macro-F1 increases future influence. Appendix B.1 defines the confusion-count update and the classes included in macro-F1. The emitted distribution is (1) with $\alpha_{k,t} = \widetilde{q}_{k,t}^{\mathrm{ts}}$ and $\widehat{K}_t^{\mathrm{F}} = \bar{p}_t$. Before eight labels, allocation uses the global Brier weights from Section 4.3, initialized at π. Signals use no class-specific offsets. Individual macro-F1 scores guide allocation but do not directly optimize the macro-F1 of the final pool.

5 Analysis: Feedback Limits and Allocation Gains

The method raises two questions. How much prediction error is unavoidable before feedback identifies a useful conditional correction? Once a correction set is available, does adaptive allocation improve on a single optimized correction or a fixed mixture? The first result answers the information question. The second constructs an allocation advantage and accounts for the time spent learning and evaluating the correction set.

5.1 WHY CONDITIONAL CORRECTION NEEDS FEEDBACK

Unlabeled inputs do not identify unrestricted conditional change, a standard obstruction in domain adaptation (Ben-David et al., 2010). Label-efficient prediction studies limited queries (Cesa-Bianchi et al., 2005), and delayed online learning quantifies late feedback (Joulani et al., 2013). The following two-world bound specializes this information argument to our predict-before-feedback protocol.

Theorem 5.1 (Conditional uncertainty before informative feedback)**.** *Two binary target worlds share the source data and i.i.d. inputs. Their labeling rules agree outside a set of target mass $\alpha > 0$ and are opposite inside. Query m labels per batch after prediction using independent uniform index samples, independent of inputs and labels. With delay $d \geq 0$, $N_t = m(t - \max\{1, d\})_+$ queried labels are available before batch t. Every learner has expected batch classification risks $R_t^{(+)}$ and $R_t^{(-)}$ satisfying*

$$\max_{w \in \{+,-\}} \frac{1}{T} \sum_{t=1}^{T} R_t^{(w)} \geq \frac{\alpha}{2T} \sum_{t=1}^{T} (1-\alpha)^{N_t}. \tag{5}$$

Both worlds have zero Bayes error. With no queries and $\alpha = 1$, the bound is $1/2$, using $0^0 = 1$.

The factor $(1-\alpha)^{N_t}$ is the probability no released label covers the changed region. Sparse queries and delay prolong this uncertainty. Labels supply the missing information. Delay restricts when learning starts. Pseudo-labels formed from the same history do not distinguish these worlds. Appendix A proves the bound and its sharpness when the two candidate rules are known.

5.2 WHEN LEARNING AND ALLOCATION OVERCOME THE FEEDBACK COST

For frozen features $Z = h_0(X)$, source weight w_0, and source second moment $A \succ 0$, define the optimized correction in regime r by

$$w_\lambda^r = \arg\min_w \left\{ \mathbb{E}_r[\ell_{\mathrm{ce}}(\sigma(w^\top Z), Y)] + \frac{\lambda}{2}(w - w_0)^\top A (w - w_0) \right\}.$$

The candidate is $p_\lambda^r(z) = \sigma((w_\lambda^r)^\top z)$, where σ is the sigmoid. Strengths satisfy $\lambda \geq \lambda_{\min} > 0$, including the source at $\lambda = \infty$. The scalar oracle selects the best strength separately in each regime. The fixed oracle optimizes constant weights over the source and two corrections, each optimized for its current regime. Both oracle risks weight the two regimes equally.

Theorem 5.2 (Learning and allocation with delayed feedback)**.** *There exist two target regimes differing in both $P(X)$ and $P(Y \mid X)$, with i.i.d. observations within each, and a three-candidate correction set whose equally weighted stationary routed risk is bounded by $U < \min\{R^*_{\mathrm{scalar}}, R^*_{\mathrm{fixed}}\}$. Use the specified discounted F1 router and $m = 3$ uniform queries per batch with finite delay d. Both corrections minimize the empirical anchored objective on the latest W released labels, independently of their weights. For every $\epsilon > 0$, finite replay size W and discount-tail horizon H exist such that, when the regimes alternate in equal segments of length D,*

$$\limsup_{T \to \infty} \mathcal{R}_T^{\mathrm{route}} \leq U + e_D, \qquad e_D = \epsilon + \frac{\lceil W/m \rceil + H + 2d}{D}. \tag{6}$$

Every fixed pool of the same learned candidates has long-run risk at least $R^*_{\mathrm{fixed}} - e_D$. Thus routing beats both the scalar population oracle and every such fixed pool whenever $e_D < \min\{R^*_{\mathrm{scalar}} - U, (R^*_{\mathrm{fixed}} - U)/2\}$.

Why the method works in this construction. Complementary members of the correction set improve one regime, while the source serves the other. All corrections learn from released labels. Saved-output feedback then changes their influence. The cost separates into candidate learning $\lceil W/m \rceil$, discount-tail horizon H, and two feedback waits $2d$. Longer regimes dilute this transient cost. The two waits are a sufficient proof allowance, not a universal lower bound on adaptation time.

Appendix A.2 specifies the construction, router, and proof. This result concerns empirical anchored minimizers and sufficiently long regimes. It does not establish optimality for the deployed neural pools, zero-anchor adaptation, or jointly trained mixtures. Appendix A.3 checks the feedback and duration effects with synthetic data. Appendix A.4 records the standard sampling and pooling identities used to interpret the method.

Table 1: Image-domain macro-F1 with 15% delayed labels (mean±SD, nine source/query combinations). Mean and rank weight datasets equally. Bold marks the best.

Method	PACS	OfficeHome	VLCS	Mean	Rank ↓
Source-only	.3701±.0120	.5161±.0008	.4703±.0251	.4521±.0122	6.67
Matched residual	.7306±.0079	.5425±.0051	.6448±.0136	.6393±.0052	2.67
Audit replay	.7474±.0027	.5271±.0060	.6412±.0135	.6386±.0063	3.67
DER++	.7111±.0085	.4740±.0064	.6215±.0142	.6022±.0066	5.67
Full covariance	.7860±.0020	.5281±.0030	.6339±.0166	.6493±.0059	3.33
Delayed OSAMD port	.5641±.0083	.5414±.0017	.5473±.0223	.5509±.0074	5.00
FORTA	**.7942±.0027**	**.5573±.0032**	**.6642±.0179**	**.6719±.0064**	**1.00**

6 Experiments

6.1 Data, protocol, and comparators

The experiments separate three questions: whether the complete correction set improves over the source and individual corrections, whether changing allocation helps when correction trajectories are held fixed, and whether feedback coverage and regime duration produce the effects predicted by the theory. Four ordered signal streams, WISDM, MIT-BIH, REALDISP, and Gas Sensor Drift, cover changes across people, recordings, sensor placements, and acquisition periods (Weiss, 2019; Moody & Mark, 2001; Banos et al., 2012; Vergara et al., 2012). The image benchmarks PACS, OfficeHome, and VLCS test transfer across real visual domains (Li et al., 2017; Venkateswara et al., 2017; Fang et al., 2013). We use a frozen TCN for signals, ResNet-18 for PACS and OfficeHome, and RN50 for VLCS. Resets at each target image domain isolate domain adaptation from temporal switching. The controlled stream separately tests changing regimes within deployment.

Evaluation scores predictions made before query selection or label release. We query 5% of signal examples and 15% of image examples uniformly after prediction and release their labels two batches later to update future predictions. Within each comparison, label-using methods share the source, target stream, queried examples, and release times. Label-free TTA uses a different observation setting and is not presented as a matched performance baseline. Tables report macro-F1 mean and standard deviation separately for each modality and budget.

The comparisons include the source predictor, individual correction mechanisms, and the complete FORTA framework. DER++ is a separate image comparator; full covariance and OSAMD are separate signal comparators. Standalone runs assess complete methods, whereas matched-portfolio controls reuse exactly the same candidate trajectories and change only their combination. We used the real benchmarks during development; they are not an independent confirmatory suite. Appendix C records selection history and reproducibility details.

6.2 Real image-domain shifts

FORTA leads the reported individual comparators on all three image benchmarks (Table 1), with mean macro-F1 of .672 compared with .649 for full covariance, the strongest shared comparator. On VLCS, FORTA improves all three target domains and raises worst-domain macro-F1 from .360 to .535. These comparisons assess the full time-varying correction set and its allocation. Detailed domain results appear in Appendix D.

6.3 Natural ordered signals

FORTA improves over the source on all four streams and achieves the strongest mean performance (Table 2). It leads on MIT-BIH and REALDISP, while audit replay remains stronger on WISDM and Gas. The strongest standalone mechanism varies across streams, motivating a set of complementary corrections. The following controls isolate allocation.

Table 2: Signal-stream macro-F1 with 5% delayed labels (mean±SD, three runs). Mean and rank weight streams equally. Bold marks the best.

Method	WISDM	MIT-BIH	REALDISP	Gas	Mean	Rank ↓
Source-only	.804±.014	.434±.013	.589±.051	.564±.014	.598±.014	6.000
Matched residual	.854±.007	.517±.017	.418±.013	.765±.011	.638±.001	3.750
Audit replay	**.858±.009**	.521±.018	.418±.030	**.777±.017**	.643±.004	3.000
DER++	.846±.008	.538±.038	.358±.017	.752±.017	.623±.011	4.250
Full covariance	.839±.007	.536±.027	.607±.055	.722±.012	.676±.016	3.750
Delayed OSAMD port	.816±.012	.487±.009	.588±.053	.574±.013	.616±.013	5.500
FORTA	.853±.004	**.544±.011**	**.661±.023**	.772±.021	**.707±.005**	**1.750**

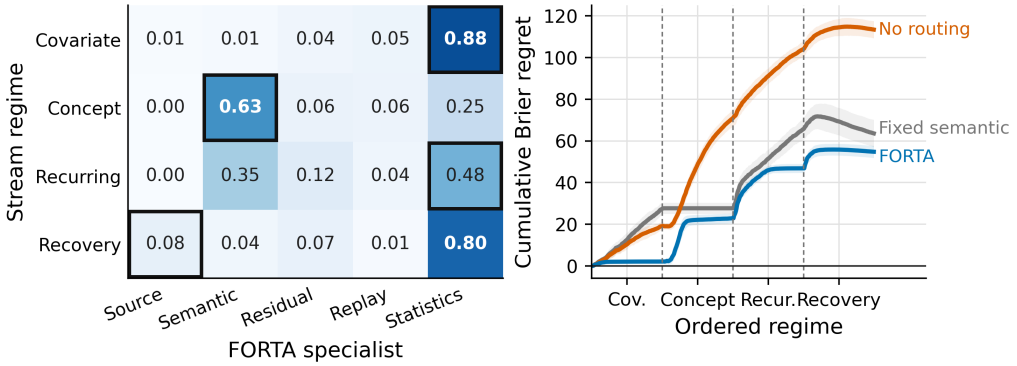


Figure 2: **Allocation across changing shift regimes** (mean±SD, ten paired seeds). Left: regime-average router mass; boxes mark the switching oracle. Right: cumulative Brier loss relative to the regime-wise specialist oracle on queried examples, scored before feedback. Lower is better.

6.4 Changing regimes: isolating allocation

The benchmark tables evaluate the full framework. To isolate allocation, we next hold the correction trajectories fixed while changing which member is useful. We construct a stream with measurement shift, conditional change, recurrence, and source recovery. The best specialist changes with the regime, from feature statistics to semantic correction and eventually the source. All adaptive methods receive the same 5% labels delayed by two batches. For the fixed-mixture control, we train the same adapters as FORTA but keep their allocation unchanged.

FORTA reaches .805 macro-F1, compared with .732 for the strongest fixed specialist and .662 for the fixed mixture. Its cumulative Brier loss relative to the regime-wise specialist oracle is also lower on the queried examples (Figure 2). With candidate learning held fixed, this gain comes from changing the combination of predictive distributions. Recovery is slower: the router retains mass on feature-statistics correction when the source becomes best. The retained source is therefore a recovery option, not a guarantee of immediate recovery. Appendix E gives the regime-wise results.

6.5 What does allocation add to a learned correction set?

The controlled stream establishes a setting where adaptive allocation helps. Figure 3 tests whether this benefit extends to natural shifts, using identical candidate predictions and delayed feedback. Fixed pools still train candidates. PACS uses class-blocked or shuffled inputs, while WISDM preserves within-person chronology. We compare per-group resets with continual adaptation, which also changes candidate learning.

On shuffled PACS, FORTA gains 6.63 and 5.71 points over the fixed prior, but 0.08 and -0.63 over uniform pooling. FORTA leads continual class-blocked PACS. Fixed-share and convex stacking lead

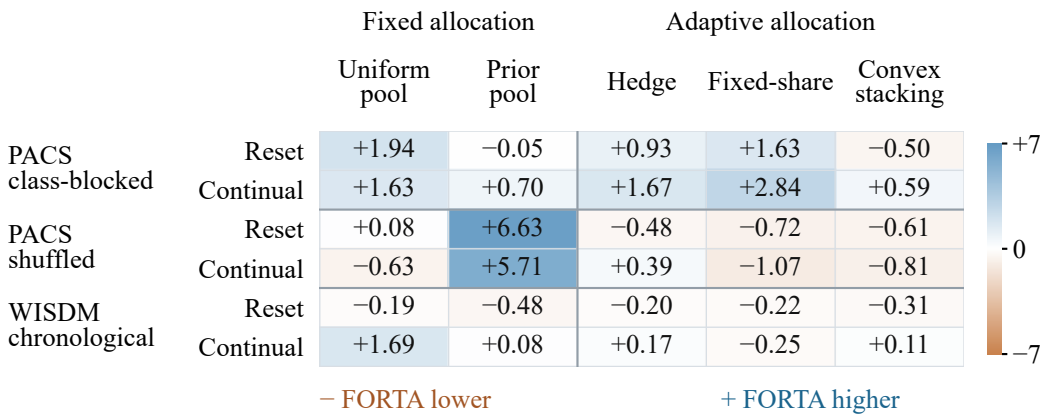


Figure 3: FORTA minus each comparator in macro-F1 points. Blue indicates gains and orange losses on a shared symmetric scale. Comparators share candidate predictions and feedback within each condition. Differences are paired within source checkpoints before aggregation.

FORTA on shuffled PACS, and fixed-share leads continual WISDM. Allocation gains depend on the condition and baseline.

Appendix F specifies the matched protocol and allocation controls. The original PACS reset rerun is .788 versus the archived .794. This reproduction discrepancy remains unresolved.

6.6 Feedback sensitivity

In the tested sweep, more feedback improves performance and longer delays reduce the gains. The largest budget improvement occurs between 1% and 5%. This sweep measures correction-set learning and allocation jointly. Appendix D gives the curves and settings.

6.7 Synthetic checks of the theory

Separate simulations test both theorem predictions. The two-world experiment checks error before an informative label arrives. The alternating-regime experiment checks whether the learned correction set and delayed allocation recover the certified advantage as segments grow longer. At delay 16, routed error falls from .491 to .217 as segments grow from 128 to 8,192 batches, beating the .225 scalar oracle. Rapid switching with long delay loses this advantage. Appendix A.3 gives the complete protocol.

7 Conclusion

FORTA treats adaptation as learning a time-varying set of predictive distribution corrections and allocating its members. The frozen source provides a stable anchor, while complementary mechanisms revise its class distribution in different ways. Sparse delayed feedback trains every supervised correction and evaluates saved pre-feedback predictions for future allocation. A suppressed correction therefore continues learning and remains available when target conditions change.

The information bound explains why unlabeled inputs and history-derived pseudo-labels do not identify unrestricted conditional change. The allocation construction shows when the gain from selecting among learned corrections outweighs the transient costs of candidate learning, delayed evaluation, and discounted routing history. Across seven benchmarks, FORTA improves over the source. Controlled switching shows the benefit of changing allocation when correction usefulness changes, while matched natural streams show that fixed pools remain competitive in some conditions. These findings frame heterogeneous TTA as the joint problem of learning a correction set and deciding which members should shape future predictions. The guarantee remains limited to the stated source-anchored construction.

REFERENCES

Anass Aghbalou and Guillaume Staerman. Hypothesis transfer learning with surrogate classification losses: Generalization bounds through algorithmic stability. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 280–303, 2023. URL https://proceedings.mlr.press/v202/aghbalou23a.html.

Oresti Banos, Mate Toth, and Oliver Amft. REALDISP activity recognition dataset. UCI Machine Learning Repository, 2012.

Pierre Bellec and Dana Yang. The cost-free nature of optimally tuning Tikhonov regularizers and other ordered smoothers. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 746–755, 2020. URL https://proceedings.mlr.press/v119/bellec20a.html.

Shai Ben-David, Tyler Lu, Teresa Luu, and David Pal. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 129–136, 2010.

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*, pp. 8344–8353, 2022.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: A strong, simple baseline. In *Advances in Neural Information Processing Systems*, volume 33, pp. 15920–15930, 2020.

Nicolò Cesa-Bianchi, Gábor Lugosi, and Gilles Stoltz. Minimizing regret with label efficient prediction. *IEEE Transactions on Information Theory*, 51(6):2152–2162, 2005. doi: 10.1109/TIT.2005.847729. URL https://stoltz.perso.math.cnrs.fr/Publications/CBLS-LabelEff.pdf.

Botos Csaba, Wenxuan Zhang, Matthias Müller, Ser-Nam Lim, Mohamed Elhoseiny, Philip H. S. Torr, and Adel Bibi. Label delay in online continual learning. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

A. Philip Dawid. Present position and potential developments: Some personal views: Statistical theory: The prequential approach. *Journal of the Royal Statistical Society: Series A*, 147(2): 278–292, 1984. doi: 10.2307/2981683.

Xiao Fan, Jingyan Jiang, Zhaoru Chen, Fanding Huang, Xiao Chen, Qinting Jiang, Bowen Zhang, Xing Tang, and Zhi Wang. Moetta: Test-time adaptation under mixed distribution shifts with moe-layernorm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21011–21019, 2026. doi: 10.1609/aaai.v40i25.39243. URL https://ojs.aaai.org/index.php/AAAI/article/view/39243.

Chen Fang, Ye Xu, and Daniel N. Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *IEEE International Conference on Computer Vision*, 2013.

Joao Gama, Indre Zliobaite, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):1–37, 2014. doi: 10.1145/2523813.

Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. NOTE: Robust continual test-time adaptation against temporal correlation. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27253–27266, 2022. URL https://papers.nips.cc/paper_files/paper/2022/hash/ae6c7dbd9429b3a75c41b5fb47e57c9e-Abstract-Conference.html.

Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. SoTTA: Robust test-time adaptation on noisy data streams. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://papers.nips.cc/paper_files/paper/2023/hash/2da53cd1abdae59150e35f4693834f32-Abstract-Conference.html.

Mark Herbster and Manfred K. Warmuth. Tracking the best expert. *Machine Learning*, 32(2): 151–178, 1998.

Nicholas J. Higham. The accuracy of floating point summation. *SIAM Journal on Scientific Computing*, 14(4):783–799, 1993. doi: 10.1137/0914050.

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvari. Online learning under delayed feedback. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1453–1461. PMLR, 2013. URL https://proceedings.mlr.press/v28/joulani13.html.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision*, 2017.

George B. Moody and Roger G. Mark. The impact of the MIT-BIH arrhythmia database. *IEEE Engineering in Medicine and Biology Magazine*, 20(3):45–50, 2001. doi: 10.1109/51.932724.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16888–16905. PMLR, 2022. URL https://proceedings.mlr.press/v162/niu22a.html.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=g2YraF75Tj.

Joaquin Quinonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence (eds.). *Dataset Shift in Machine Learning*. MIT Press, 2009.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Samarth Sinha, Peter Gehler, Francesco Locatello, and Bernt Schiele. Test: Test-time self-training under distribution shift. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023. URL https://openaccess.thecvf.com/content/WACV2023/papers/Sinha_TeST_Test-Time_Self-Training_Under_Distribution_Shift_WACV_2023_paper.pdf.

Zixian Su, Jingwei Guo, Xi Yang, Qiufeng Wang, and Kaizhu Huang. Un-mixing test-time adaptation under heterogeneous data streams, 2024. URL https://arxiv.org/abs/2411.15173.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Alexander Vergara, Shankar Vembu, Tuba Ayhan, Margaret A. Ryan, Margie L. Homer, and Ramon Huerta. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166–167:320–329, 2012. doi: 10.1016/j.snb.2012.01.074.

Guillaume Vray, Devavrat Tomar, Xufeng Gao, Jean-Philippe Thiran, Evan Shelhamer, and Behzad Bozorgtabar. ReservoirTTA: Prolonged test-time adaptation for evolving and recurring domains. In *Advances in Neural Information Processing Systems*, volume 38, pp. 63357–63403, 2025.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

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*, pp. 7201–7211, 2022.

Shuoyuan Wang, Jindong Wang, Huajun Xi, Bob Zhang, Lei Zhang, and Hongxin Wei. Optimization-free test-time adaptation for cross-person activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 7(4):1–27, 2023. doi: 10.1145/3631450.

Gary M. Weiss. WISDM smartphone and smartwatch activity and biometrics dataset. UCI Machine Learning Repository, 2019.

David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992. doi: 10.1016/S0893-6080(05)80023-1.

Teresa Yeo, Oğuzhan Fatih Kar, Zahra Sodagar, and Amir Zamir. Rapid network adaptation: Learning to adapt neural networks using test-time feedback. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4674–4687, 2023.

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*, pp. 15922–15932, 2023.

Shiji Zhou, Han Zhao, Shanghang Zhang, Lianzhe Wang, Heng Chang, Zhi Wang, and Wenwu Zhu. Online continual adaptation with active self-training. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 8852–8883. PMLR, 2022. URL https://proceedings.mlr.press/v151/zhou22d.html.

Lukas Zierahn, Dirk van der Hoeven, Tal Lancewicki, Aviv Rosenberg, and Nicolò Cesa-Bianchi. A unified analysis of nonstochastic delayed feedback for combinatorial semi-bandits, linear bandits, and MDPs. *Journal of Machine Learning Research*, 26(104):1–60, 2025. URL https://www.jmlr.org/papers/v26/24-0496.html.

A Proofs and Supporting Analysis

A.1 Proof of Theorem 5.1: feedback availability

Write the joint law as $P_t(dx, dy) = P_t^X(dx)K_t(x, dy)$, fixing a conditional version on the target support. The two-world experiment holds P_t^X and the source observations fixed while changing K_t. The proof therefore concerns information about the conditional, even when the learner has unlimited unlabeled inputs.

Sparse delayed feedback. Theorem 5.1 is a protocol-specific two-world argument, building on domain-adaptation impossibility (Ben-David et al., 2010) and the study of limited and delayed feedback (Cesa-Bianchi et al., 2005; Joulani et al., 2013). It does not claim a new label-free impossibility principle. The bound accounts for how many informative labels are available before each prediction.

Proof of Theorem 5.1. Let $G \subseteq \mathcal{X}$ be measurable with $P^X(G) = \alpha$. Fix a binary rule g. Both worlds share the source law, learner randomness, and independent target inputs with marginal P^X. Their deterministic labels are

$$Y^{(+)} = g(X), \qquad Y^{(-)} = \begin{cases} 1 - g(X), & X \in G, \\ g(X), & X \notin G. \end{cases}$$

Batches contain at least m examples. Choose a uniform subset of m indices after prediction, independently of inputs, labels, and other batches. Couple the index choices across worlds. Sampling distinct indices from an i.i.d. batch gives m independent queried inputs marginally. Across batches these inputs also remain independent.

For $d \geq 1$, queries from batch s arrive before prediction in batch $s + d$. For $d = 0$, feedback follows prediction in batch s and first influences batch $s + 1$. Thus, with $\ell = \max\{1, d\}$, exactly $N_t = m(t - \ell)_+$ queried labels are available before batch t. The event E_t of no released query in G has probability

$$P(E_t) = (1 - \alpha)^{N_t}.$$

On E_t, all released labels agree across worlds. All observable histories, including current inputs and any pseudo-labels, have the same coupled realization. Their predictions therefore agree. On each current input in G, these predictions make an error in exactly one world. The current batch is independent of E_t, which depends only on past queried inputs. Consequently,

$$R_t^{(+)} + R_t^{(-)} \geq \mathbb{E}\left[\mathbf{1}_{E_t} \frac{1}{n_t} \sum_{j=1}^{n_t} \mathbf{1}\{X_{tj} \in G\}\right] = \alpha(1 - \alpha)^{N_t},$$

where n_t denotes batch size, distinct from the available label count N_t. Summing and bounding the maximum of the two cumulative risks by their average proves (5). Both rules are deterministic, so both Bayes risks are zero. For $m = 0$, the exponent is zero and the probability is one, including at $\alpha = 1$. □

Sharpness and interpretation. Suppose the learner knows g and G, but not the world. Before the first released query in G, choose either world with equal probability and predict its rule. The first such label identifies the world exactly. This learner attains risk $\alpha(1 - \alpha)^{N_t}/2$ in each world at every batch. The bound is therefore exact for this two-rule experiment. It does not imply attainability for an unknown high-dimensional conditional. For $m \geq 1$, write $r = (1 - \alpha)^m < 1$. The total expected error before identification, summed in batch-average risk units, is

$$\frac{\alpha}{2} \sum_{t=1}^{\infty} r^{(t-\ell)_+} = \frac{\alpha}{2} \left(\ell - 1 + \frac{1}{1 - r}\right).$$

Here $1/(1 - r)$ is the expected wait for an informative queried batch, and $\ell - 1$ is the additional wait relative to post-prediction immediate feedback. Delay does not supply information. Released labels do. The same coupling gives $\alpha(1 - \alpha)^{N_t}/2$ for any predetermined query schedule, with N_t the number of labels available before batch t. Adaptive query policies require a separate coverage argument.

Label-free observations and pseudo-labels. With no queries and disagreement on the full support, couple the source, target inputs, and learner randomness across the two worlds. Their emitted predictions agree, while their true labels are opposite, giving $R_t^{(+)} + R_t^{(-)} = 1$. This special case requires no independence assumption on the target input process. A pseudo-label computed from the observable history has the same coupled value in both worlds. Induction preserves this equality through later updates, so pseudo-labeling does not remove the lower bound. Structural restrictions on the target conditional would define a different observation experiment.

A.2 Proof of Theorem 5.2: learning and routing

The theorem is an attainability result for the specified global F1 rule over a three-member, time-varying correction set. In the predictive-kernel formulation, each binary candidate is $\widehat{K}_k^r(1 \mid z) = p_k^r(z)$, with class-0 probability $1 - p_k^r(z)$. Thus the probability pool below is exactly $\widehat{K}^{\mathrm{F}}$; the residual parameterization constructs the candidates and does not assert a guarantee for arbitrary repair operators. Its comparators receive residuals optimized for the current regime; neither comparison freezes the learned residuals across a shift. The proof gives an explicit construction, establishes the two oracle values, and bounds the risk of the feedback-selected mixture. The numerical inequalities below are verified with outward-rounded interval arithmetic and finite positive convolutions. The accompanying `supplement/theorem52` contains the code, inputs, probability arrays, and a verification driver. These are proof certificates, not estimates from benchmark data or Monte Carlo trials.

Complete setup and quantitative statement. For a target regime r, consider binary linear residuals on frozen features Z, source logit $w_0^\top Z$, and source-anchor second moment $A \succ 0$. Define

$$w_\lambda^r = \arg\min_w \left\{ \mathbb{E}_r[\log(1 + e^{w^\top Z}) - Y w^\top Z] + \frac{\lambda}{2}(w - w_0)^\top A (w - w_0) \right\}, \tag{7}$$

where $(w - w_0)^\top Z$ is the residual logit, $p_\lambda^r(z) = \sigma((w_\lambda^r)^\top z)$, and $p_\infty^r = p_0 = \sigma(w_0^\top z)$ retains the source. Write $R_r(p) = P_r(\mathbf{1}\{p(Z) \geq 1/2\} \neq Y)$. The analyzed bank is $(p_0, p_{.02}^r, p_3^r)$. Its global signal-routing map is

$$q_L^r = .98\ \mathrm{softmax}(\log \pi + 32 \widehat{F}_L^r) + .02\pi, \qquad \pi = (.08, .46, .46), \tag{8}$$

where $\widehat{F}_L^r$ uses confusion counts discounted by .98 per release and predictions saved before feedback. Let $\overline{R}_{\mathrm{route}} = \frac{1}{2} \sum_{r \in \{A,B\}} \mathbb{E} R_r(\sum_k q_{L,k}^r p_k^r)$.

There exist a well-specified immutable source, $A \succ 0$, and two target regimes differing in both $P(X)$ and $P(Y \mid X)$ for which the following holds. Optimize every residual in its current regime as in (7). For i.i.d. observations within each regime, three uniformly queried labels per release, any fixed finite delay, and $L \geq 500$ releases in stationary comparisons initialized with zero confusion counts, the router in (8) satisfies

$$\begin{aligned} &\overline{R}_{\mathrm{route}} < .219737 < \min\{R^*_{\mathrm{scalar}}, R^*_{\mathrm{fixed}}\}, \\ &R^*_{\mathrm{scalar}} := \inf_{\lambda_A, \lambda_B \in [.02, \infty]} \frac{1}{2} \sum_{r \in \{A,B\}} R_r(p_{\lambda_r}^r) = .225, \\ &R^*_{\mathrm{fixed}} := \inf_{q \in \Delta_3} \frac{1}{2} \sum_{r \in \{A,B\}} R_r\left(\sum_k q_k p_k^r \right) = .30. \end{aligned} \tag{9}$$

The strict comparisons persist in the stationary consistent-learning limit when all residuals train on the same released labels independently of their weights. They also hold with sufficiently large finite replay windows and sufficiently long alternating regimes.

The stated finite-feedback count concerns evaluation of population-optimal heads. It does not assert that the same count is sufficient to train them. The consistent-learning and finite-window claims are proved separately below.

Construction. Take frozen features in $\mathbb{R}^3$, with base atoms

$$x_0 = (1, 0, 0), \quad x_1 = (.3, 1, 0), \quad x_2 = (.3, -1, 0), \quad x_3 = (.15, 0, 1), \quad x_4 = (.15, 0, -1).$$

For each j, the atoms x_j and $-x_j$ have equal mass and complementary labels. Their total pair masses and the labels at the positive base atoms are

$$m_A = (.5, .1, .1, .15, .15), \qquad y_A = (1, 1, 1, 1, 1),$$
$$m_B = (.55, .125, .125, .1, .1), \qquad y_B = (0, 1, 0, 0, 0).$$

Thus both regimes are balanced and linearly separable, their feature marginals have total variation distance .10, and their conditionals differ. Let $g_r(w)$ denote the regime-r cross-entropy gradient, and set

$$w_1 = (1, 1, 0), \quad w_2 = (.15, -1, 0), \quad (\lambda_1, \lambda_2) = (.02, 3),$$
$$u = w_1 - w_2, \quad z = -g_A(w_1)/\lambda_1 + g_A(w_2)/\lambda_2, \quad v = (-2, .85, 1.5),$$
$$A = \frac{zz^\top}{u^\top z} + I - \frac{uu^\top}{\|u\|^2} + 10vv^\top, \qquad w_0 = w_1 + A^{-1}g_A(w_1)/\lambda_1.$$

Here $u^\top z > 0$, $v^\top u = 0$, and $Au = z$. The first two terms in A are positive definite together: a vector annihilated by the second is parallel to u, and then cannot be annihilated by the first unless it is zero. Consequently $A \succ 0$, and the stationarity equations show that w_1, w_2 are the exact unique optima at the two specified strengths in A. A source with $Z \sim N(0, A)$ and $Y \mid Z \sim \mathrm{Bernoulli}(\sigma(w_0^\top Z))$ is well specified and has source second moment A. Its predictor is retained unchanged. For orientation, $w_0 \approx (-.015123, -1.601202, .115389)$ and $\mathrm{cond}(A) \approx 1499$; the certificates use the exact construction above, not these rounded values.

In B, refitting at the same strengths gives centers

$$w^B_{.02} \approx (.111541, .527151, -.797087), \qquad w^B_3 \approx (.119428, -1.088225, .008832).$$

If $\widetilde{w}$ is a numerical center and $\mu_\lambda = \lambda\lambda_{\min}(A)$, strong convexity encloses the exact optimizer by

$$\|w^r_\lambda - \widetilde{w}\| \le \frac{\|g_r(\widetilde{w}) + \lambda A(\widetilde{w} - w_0)\|}{\mu_\lambda}. \tag{10}$$

Outward-rounded enclosures certify the following hard-prediction patterns on $(x_0, \ldots, x_4)$ and their resulting risks:

	Source	Weak anchor	Strong anchor
A: predictions	00110	11011	10111
A: error	.75	.10	.10
B: predictions	00110	11001	10111
B: error	.35	.65	1

Every logit has a strict sign. In A the residual errors are disjoint; in B the weak head and source disagree everywhere. Both facts will enter the *joint* feedback calculation.

The entire scalar anchoring path. First consider A and write $w = (s, e_1, e_2)$. Having no strictly wrong base atom requires

$$s \ge 0, \qquad |e_1| \le .3s, \qquad |e_2| \le .15s.$$

Partition A into its bottom-right block B and bottom-left column h, and put $g = (Aw_0)_{2:3}$ and $a = (g_2, -g_1)$. The last two cross-entropy gradient coordinates equal $\mathrm{diag}(\kappa_1, \kappa_2)e$, where symmetry and $\sigma' \le 1/4$ give $0 \le \kappa_1 \le .1/2$ and $0 \le \kappa_2 \le .15/2$. Projecting stationarity onto a, which satisfies $a^\top g = 0$, would require, for $s > 0$ and $\lambda \ge .02$,

$$|a^\top h| \le .3|(Ba)_1| + .15|(Ba)_2| + \frac{.1 \cdot .3}{2 \cdot .02}|a_1| + \frac{.15 \cdot .15}{2 \cdot .02}|a_2|.$$

The interval-verified left side minus right side exceeds .3604758277. If $s = 0$, then $e = 0$, contradicting $g \neq 0$ in stationarity. Thus every finite admissible strength has a strictly wrong pair and risk at least .10; $\lambda = .02$ attains .10. The source endpoint also satisfies this lower bound.

For B, 128 adjacent parameter intervals cover $[.02, 10^4]$. On each interval I, let $F_\lambda(w) = g_B(w) + \lambda A(w - w_0)$. The certificate chooses a center w_c, a fixed invertible approximate inverse C, and a box $K = w_c + [-r, r]$, and verifies

$$w - CF_\lambda(w) \in \mathrm{int}(K) \quad \text{for every } w \in K,\ \lambda \in I.$$

The inclusion is checked by enclosing $-CF_\lambda(w_c)$ and $I - C\nabla F_\lambda(K)$ with interval arithmetic. Brouwer's theorem gives a zero for every parameter in I, and strict convexity makes that zero the unique optimizer. Each box fixes enough strictly wrong atoms to have total pair mass at least .35. Beyond 10^4,

$$\|w_\lambda^B - w_0\| \le \frac{\|g_B(w_0)\|}{\lambda \lambda_{\min}(A)} < .000587.$$

This ball preserves the source's wrong atoms $1, 2, 3$, of total mass .35. The source attains .35. The covered intervals include every endpoint; this is not a check on a sampled strength grid. Therefore $R^*_{\mathrm{scalar}} = (.10 + .35)/2 = .225$.

Every fixed probability mixture. Write $q = (\alpha, b, 1 - \alpha - b)$ for the source, weak, and strong weights. The certified probability enclosures imply the following strict-cell bounds. If B's central atom is correct, then $\alpha > .88$; B atoms $1, 2, 3$ remain wrong (mass .35), and A atoms $1, 4$ remain wrong (mass .25). The summed risk is at least .60. If B's central atom is wrong, it contributes .55. For $b \le .6$, B atoms $1, 2$ are also wrong; for $b > .6$, atom 4 is also wrong. Thus B alone has risk at least .65 in this case. All these implications are affine inequalities in the mixture weights; the interval certificate checks their extremal probabilities. The fixed weights $(.9, .1, 0)$ attain errors $(.25, .35)$, proving $R^*_{\mathrm{fixed}} = .30$ away from ties.

The lower bound extends to ties as follows. Perturb a mixture symmetrically within its minimal simplex face. Each tied affine margin is nonconstant on that face, since no expert probability at a base atom equals $1/2$. Its two signs are therefore equiprobable under the perturbation. For a centrally symmetric atom pair, a tie under the stated threshold rule contributes exactly half its mass. Averaging nearby strict-cell bounds consequently recovers the tied mixture's risk and proves the same lower bound on the entire closed simplex.

Saved-output F1 scores. After L releases with $m = 3$ labels each, an observation of release age j has normalized weight

$$a_j = \frac{1 - \rho}{m(1 - \rho^L)} \rho^j, \qquad \rho = .98, \qquad j = 0, \dots, L - 1.$$

Each weight occurs m times. For $L \ge 500$,

$$N_{\mathrm{eff}} := \frac{1}{\sum_i a_i^2} = \frac{m(1 + \rho)}{1 - \rho} \frac{1 - \rho^L}{1 + \rho^L} > 296, \qquad a_{\max} < \frac{593}{88800}. \tag{11}$$

Uniform selection from independent observations gives independent queried pairs. A fixed finite delay only shifts their release times in a stationary regime. The feedback predictions remain those saved before their labels arrived.

Let E_{ki} indicate a wrong prediction, and let D_{ki} be $+1$ for a correct class-1 prediction, -1 for a correct class-0 prediction, and zero otherwise. Set $e_k = \sum_i a_i E_{ki}$ and $d_k = \sum_i a_i D_{ki}$. When both true classes are represented, direct substitution into macro-F1 gives the useful identity

$$\widehat{F}_k = 1 - \frac{e_k}{1 - d_k^2}. \tag{12}$$

The missing-class event has probability at most 2^{1-3L} and is charged entirely as failure probability. We conservatively use 2^{1-296} in all numerical bounds.

A reusable paired tail bound controls excessive weight on an inferior expert. Let its error rate be r_b, a better expert's rate be r_g, and their joint error rate be I_{bg}. For cutoff c, put

$$z_c = \frac{c - \gamma \pi_b}{1 - \gamma}, \qquad h_c = \frac{1}{\beta} \log \frac{\pi_b(1 - z_c)}{\pi_g z_c}, \quad (\beta, \gamma) = (32, .02).$$

Softmax odds imply $q_b > c \Rightarrow \widehat{F}_g - \widehat{F}_b < h_c$. On $|d_g| \le u$, equation (12) gives $\widehat{F}_g - \widehat{F}_b \ge e_b - e_g/(1-u^2)$. For $X = E_g/(1-u^2) - E_b$, define

$$\mu = \frac{r_g}{1-u^2} - r_b, \quad v = \frac{r_g}{(1-u^2)^2} + r_b - \frac{2I_{bg}}{1-u^2} - \mu^2, \quad M_X = \frac{1}{1-u^2} - \mu.$$

The weighted Bernstein moment bound and Hoeffding's inequality for d_g yield, when $ta_{\max}M_X < 3$,

$$\Pr(q_b > c) \le \exp\left[th_c + t\mu + \frac{t^2 v}{2N_{\text{eff}}(1 - ta_{\max}M_X/3)}\right] + 2e^{-N_{\text{eff}}u^2/2} + 2^{1-296}. \tag{13}$$

Joint errors enter through I_{bg}: the expert scores are not assumed independent. In A, taking the source as b and the weak residual as g, $(r_b, r_g, I_{bg}) = (.75, .10, 0)$, $c = .002$, $t = 100$, and $u = .458$ gives $\Pr(q_0 > .002) < 3.900 \times 10^{-11}$.

A: feedback retains complementary corrections. If $q_0 \le .002$, at most one of A's two residual-disagreement pairs is wrong, so risk is at most .10. If additionally $|\widehat{F}_1 - \widehat{F}_2| \le .0077$, every atom is correct. To verify the latter, the equal residual priors imply that their relative weight lies between $\sigma(-32 \cdot .0077)$ and $\sigma(32 \cdot .0077)$; the share term moves this ratio toward $1/2$. Checking the resulting probability rectangle gives a minimum margin above $1/2$ exceeding 5.795×10^{-5}.

The probability of this useful score agreement must account for the correlation of the two residuals. Put $S = e_2 - e_1$, $U = d_2 - d_1$, $V = d_2 + d_1$; then

$$\widehat{F}_1 - \widehat{F}_2 = \frac{S}{1-d_1^2} + \frac{e_2 UV}{(1-d_1^2)(1-d_2^2)}.$$

For $s_\ell = \sum_i a_i^\ell$, expanding the six factors under the joint atom law gives

$$\begin{aligned}\mathbb{E}[e_2^2 U^2 V^2] &= .0612 s_2^3 + .0068 s_2^2 + .0448 s_2 s_3 + .136 s_2 s_4 + .0016 s_3^2 \\ &\quad - .0048 s_4 - .0288 s_5 - .1168 s_6 \\ &\le \frac{.0612}{N^3} + \frac{.0068}{N^2} + \frac{.0448 a_{\max}}{N^2} + \frac{.1376 a_{\max}^2}{N^2} =: M, \qquad N = 296.\end{aligned}$$

The law used in this expansion has error-pair probabilities $P(0,0) = .8$, $P(1,0) = P(0,1) = .1$; conditional signs are symmetric. The supplement independently enumerates the moment with rational arithmetic. For $u = .23$, $\zeta = .0017$, and $r_* = .0077(1-u^2) - \zeta/(1-u^2)$, Markov's and Hoeffding's inequalities give

$$\Pr(|\widehat{F}_1 - \widehat{F}_2| \le .0077) \ge \Pr(|S| \le r_*) - \frac{M}{\zeta^2} - 4e^{-Nu^2/2} - 2^{1-296} > .1350564938. \tag{14}$$

The last step uses the finite-convolution certificate described below. Consequently, uniformly for $L \ge 500$,

$$\mathbb{E}R_A\Big(\sum_k q_k p_k^A\Big) \le .10\big[1 - \Pr(|\widehat{F}_1 - \widehat{F}_2| \le .0077)\big] + \Pr(q_0 > .002) < .086494351. \tag{15}$$

B: feedback restores the unchanged source. The probability region $q_{\text{weak}} \le .10$, $q_{\text{strong}} \le .01$ reproduces every source decision, by checking the rectangle's vertices. The weak residual and source have opposite predictions at every atom, so $e_{\text{weak}} = 1 - e_0$. On $|d_0| \le u$,

$$\widehat{F}_0 - \widehat{F}_{\text{weak}} \ge 1 - e_0 \frac{2-u^2}{1-u^2}.$$

Using $u = .2332$ and the odds threshold for weak weight .10, a violation requires $e_0 > .4248026233$, except on imbalance or missing-class events. The source-error convolution below gives

$$\Pr(q_{\text{weak}} > .10) < .004583888564.$$

For the strong residual, equation (13) with $(r_b, r_g, I_{bg}) = (1, .35, .35)$, $c = .01$, $t = 113$, and $u = .398$ gives $\Pr(q_{\text{strong}} > .01) < 3.553 \times 10^{-10}$. It follows that

$$\begin{aligned}\mathbb{E}R_B\Big(\sum_k q_k p_k^B\Big) &\le .35 + .65 \Pr(q_{\text{weak}} > .10 \text{ or } q_{\text{strong}} > .01) < .352979528, \\ \overline{R}_{\text{route}} &< (.086494351 + .352979528)/2 < .219737 < .225 < .30.\end{aligned} \tag{16}$$

This proves the stationary population comparison used in Theorem 5.2.

Positive-convolution certificate and rounding error. For $H = 500$, round the exact weights $a_j = (1-\rho)\rho^j/[3(1-\rho^H)]$ to integers k_j on a lattice with spacing $h = 2\times10^{-7}$. Interval arithmetic verifies

$$3\sum_{j=0}^{H-1}|a_j - hk_j| < .00007357.$$

In A, the paired difference from three labels has integer increments $-3,\dots,3$ with exact probabilities $(1,24,195,560,195,24,1)/1000$. In B, the source error count is binomial with three trials and parameter .35, with probabilities $(2197,3549,1911,343)/8000$. Convolve at each lattice step k_j; for B, subtract one from each release count and restore the deterministic offset $h\sum_j k_j$ at the end. Arrays retain the interval $[-.15,.15]$ with 1,500,001 entries. Paths leaving it are discarded. All coefficients are positive, so a retained interval mass or CDF is a lower probability bound, and one minus that CDF includes all discarded mass in the upper tail.

With binary64 unit roundoff $u_f = 2^{-53}$, each stage has at most seven coefficient products and additions. A conservative $20u_f$ allowance per stage covers their error, including coefficient representation. The substochastic positive operators have ℓ_1 norm at most one. Thus recurrence error, final summation error, and all possible underflow are bounded by

$$[(1+20u_f)^H - 1] + \frac{nu_f}{1-nu_f}(1+20u_f)^H + 10^{-300} < 1.677\times10^{-10}, \qquad n = 1{,}500{,}001.$$

This is the standard absolute-error argument for positive summation (Higham, 1993). Every reported lower probability instead subtracts 10^{-8}. Short-path recurrences, including truncation, are also checked against exact rational enumeration. The array hashes and all numeric enclosures are recorded by the verification driver.

To cover every $L \ge H$, decompose the statistic into a recent H-release term multiplied by a number in $(0,1]$, plus older feedback. For the older observation weights,

$$a_o \le \frac{(1-\rho)\rho^H}{3(1-\rho^H)}, \qquad s_{2,o} \le \frac{(1-\rho)^2\rho^{2H}}{3(1-\rho^H)^2(1-\rho^2)}.$$

Bernstein's inequality at cutoff 10^{-5} bounds the old centered paired difference by 4.820×10^{-11} (two-sided), and the old centered source error by 2.891×10^{-12} (upper tail). For A, shrinking the recent lattice radius by the rounding and old-tail cutoffs leaves radius $27068h = .0054136$. Its retained probability, after all allowances, exceeds .1655257112; substituting into (14) gives the stated .1350564938 bound. Scaling toward zero preserves this symmetric interval. For B, the retained recent CDF below the shrunken threshold exceeds .9960552275; adding imbalance, missing-class, and old-tail bounds gives the stated weak-weight tail. The threshold is above the mean .35, so contracting the recent centered statistic also preserves the required upper bound. This stationary argument does not presume that 500 releases erase arbitrary pre-switch confusion counts.

Learning on the same labels. The remaining argument transfers the population certificate to candidates trained from the released labels. It has three parts: consistency of the anchored fits, continuity of the routing scores, and preservation of the certified decision regions.

Let $\widehat{w}_\lambda$ minimize the empirical anchored objective on n current-regime labels. Strong convexity with modulus $\lambda\lambda_{\min}(A)$ gives

$$\|\widehat{w}_\lambda - w^r_\lambda\| \le \frac{\|\nabla\widehat{J}_\lambda(w^r_\lambda) - \nabla J_\lambda(w^r_\lambda)\|}{\lambda\lambda_{\min}(A)}.$$

If each of the ten signed-atom frequencies differs from its probability by at most a, then

$$\|\widehat{w}_\lambda - w^r_\lambda\| \le aC_{\lambda,r}, \qquad C_{\lambda,r} = \frac{\sum_{(x,y)\in\mathrm{supp}(P_r)}\|x(\sigma((w^r_\lambda)^\top x) - y)\|}{\lambda\lambda_{\min}(A)}. \tag{17}$$

Hoeffding's inequality and a union bound give frequency-failure probability at most $20e^{-2na^2}$. The same frequency event controls both strengths. All population logits have strict signs. For sufficiently small a, both learned heads therefore have the population hard predictions, and $\sigma' \le 1/4$ also bounds their probability perturbations uniformly on the finite support.

Couple learned and population routers on the same queried inputs and labels. For a fixed recent horizon H, a union bound over the H saved-output batches and the current fit bounds the probability of any inaccurate head by $20(H+1)e^{-2n_{\min}a^2}$. Here $n_{\min}$ is the smallest training sample size among those fits. Overlap between their training sets does not require independence. On the complementary event, every recent saved hard prediction agrees with its population counterpart.

It remains to control F1 ratios when old feedback differs. Normalize each confusion matrix to total mass one. After at least H releases, the normalized mass older than H is at most $\rho^H/(1-\rho^L)$, where $L \geq H$ counts current-regime releases. For a fixed $J \leq H$, both true classes occur in the last J releases except on an event of probability 2^{1-3J}. On this event's complement, each true-class mass is at least

$$c_J = \frac{(1-\rho)\rho^{J-1}}{3} > 0.$$

Every F1 denominator, true-class mass plus predicted-class mass, is therefore at least c_J. The F1 map is uniformly continuous on this compact set, as is the softmax-plus-prior routing map. First choose J to make the missing-class probability small. Then increase H to make the old-mass perturbation small relative to c_J. Finally increase $n_{\min}$ to control all recent head fits. This order avoids assuming continuity at zero denominators or independence between training and evaluation labels.

The population certificate uses strict probability margins for the good routing rectangles and strict affine implications for all fixed weights. Sufficiently small perturbations preserve these conclusions. For the fixed-pool lower bound, the implications hold uniformly on the closed simplex, with the same symmetric-pair tie argument as above. Consequently, for every $\epsilon > 0$, the choices above make the expected routed-risk upper bound at most $U + \epsilon$, with $U = .219737$, and the equal-regime fixed-pool lower bound at least $.30 - \epsilon$. Failure events are charged at loss one. These bounds retain the dependency structure of the shared label stream. They establish the stationary consistent-learning extension.

Finite replay and alternating regimes. Fit both residuals at each release by minimizing the empirical anchored objective on the same most recent W labels. Their updates are independent of mixture weights. Alternate A and B in equal segments of D batches, with independent observations within each segment. Each batch queries $m = 3$ labels uniformly after prediction. The batch size need only exceed three, so the queried fraction is arbitrarily small without changing the label-count argument.

Fix $\epsilon > 0$. Choose J and then a finite horizon H as above, so missing-class events and older confusion mass fit within the error allowance. Then choose finite W large enough to control the empirical-head error for the current prediction and the H recent saved-output batches. To see this explicitly, for any atom-frequency tolerance $a > 0$, the probability of a bad fit among these batches is at most $20(H+1)\exp(-2Wa^2)$. Choose a small enough to preserve the hard-prediction and probability margins, then choose W to make this failure contribution small. Overlapping replay windows do not invalidate the union bound. Combined with the previous coupling and the strict certificate slack, these choices give risk perturbation at most ϵ away from segment boundaries, uniformly in time.

A sufficient boundary allowance is

$$B_0 = \lceil W/m \rceil + H + 2d. \tag{18}$$

After $\lceil W/m \rceil$ queried batches and a delay, replay contains only current-regime labels. The next H batches supply predictions from these updated heads. A second delay supplies their evaluation labels. This count is conservative under the before-batch release convention for $d \geq 1$. For $d = 0$, labels follow prediction and affect the next batch. In both cases, predictions after the first B_0 batches satisfy the coupling. Older routing mass is bounded by discounting, without a router reset. The term $2d$ is a sufficient learning-then-evaluation allowance, not an information-theoretic lower bound on every adaptation algorithm.

Let $\mathcal{R}_T$ denote expected classification risk averaged over T batches as in Section 3. The two regimes occupy equal fractions of the long-run stream. Bounding boundary losses by one gives

$$\limsup_{T\to\infty} \mathcal{R}_T^{\text{route}} \leq U + e_D, \qquad U = .219737, \qquad e_D = \epsilon + B_0/D, \tag{19}$$

$$\inf_{q\in\Delta_3} \liminf_{T\to\infty} \mathcal{R}_T^q \geq R^*_{\text{fixed}} - e_D, \qquad R^*_{\text{fixed}} = .30. \tag{20}$$

The second inequality uses the uniform fixed-mixture margin argument above on the interior batches and loses at most B_0/D at boundaries. All fixed allocations use the same learned head trajectories because residual updates do not depend on mixture weights. The scalar population oracle has risk $R^*_{\text{scalar}} = .225$, including at regime boundaries since its candidates are already optimized for the new regime. Therefore

$$e_D < \min\{R^*_{\text{scalar}} - U, (R^*_{\text{fixed}} - U)/2\}$$

ensures strict improvement over both comparators. Choose ϵ below this positive gap, then finite W, H as above, and finally choose finite D large enough. This proves Theorem 5.2. The result gives existence of a finite replay size and regime duration. It does not provide a sharp training sample bound for the experimental optimizer or establish the same advantage for arbitrary shifts.

Scope. The positive anchoring floor is essential. The comparison excludes an unrestricted zero-anchor Bayes solution, input-dependent anchor selection, and joint mixture training. The population certificate concerns the specified source and two regimes. The learning extension requires empirical anchored minimizers and sufficient regime duration. Its role is to establish an allocation advantage after candidate optimization, rather than optimality of the deployed neural models.

A.3 Synthetic checks of feedback and allocation

Feedback availability. We simulate the two-rule experiment in Theorem 5.1, with changed-region mass $\alpha = .02$. The learner knows the two possible rules and uses a fair prediction on their disagreement region until an informative label arrives. Each condition uses 100,000 query streams of 256 batches. Query counts are $m \in \{1, 3, 8\}$ and delays are $d \in \{0, 2, 16, 64\}$. The simulation samples queried inputs and tracks when feedback identifies the rule. Conditional prediction risk is then $\alpha/2$ before identification and zero afterward. Simulated means match (5) within Monte Carlo variation (Figure 4, top). This checks the information and timing calculation for a known two-rule experiment, rather than the learnability of an unknown conditional.

Correction-set learning and allocation. We simulate the exact ten-atom construction in Appendix A.2. Both residuals minimize empirical anchored cross-entropy on the same finite replay window. The router scores saved pre-feedback predictions and discounts confusion counts by .98 per release. Predictions precede label queries. No model or router resets at regime boundaries. Every emitted prediction contributes to the reported error, including startup and transitions. The evaluator integrates error over all ten current-regime atoms, avoiding evaluation-sample noise.

The primary sweep uses four alternating regimes, $m = 3$ queried labels per batch, $W = 1536$ replay labels, regime lengths $D \in \{128, 512, 2048, 8192\}$, and delays $d \in \{0, 16, 64\}$. Each condition uses 24 independent seeds. Queried examples follow the same law as a uniform subset of a 64-example i.i.d. batch. The source, individual corrections, fixed pools, and adaptive pool share candidate trajectories and released labels. A separate diagnostic supplies population-optimal heads to the router, isolating the cost of learning the candidates. This diagnostic has access to the current regime's optimized heads.

At delay 16, routed error decreases from .491 to .217 as D grows from 128 to 8192. The latter is below both population oracle values, .225 for per-regime anchor tuning and .300 for fixed pooling. Applying the population-optimal fixed weights $(.9, .1, 0)$ to the learned heads gives .360 at $D = 8192$. These weights are not claimed optimal for finite learned trajectories. With population heads, routing gives .214, separating the remaining training cost from allocation. Short regimes expose the feedback constraint: at $D = 128, d = 64$, routed error is .642, above the source's .550.

Additional checks and interpretation. At $D = 2048, d = 16$, increasing queries from one to three to eight per batch reduces routed error from .259 to .233 to .225. Replay also has a tradeoff. At this duration, windows of 384, 1536, and 6144 labels give errors of .234, .233, and .240. At $D = 8192$, they give .222, .217, and .218. A larger window improves estimation but retains old-regime labels longer. All 18 conditions and 432 runs are retained in the supplement. These results support the predicted role of informative feedback and regime duration. They do not establish the exact coefficient of the theorem's conservative $2d/D$ allowance or a guarantee outside the construction.

The reproducible script, per-seed results, and plots are in `supplement/feedback_simulation`. Independent optimization checks match the

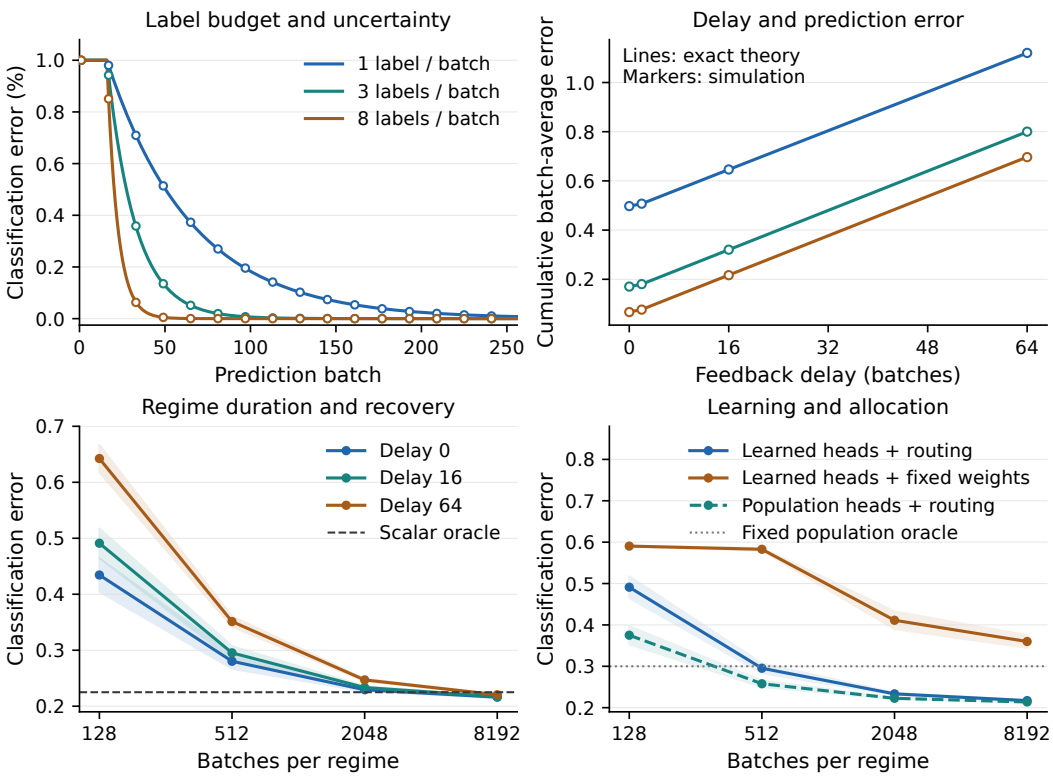


Figure 4: Synthetic checks of Theorems 5.1 and 5.2. Top: exact theory (lines) and simulated means (markers), with delay 16 on the left and cumulative risk over 256 batches on the right. Bottom: mean classification error across 24 seeds, with one-standard-deviation bands. The duration sweep uses three queried labels per batch and 1536 replay labels. The right panel fixes delay at 16. The scalar and fixed population oracles receive already optimized heads. Fixed weights on learned heads use the population oracle's $(.9, .1, 0)$.

candidate objectives, and every fitted objective has gradient residual below 10^{-8}. Timing checks confirm unchanged predictions before the first release and preserve the immutable source. The existing interval certificates remain the proof of the population oracle comparisons.

A.4 Sampling and pooling identities

These standard identities explain the feedback protocol and global pool. They are not separate theoretical contributions.

For a batch of n saved predictions with losses ℓ_j, a uniform query subset S of size $m > 0$ gives

$$\mathbb{E}\left[\frac{1}{m}\sum_{j\in S}\ell_j \,\middle|\, \ell_{1:n}\right] = \frac{1}{n}\sum_{j=1}^{n}\ell_j.$$

The equality follows from inclusion probability $P(j \in S) = m/n$. Conditioning fixes latent losses for analysis without exposing unqueried labels to the learner. Delay changes when this estimate is available. The result does not make discounted ratios or macro-F1 unbiased.

For global weights $q \in \Delta_M$, $\bar{p} = \sum_k q_k p_k$, and $L(p, y) = \frac{1}{2}\|p - e_y\|_2^2$, expanding the square gives

$$L(\bar{p}, y) = \sum_k q_k L(p_k, y) - \frac{1}{2}\sum_k q_k \|p_k - \bar{p}\|_2^2.$$

The cross term vanishes because $\sum_k q_k(p_k - \bar{p}) = 0$. This identity applies to common-weight global pooling. It does not imply improvement over the best candidate, a macro-F1 gain, or the same

identity for normalized class-specific weights. Theorem 5.2 provides the separate classification-risk comparison.

B Time-Varying Correction Set and Implementation

At time t, the candidate mechanisms form the correction set $\mathcal{K}_t$. Each mechanism parameterizes a different update of the predictive distribution induced by the shared frozen embedding. All label-trained states receive the same released tokens and add residual corrections to immutable source logits. The label-free feature-statistics state is evaluated on these tokens but updates only from past input batches. A token contains the frozen feature, source logits, each strategy prediction, query time, and audit identity. The image bank contains the source and five corrections. The signal bank contains the source, three label-trained corrections, and one feature-statistics correction.

DER++ replay. The signal-only DER++ candidate learns a zero-initialized residual from released-label replay while distilling logits saved when each item arrived. Its optimizer, buffer size, and loss weights appear in Table 3.

Class-balanced semantic replay. For $q_\theta(h, z^0) = z^0 + Bh + b$, a FIFO buffer retains at most 256 released audits. Replay is balanced over observed classes, with within-class recency weight proportional to $2^{-a_i/8}$. The objective combines cross-entropy, soft macro-F1, a source-anchor correction penalty, and $10^{-4}\|\theta\|_2^2$.

Deterministic recency replay. Every stored audit has normalized weight

$$\omega_i \propto 2^{-a_i/64}/n_{y_i}.$$

The deterministic weighting removes replay-sampling noise while preserving class balance and a fixed recency preference.

Full covariance. Let $v_i = [\widetilde{h}_i; 1]$, with saved standardized feature $\widetilde{h}_i$. Let P^- denote covariance before release, G the class-balanced log-loss gradient, and V the curvature-weighted design. With release lag ℓ and retention $\rho_s = .995$, the update is

$$P = P^- - P^- V^\top (I + V P^- V^\top)^{-1} V P^-, \qquad \Theta \leftarrow \Theta - \rho_s^\ell G P.$$

We truncate the correction to rank four and bound its norm. This is a local approximation, not an exact Bayesian posterior.

Delayed OSAMD-compatible correction. This image-only specialist ports sparse-feedback online adaptation to the same frozen feature interface and delay queue. A standalone version serves as a comparator. The term *port* reflects differences in source architecture and stream protocol from the original implementation.

Nonlinear audit correction. The fifth image correction is $r(h) = B\tanh(Ah) + b$ with rank 32. We initialize matrix B and bias b to zero, so the initial prediction is exactly the source prediction. Only released audits train this bounded replay state.

Feature-statistics correction from past inputs. This signal-only state computes source feature location and scale from frozen anchors. Before batch t, it transforms features using the target location and within-batch scale accumulated through $t-1$, maps the feature difference through the frozen classifier weight, and only then updates its target moments with rate .10. Componentwise scales are clamped below at 10^{-4}. Its `observe` operation is empty: audit labels score it for routing but never train it.

With source moments (μ_s, σ_s), past-target moments $(\mu_{t-1}, \sigma_{t-1})$, and componentwise operations, the correction is

$$\widetilde{h}_t = (h_t - \mu_{t-1})\sigma_s/\sigma_{t-1} + \mu_s, \qquad z_t^{\mathrm{stat}} = z_t^0 + W_0(\widetilde{h}_t - h_t).$$

The feature transform therefore changes the class distribution through the frozen classifier without changing the encoder.

Write v_t for the running within-batch variance and $(\bar{h}_t, s_t^2)$ for the current batch mean and population variance. After prediction, the running moments update as

$$\mu_t = .9\mu_{t-1} + .1\bar{h}_t, \qquad v_t = .9v_{t-1} + .1s_t^2, \qquad \sigma_t = \max\{\sqrt{v_t}, 10^{-4}\}$$

componentwise, starting from $\mu_0 = \mu_s$ and $v_0 = \sigma_s^2$. This formula uses correction strength one. These variances average within-batch spread rather than total variance across batch means.

B.1 Allocation states and update details

Global Brier evidence. Using the saved losses in Section 4.3, the router updates discounted loss and class mass at every nonempty release:

$$L_{kc,t} = \rho L_{kc,t-1} + \sum_{i \in \mathcal{A}_t} \mathbf{1}\{y_i = c\}\ell_{ki}, \qquad N_{c,t} = \rho N_{c,t-1} + \sum_{i \in \mathcal{A}_t} \mathbf{1}\{y_i = c\},$$

Here $\mathcal{A}_t$ contains labels released before predicting batch t, and $\rho = .995$. Empty releases preserve the previous state. Initialization sets $L = N = u = 0$, with $q_t = \widetilde{q}_t = \pi$ before the first release. Let $\mathcal{C}_t = \{c : N_{c,t} > 10^{-12}\}$. Averaging within each represented class before averaging across classes gives $\bar{\ell}_{k,t} = |\mathcal{C}_t|^{-1} \sum_{c \in \mathcal{C}_t} L_{kc,t}/N_{c,t}$. The global allocation is

$$q_{k,t} \propto \pi_k \exp\left[-\frac{\eta}{|\mathcal{C}_t|} \sum_{c \in \mathcal{C}_t} L_{kc,t}/N_{c,t}\right], \qquad \widetilde{q}_{k,t} = (1-\gamma)q_{k,t} + \gamma\pi_k.$$

We use $\eta = 4$, $\gamma = .01$, and positive normalized priors π. Then $\widetilde{q}_{k,t} \geq \gamma\pi_k > 0$.

Class-specific image pooling. The normalized pool and its objective appear in Section 4.3. The latest 512 released rows form the buffer. Each release takes two Adam steps with learning rate .05 and gradient norm cap 2. The penalty is $\lambda_u = .01$. Offsets start at zero. Every buffered row uses the current global weights, held fixed along with saved candidate probabilities during optimization. The implementation floors pooled probabilities at 10^{-12} before normalization. Absent classes receive no positive-label weight, while their predicted mass still enters the normalization.

Signal confusion counts. For each candidate, counts start at $C_{k,0} = 0$ and update at every nonempty release from the first released label:

$$C_{k,t}(a,b) = .98C_{k,t-1}(a,b) + \sum_{i \in \mathcal{A}_t} \mathbf{1}\{y_i = a, \widehat{y}_{ki} = b\}.$$

Empty releases preserve the state. With $\mathcal{C}_t^{\mathrm{ts}} = \{c : \sum_b C_{0,t}(c,b) > 10^{-12}\}$,

$$F_{k,t} = \frac{1}{|\mathcal{C}_t^{\mathrm{ts}}|} \sum_{c \in \mathcal{C}_t^{\mathrm{ts}}} \frac{2C_{k,t}(c,c)}{\sum_b C_{k,t}(c,b) + \sum_a C_{k,t}(a,c)}.$$

Denominators are floored at 10^{-12}. Global Brier weights operate until eight labels arrive. The macro-F1 rule in Section 4.4 then supplies global weights. Signals use no class-specific offsets.

C Evaluation Protocol and Reproducibility

Available information. Let Q_{tj} indicate whether example j in batch t is queried. Queried labels arrive after d batches. Let U collect model randomness, independent of the query sampler. After reading batch t and receiving due labels, the available information is

$$\mathcal{F}_{t^-} = \sigma(D_0, U, X_{1:t}, Q_{1:t-1}, \{Y_{ij} : Q_{ij} = 1,\ i < t,\ i + d \leq t\}).$$

Every state and prediction for B_t is $\mathcal{F}_{t^-}$-measurable. Current inputs and previously released labels are available, while current labels and query selections are not. Once emitted, predictions remain fixed for scoring. Feedback updates future predictions only.

Table 3: Complete frozen FORTA adaptation settings. *Share* denotes shrinkage toward the displayed prior, not a trainable parameter.

Quantity	Setting
Audit fraction / delay	5% signals; 15% images / two batches
Audit seeds / source seeds	404, 505, 606 / 11, 22, 33
Image loss-state discount ρ	.995 per nonempty release event
Image score / temperature / share	Brier/2; 4.0; .01
Image priors (source, semantic, audit, covariance, OSAMD, nonlinear)	(.05,.05,.15,.60,.03,.12)
Signal priors (source, semantic, audit, DER++, statistics)	(.08,.08,.24,.40,.20)
Signal router	Brier weights before 8 labels; then macro-F1 with discount .98, temperature 32, share .02
Image pool optimizer	Adam, two steps, learning rate .05
Image pool anchor / capacity	.01 / 512 audited rows
Semantic replay	AdamW; lr .01; capacity 256; batch 64; four steps; class-balanced; half-life 8; anchor .20; 10^{-4} parameter penalty
Deterministic audit replay	Cross-entropy; same optimizer, sizes, and penalties; exact recency/class weights; half-life 64; no soft macro-F1
Signal DER++	AdamW, lr .01; capacity 256; batch 64; four steps; $\alpha = .5, \beta = 1$; no source-anchor penalty
Full covariance	rank 4; retention .995; initial/process/ maximum variance .02/10^{-4}/.10; curvature .25
Covariate state bounds	momenta .90/.90; strength .25; z-cap 3; residual-norm cap 4; minimum uncertainty gain .20
OSAMD port	conservative lr .01; one pseudo step; aggressive scale .01; step cap .02; margin 1; residual-norm cap 4
Nonlinear audit correction	rank 32; AdamW lr .003; weight decay 10^{-4}; capacity 2048; batch 128; two steps; delta-norm cap 8
Feature statistics	momentum .10; strength 1.0; scale floor 10^{-4}

Prequential evaluation. Before each batch, due feedback updates the model from earlier predictions. The model then predicts the batch, followed by uniform query sampling. Each queried label scores the stored pre-feedback prediction and can affect only subsequent predictions. Metrics include queried examples because their scored predictions preceded their labels. Within each comparison, methods share source checkpoints, representations, target order, reset points, audit identities, and delays. A credit process chooses the number of queries per batch by carrying fractional rounding credit only. Starting from $c_0 = 0$, the recursion sets $m_t = \lfloor c_{t-1} + bn_t \rfloor$ and $c_t = c_{t-1} + bn_t - m_t$. This gives $\sum_{t \le T} m_t = \lfloor b \sum_{t \le T} n_t \rfloor$ (up to floating-point rounding in code). A separate seeded random generator draws the query indices. There is no confidence-based budget allocation. For $m_t > 0$, uniform within-batch sampling makes the saved sample-mean loss unbiased for that batch's mean conditional on its predictions. This does not make discounted class ratios or macro-F1 unbiased finite-sample estimators. No inverse-propensity correction is required for uniform sampling. For images, we cross three source and three audit seeds. For signals, we pair source checkpoints 11, 22, and 33 with audit seeds 404, 505, and 606.

C.1 Complete online algorithm

Algorithm 1 updates and evaluates the time-varying correction set. Algorithm 2 processes one nonempty feedback release. Both algorithms share persistent states. $\mathcal{H}$ indexes label-trained candi-

Algorithm 1 FORTA: prediction, uniform queries, and delayed release

Require: Frozen h_0, z^0, candidate bank, positive normalized prior π, budget $b \in [0, 1]$, delay $d \geq 0$.

Initialize candidate parameters θ_k, input states ξ_k, and optimizer states.
$L, N, C, u \leftarrow 0$, $n_{\text{lab}}, v \leftarrow 0$, $\alpha \leftarrow \pi$, $\mathcal{B}, \mathcal{Q} \leftarrow \varnothing$.
for batch $B_t = \{x_{tj}\}_{j=1}^{n_t}$ **do**
  Advance candidate time states to t before processing releases.
  for saved batch $(\mathcal{T}_s, Y_{\mathcal{S}_s})$ due at t, in query-time order **do**
    Remove its queued item and call $\mathrm{Feedback}(\mathcal{T}_s, Y_{\mathcal{S}_s})$.
  end for
  $H_t \leftarrow h_0(B_t)$, $Z_t^0 \leftarrow z^0(B_t)$.
  for candidate k **do**
    Evaluate its input-driven update and residual R_t^k from $(H_t, Z_t^0, \theta_k, \xi_k)$.
    $p_t^k \leftarrow \mathrm{softmax}(Z_t^0 + R_t^k)$, with $R_t^0 = 0$.
    Retain each prediction-time feedback state τ_{tj}^k.
  end for
  if images **then**
    $a_{kc} \leftarrow \mathrm{softmax}_k(\log \alpha_k + u_{kc})$.
    $\bar{p}_t(c \mid x) \leftarrow \dfrac{\sum_k a_{kc} p_t^k(c \mid x)}{\sum_v \sum_k a_{kv} p_t^k(v \mid x)}$.
  else
    $\bar{p}_t(\cdot \mid x) \leftarrow \sum_k \alpha_k p_t^k(\cdot \mid x)$.
  end if
  Emit $\widehat{y}_{tj} = \arg\max_c \bar{p}_t(c \mid x_{tj})$ for all j.
  $m_t \leftarrow \lfloor v + b n_t \rfloor$, $v \leftarrow v + b n_t - m_t$.
  Draw a uniform subset $\mathcal{S}_t \subseteq [n_t]$ of size m_t, independently of predictions.
  if $m_t > 0$ **then**
    $\mathcal{T}_t \leftarrow \{(H_{tj}, Z_{tj}^0, (p_t^k(x_{tj}), \tau_{tj}^k)_k, t, j) : j \in \mathcal{S}_t\}$.
    Queue $(t + d, \mathcal{T}_t)$ for label release.
    if $d = 0$ **then**
      Remove this queued item and call $\mathrm{Feedback}(\mathcal{T}_t, Y_{\mathcal{S}_t})$.
    end if
  end if
end for

dates, ξ_k holds input statistics, and τ_{tj}^k stores the candidate state needed when feedback arrives. L, N accumulate Brier losses and class counts. C_k is a confusion matrix. The image pool stores saved probabilities in $\mathcal{B}$ and learns offsets u. The label queue $\mathcal{Q}$ contains query identifiers and saved states, with labels supplied only at their scheduled release.

Candidate evaluation follows each mechanism's update order. Feature-statistics alignment predicts from past moments, then updates those moments with the current inputs. Full covariance uses current-batch feature statistics. The OSAMD port also performs its pseudo-label update. These steps use no current true labels. Source parameters never change. Empty query batches create no release and do not advance release-indexed discounting.

The feedback routine reads saved p_{ki} throughout scoring and offset optimization. It does not recompute past predictions from updated candidates. For image offsets, the two Adam steps use learning rate .05, fixed current global weights, and the class-balanced objective in (4). Tail_{512} retains the latest 512 released rows. Signal macro-F1 includes represented true classes, as specified in Appendix B.1. Candidate updates $\mathcal{U}_k$ include each mechanism's replay, optimizer, and saved-state rules from Appendix B. All settings appear in Table 3. A zero-delay release occurs after emission, so its label first affects the next prediction.

Algorithm 2 FORTA feedback routine: score saved outputs, allocate, then train

Require: Nonempty saved token batch $\mathcal{T}_s$, released labels $Y_{\mathcal{S}_s}$, and persistent states from Algorithm 1.

function Feedback($\mathcal{T}_s, Y_{\mathcal{S}_s}$)
  $\ell_{ki} \leftarrow \frac{1}{2}\|p_{ki} - e_{y_i}\|_2^2$, $\widehat{y}_{ki} \leftarrow \arg\max_c p_{ki}(c)$, $i \in \mathcal{S}_s$.
  $L_{kc} \leftarrow .995 L_{kc} + \sum_{i \in \mathcal{S}_s} \mathbf{1}\{y_i = c\}\ell_{ki}$, $N_c \leftarrow .995 N_c + \sum_{i \in \mathcal{S}_s} \mathbf{1}\{y_i = c\}$.
  $\mathcal{C} \leftarrow \{c : N_c > 10^{-12}\}$, $\bar{\ell}_k \leftarrow |\mathcal{C}|^{-1} \sum_{c \in \mathcal{C}} L_{kc}/N_c$.
  $\widetilde{q} \leftarrow .99\,\mathrm{softmax}(\log \pi - 4\bar{\ell}) + .01\pi$.
  $n_{\mathrm{lab}} \leftarrow n_{\mathrm{lab}} + |\mathcal{S}_s|$.
  if images **then**
    $\mathcal{B} \leftarrow \mathrm{Tail}_{512}(\mathcal{B} \,\|\, \{((p_{ki})_k, y_i)\}_{i \in \mathcal{S}_s})$, $\alpha \leftarrow \widetilde{q}$.
    $u \leftarrow \mathrm{Adam}^{(2)}(u, \nabla_u \mathcal{J}(u; \mathcal{B}, \alpha))$ using (4).
  else
    $C_k \leftarrow .98 C_k + \sum_{i \in \mathcal{S}_s} e_{y_i} e_{\widehat{y}_{ki}}^\top$, $F_k \leftarrow \mathrm{macroF1}(C_k)$.
    $\alpha \leftarrow \begin{cases} \widetilde{q}, & n_{\mathrm{lab}} < 8, \\ .98\,\mathrm{softmax}(\log \pi + 32F) + .02\pi, & n_{\mathrm{lab}} \geq 8. \end{cases}$
  end if
  for $k \in \mathcal{H}$, regardless of α_k **do**
    $\theta_k \leftarrow \mathcal{U}_k(\theta_k; \mathcal{T}_s, Y_{\mathcal{S}_s})$, including its replay or feedback update.
    Keep the source fixed and leave allocation unchanged during this training step.
  end for
end function

Source optimization and batching. We use a GroupNorm TCN for signal sources, with hidden width 64, adapter width 16, and dropout .10. WISDM, MIT-BIH, REALDISP, and Gas use, respectively, 4/6, 1/4, 54/33, and 8/6 input-channel/class counts. We optimize source models with AdamW, using learning rate 10^{-3}, weight decay 10^{-4}, square-root inverse-frequency class weights, label smoothing .03, gradient-norm cap 5, and a fixed 85/15 stratified split (seed 4901). Epoch/batch pairs are 12/128, 8/256, 50/64, and 40/128 in the same dataset order. Target stream batch sizes are 16, 64, 32, and 128.

Source-head training uses fixed image features: ImageNet ResNet-18 for PACS and OfficeHome, and OpenAI RN50 layer-4 average-pooled features for VLCS. We train each linear head with AdamW, learning rate .003, weight decay 10^{-4}, and label smoothing .02 for at most 100 epochs. Early stopping monitors validation macro-F1 with patience 10 and improvement tolerance 10^{-4}. Training batch size is 128 for PACS/VLCS and 256 for OfficeHome; target stream batch size is 32. All compared methods reuse these checkpoints and feature arrays.

Development and evaluation status. WISDM, MIT-BIH, REALDISP, Gas, PACS, and OfficeHome influenced method development. We selected the covariance-first low-feedback prior on the three-checkpoint signal development panel and one shared feature-statistics configuration on early WISDM/REALDISP/Gas groups with checkpoints 11 and 22, then evaluated it without retuning on later groups with checkpoint 33. The later-group improvement was concentrated on REALDISP, with a small decrease on WISDM and no change on Gas. We excluded MIT-BIH from this selection and recorded the VLCS evaluation protocol before accessing its feature and label arrays. A precursor evaluation on VLCS preceded the final prior revision, so VLCS is not an untouched confirmatory test. The recorded protocol fixes the reported run settings. Source-head selection uses source validation labels, but this does not make the adaptation design independent of target benchmarks that informed development. We did not use nested target-held-out selection for the full reported method. The follow-up controls use these same development benchmarks.

Natural signal streams. WISDM preserves chronology and separates source and target people. MIT-BIH separates patient records under non-paced AAMI4. REALDISP uses subject 15 valid self-placement trials; mutual placements M5–M7 are excluded. Gas preserves acquisition-batch order and places later batches in the target stream. No synthetic corruption is applied to this panel.

Image domains. PACS uses photo as source and art, cartoon, and sketch as targets. OfficeHome uses Real World as source and Art, Clipart, and Product as targets. VLCS uses Caltech101 as source and LabelMe, SUN09, and VOC2007 as targets. Feature extraction follows the settings above. Each target domain starts with a model reset and forms an ordered stream with 15% uniform delayed labels.

Dataset inclusion. We exclude near-ceiling portability streams with insufficient source degradation from the principal comparisons. The controlled scenarios are separate from the seven real benchmarks. Their adaptation methods share feedback schedules, and the switching protocol is specified in Appendix E.

Baseline fidelity. Feedback comparators use the same frozen features, queried examples, and release times. The delayed OSAMD signal row is a modality-compatible port rather than an exact reproduction of its original pipeline.

D Additional Benchmark Results

The domain-level results below show how the image gains are distributed across targets. Tables report mean performance and variability across runs.

Table 4: Complete matched delayed-feedback cross-modality comparison on VLCS. All rows receive the same 15% audits after two batches. Values are mean±SD over nine paired runs. Rank averages the three target-domain ranks; lower is better.

Method	Mean	LabelMe	SUN09	VOC2007	Worst	Rank ↓
Source-only	.4703±.0251	.3602±.0267	.4419±.0128	.6087±.0393	.3602±.0267	8.00
Matched residual	.6448±.0136	.5149±.0197	.6033±.0157	.8162±.0097	.5149±.0197	2.67
Audit replay	.6412±.0135	.5105±.0099	.6067±.0238	.8065±.0122	.5105±.0099	3.33
DER++	.6215±.0142	.5116±.0219	.5718±.0146	.7811±.0123	.5116±.0219	4.33
Full covariance	.6339±.0166	.4958±.0280	.6171±.0271	.7889±.0170	.4958±.0280	3.67
OSAMD port	.5473±.0223	.4559±.0274	.4974±.0118	.6885±.0324	.4559±.0274	6.67
Nonlinear audit	.5426±.0236	.4711±.0266	.5227±.0167	.6341±.0458	.4711±.0266	6.33
FORTA	**.6642±.0179**	**.5347±.0135**	**.6279±.0319**	**.8301±.0163**	**.5347±.0135**	**1.00**

Feedback-budget and delay sensitivity. The sensitivity study varies the label budget at a fixed two-batch delay and varies delay at the standard signal label budget. Figure 5 reports mean performance and variability over three paired runs. Additional labels improve performance with diminishing gains, while longer delays reduce the benefit of adaptation.

E Heterogeneous Switching Diagnostic

The switching study tests whether allocation responds when the best candidate changes. A frozen identity representation has dimension six, three classes, source anchors $h \sim \mathcal{N}(0, I)$, and source logits Wh. Four consecutive segments contain 7,680 examples each. Segments one and three apply the same invertible affine sensor transformation to the observed feature while retaining the latent source labels. Segment two restores the source input law but rotates the three class rules, so $P(X)$ is unchanged while $P(Y \mid X)$ changes. Segment four restores the source process. The affine segments model measurement shift rather than serving as a strict pure-covariate identifiability construction.

Every adaptive row receives the same 5% uniformly sampled audits after two batches. Ten paired runs evaluate the reported signal model. Fixed rows run one specialist through the entire sequence. The no-routing row updates the same four FORTA specialists but retains their frozen prior mixture. The switching oracle retrospectively selects among the immutable source and the same four internal FORTA specialists in each segment. This is a retrospective macro-F1 diagnostic, not a deployable

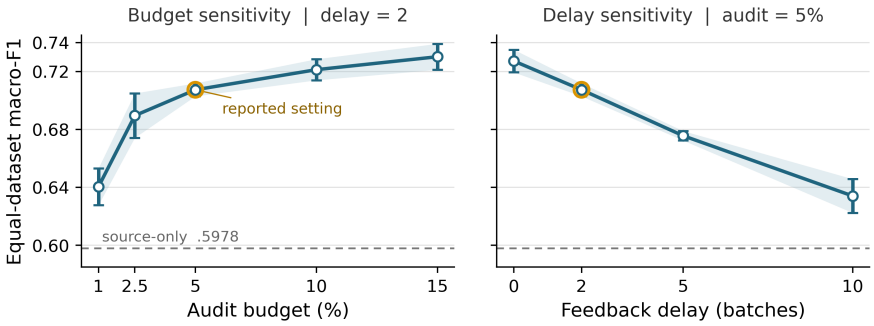


Figure 5: **Information sensitivity on four natural signal streams.** Equal-dataset macro-F1 (mean±SD, three paired seeds) versus audit budget at delay two and delay at 5% audits. Gold marks the reported setting; dashed lines mark source-only.

method. The allocation theorem concerns a separate source-anchored construction and does not bound this macro-F1 oracle gap. We retain full covariance as an external fixed comparator and exclude it from this oracle.

FORTA shifts allocation toward semantic correction after conditional change, returns to feature statistics on recurrence, but retains substantial statistics weight during recovery even though the source is best. It outperforms the strongest fixed specialist and the fixed mixture, while remaining below the retrospective switching oracle (Table 5). Its cumulative feedback loss is also lower (Figure 2). Tracking is slowest after recurrence, showing that delayed evidence can prolong an unsuitable allocation.

Covariate-shift control. We also vary $P(X)$ while keeping $P(Y \mid X)$ fixed and the source Bayes-correct. There is no accuracy gain to obtain in this setting, and feature-statistics correction can harm performance. For a separate diagnostic, we retain the source while feedback is insufficient or when its measured performance leads the corrections. It preserves source accuracy in this control, but is not part of the FORTA model used in the benchmark tables.

F Matched Allocation: Stream Protocols and Diagnostics

What is held fixed. A portfolio run saves every candidate probability before query selection. All combiners replay those same outputs, queried labels, and release times. Allocation does not affect candidate training because every supervised candidate learns from the same feedback. Fixed

Table 5: Controlled heterogeneous-switching macro-F1. Values are mean±SD across ten paired data/audit seeds. All adaptive rows receive the same 5% audits delayed by two batches. The oracle is retrospective.

Method	Measurement	Concept	Recurring	Recovery	Overall
Source-only	.4291±.0027	.0000±.0000	.4310±.0042	1.0000±.0000	.4977±.0015
Semantic	.6698±.0098	**.6919±.0182**	.6336±.0073	.8959±.0089	.7320±.0063
Audit residual	.6576±.0109	.5546±.0159	.5942±.0176	.8855±.0109	.6882±.0069
DER++	.6915±.0161	.4646±.0175	.5234±.0213	.7598±.0205	.6270±.0094
Feature statistics	**.9225±.0049**	.0404±.0043	**.9242±.0052**	.9196±.0032	.7031±.0026
Full covariance	.6476±.0112	.5965±.0150	.5817±.0233	.8644±.0093	.6891±.0051
No routing	.7047±.0091	.3869±.0121	.6338±.0171	.8697±.0086	.6623±.0052
FORTA	.9115±.0069	.6085±.0156	.7748±.0104	.9192±.0034	**.8049±.0043**
Switching oracle	.9225±.0049	.6919±.0182	.9242±.0052	**1.0000±.0000**	.8851±.0051

pools therefore retain candidate learning. Their only fixed component is the mixture weight. Every replay preserves query order within releases, including buffer eviction. The complete six-condition study contains 126 portfolio runs. Replayed FORTA outputs match the captured probabilities up to floating-point roundoff.

Input order and state retention. PACS uses photo as source and the three original target domains. The class-block condition retains file order, which places the seven classes in consecutive blocks. The shuffled condition uses label-independent random permutations within domains. Three orders cyclically rotate the domains, so each appears first once. Permutation seeds are 9101, 9102, and 9103. WISDM preserves chronology within each of 18 target participants. Its three orders rotate the participant sequence by zero, six, and twelve positions. These are constructed sequences of real target groups. The learner receives no group identities or boundary indicators.

Reset initializes candidate and router state for each group and discards pending feedback from the completed group. Continual preserves candidate, router, replay, and pending-label state across boundaries. Candidates train on the resulting stream, rather than reusing predictions from reset runs. Budgets remain 15% for PACS and 5% for WISDM, with labels delayed by two batches. The query generator and fractional credit restart per group under both protocols, preserving the same selected examples for a given order and query seed. Continual learning also receives labels crossing a boundary. Within-condition comparisons isolate allocation. Reset-versus-continual comparisons change candidate learning and retained feedback as well.

Replication and aggregation. The protocol and model settings were fixed before these reruns. These remain development datasets. Each condition crosses three source checkpoints with three query seeds. Shuffled PACS and all continual conditions also use the three stream orders, giving 27 runs each. The original reset conditions use nine runs because their aggregate scores do not depend on group order. The reset reruns reproduce all original follow-up combiner scores. WISDM's macro-F1 router begins after 176 examples in each reset participant stream, when feedback covers only two or three of six classes. About 35% of all WISDM examples precede this switch. Brier allocation operates during this phase.

PACS macro-F1 averages domains equally. WISDM macro-F1 scores concatenated participant predictions. For Figure 3, query and order runs are first averaged within each source checkpoint. The figure subtracts each comparator from FORTA within the same run, averages differences within each source checkpoint, then averages the three source differences. Multiplying by 100 gives macro-F1 points. This aggregation does not treat repeated query seeds from one source model as independent source models. Complete absolute scores and run-level metrics accompany the experiment record rather than repeating the main comparison in another table.

Combiners. Uniform and prior pools keep global weights fixed while all candidates continue learning. Delayed Hedge starts at the FORTA prior and, on each released label, uses the saved Brier/2 losses in $w_k \propto w_k \exp(-\eta \ell_k)$. Delayed fixed-share additionally replaces w by $.99w + .01\pi$ after each label. Both use $\eta = 1$, with .25 and 4 included in sensitivity analysis. These established weighting rules act at label-release time (Herbster & Warmuth, 1998; Joulani et al., 2013).

The online linear stacker applies a trainable K-class softmax layer to concatenated candidate log probabilities, initialized to the prior geometric pool. It uses class-balanced cross-entropy on the latest 512 released rows, two Adam steps per release, and a 10^{-4} mean-square parameter penalty. Learning rates are .01, .001, and .0001. Convex stacking learns one shared simplex by class-balanced NLL on the same buffer, starting at the FORTA prior, with two Adam steps at .05 and a .01 penalty on log-weight offsets. Both fit only pre-feedback candidate outputs and provide supervised combination controls inspired by stacking (Wolpert, 1992).

Allocation diagnostics. Post hoc replays vary the prior, image class offsets, router memory, score scale, prior share, offset penalty, buffer size, and optimizer steps while holding candidate predictions and feedback fixed. Uniform priors improve FORTA on shuffled PACS but reduce its WISDM scores. Removing image class offsets lowers shuffled-PACS macro-F1, while several router settings change NLL more than macro-F1. Convex stacking and fixed-share lead in some natural conditions. These diagnostics show sensitivity to the allocation rule and do not identify one combiner as uniformly best. Complete settings and outputs remain in the experiment record.

Reproduction and cost. The WISDM reset reruns reproduce the archived FORTA mean. The class-blocked PACS reset rerun gives .788 rather than the archived .794. The current stream driver reproduces the new predictions exactly, but the archived run lacks per-example outputs needed to localize the difference. Allocation claims therefore use the new within-run controls. The discrepancy remains unresolved. Cached-feature timing, saved probabilities, source and query identities, environment checksums, and run-level metrics accompany the experiment record.

G Broader Impact

Sparse audits reduce labeling effort but cannot undo errors emitted before feedback. Queued features, logits, and labels require access control and deletion. High-stakes use requires human oversight; these benchmarks establish neither privacy nor clinical safety.

AI Use Statement

Generative AI tools assisted with writing polish, code organization, and research. The authors are responsible for verifying the AI-assisted content and for the final manuscript and results.