Learning to Defer in Non-Stationary Time Series via Switching State-Space Models

Anonymous Authors[1]

Abstract

We study sequential expert routing in non-stationary time series with censored (bandit-style) feedback and time-varying expert availability: at each round, the router observes the target but only the queried expert's prediction. We model signed expert residuals with a factorized switching linear-Gaussian state-space model comprising a context-dependent regime process, a shared global factor, and per-expert idiosyncratic states. To scale inference to large and evolving expert registries, we derive an IMM-style filter with factorized updates that maintains per-expert marginals, supports expert entry and pruning, and jointly updates only the queried expert and the shared factor. Using one-step-ahead predictive beliefs, we apply an information-directed routing rule that trades off predicted cost against information gain about the latent regime and shared factor. We show experimentally that our framework outperforms both contextual bandits and adapted offline learning-to-defer methods.

1. Introduction

Learning-to-defer (L2D) studies decision systems that *route* each query to one of several experts and incur expert-dependent *consultation costs* (Madras et al., 2018; Mozannar & Sontag, 2020; Narasimhan et al., 2022; Mao et al., 2023; Montreuil et al., 2025a). Most L2D work is studied in an *offline* regime: a routing policy is learned from a fixed dataset, typically under i.i.d. assumptions, and training often relies on supervision that is unavailable online, such as access to *all* experts' predictions or losses for the same input.

In sequential problems, decisions and observations are interleaved over time and the offline assumptions above become impractical. At round t, the router observes a context $\mathbf{x}_t$ and a set of available experts $\mathcal{E}_t$, selects an expert $I_t \in \mathcal{E}_t$, and then observes the target $\mathbf{y}_t$ together with the queried prediction $\widehat{\mathbf{y}}_{t,I_t}$. Feedback is *censored*: the predictions of unqueried experts remain unobserved. Moreover, the stream is *non-i.i.d.* and often non-stationary (Hamilton, 2020; Sezer et al., 2020), so expert capability and cross-expert dependence can drift or switch regimes over time. The expert pool can also change, with experts becoming unavailable or newly arriving, and in operational settings experts may be scarce resources that must be allocated under availability constraints. These features make a direct transfer of offline L2D formulations insufficient and motivate online methods that explicitly reason over time, uncertainty, and resource constraints.

To address these challenges, we develop a probabilistic routing framework for non-stationary time series under censored feedback and a dynamic expert pool. We model expert residuals with a switching linear-Gaussian state-space (Ghahramani & Hinton, 2000; Linderman et al., 2016; Hu et al., 2024) model that couples a shared global factor with expert-specific idiosyncratic states and a discrete regime process, enabling time-varying cross-expert dependence. Faithful to practical settings, we support adding or removing experts without affecting the maintained marginals of retained experts. We also propose an exploration rule based on the IDS framework (Russo & Van Roy, 2014) that trades off predictive cost and information gain about latent states and regimes.

2. Related Work

L2D extends selective prediction (Chow, 1970; Bartlett & Wegkamp, 2008; Cortes et al., 2016; Geifman & El-Yaniv, 2017; Cao et al., 2022; Cortes et al., 2024) by allowing a learner to defer uncertain inputs to external experts (Madras et al., 2018; Mozannar & Sontag, 2020; Verma et al., 2022). An important line of work develops surrogate losses and statistical guarantees (Mozannar & Sontag, 2020; Verma et al., 2022; Cao et al., 2024; Mozannar et al., 2023; Mao et al., 2024b; 2025; Charusaie et al., 2022; Mao et al., 2024a; Wei et al., 2024). L2D has also been extended to regression

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

and multi-task settings and applied in real systems (Mao et al., 2024c; Strong et al., 2024; Palomba et al., 2025; Montreuil et al., 2025b;c). Missing expert predictions have been studied in offline/batch learning (Nguyen et al., 2025). Sequential L2D has been studied in a different setting: Joshi et al. (2021) formulate deferral in a non-stationary MDP and learn a *pre-emptive* deferral policy by comparing the long-term value of deferring now versus delaying deferral.

In contrast, we study time-series expert routing where the router selects among available experts *online* under censored (bandit-style) feedback, with potentially non-stationary data and a time-varying expert pool. We are not aware of existing L2D formulations that jointly address non-stationarity, censored observations, and dynamic expert availability.

3. Background

3.1. Offline Learning-to-Defer

We briefly recall the standard *offline* learning-to-defer (L2D) setup (Madras et al., 2018; Mozannar & Sontag, 2020; Narasimhan et al., 2022; Mao et al., 2024c). In its simplest form, one observes i.i.d. samples $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$, where $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ and $\mathbf{y} \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$. There is a fixed registry $\mathcal{K} = \{1, \dots, K\}$ of experts (or predictors), each providing a prediction $\widehat{\mathbf{y}}_k(\mathbf{x}) \in \mathcal{Y}$ when queried. Given a per-expert consultation fee $\beta_k \geq 0$ and a loss on the prediction error $\psi : \mathbb{R}^{d_y} \to \mathbb{R}_{\geq 0}$, the incurred cost of routing $(\mathbf{x}, \mathbf{y})$ to expert k is

$$C_k(\mathbf{x}, \mathbf{y}) := \psi\big(\widehat{\mathbf{y}}_k(\mathbf{x}) - \mathbf{y}\big) + \beta_k. \tag{1}$$

A router is a policy $\pi : \mathcal{X} \to \Delta^{K-1}$ mapping each input to a distribution over experts. Its population objective is the expected routing cost

$$\mathcal{R}(\pi) := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\sum_{k=1}^{K} \pi(k \mid \mathbf{x})\, C_k(\mathbf{x}, \mathbf{y})\right]. \tag{2}$$

Conditioned on $\mathbf{x}$, the Bayes-optimal deterministic router selects

$$k^\star(\mathbf{x}) \in \arg\min_{k\in\mathcal{K}} \mathbb{E}\left[C_k(\mathbf{x}, \mathbf{y}) \mid \mathbf{x}\right]. \tag{3}$$

If the router selects expert $I \in \mathcal{K}$ on input $\mathbf{x}$ with outcome $\mathbf{y}$, the incurred cost is $C_I(\mathbf{x}, \mathbf{y})$. Thus, conditioned on $\mathbf{x}$, the Bayes-optimal deterministic router chooses the expert with the smallest conditional expected cost, as in (3).

Most prior works learn π from a fixed dataset by empirical risk minimization on a dedicated surrogate loss (Mozannar & Sontag, 2020), often assuming access to all experts' predictions $(\widehat{\mathbf{y}}_k(\mathbf{x}_i))_{k=1}^K$ (or equivalently all costs $(C_k(\mathbf{x}_i, \mathbf{y}_i))_{k=1}^K$) for each training sample. Practical algorithms parameterize π with a model (e.g., a neural network) and may use surrogates or relaxations to handle discrete routing decisions and to obtain statistical guarantees (Mozannar & Sontag, 2020; Verma et al., 2022; Mao et al., 2024c).

3.2. Non-Stationary Time Series and SSMs

The offline L2D formulation above assumes i.i.d. data under a fixed distribution $\mathcal{D}$. In time-series, the data-generating process is typically *non-stationary*: the joint law of a process need not be invariant to time shifts (Hamilton, 2020). In many learning problems with observed contexts, this manifests as *time-varying conditional laws* (concept drift), i.e., the conditional distribution of $\mathbf{y}_t$ given $\mathbf{x}_t$ can evolve with t.

State-space models (SSMs) provide a standard probabilistic representation of such non-stationarity by introducing a latent state $\mathbf{h}_t$ capturing time-varying conditions (Rabiner & Juang, 2003; Shumway, 2006). In our setting, the observation will later correspond to an expert residual. In a linear-Gaussian SSM,

$$\mathbf{h}_t = A\mathbf{h}_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0, Q), \tag{4}$$

$$r_t = C\mathbf{h}_t + v_t, \qquad v_t \sim \mathcal{N}(0, R), \tag{5}$$

and the Kalman filter (Kalman, 1960; Welch et al., 1995) yields tractable online posteriors and predictive uncertainties. Switching linear dynamical systems (SLDSs) (Bengio & Frasconi, 1994; Ghahramani & Hinton, 2000; Fox et al., 2008; Hu et al., 2024; Geadah et al., 2024) enrich this model with a discrete regime variable $z_t \in \{1, \dots, M\}$ selecting among multiple linear-Gaussian dynamics; conditioned on $z_t = m$, (A, Q, C, R) are replaced by (A_m, Q_m, C_m, R_m).

4. Context-Aware Routing in Non-Stationary Environments

4.1. Problem Formulation

Building on the offline learning-to-defer setup in Section 3.1, we study *sequential* expert routing in *non-stationary* time series under *censored feedback* (Neu et al., 2010; Dani et al., 2008).

Primitives. Time is indexed by a finite horizon $t \in [T] := \{1, \dots, T\}$. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space supporting all random variables. At each round t, the environment produces a context $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$, a target $\mathbf{y}_t \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$ with $d_y \geq 1$, and a non-empty finite set of available expert identities $\mathcal{E}_t$. We allow $\mathcal{E}_t$ to vary with t, capturing both temporary unavailability and newly arriving experts. The router maintains a time-varying *expert registry* $\mathcal{K}_t$, containing the experts for which it stores per-expert state, with $\mathcal{E}_t \subseteq \mathcal{K}_t$ at decision time. For scalability, $\mathcal{K}_t$ may discard stale experts and reinitialize them upon re-entry (details in Section 4.2.4).

Each identity $k \in \mathcal{K}_t$ corresponds to a persistent expert that, when queried at time t, outputs a prediction $\widehat{\mathbf{y}}_{t,k} \in \mathcal{Y}$.

Residuals, loss, and cost. As in (1), routing to expert k incurs a prediction error loss plus a query fee. We track experts via their signed residuals (prediction minus target). We define the *potential* residual of expert k at time t as

$$e_{t,k} := \widehat{\mathbf{y}}_{t,k} - \mathbf{y}_t. \tag{6}$$

When $I_t = k$ is queried, the realized observation is $e_t := e_{t,I_t}$. We model residuals (rather than the nonnegative loss $\psi(e_{t,k})$) because the state-space emission model is defined on $\mathbb{R}^{d_y}$, preserving signed deviations (over- vs. under-prediction) that would be lost after applying ψ. The corresponding (potential) routing cost is

$$C_{t,k} := \psi(e_{t,k}) + \beta_k. \tag{7}$$

where $\psi : \mathbb{R}^{d_y} \to \mathbb{R}_{\geq 0}$ is a known convex loss (e.g., squared error for $d_y = 1$ or squared norm $\psi(e) = \|e\|_2^2$ in general) and $\beta_k \geq 0$ is a known, expert-specific query fee. When $I_t = k$ is queried, the realized cost is $C_t := C_{t,I_t}$.

Observation model (censoring). At each round, the router selects an expert index $I_t \in \mathcal{E}_t$. Due to bandit-style feedback, it observes only the queried prediction $\widehat{\mathbf{y}}_{t,I_t}$ (and hence only e_{t,I_t} and C_{t,I_t}); for $k \in \mathcal{E}_t \setminus \{I_t\}$, $(\widehat{\mathbf{y}}_{t,k}, e_{t,k}, C_{t,k})$ remain unobserved. We denote the post-action feedback tuple by $O_t := (I_t, \widehat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$.

Filtrations and policies. Let $\mathcal{H}_t := \big((\mathbf{x}_\tau, \mathcal{E}_\tau, O_\tau)\big)_{\tau=1}^{t}$ be the interaction history up to the end of round t. Decisions are non-anticipative, i.e., made before observing O_t. We define the *decision-time* sigma-algebra as $\mathcal{F}_t := \sigma\left(\mathcal{H}_{t-1}, \mathbf{x}_t, \mathcal{E}_t\right)$.

A policy $\pi = (\pi_t)_{t=1}^{T}$ is a sequence of decision rules where $\pi_t(\cdot \mid \mathcal{F}_t)$ is an $\mathcal{F}_t$-measurable distribution over $\mathcal{E}_t$. The action is sampled as $I_t \sim \pi_t(\cdot \mid \mathcal{F}_t)$, so that $I_t \in \mathcal{E}_t$ almost surely.

Interaction protocol. The process unfolds in discrete rounds. At each time t:

1. **Decision-time revelation:** the environment reveals $(\mathbf{x}_t, \mathcal{E}_t)$, thereby determining $\mathcal{F}_t$.
2. **Action:** the router samples $I_t \sim \pi_t(\cdot \mid \mathcal{F}_t)$.
3. **Censored feedback:** the router observes $O_t = (I_t, \widehat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$ and can evaluate the realized residual e_{t,I_t} and cost C_{t,I_t}.

Non-stationarity and exogeneity. We do not assume i.i.d. data: the joint law of $(\mathbf{x}_t, \mathcal{E}_t, \mathbf{y}_t)$ may drift over time (Section 3.2). Concretely, we allow a sequence of time-varying conditional laws $\{\mathcal{D}_t\}_{t\geq 1}$ such that

$$(\mathbf{x}_t, \mathcal{E}_t, \mathbf{y}_t) \mid \sigma\big((\mathbf{x}_\tau, \mathcal{E}_\tau, \mathbf{y}_\tau)_{\tau<t}\big) \sim \mathcal{D}_t\left(\cdot | (\mathbf{x}_\tau, \mathcal{E}_\tau, \mathbf{y}_\tau)_{\tau<t}\right).$$

This captures non-stationarity (e.g., concept drift or regime shifts). We additionally assume *exogeneity*: past routing actions affect which expert predictions are observed, but do not influence the data-generating process. Equivalently, $(\mathbf{x}_t, \mathcal{E}_t, \mathbf{y}_t)$ is conditionally independent of past actions $I_{1:t-1}$ given $\sigma\big((\mathbf{x}_\tau, \mathcal{E}_\tau, \mathbf{y}_\tau)_{\tau<t}\big)$.

Objective and myopic Bayes selector. Our goal is to minimize expected cumulative routing cost

$$J(\pi) := \mathbb{E}\left[\sum_{t=1}^{T} C_{t,I_t}\right]. \tag{8}$$

As an idealized one-step benchmark, the *myopic Bayes selector* chooses

$$k_t^\star \in \arg\min_{k\in\mathcal{E}_t} \mathbb{E}\left[C_{t,k} \mid \mathcal{F}_t\right]. \tag{9}$$

Under full feedback, (9) is directly evaluable from contemporaneous observations of all experts' costs. Under censoring, however, $C_{t,k}$ is observed only for the queried expert (Neu et al., 2010), so (9) is not directly computable. Since β_k is known, evaluating (9) reduces to forecasting $\mathbb{E}[\psi(e_{t,k}) \mid \mathcal{F}_t]$ for unqueried experts. In subsequent sections, we introduce a latent-state model that yields tractable one-step-ahead predictive beliefs $p(e_{t,k} \mid \mathcal{F}_t)$.

4.2. Generative Model: Factorized Switching LDS

Section 4.1 highlights that censored feedback and non-stationarity make the myopic selector (9) intractable without a predictive belief over *unobserved* expert residuals.

We therefore model the *potential residuals* $e_{t,k} = \widehat{\mathbf{y}}_{t,k} - \mathbf{y}_t$ from Section 4.1 as emissions of a **factorized switching linear dynamical system** (Bengio & Frasconi, 1994; Linderman et al., 2016; Hu et al., 2024). The central bottleneck is censoring: at round t we observe only the queried residual $e_t := e_{t,I_t}$, while $(e_{t,k})_{k\neq I_t}$ remain counterfactual. We address this by combining (i) a *switching* latent regime z_t to capture abrupt changes, (ii) a *shared* global factor $\mathbf{g}_t$ that couples experts and enables information transfer, and (iii) *idiosyncratic* expert-specific dynamics $\mathbf{u}_{t,k}$. For scalability under a growing registry, our inference later maintains per-expert marginals via a factorized filtering approximation. The resulting linear-Gaussian structure yields Kalman-style updates and closed-form information quantities used in our routing rule.

4.2.1. Latent state hierarchy

We represent non-stationarity via a two-level hierarchy separating systemic shifts from expert-specific drifts. The hierarchy is designed so that a single queried residual can update a *shared* latent factor $\mathbf{g}_t$, which immediately refines predictions for all experts. Expert-specific states $\mathbf{u}_{t,k}$ then capture persistent idiosyncratic deviations that cannot be explained by global conditions alone.

Context-dependent regime switching. A discrete regime $z_t \in \{1, \dots, M\}$ selects the active dynamical law (e.g., "bull" vs. "crisis"). While classical SLDSs often use a time-homogeneous transition matrix, we allow transition probabilities to depend on the observed context $\mathbf{x}_t$ (input-driven switching; e.g., Bengio & Frasconi (1994)). Let $\Pi_\theta(\mathbf{x}_t) \in [0,1]^{M \times M}$ be a row-stochastic matrix with

$$\mathbb{P}(z_t = m \mid z_{t-1} = \ell, \mathbf{x}_t) = \Pi_\theta(\mathbf{x}_t)_{\ell m}.$$

This lets the filter incorporate exogenous signals in $\mathbf{x}_t$ to update its regime belief before observing the queried residual e_t. Contexts that shift mass toward regime m favor experts with low mode-m predicted cost, yielding an interpretable link between $\mathbf{x}_t$, regimes, and expert specialization.

We parameterize the logits of $\Pi_\theta(\mathbf{x}_t)$ with a low-rank scaled-attention form to control statistical and computational complexity (Vaswani et al., 2017; Kossen et al., 2021; Mehta et al., 2022). Specifically, for a chosen bottleneck dimension d_{attn}, we compute $Q_\theta(\mathbf{x}_t), K_\theta(\mathbf{x}_t) \in \mathbb{R}^{M \times d_{\text{attn}}}$ and set

$$S(\mathbf{x}_t) := \frac{1}{\sqrt{d_{\text{attn}}}} Q_\theta(\mathbf{x}_t) K_\theta(\mathbf{x}_t)^\top,$$

so that $\operatorname{rank}(S(\mathbf{x}_t)) \le d_{\text{attn}}$. Applying a row-wise softmax yields the transition matrix:

$$\mathbb{P}(z_t = m \mid z_{t-1} = \ell, \mathbf{x}_t) = \frac{\exp(S_{\ell m}(\mathbf{x}_t))}{\sum_{j=1}^{M} \exp(S_{\ell j}(\mathbf{x}_t))}. \quad (10)$$

Global factor dynamics. Under censored feedback, the only way to learn about *unqueried* experts is to exploit structure that couples them (see Proposition 2). We therefore introduce a continuous *shared* latent state $\mathbf{g}_t \in \mathbb{R}^{d_g}$ representing system-wide conditions (e.g., overall difficulty, market volatility, sensor drift) that affect many experts simultaneously. Because $\mathbf{g}_t$ appears in every expert's residual model, updating $\mathbf{g}_t$ from the single observed residual $e_t = e_{t,I_t}$ tightens the predictive beliefs for other experts $k \neq I_t$, providing the cross-expert information transfer needed for routing.

Conditioned on $z_t = m$, we model $\mathbf{g}_t$ with linear-Gaussian dynamics to retain Kalman-style updates and closed-form predictive quantities used later for exploration:

$$\mathbf{g}_t = \mathbf{A}_m^{(g)} \mathbf{g}_{t-1} + \mathbf{w}_t^{(g)}, \qquad \mathbf{w}_t^{(g)} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_m^{(g)}), \quad (11)$$

where $\mathbf{A}_m^{(g)} \in \mathbb{R}^{d_g \times d_g}$ and $\mathbf{Q}_m^{(g)} \in \mathbb{S}_{++}^{d_g}$. We assume $(\mathbf{w}_t^{(g)})_t$ are independent across time and independent of all other process and emission noise terms.

Expert-specific dynamics. Not all variation is shared: experts can drift due to recalibration, local overfitting, model updates, or intermittent failures. We capture these *idiosyncratic* effects with a per-expert latent state $\mathbf{u}_{t,k} \in \mathbb{R}^{d_\alpha}$. Conditioned on $z_t = m$,

$$\mathbf{u}_{t,k} = \mathbf{A}_m^{(u)} \mathbf{u}_{t-1,k} + \mathbf{w}_{t,k}^{(u)}, \quad \mathbf{w}_{t,k}^{(u)} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_m^{(u)}), \quad (12)$$

where conditional on (z_t), the noise terms are independent across experts and time. To maintain statistical strength under sparse feedback, we share the dynamics parameters $(\mathbf{A}_m^{(u)}, \mathbf{Q}_m^{(u)})$ across experts.

Reliability composition and residual emission. Expert heterogeneity is then expressed through (i) the expert-specific state realization $\mathbf{u}_{t,k}$ and (ii) expert-specific loadings $\mathbf{B}_k$, which determine how each expert responds to the shared factor $\mathbf{g}_t$.

Definition 1 (L2D-SLDS reliability and residual emission)**.** *Fix latent dimensions d_g and d_α and a feature map $\Phi : \mathcal{X} \to \mathbb{R}^{d_\alpha \times d_y}$. For each expert k, define its latent* reliability *vector at time t by*

$$\boldsymbol{\alpha}_{t,k} := \mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k}, \qquad \mathbf{B}_k \in \mathbb{R}^{d_\alpha \times d_g}. \quad (13)$$

Given regime $z_t = m$, context $\mathbf{x}_t$, and latent states $(\mathbf{g}_t, \mathbf{u}_{t,k})$, the signed residual $e_{t,k} = \widehat{\mathbf{y}}_{t,k} - \mathbf{y}_t$ is generated by the linear-Gaussian emission

$$e_{t,k} \mid (z_t = m, \mathbf{g}_t, \mathbf{u}_{t,k}, \mathbf{x}_t) \sim \mathcal{N}\big(\Phi(\mathbf{x}_t)^\top \boldsymbol{\alpha}_{t,k}, \mathbf{R}_{m,k}\big), \quad (14)$$

where $\mathbf{R}_{m,k} \in \mathbb{S}_{++}^{d_y}$ is an expert- and regime-specific noise covariance.

Definition 1 is the *residual emission* component of our L2D-SLDS: it makes expert performance depend on the observed context via $\Phi(\mathbf{x}_t)$ while preserving linear-Gaussian structure (hence Kalman-style updates and closed-form predictive quantities). We assume emission noise is conditionally independent across experts and time given $(z_t, \mathbf{g}_t, (\mathbf{u}_{t,k})_k)$. We assume an initial distribution $p(z_1)$ and Gaussian priors for $\mathbf{g}_0$ and $\mathbf{u}_{0,k}$; inference only requires these to be specified and known.

4.2.2. Implications of the Hierarchy

Selective information transfer via factorization. The hierarchy is constructed so that routing can generalize across experts through the shared factor $\mathbf{g}_t$, while $\mathbf{u}_{t,k}$ captures persistent expert-specific drift. In the exact Bayesian filter, conditioning on the single observed residual $e_t = e_{t,I_t}$ couples $\mathbf{g}_t$ with $(\mathbf{u}_{t,k})_k$, and hence couples experts with each other; maintaining the full joint posterior becomes prohibitive as the registry grows.

For scalability, our inference maintains a *factorized* filtering approximation: after each update, we project the belief onto a family in which (conditional on z_t) the idiosyncratic states are independent across experts and independent of $\mathbf{g}_t$; see Appendix D.3 for the corresponding non-factorized update.

This projection discards posterior cross-covariances, but preserves the mechanism needed under censoring: querying a single expert updates $\mathbf{g}_t$, which shifts the predictive residual distributions of *all* experts through $\mathbf{B}_k$. The proposition below makes the resulting information transfer criterion explicit.

Proposition 2 (Information transfer under a shared factor)**.** *Fix t and $z_t = m$, and let $\mathcal{G}_t := \sigma(\mathcal{F}_t, I_t, z_t = m)$. Let $j \neq I_t$ and let $(e^{\text{pred}}_{t,j}, e^{\text{pred}}_{t,I_t})$ denote the one-step-ahead predictive residuals under $p(e_{t,\cdot} \mid \mathcal{F}_t, z_t = m)$. Assume that this predictive pair is jointly Gaussian conditional on $\mathcal{G}_t$ and that $\mathrm{Cov}(e^{\text{pred}}_{t,I_t} \mid \mathcal{G}_t)$ is non-singular (e.g., $\mathbf{R}_{m,I_t} \succ \mathbf{0}$). Then*

$$\mathbb{E}\left[e^{\text{pred}}_{t,j} \mid e_t, \mathcal{G}_t\right] = \mathbb{E}\left[e^{\text{pred}}_{t,j} \mid \mathcal{G}_t\right] \iff \mathrm{Cov}\left(e^{\text{pred}}_{t,j}, e^{\text{pred}}_{t,I_t} \mid \mathcal{G}_t\right) = \mathbf{0}.$$

In particular, if the covariance is non-zero, then observing $e_t = e_{t,I_t}$ updates the posterior predictive mean of $e^{\text{pred}}_{t,j}$.

We prove Proposition 2 in Appendix F.1. Observing the queried residual affects unqueried experts exactly when their predictive residuals are correlated. In our factorized SLDS, this correlation is induced by the shared factor $\mathbf{g}_t$. Under the linear-Gaussian model, the predictive residuals are jointly Gaussian, and their cross-covariance can be read directly from the shared-factor channel. For example, conditional on $(\mathcal{F}_t, z_t = m)$, $\mathrm{Cov}(e^{\text{pred}}_{t,j}, e^{\text{pred}}_{t,i})$ contains the shared-factor term

$$\Phi(\mathbf{x}_t)^\top \mathbf{B}_j \Sigma^{(m)}_{g,t|t-1} \mathbf{B}_i^\top \Phi(\mathbf{x}_t),$$

where $\Sigma^{(m)}_{g,t|t-1}$ is the one-step predictive covariance of $\mathbf{g}_t$ under regime m. Thus, querying $i = I_t$ tightens expert j's predictive distribution whenever the coupling through $\mathbf{g}_t$ is non-negligible in the directions probed by $\Phi(\mathbf{x}_t)$. Conversely, if this term vanishes, then under the factorized predictive belief there is no information transfer from I_t to j at time t.

4.2.3. Exploration via Information-Directed Sampling

Under censored feedback, greedily selecting the expert with the lowest predicted cost can slow adaptation by repeatedly querying a "safe" expert. We therefore use *Information-Directed Sampling (IDS)* (Russo & Van Roy, 2014) to trade off predicted cost against information about the latent state $(z_t, \mathbf{g}_t)$.

Exploitation: predicted cost and gap. For each $k \in \mathcal{E}_t$, the model provides a one-step-ahead predictive residual $e^{\text{pred}}_{t,k} \sim p(e_{t,k} \mid \mathcal{F}_t)$ and predicted cost

$$\bar{C}^{\text{pred}}_{t,k} := \mathbb{E}\left[\psi(e^{\text{pred}}_{t,k}) \,\middle|\, \mathcal{F}_t\right] + \beta_k.$$

Let $k^{\text{pred}}_t \in \arg\min_{k \in \mathcal{E}_t} \bar{C}^{\text{pred}}_{t,k}$ be the myopic predictor. We define the predictive value gap

$$\Delta_t(k) := \bar{C}^{\text{pred}}_{t,k} - \bar{C}^{\text{pred}}_{t,k^{\text{pred}}_t} \geq 0. \tag{15}$$

Exploration: informativeness of a query. We quantify the informativeness of querying k by the mutual information between the latent state and the (hypothetical) queried residual:

$$\mathrm{IG}_t(k) := \mathcal{I}\left((z_t, \mathbf{g}_t); e^{\text{pred}}_{t,k} \,\middle|\, \mathcal{F}_t\right). \tag{16}$$

For our model, the shared-factor component admits a closed form, while the regime-identification component is estimated with a lightweight Monte Carlo routine; see Remark 5 (Appendix) and Algorithm 1.

Minimizing the information ratio. IDS selects the routing action by minimizing the squared information ratio

$$I_t \in \arg\min_{k \in \mathcal{E}_t} \frac{\Delta_t(k)^2}{\mathrm{IG}_t(k)}. \tag{17}$$

We interpret the ratio as $+\infty$ when $\mathrm{IG}_t(k) = 0$ unless $\Delta_t(k) = 0$; if all $\mathrm{IG}_t(k) = 0$, IDS reduces to the myopic choice k^{pred}_t.

4.2.4. Dynamic Registry Management

In many deployments, expert availability varies and the pool evolves over time. A static learning-to-defer router (Madras et al., 2018; Mozannar & Sontag, 2020) trained on a fixed expert catalog does not naturally support adding experts without retraining, nor dropping expert-specific components to reclaim memory/compute.

Our state-space approach makes this issue explicit: each expert k carries an idiosyncratic latent state $\mathbf{u}_{t,k}$ that must be stored and propagated for prediction. When the pool is large or long-lived, we cannot maintain $\mathbf{u}_{t,k}$ for every expert ever encountered. We therefore treat expert-specific state as a *cache* and manage it online.

Recall that $\mathcal{K}_t$ denotes the router's maintained expert registry (Section 4.1): experts for which we store per-expert filtering marginals, i.e., maintain $\mathbf{u}_{t,k}$. The registry is not cumulative: experts may be removed when stale and re-added upon re-entry, while maintaining $\mathcal{E}_t \subseteq \mathcal{K}_t$ at decision time and keeping $|\mathcal{K}_t|$ bounded.

Pruning. Let $\tau_{\text{last}}(k) \in \{0, 1, \ldots, t-1\}$ be the last round at which expert k was queried (with the convention $\tau_{\text{last}}(k) = 0$ if k has never been queried). We call an expert *stale* if it is currently unavailable and has not been queried for more than $\Delta_{\max}$ steps, where $\Delta_{\max} \geq 1$ is a user-chosen staleness horizon:

$$\mathcal{K}^{\text{stale}}_t := \{k \in \mathcal{K}_{t-1} \setminus \mathcal{E}_t : \; t - \tau_{\text{last}}(k) > \Delta_{\max}\}. \tag{18}$$

We update the registry by first adding currently available experts and then pruning stale ones:

$$\mathcal{K}_t := (\mathcal{K}_{t-1} \cup \mathcal{E}_t) \setminus \mathcal{K}_t^{\text{stale}}, \qquad \mathcal{K}_0 = \varnothing. \tag{19}$$

Since $\mathcal{K}_t^{\text{stale}} \subseteq \mathcal{K}_{t-1} \setminus \mathcal{E}_t$ by construction, (19) guarantees $\mathcal{E}_t \subseteq \mathcal{K}_t$. Operationally, pruning means we stop storing the idiosyncratic filtering marginal(s) associated with $\mathbf{u}_{t-1,k}$ (and hence do not propagate it forward) for $k \in \mathcal{K}_t^{\text{stale}}$.

Pruning does *not* alter the maintained belief over retained variables: it is exact marginalization of dropped coordinates in the filtering distribution.

Proposition 3 (Pruning does not affect retained experts)**.** *Fix time t and let $P_t \subseteq \mathcal{K}_{t-1}$ be any set of experts to be pruned. Let $q_{t-1|t-1}\big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell \in \mathcal{K}_{t-1}}\big)$ denote the (exact or approximate) filtering belief at the end of round $t-1$ conditioned on the realized history. Define the pruned belief by marginalization:*

$$q_{t-1|t-1}^{\text{pr}(P_t)}\big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell \in \mathcal{K}_{t-1} \setminus P_t}\big) := \int q_{t-1|t-1}\big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell \in \mathcal{K}_{t-1}}\big) \prod_{k \in P_t} d\mathbf{u}_{t-1,k}.$$

Then $q_{t-1|t-1}^{\text{pr}(P_t)}$ equals the marginal of $q_{t-1|t-1}$ on the retained variables. Consequently, after applying the standard SLDS time update to obtain the predictive belief at round t, the predictive distribution of $\boldsymbol{\alpha}_{t,\ell}$ and the one-step predictive law of $e_{t,\ell}^{\text{pred}}$ are identical before and after pruning, for every retained $\ell \notin P_t$.

We defer the proof to Appendix F.2. If a pruned expert later reappears, we treat it as a re-entry and reinitialize its idiosyncratic state; $\Delta_{\max}$ controls the resulting memory–accuracy trade-off.

Birth and re-entry. Let $\mathcal{E}_t^{\text{init}} := \mathcal{E}_t \setminus \mathcal{K}_{t-1}$ denote experts that *enter* the maintained registry at time t (either newly observed or re-entering after pruning). For each $j \in \mathcal{E}_t^{\text{init}}$, the filter must instantiate an idiosyncratic state $\mathbf{u}_{t,j}$ before the router can assign a calibrated predictive belief to $e_{t,j}$. We do so at the *predictive* time (before observing any residual at round t).

For each entering expert j and each regime $m \in [M]$, we assume an initialization prior

$$\mathbf{u}_{t-1,j} \mid (z_t = m) \sim \mathcal{N}\big(\mu_{\text{init},j}^{(m)}, \Sigma_{\text{init},j}^{(m)}\big). \tag{20}$$

The parameters $(\mu_{\text{init},j}^{(m)}, \Sigma_{\text{init},j}^{(m)})$ can be set from side information when available, or to a conservative default. On entry, we assume the router is provided with β_j, an emission-noise specification $\mathbf{R}_{m,j}$ (or a shared $\mathbf{R}_m$), and a loading matrix $\mathbf{B}_j$ (or a default initialization), so the expert can immediately benefit from the shared factor via $\boldsymbol{\alpha}_{t,j} = \mathbf{B}_j \mathbf{g}_t + \mathbf{u}_{t,j}$.

Proposition 4 (Coupling at birth through the shared factor)**.** *Fix time t and condition on $(\mathcal{F}_t, z_t = m)$. Under the Factorized SLDS one-step predictive belief (i.e., with $\operatorname{Cov}(\mathbf{g}_t, \mathbf{u}_{t,k} \mid \cdot) = \mathbf{0}$ and $\operatorname{Cov}(\mathbf{u}_{t,i}, \mathbf{u}_{t,j} \mid \cdot) = \mathbf{0}$ for $i \neq j$), for any experts $j \neq k$,*

$$\operatorname{Cov}\left(\boldsymbol{\alpha}_{t,j}, \boldsymbol{\alpha}_{t,k} \mid \mathcal{F}_t, z_t = m\right) = \mathbf{B}_j \Sigma_{g,t|t-1}^{(m)} \mathbf{B}_k^\top,$$

where $\Sigma_{g,t|t-1}^{(m)}$ is the regime-m one-step predictive covariance of $\mathbf{g}_t$. In particular, if the joint predictive law is Gaussian and $\mathbf{B}_j \Sigma_{g,t|t-1}^{(m)} \mathbf{B}_k^\top \neq \mathbf{0}$, then $\boldsymbol{\alpha}_{t,j}$ and $\boldsymbol{\alpha}_{t,k}$ are not independent and hence $\mathcal{I}(\boldsymbol{\alpha}_{t,j}; \boldsymbol{\alpha}_{t,k} \mid \mathcal{F}_t, z_t = m) > 0$.

We give the proof in Appendix F.3.

5. Experiments

We evaluate the proposed factorized Switching LDS router (Section 4.2) in the online learning-to-defer setting of Section 4.1, focusing on three failure modes of offline L2D: *censored (bandit) feedback*, *non-stationarity*, and *dynamic expert availability*. Throughout, at each round t the router observes $(\mathbf{x}_t, \mathcal{E}_t)$, selects $I_t \in \mathcal{E}_t$, and then observes $\mathbf{y}_t$ and the queried prediction $\widehat{\mathbf{y}}_{t,I_t}$ (hence the realized residual $e_t = e_{t,I_t}$). Unless stated otherwise we use squared loss and no query fees ($\psi(e) = \|e\|_2^2$, $\beta_k \equiv 0$). We report averages over multiple random seeds with standard errors. Following standard Learning-to-Defer evaluations, we treat each expert as a fixed prediction rule f_k and focus on learning the *router* under partial feedback (e.g., Mozannar & Sontag (2020); Narasimhan et al. (2022); Mao et al. (2024c)). We are then interested on observing the difference between using seperate experts and using the routing system as prediction rules. Additional experimental details are in Appendix G.

5.1. Synthetic: Regime-Dependent Correlation and Information Transfer

Design goal. We construct a controlled routing instance in which (i) experts are *correlated* in a regime-dependent way, so that observing one expert should update beliefs about others (information transfer; Proposition 2); and (ii) one expert temporarily disappears and re-enters, so that the maintained registry $\mathcal{K}_t$ matters (see Appendix).

Environment (regimes, target, context). We use $M = 2$ regimes and deterministic switching in blocks of length $L = 150$ over horizon $T = 3000$ such as $z_t := 1 + \left\lfloor \frac{t-1}{L} \right\rfloor \bmod 2$. The target follows a regime-dependent AR(1), and the context is the one-step lag:

$$y_t = 0.8\, y_{t-1} + d_{z_t} + \eta_t, \qquad \eta_t \sim \mathcal{N}(0, \sigma_y^2). \tag{21}$$

We set the router's context to $x_t := y_{t-1}$. The regime z_t is latent to the router: the router observes only x_t (before acting) and the single queried prediction $\hat{y}_{t,I_t}$ (after acting).

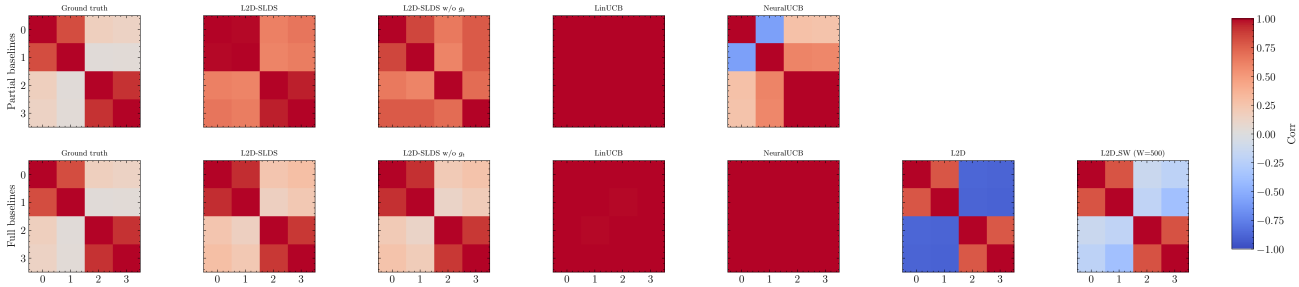

Figure 1. Regime-0 expert dependence in the synthetic transfer experiment. Each heatmap shows the pairwise Pearson correlation (color: $[-1, 1]$) between experts' per-round losses (experts indexed 0–3). Top row: partial feedback (only queried losses observed); bottom row: full feedback. Columns (left-to-right) show the ground-truth correlation implied by the simulator and the correlations estimated by each method. L2D-SLDS best recovers the block-structured correlations (experts $\{0, 1\}$ vs. $\{2, 3\}$), highlighting the benefit of modeling shared latent factors for cross-expert information transfer under censoring.

Experts. We use $K = 4$ experts indexed $k \in \{0, 1, 2, 3\}$. Expert $k = 1$ is removed from the available set $\mathcal{E}_t$ for a contiguous interval $t \in [2000, 2500]$ and then re-enters. Each expert is a one-step forecaster $\hat{y}_{t,k} = f_k(x_t)$ with a shared slope and expert-specific intercept plus noise:

$$\hat{y}_{t,k} := 0.8\, y_{t-1} + b_k + \varepsilon_{t,k}. \qquad (22)$$

We set $(b_0, b_1, b_2, b_3) = (d_1, d_1, d_2, d_2)$, so experts $\{0, 1\}$ are well-calibrated in regime $z_t = 1$ and experts $\{2, 3\}$ are well-calibrated in regime $z_t = 2$.

To induce *regime-dependent correlation* under bandit feedback, we generate the expert noises as

$$\varepsilon_{t,k} := s_{t,g(k)} + \tilde{\varepsilon}_{t,k}, \qquad g(k) := 1 + \mathbf{1}\{k \in \{2, 3\}\},$$

with independent components $s_{t,1}, s_{t,2}, (\tilde{\varepsilon}_{t,k})_k$ and regime-dependent variances $s_{t,1} \sim \mathcal{N}(0, \sigma^2_{z_t,1}), s_{t,2} \sim \mathcal{N}(0, \sigma^2_{z_t,2}), \tilde{\varepsilon}_{t,k} \sim \mathcal{N}(0, \sigma^2_{\mathrm{id}})$, where $(\sigma^2_{1,1}, \sigma^2_{1,2}) = (\sigma^2_{\mathrm{hi}}, \sigma^2_{\mathrm{lo}})$ and $(\sigma^2_{2,1}, \sigma^2_{2,2}) = (\sigma^2_{\mathrm{lo}}, \sigma^2_{\mathrm{hi}})$ with $\sigma^2_{\mathrm{hi}} \gg \sigma^2_{\mathrm{lo}}$. This makes experts $\{0, 1\}$ strongly correlated in regime 1 and experts $\{2, 3\}$ strongly correlated in regime 2. We report the MSE of each expert in Table 1.

Compared methods. We compare our **L2D-SLDS** router under bandit feedback to the following baselines. *(i) Ablation:* L2D-SLDS without the shared global factor (set $d_g = 0$). *(ii) Contextual bandits:* LinUCB (Li et al., 2010) and NeuralUCB (Zhou et al., 2020). *(iii) Full-feedback:* a full-feedback variant of L2D-SLDS and online Learning-to-Defer baselines (Mao et al., 2024c; Narasimhan et al., 2022) that assume access to all experts' predictions each round (hence are not feasible under censoring): standard L2D (Narasimhan et al., 2022; Mao et al., 2024c), and a sliding-window L2D (L2D_SW) with $W = 500$ taking the most recent data to handle non-stationarity. We use an RNN encoder (Rumelhart et al., 1985) as a drop-in context representation for methods that require learned features.

Table 1. Averaged cumulative cost (8) on experiment (Section 5.1). We report mean $\pm$ standard error across five runs. Lower is better.

Method	Partial feedback	Full feedback
L2D-SLDS	$\mathbf{13.58 \pm 0.07}$	$\mathbf{10.17 \pm 0.01}$
L2D-SLDS w/o $\mathbf{g}_t$	14.68 ± 0.01	10.18 ± 0.01
L2D	–	16.69 ± 0.25
L2D_SW ($W = 500$)	–	13.26 ± 0.11
LinUCB	22.94 ± 0.01	23.24 ± 0.01
NeuralUCB	21.92 ± 0.31	21.39 ± 1.89
Random	26.13 ± 0.25	26.13 ± 0.25
Always expert 0	23.07	–
Always expert 1	28.66	–
Always expert 2	23.05	–
Always expert 3	29.36	–
Oracle	9.04	9.04

Correlation recovery. Figure 1 compares the regime-0 loss correlation structure. The ground truth exhibits a clear block structure: experts $\{0, 1\}$ form one correlated group while experts $\{2, 3\}$ form another. Under partial feedback, L2D-SLDS is the only method that reliably recovers this clustering from partial observations, whereas removing the shared factor $\mathbf{g}_t$ blurs the separation and inflates cross-group correlations, consistent with losing cross-expert information transfer. In contrast, LinUCB/NeuralUCB yield near-degenerate correlation estimates (e.g., overly uniform or unstable patterns), reflecting that purely discriminative bandit updates do not maintain a coherent joint belief over experts' latent error processes. Under full feedback, the gap between L2D-SLDS and its ablation largely closes, as observing all experts makes explicit transfer less critical; however, the remaining baselines can still exhibit spurious structure, highlighting that modeling regime-wise coupling is beneficial beyond simply having access to more feedback.

Results. Table 1 shows that **L2D-SLDS** achieves the lowest routing cost under partial feedback (13.58 ± 0.07), improving over LinUCB/NeuralUCB by a wide margin and also outperforming the best fixed expert. Crucially, it also beats the ablation that removes the shared factor $\mathbf{g}_t$ (14.68 ± 0.01), a $\approx 7.5\%$ reduction, which directly supports our central claim: under censoring, modeling a *global* latent component enables *cross-expert information transfer* from a single queried residual (see Proposition 4). Intuitively, $\mathbf{g}_t$ captures regime-dependent common shocks that couple experts; thus, querying one expert updates beliefs about unqueried experts in a way that contextual bandits (which treat arms largely independently) and independent per-expert dynamics cannot replicate. Under full feedback, the gap between L2D-SLDS and its ablation essentially vanishes (≈ 10.17), as expected when all experts are observed and explicit transfer is no longer the bottleneck; in this setting L2D-SLDS is close to the oracle and substantially improves over full-feedback L2D and L2D_SW.

In Appendix, we provide additional experiments and study in depth this regime-dependant experiment notably by studying how our approach treat the pruning and the re-birth of experts.

6. Conclusion

7. Impact Statement

References

Bartlett, P. L. and Wegkamp, M. H. Classification with a reject option using a hinge loss. *The Journal of Machine Learning Research*, 9:1823–1840, June 2008.

Bengio, Y. and Frasconi, P. An input output hmm architecture. *Advances in neural information processing systems*, 7, 1994.

Cao, Y., Cai, T., Feng, L., Gu, L., Gu, J., An, B., Niu, G., and Sugiyama, M. Generalizing consistent multi-class classification with rejection to be compatible with arbitrary losses. *Advances in neural information processing systems*, 35:521–534, 2022.

Cao, Y., Mozannar, H., Feng, L., Wei, H., and An, B. In defense of softmax parametrization for calibrated and consistent learning to defer. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2024. Curran Associates Inc.

Charusaie, M.-A., Mozannar, H., Sontag, D., and Samadi, S. Sample efficient learning of predictors that complement humans, 2022.

Chow, C. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, January 1970. doi: 10.1109/TIT.1970.1054406.

Cortes, C., DeSalvo, G., and Mohri, M. Learning with rejection. In Ortner, R., Simon, H. U., and Zilles, S. (eds.), *Algorithmic Learning Theory*, pp. 67–82, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46379-7.

Cortes, C., Mao, A., Mohri, C., Mohri, M., and Zhong, Y. Cardinality-aware set prediction and top-k classification. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=WAT3qu737X.

Dani, V., Hayes, T. P., and Kakade, S. M. Stochastic linear optimization under bandit feedback. In *21st Annual Conference on Learning Theory*, number 101, pp. 355–366, 2008.

Fox, E., Sudderth, E., Jordan, M., and Willsky, A. Nonparametric bayesian learning of switching linear dynamical systems. *Advances in neural information processing systems*, 21, 2008.

Geadah, V., Pillow, J. W., et al. Parsing neural dynamics with infinite recurrent switching linear dynamical systems. In *The Twelfth International Conference on Learning Representations*, 2024.

Geifman, Y. and El-Yaniv, R. Selective classification for deep neural networks. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/4a8423d5e91fda00bb7e46540e2b0cf1-Paper.pdf.

Ghahramani, Z. and Hinton, G. E. Variational learning for switching state-space models. *Neural computation*, 12 (4):831–864, 2000.

Hamilton, J. D. *Time series analysis*. Princeton university press, 2020.

Hu, A., Zoltowski, D., Nair, A., Anderson, D., Duncker, L., and Linderman, S. Modeling latent neural dynamics with gaussian process switching linear dynamical systems. *Advances in Neural Information Processing Systems*, 37: 33805–33835, 2024.

Joshi, S., Parbhoo, S., and Doshi-Velez, F. Learning-to-defer for sequential medical decision-making under uncertainty. *arXiv preprint arXiv:2109.06312*, 2021.

Kalman, R. E. A new approach to linear filtering and prediction problems. 1960.

Kossen, J., Band, N., Lyle, C., Gomez, A. N., Rainforth, T., and Gal, Y. Self-attention between datapoints: Going beyond individual input-output pairs in deep learning. *Advances in Neural Information Processing Systems*, 34: 28742–28756, 2021.

Li, L., Chu, W., Langford, J., and Schapire, R. E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 661–670, 2010.

Linderman, S. W., Miller, A. C., Adams, R. P., Blei, D. M., Paninski, L., and Johnson, M. J. Recurrent switching linear dynamical systems. *arXiv preprint arXiv:1610.08466*, 2016.

Madras, D., Pitassi, T., and Zemel, R. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in neural information processing systems*, 31, 2018.

Mao, A., Mohri, C., Mohri, M., and Zhong, Y. Two-stage learning to defer with multiple experts. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=GIlsH0T4b2.

Mao, A., Mohri, M., and Zhong, Y. Principled approaches for learning to defer with multiple experts. In *ISAIM*, 2024a.

Mao, A., Mohri, M., and Zhong, Y. Realizable h-consistent and bayes-consistent loss functions for learning to defer. *Advances in neural information processing systems*, 37: 73638–73671, 2024b.

Mao, A., Mohri, M., and Zhong, Y. Regression with multi-expert deferral. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024c.

Mao, A., Mohri, M., and Zhong, Y. Mastering multiple-expert routing: Realizable h-consistency and strong guarantees for learning to defer. In *Forty-second International Conference on Machine Learning*, 2025.

Mehta, S., Székely, É., Beskow, J., and Henter, G. E. Neural hmms are all you need (for high-quality attention-free tts). In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7457–7461. IEEE, 2022.

Montreuil, Y., Carlier, A., Ng, L. X., and Ooi, W. T. Why ask one when you can ask k? two-stage learning-to-defer to the top-k experts. *arXiv preprint arXiv:2504.12988*, 2025a.

Montreuil, Y., Heng, Y. S., Carlier, A., Ng, L. X., and Ooi, W. T. A two-stage learning-to-defer approach for multi-task learning. In *Forty-second International Conference on Machine Learning*, 2025b.

Montreuil, Y., Yeo, S. H., Carlier, A., Ng, L. X., and Ooi, W. T. Optimal query allocation in extractive qa with llms: A learning-to-defer framework with theoretical guarantees. *arXiv preprint arXiv:2410.15761*, 2025c.

Mozannar, H. and Sontag, D. Consistent estimators for learning to defer to an expert. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Mozannar, H., Lang, H., Wei, D., Sattigeri, P., Das, S., and Sontag, D. A. Who should predict? exact algorithms for learning to defer to humans. In *International Conference on Artificial Intelligence and Statistics*, 2023. URL https://api.semanticscholar.org/CorpusID:255941521.

Narasimhan, H., Jitkrittum, W., Menon, A. K., Rawat, A., and Kumar, S. Post-hoc estimators for learning to defer to an expert. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 29292–29304. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/bc8f76d9caadd48f77025b1c889d2e2d-Paper-Conference.pdf.

Neu, G., Antos, A., György, A., and Szepesvári, C. Online markov decision processes under bandit feedback. *Advances in Neural Information Processing Systems*, 23, 2010.

Nguyen, C. C., Do, T.-T., and Carneiro, G. Probabilistic learning to defer: Handling missing expert annotations and controlling workload distribution. In *The Thirteenth International Conference on Learning Representations*, 2025.

Palomba, F., Pugnana, A., Alvarez, J. M., and Ruggieri, S. A causal framework for evaluating deferring systems. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025. URL https://openreview.net/forum?id=mkkFubLdNW.

Rabiner, L. and Juang, B. An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16, 2003.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning internal representations by error propagation. Technical report, 1985.

Russo, D. and Van Roy, B. Learning to optimize via information-directed sampling. *Advances in neural information processing systems*, 27, 2014.

Sezer, O. B., Gudelek, M. U., and Ozbayoglu, A. M. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied soft computing*, 90:106181, 2020.

Shumway, R. H. Time series analysis and its applications: With r examples, 2006.

Strong, J., Men, Q., and Noble, A. Towards human-AI collaboration in healthcare: Guided deferral systems with large language models. In *ICML 2024 Workshop on LLMs and Cognition*, 2024. URL https://openreview.net/forum?id=4c5rg9y4me.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Verma, R., Barrejon, D., and Nalisnick, E. Learning to defer to multiple experts: Consistent surrogate losses, confidence calibration, and conformal ensembles. In *International Conference on Artificial Intelligence and Statistics*, 2022. URL https://api.semanticscholar.org/CorpusID:253237048.

Wei, Z., Cao, Y., and Feng, L. Exploiting human-ai dependence for learning to defer. In *Forty-first International Conference on Machine Learning*, 2024.

Welch, G., Bishop, G., et al. An introduction to the kalman filter. 1995.

Zhou, D., Li, L., and Gu, Q. Neural contextual bandits with ucb-based exploration, 2020. URL https://arxiv.org/abs/1911.04462.

A. Appendix Roadmap

This appendix collects (i) implementation-ready algorithms for the router and learning routines, (ii) derivations underlying our exploration score, and (iii) proofs deferred from the main text. It is organized as follows:

- Appendix B: notation table for the main paper.
- Section D: end-to-end router/filtering pseudocode and optional learning updates.
- Appendix D.3: exact (non-factorized) Kalman update cross-covariance for the queried update.
- Section E: information-gain derivations used for IDS-style exploration.
- Appendix F.1–F.3: proofs of propositions.

B. Notation

Symbol	Meaning
Time, data, and actions	
$t \in [T]$	Round index; finite horizon T.
$\mathbf{x}_t \in \mathbb{R}^d$	Observed context at round t.
$\mathbf{y}_t \in \mathbb{R}^{d_y}$	Target/label at round t.
$\mathcal{E}_t$	Set of available experts at round t (may vary with time).
$I_t \in \mathcal{E}_t$	Queried expert at round t.
$\widehat{\mathbf{y}}_{t,k} \in \mathbb{R}^{d_y}$	Prediction of expert k at round t.
$O_t = (I_t, \widehat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$	Post-action feedback tuple at round t.
$\mathcal{H}_t$	Interaction history through the end of round t.
$\mathcal{F}_t$	Decision-time sigma-algebra (information before choosing I_t).
Residuals, costs, and objective	
$e_{t,k} = \widehat{\mathbf{y}}_{t,k} - \mathbf{y}_t$	Signed residual of expert k at time t; realized residual $e_t = e_{t,I_t}$.
$\psi(\cdot)$	Convex loss applied to residuals (e.g., $\\|\cdot\\|_2^2$).
$\beta_k \geq 0$	Expert-specific query fee.
$C_{t,k} = \psi(e_{t,k}) + \beta_k$	Routing cost for expert k; realized cost $C_t = C_{t,I_t}$.
$J(\pi) = \mathbb{E}\left[\sum_{t=1}^{T} C_{t,I_t}\right]$	Expected cumulative cost of policy π.
$k_t^\star$	Myopic Bayes benchmark minimizing $\mathbb{E}[C_{t,k} \mid \mathcal{F}_t]$ over $k \in \mathcal{E}_t$.
Latent-state model (factorized switching LDS)	
$z_t \in \{1, \dots, M\}$	Discrete latent regime at round t; M regimes.
$\Pi_\theta(\mathbf{x}_t) \in [0,1]^{M \times M}$	Context-dependent transition matrix; $\mathbb{P}(z_t = m \mid z_{t-1} = \ell, \mathbf{x}_t) = \Pi_\theta(\mathbf{x}_t)_{\ell m}$.
θ	Parameters of the context-dependent transition model $\Pi_\theta(\mathbf{x}_t)$.
d_{attn}	Bottleneck dimension in the low-rank transition-parameterization.
$\mathbf{g}_t \in \mathbb{R}^{d_g}$	Shared global latent factor coupling experts.
$\mathbf{u}_{t,k} \in \mathbb{R}^{d_\alpha}$	Expert-specific idiosyncratic latent state.
$\mathbf{A}_m^{(g)}, \mathbf{Q}_m^{(g)}$	Regime-m dynamics matrix and process noise covariance for $\mathbf{g}_t$.
$\mathbf{A}_m^{(u)}, \mathbf{Q}_m^{(u)}$	Regime-m dynamics matrix and process noise covariance for $\mathbf{u}_{t,k}$ (shared across experts).
$\Phi(\mathbf{x}_t)$	Feature map used in the residual emission mean.
$\mathbf{B}_k \in \mathbb{R}^{d_\alpha \times d_g}$	Expert-specific loading matrix coupling $\mathbf{g}_t$ into expert k's residual model.
$\boldsymbol{\alpha}_{t,k} = \mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k}$	Latent "reliability" vector of expert k at time t.

Symbol	**Meaning**
$\mathbf{R}_{m,k} \in \mathbb{S}_{++}^{d_y}$	Regime- and expert-specific emission noise covariance.
Θ	Collection of model parameters (e.g., Π_θ, $(\mathbf{A}_m^{(g)}, \mathbf{Q}_m^{(g)})_m$, $(\mathbf{A}_m^{(u)}, \mathbf{Q}_m^{(u)})_m$, $(\mathbf{B}_k)_k$, $(\mathbf{R}_{m,k})_{m,k}$).
Filtering, prediction, and routing scores	
$\bar{w}_t^{(m)} = \mathbb{P}(z_t = m \mid \mathcal{F}_t)$	Predictive (pre-observation) regime weight.
$w_t^{(m)} = \mathbb{P}(z_t = m \mid \mathcal{F}_t, I_t, e_t)$	Filtering (post-observation) regime weight.
$\gamma_t^{(m)}$	Posterior regime responsibility used in (Monte Carlo) EM.
$\xi_{t-1}^{(\ell,m)}$	Posterior transition responsibility used in (Monte Carlo) EM.
$e_{t,k}^{\text{pred}}$	One-step-ahead predictive residual random variable for expert k.
$\bar{C}_{t,k}^{\text{pred}}$	Predicted cost: $\mathbb{E}[\psi(e_{t,k}^{\text{pred}}) \mid \mathcal{F}_t] + \beta_k$.
k_t^{pred}	Myopic predicted-cost minimizer in $\mathcal{E}_t$.
$\Delta_t(k)$	Predicted cost gap relative to k_t^{pred}.
$\mathrm{IG}_t(k)$	Information gain: $\mathcal{I}((z_t, \mathbf{g}_t); e_{t,k}^{\text{pred}} \mid \mathcal{F}_t)$.
ϵ_w	Mixing floor for predictive mode weights $\bar{w}_t^{(m)}$ in IMM updates.
ϵ_{IG}	Information-gain floor used in IDS (avoids division by zero and clamps Monte Carlo noise).
S	Monte Carlo sample size used to estimate the mode-identification term in $\mathrm{IG}_t(k)$.
Dynamic registry management	
$\mathcal{K}_t$	Maintained expert registry: experts for which per-expert filtering marginals (hence $\mathbf{u}_{t,k}$) are stored.
$\mathcal{E}_t^{\text{init}} = \mathcal{E}_t \setminus \mathcal{K}_{t-1}$	Entering experts at round t (new or re-entering after pruning).
$\tau_{\text{last}}(k)$	Last round at which expert k was queried.
$\Delta_{\max}$	Staleness horizon controlling pruning.
$\mathcal{K}_t^{\text{stale}}$	Stale experts eligible for pruning.

C. L2D-SLDS Probabilistic Model

We report the complete probabilistic graphical model of our L2D-SLDS with censored feedback and context-dependent regime switching in Figure 2.

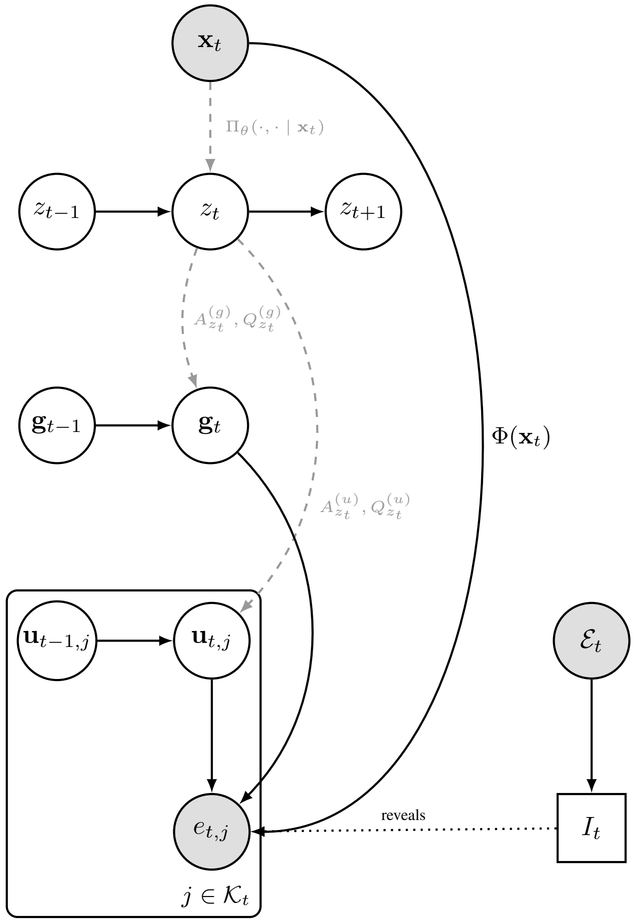

Figure 2. L2D-SLDS with bandit feedback and *context-dependent* regime switching: $p(z_t \mid z_{t-1}, \mathbf{x}_t)$. The plate $j \in \mathcal{K}_t$ indexes experts whose idiosyncratic states are stored. Each $e_{t,j}$ is a *potential* residual, but only e_{t,I_t} is revealed at round t.

D. Algorithms

D.1. Router and Filtering Recursion

Scope. This subsection provides implementation-ready pseudocode for the per-round router (Algorithm 1) and the queried update (Algorithm 2). We assume parameters Θ and an initial belief are provided (learnable via Algorithm 3).

Algorithm 1 Context-Aware Router (Factorized SLDS + IMM + IDS)

Input: horizon T; parameters Θ; feature map Φ; loss ψ; fees $(\beta_k)_k$; default entry priors $(\mu^{(m)}_{\text{init,def}}, \Sigma^{(m)}_{\text{init,def}})_{m=1}^{M}$; staleness $\Delta_{\max}$; floors $(\epsilon_w, \epsilon_{\text{IG}})$; Monte Carlo budget S for $\text{IG}_t(k)$ (Appendix E).
Initialize: $w_0^{(m)} \leftarrow \mathbb{P}(z_1 = m)$; $(\mu^{(m)}_{g,0|0}, \Sigma^{(m)}_{g,0|0})_{m=1}^{M}$; $\mathcal{K}_0 \leftarrow \varnothing$; $\tau_{\text{last}}(k) \leftarrow 0$ for all k.
for $t = 1$ to T **do**
  Observe $(\mathbf{x}_t, \mathcal{E}_t)$.
  Registry: $\mathcal{E}_t^{\text{init}} \leftarrow \mathcal{E}_t \setminus \mathcal{K}_{t-1}$.
  Registry: $\mathcal{K}_t^{\text{stale}} \leftarrow \{k \in \mathcal{K}_{t-1} \setminus \mathcal{E}_t : \ t - \tau_{\text{last}}(k) > \Delta_{\max}\}$.
  Registry: $\mathcal{K}_t \leftarrow (\mathcal{K}_{t-1} \cup \mathcal{E}_t) \setminus \mathcal{K}_t^{\text{stale}}$. {Prune stale $\mathbf{u}_{\cdot,k}$ marginals}
  For each $k \in \mathcal{E}_t^{\text{init}}$, set $(\mu^{(m)}_{\text{init},k}, \Sigma^{(m)}_{\text{init},k})_{m=1}^{M}$ (default: $(\mu^{(m)}_{\text{init,def}}, \Sigma^{(m)}_{\text{init,def}})$).
  {**IMM predictive step:** compute $\bar{w}_t^{(m)} = \mathbb{P}(z_t = m \mid \mathcal{F}_t)$ from w_{t-1} and $\Pi_\theta(\mathbf{x}_t)$ (Eq. 10), with flooring ϵ_w, and moment-match mixed priors at time $t-1$.}
  {**Time update:** apply Eqs. 11, 12, and 21 to obtain $(\mu^{(m)}_{g,t|t-1}, \Sigma^{(m)}_{g,t|t-1})$ and $(\mu^{(m)}_{u,k,t|t-1}, \Sigma^{(m)}_{u,k,t|t-1})$ for $k \in \mathcal{K}_t$.}
  For each $m \in [M]$ and $k \in \mathcal{E}_t$, compute $(\bar{e}^{\text{pred},(m)}_{t,k}, \Sigma^{\text{pred},(m)}_{t,k})$ from Eq. 14.
  For each $k \in \mathcal{E}_t$, set $\bar{C}^{\text{pred}}_{t,k} \leftarrow \sum_{m=1}^{M} \bar{w}_t^{(m)} \big(\mathbb{E}_{e \sim \mathcal{N}(\bar{e}^{\text{pred},(m)}_{t,k}, \Sigma^{\text{pred},(m)}_{t,k})}[\psi(e)] + \beta_k\big)$.
  $k_t^{\text{pred}} \in \arg\min_{k \in \mathcal{E}_t} \bar{C}^{\text{pred}}_{t,k}$; $\Delta_t(k) \leftarrow \bar{C}^{\text{pred}}_{t,k} - \bar{C}^{\text{pred}}_{t,k_t^{\text{pred}}}$ for all $k \in \mathcal{E}_t$.
  Compute $\text{IG}_t(k) = \mathcal{I}((z_t, \mathbf{g}_t); e^{\text{pred}}_{t,k} \mid \mathcal{F}_t)$ as in Appendix E; clamp $\text{IG}_t(k) \leftarrow \max(\text{IG}_t(k), \epsilon_{\text{IG}})$.
  Choose $I_t \in \arg\min_{k \in \mathcal{E}_t} \Delta_t(k)^2 / \text{IG}_t(k)$.
  Observe $(\widehat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$, set $e_t \leftarrow \widehat{\mathbf{y}}_{t,I_t} - \mathbf{y}_t$, and update $\tau_{\text{last}}(I_t) \leftarrow t$.
  Run Algorithm 2 to obtain w_t and updated posteriors for $\mathbf{g}_t$ and $(\mathbf{u}_{t,k})_{k \in \mathcal{K}_t}$.
  Optional: update Θ via Algorithm 4.
end for

Algorithm 2 CORRECT: Queried Kalman Update and Mode Posterior

Input: $\mathbf{x}_t$, queried residual e_t, queried expert I_t; predictive weights $\bar{w}_t$; predictive states $\{\mu^{(m)}_{g,t|t-1}, \Sigma^{(m)}_{g,t|t-1}\}_{m=1}^{M}$ and $\{\mu^{(m)}_{u,k,t|t-1}, \Sigma^{(m)}_{u,k,t|t-1}\}_{m \in [M],\, k \in \mathcal{K}_t}$; parameters $(\mathbf{B}_{I_t}, (\mathbf{R}_{m,I_t})_{m=1}^{M})$.
$H_t \leftarrow [\Phi(\mathbf{x}_t)^\top \mathbf{B}_{I_t} \ \ \Phi(\mathbf{x}_t)^\top]$.
for $m = 1$ to M **do**
  $\mu^{(m)}_{s,t|t-1} \leftarrow [(\mu^{(m)}_{g,t|t-1})^\top \ (\mu^{(m)}_{u,I_t,t|t-1})^\top]^\top$.
  $\Sigma^{(m)}_{s,t|t-1} \leftarrow \operatorname{diag}(\Sigma^{(m)}_{g,t|t-1}, \Sigma^{(m)}_{u,I_t,t|t-1})$.
  $\bar{e}^{\text{pred},(m)}_{t,I_t} \leftarrow H_t \mu^{(m)}_{s,t|t-1}, \quad \Sigma^{\text{pred},(m)}_{t,I_t} \leftarrow H_t \Sigma^{(m)}_{s,t|t-1} H_t^\top + \mathbf{R}_{m,I_t}$.
  $K_t^{(m)} \leftarrow \Sigma^{(m)}_{s,t|t-1} H_t^\top (\Sigma^{\text{pred},(m)}_{t,I_t})^{-1}$.
  $\mu^{(m)}_{s,t|t} \leftarrow \mu^{(m)}_{s,t|t-1} + K_t^{(m)}(e_t - \bar{e}^{\text{pred},(m)}_{t,I_t})$.
  $\Sigma^{(m)}_{s,t|t} \leftarrow \Sigma^{(m)}_{s,t|t-1} - K_t^{(m)} \Sigma^{\text{pred},(m)}_{t,I_t} (K_t^{(m)})^\top$.
  Project to factorized marginals: keep only the diagonal blocks for $\mathbf{g}_t$ and $\mathbf{u}_{t,I_t}$; set $(\mu^{(m)}_{u,k,t|t}, \Sigma^{(m)}_{u,k,t|t}) \leftarrow (\mu^{(m)}_{u,k,t|t-1}, \Sigma^{(m)}_{u,k,t|t-1})$ for $k \neq I_t$.
  $\mathcal{L}_t^{(m)} \leftarrow \mathcal{N}\big(e_t; \bar{e}^{\text{pred},(m)}_{t,I_t}, \Sigma^{\text{pred},(m)}_{t,I_t}\big)$.
end for
$w_t^{(m)} \leftarrow \frac{\mathcal{L}_t^{(m)} \bar{w}_t^{(m)}}{\sum_{\ell=1}^{M} \mathcal{L}_t^{(\ell)} \bar{w}_t^{(\ell)}}$ for all $m \in [M]$.
Return: w_t and updated posteriors $\{\mu^{(m)}_{g,t|t}, \Sigma^{(m)}_{g,t|t}\}_{m=1}^{M}$, $\{\mu^{(m)}_{u,k,t|t}, \Sigma^{(m)}_{u,k,t|t}\}_{m \in [M],\, k \in \mathcal{K}_t}$.

D.2. Parameter Learning and Online Adaptation

Scope. This subsection describes optional model-learning routines (offline initialization and sliding-window adaptation). The main router only requires a filtering belief and the learned parameters.

Algorithm 3 LEARNPARAMETERS_MCEM: Monte Carlo EM for the Factorized SLDS (windowed batch)

Input: window $\mathcal{T} = \{t_a, \ldots, t_b\}$; stream $(\mathbf{x}_t, I_t, e_t)_{t\in\mathcal{T}}$ with $e_t = \widehat{\mathbf{y}}_{t,I_t} - \mathbf{y}_t$; feature map Φ; EM iterations N_{EM}; MCMC settings $(N_{\mathrm{samp}}, N_{\mathrm{burn}})$; occupancy floor $\epsilon_N > 0$; (optional) regularization $(\lambda_\theta, \lambda_B)$ for $(\Pi_\theta, \mathbf{B})$.
$\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}} \leftarrow \{I_t : t \in \mathcal{T}\}$. {Experts queried in the window}
Initialize: parameters $\Theta^{(0)}$ and priors for z_{t_a}, $\mathbf{g}_{t_a}$, and $\{\mathbf{u}_{t_a,k}\}_{k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}}$.
for iteration $i = 1$ to N_{EM} **do**
  // E-step: Monte Carlo posterior (blocked Gibbs)
  Draw samples from $p(z_{t_a:t_b}, \mathbf{g}_{t_a:t_b}, (\mathbf{u}_{t_a:t_b,k})_{k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}} \mid (\mathbf{x}_t, I_t, e_t)_{t\in\mathcal{T}}, \Theta^{(i-1)})$ by alternating:
    1) sample $z_{t_a:t_b}$ via FFBS from the conditional HMM given $\mathbf{g}_{t_a:t_b}$ and $(\mathbf{u}_{t_a:t_b,k})_k$;
    2) sample $\mathbf{g}_{t_a:t_b}$ via Kalman smoothing given $z_{t_a:t_b}$ and $(\mathbf{u}_{t,I_t})_{t\in\mathcal{T}}$;
    3) for each $k \in \mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}$, sample $\mathbf{u}_{t_a:t_b,k}$ via Kalman smoothing using only $\{(t, e_t) : I_t = k\}$.
  From post-burn-in samples, estimate $\gamma_t^{(m)} \approx \mathbb{P}(z_t = m \mid \cdot)$, $\xi_{t-1}^{(\ell,m)} \approx \mathbb{P}(z_{t-1} = \ell, z_t = m \mid \cdot)$, and the moments used in the M-step.
  // M-step: MAP / regularized updates (factorized moments)
  Update $(\mathbf{A}^{(g)}_m, \mathbf{Q}^{(g)}_m)_{m=1}^M$ and $(\mathbf{A}^{(u)}_m, \mathbf{Q}^{(u)}_m)_{m=1}^M$ using weighted least-squares/covariance matching (skip updates when the effective count is $\le \epsilon_N$; see below).
  Update $(\mathbf{B}_k)_{k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}}$ and $(\mathbf{R}_{m,k})_{m\in[M],\,k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}}$ via weighted linear-Gaussian regression (skip updates when the effective count is $\le \epsilon_N$; see below).
  Update θ by maximizing $\sum_{t\in\mathcal{T}\setminus\{t_a\}} \sum_{\ell,m} \xi_{t-1}^{(\ell,m)} \log \Pi_\theta(\mathbf{x}_t)_{\ell m} - \frac{\lambda_\theta}{2}\|\theta\|_2^2$.
end for
Return: $\Theta^{(N_{\mathrm{EM}})}$

Implementation notes (E-step). In step 1, FFBS samples $z_{t_a:t_b}$ from the conditional distribution induced by the Markov transition $\Pi_\theta(\mathbf{x}_t)$ (Eq. 10) and the linear-Gaussian dynamics/emission terms (Eqs. 11, 12, 14) evaluated at the current $\mathbf{g}_{t_a:t_b}$ and $(\mathbf{u}_{t_a:t_b,k})_k$. In step 2, conditioned on $z_{t_a:t_b}$ and $(\mathbf{u}_{t,I_t})_t$, the observation model for $\mathbf{g}_t$ is $e_t - \Phi(\mathbf{x}_t)^\top \mathbf{u}_{t,I_t} = \Phi(\mathbf{x}_t)^\top \mathbf{B}_{I_t}\mathbf{g}_t + v_t$ with $v_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{z_t,I_t})$. In step 3, for a fixed expert k, conditioning on $z_{t_a:t_b}$ and $\mathbf{g}_{t_a:t_b}$, the observations at times $\{t : I_t = k\}$ satisfy $e_t - \Phi(\mathbf{x}_t)^\top \mathbf{B}_k \mathbf{g}_t = \Phi(\mathbf{x}_t)^\top \mathbf{u}_{t,k} + v_t$ with the same v_t.

M-step updates. Let $\langle\cdot\rangle$ denote the average over post-burn-in samples. For each regime m, define $N_m := \sum_{t=t_a+1}^{t_b} \gamma_t^{(m)}$ and the sufficient statistics

$$S^{(m)}_{gg^-} := \sum_{t=t_a+1}^{t_b} \left\langle \mathbf{1}\{z_t = m\}\mathbf{g}_t\mathbf{g}_{t-1}^\top \right\rangle, \qquad S^{(m)}_{g^-g^-} := \sum_{t=t_a+1}^{t_b} \left\langle \mathbf{1}\{z_t = m\}\mathbf{g}_{t-1}\mathbf{g}_{t-1}^\top \right\rangle.$$

If $N_m > \epsilon_N$, set $\mathbf{A}^{(g)}_m \leftarrow S^{(m)}_{gg^-}\left(S^{(m)}_{g^-g^-}\right)^{-1}$ and

$$\mathbf{Q}^{(g)}_m \leftarrow \frac{1}{N_m}\sum_{t=t_a+1}^{t_b} \left\langle \mathbf{1}\{z_t = m\}\left(\mathbf{g}_t - \mathbf{A}^{(g)}_m\mathbf{g}_{t-1}\right)\left(\mathbf{g}_t - \mathbf{A}^{(g)}_m\mathbf{g}_{t-1}\right)^\top \right\rangle.$$

Define $N^{(u)}_m := \sum_{t=t_a+1}^{t_b}\sum_{k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}} \gamma_t^{(m)}$ and

$$S^{(m)}_{uu^-} := \sum_{t=t_a+1}^{t_b}\sum_{k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}} \left\langle \mathbf{1}\{z_t = m\}\mathbf{u}_{t,k}\mathbf{u}_{t-1,k}^\top \right\rangle, \qquad S^{(m)}_{u^-u^-} := \sum_{t=t_a+1}^{t_b}\sum_{k\in\mathcal{K}^{\mathrm{qry}}_{\mathcal{T}}} \left\langle \mathbf{1}\{z_t = m\}\mathbf{u}_{t-1,k}\mathbf{u}_{t-1,k}^\top \right\rangle.$$

If $N_m^{(u)} > \epsilon_N$, set $\mathbf{A}_m^{(u)} \leftarrow S_{uu^-}^{(m)} \left(S_{u^-u^-}^{(m)}\right)^{-1}$ and

$$\mathbf{Q}_m^{(u)} \leftarrow \frac{1}{N_m^{(u)}} \sum_{t=t_a+1}^{t_b} \sum_{k \in \mathcal{K}_{\mathcal{T}}^{\mathrm{qry}}} \left\langle \mathbf{1}\{z_t = m\} \left(\mathbf{u}_{t,k} - \mathbf{A}_m^{(u)} \mathbf{u}_{t-1,k}\right) \left(\mathbf{u}_{t,k} - \mathbf{A}_m^{(u)} \mathbf{u}_{t-1,k}\right)^\top \right\rangle.$$

Emission parameters $(\mathbf{B}_k, \mathbf{R}_{m,k})$. Fix an expert $k \in \mathcal{K}_{\mathcal{T}}^{\mathrm{qry}}$ and denote $\Phi_t := \Phi(\mathbf{x}_t)$. For each $t \in \mathcal{T}$ with $I_t = k$, define the residual after removing the idiosyncratic term $y_t := e_t - \Phi_t^\top \mathbf{u}_{t,k} \in \mathbb{R}^{d_y}$ and the design matrix $X_t := (\mathbf{g}_t^\top \otimes \Phi_t^\top) \in \mathbb{R}^{d_y \times (d_g d_\alpha)}$, so that $y_t = X_t \operatorname{vec}(\mathbf{B}_k) + v_t$ with $v_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{z_t,k})$. Here $\otimes$ is the Kronecker product and $\operatorname{vec}(\cdot)$ stacks matrix columns. Given current $(\mathbf{R}_{m,k})_{m=1}^M$, a (ridge) generalized least-squares update is

$$\operatorname{vec}(\mathbf{B}_k) \leftarrow \Big(\sum_{t \in \mathcal{T}: I_t = k} \sum_{m=1}^{M} \left\langle \mathbf{1}\{z_t = m\}\, X_t^\top \mathbf{R}_{m,k}^{-1} X_t \right\rangle + \lambda_B \mathbf{I} \Big)^{-1} \Big(\sum_{t \in \mathcal{T}: I_t = k} \sum_{m=1}^{M} \left\langle \mathbf{1}\{z_t = m\}\, X_t^\top \mathbf{R}_{m,k}^{-1} y_t \right\rangle \Big).$$

For each regime m, define the effective count $N_{m,k} := \sum_{t \in \mathcal{T}: I_t = k} \gamma_t^{(m)}$. If $N_{m,k} > \epsilon_N$, update the emission covariance by weighted covariance matching:

$$\mathbf{R}_{m,k} \leftarrow \frac{1}{N_{m,k}} \sum_{t \in \mathcal{T}: I_t = k} \left\langle \mathbf{1}\{z_t = m\}\, r_{t,k} r_{t,k}^\top \right\rangle, \quad r_{t,k} := e_t - \Phi_t^\top (\mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k}).$$

Algorithm 4 ONLINEUPDATE: Sliding-Window Monte Carlo EM (non-stationary adaptation)

Input: current time t; stream $(\mathbf{x}_\tau, \mathcal{E}_\tau, I_\tau, e_\tau)_{\tau \le t}$; current parameters $\Theta^{(t-1)}$; window length W; update period K; EM iterations $N_{\mathrm{EM}}^{\mathrm{win}}$; MCMC settings; occupancy floor ϵ_N; hyperparameters as in Algorithm 3.
$\tau_0 \leftarrow t - W + 1$.
if $t < W$ **or** $t \bmod K \neq 0$ **then**
  $\Theta^{(t)} \leftarrow \Theta^{(t-1)}$ and **return**.
end if
Define window $\mathcal{T}_t \leftarrow \{\tau_0, \dots, t\}$ and $\mathcal{K}_{\mathcal{T}_t}^{\mathrm{qry}} \leftarrow \{I_\tau : \tau \in \mathcal{T}_t\}$.
Initialize priors for z_{τ_0}, $\mathbf{g}_{\tau_0}$, and $\{\mathbf{u}_{\tau_0,k}\}_{k \in \mathcal{K}_{\mathcal{T}_t}^{\mathrm{qry}}}$ from the stored filtering belief at time $\tau_0 - 1$ (plus one time-update); if unavailable, use conservative default priors.
Run Algorithm 3 on $\mathcal{T}_t$ with initialization $\Theta^{(t-1)}$, floor ϵ_N, and $N_{\mathrm{EM}}^{\mathrm{win}}$ iterations.
Re-run a forward filtering pass over $\mathcal{T}_t$ under $\Theta^{(t)}$ to refresh the belief at time t (starting from the window-initial prior).
Return: updated parameters $\Theta^{(t)}$.

D.3. Cross-Covariance in the Exact Update

The Kalman update in Algorithm 2 is performed on the joint state $\mathbf{s}_t := (\mathbf{g}_t, \mathbf{u}_{t,I_t})$. For readability in this subsection, set $\mathbf{u}_t := \mathbf{u}_{t,I_t}$ and write $\mathbf{s}_t = (\mathbf{g}_t, \mathbf{u}_t)$. Even if the predictive covariance is block-diagonal (our factorized predictive belief), the *exact* posterior covariance after conditioning on the queried residual e_t generally has non-zero off-diagonal blocks:

$$\Sigma_{s,t|t}^{(m)} = \begin{bmatrix} \Sigma_{g,t|t}^{(m)} & \Sigma_{gu,t|t}^{(m)} \\ (\Sigma_{gu,t|t}^{(m)})^\top & \Sigma_{u,t|t}^{(m)} \end{bmatrix}, \qquad \Sigma_{gu,t|t}^{(m)} \neq \mathbf{0} \text{ in general.}$$

These cross terms arise because the observation matrix $H_t = [\Phi(\mathbf{x}_t)^\top \mathbf{B}_{I_t} \;\; \Phi(\mathbf{x}_t)^\top]$ couples $\mathbf{g}_t$ and $\mathbf{u}_{t,I_t}$. Retaining $\Sigma_{gu,t|t}^{(m)}$ would propagate correlation into subsequent steps and into cross-expert predictive covariances.

Closed-form cross-covariance. Write the Kalman gain in block form $K_t^{(m)} = \left[(K_{g,t}^{(m)})^\top \; (K_{u,t}^{(m)})^\top\right]^\top$, and let $\Sigma_{t,I_t}^{\mathrm{pred},(m)}$ denote the innovation covariance of the queried residual as in Algorithm 2: $\Sigma_{t,I_t}^{\mathrm{pred},(m)} = H_t \Sigma_{s,t|t-1}^{(m)} H_t^\top + \mathbf{R}_{m,I_t}$. Then the covariance update can be written as $\Sigma_{s,t|t}^{(m)} = \Sigma_{s,t|t-1}^{(m)} - K_t^{(m)} \Sigma_{t,I_t}^{\mathrm{pred},(m)} (K_t^{(m)})^\top$. If the predictive covariance is block-diagonal, then the off-diagonal block is

$$\Sigma_{gu,t|t}^{(m)} = -K_{g,t}^{(m)} \Sigma_{t,I_t}^{\mathrm{pred},(m)} (K_{u,t}^{(m)})^\top = -\Sigma_{g,t|t-1}^{(m)} H_{g,t}^\top (\Sigma_{t,I_t}^{\mathrm{pred},(m)})^{-1} H_{u,t} \Sigma_{u,t|t-1}^{(m)},$$

where $H_{g,t} = \Phi(\mathbf{x}_t)^\top \mathbf{B}_{I_t} \in \mathbb{R}^{d_y \times d_g}$ and $H_{u,t} = \Phi(\mathbf{x}_t)^\top \in \mathbb{R}^{d_y \times d_\alpha}$. Unless one of these terms is zero, the cross-covariance is non-zero.

Why we discard it. Keeping $\Sigma^{(m)}_{gu,t|t}$ is exact but undermines the factorized SLDS approximation that enables scalable inference under a growing expert registry. Once $\mathbf{g}_t$ becomes correlated with $\mathbf{u}_{t,I_t}$, future prediction steps introduce non-zero cross-covariances between $\mathbf{g}_t$ and every $\mathbf{u}_{t,k}$ that shares dynamics with $\mathbf{u}_{t,I_t}$, and, through the shared factor, induce dependence across many experts. This breaks the stored-marginal structure, increases both compute and memory (scaling with the full registry size), and complicates closed-form quantities used in Section 4.2.3 (e.g., the Gaussian channel form and information gain). For these reasons, we project back to a factorized belief after each update and retain only the diagonal blocks as in Algorithm 2.

E. Information Gain for Exploration

Remark 5 ($(z_t, \mathbf{g}_t)$-Information Gain for Non-Stationary Routing)**.** *Algorithm 1 uses the* full $(z_t, \mathbf{g}_t)$-*information gain rather than conditioning only on* $\mathbf{g}_t$*. By the chain rule for mutual information:*

$$\mathcal{I}\Big((z_t, \mathbf{g}_t); e^{\text{pred}}_{t,k} \;\Big|\; \mathcal{F}_t\Big) = \underbrace{\mathcal{I}\Big(z_t; e^{\text{pred}}_{t,k} \;\Big|\; \mathcal{F}_t\Big)}_{\textit{mode-identification}} + \underbrace{\mathcal{I}\Big(\mathbf{g}_t; e^{\text{pred}}_{t,k} \;\Big|\; z_t, \mathcal{F}_t\Big)}_{\textit{shared-factor refinement}}. \tag{23}$$

The first term measures how much observing the residual $e^{\text{pred}}_{t,k}$ helps identify the current regime z_t. This is crucial for non-stationarity: when modes have distinct predictive distributions, querying an expert whose residual discriminates between regimes accelerates adaptation to regime changes.

Why both terms matter:

- *Shared-factor refinement* (closed-form): Reduces posterior uncertainty about $\mathbf{g}_t$, improving predictions for *all* experts via Proposition 2.
- *Mode-identification* (Monte Carlo): Reduces uncertainty about z_t, ensuring the router uses the correct dynamics parameters $(\mathbf{A}^{(g)}_m, \mathbf{Q}^{(g)}_m, \mathbf{A}^{(u)}_m, \mathbf{Q}^{(u)}_m, \mathbf{R}_{m,k})$.

Computational note: The mode-identification term requires Monte Carlo estimation because the predictive distribution $p(e^{\text{pred}}_{t,k} \mid \mathcal{F}_t)$ is a Gaussian mixture, for which KL divergence has no closed form. The LogSumExp trick ensures numerical stability. With $S = 50$ samples per expert, the overhead is negligible compared to the SLDS update cost.

E.1. Exploration via $(z_t, \mathbf{g}_t)$-information

Bandit feedback reveals only the queried expert's residual, so the router must trade off *exploitation* (low immediate cost) against *learning* (reducing posterior uncertainty to improve future decisions). In our IMM-factorized SLDS, two latent objects drive both non-stationarity and cross-expert transfer: the regime $z_t \in \{1, \dots, M\}$ and the shared factor $\mathbf{g}_t$ (Proposition 2). We therefore score exploration by the information gained about the *joint* latent state $(z_t, \mathbf{g}_t)$ from the (potential) queried residual. Throughout, logarithms are natural unless stated otherwise, so mutual information is measured in nats (replace $\log$ by $\log_2$ to obtain bits). We reuse the core SLDS/IMM notation from the main text: $\Phi(\mathbf{x}_t)$, $\mathbf{B}_k$, $\bar{w}^{(m)}_t = \mathbb{P}(z_t = m \mid \mathcal{F}_t)$, and the predictive moments $(\mu^{(m)}_{g,t|t-1}, \Sigma^{(m)}_{g,t|t-1})$, $(\mu^{(m)}_{u,k,t|t-1}, \Sigma^{(m)}_{u,k,t|t-1})$, and $\mathbf{R}_{m,k}$. For Monte Carlo, we use $\tilde{\cdot}$ to denote sampled quantities and write $\tilde{z} \sim \mathrm{Cat}((\bar{w}^{(m)}_t)^M_{m=1})$ for a categorical draw from the mode weights.

Decision-time predictive random variables. At round t, the decision-time sigma-algebra is $\mathcal{F}_t = \sigma(\mathcal{H}_{t-1}, \mathbf{x}_t, \mathcal{E}_t)$ and the router chooses $I_t \in \mathcal{E}_t$. For each available expert $k \in \mathcal{E}_t$, define the pre-query predictive residual random variable

$$e^{\text{pred}}_{t,k} \sim p(e_{t,k} \mid \mathcal{F}_t). \tag{24}$$

If $I_t = k$, the realized observation is $e_t = e_{t,k}$ and $e_t \mid (\mathcal{F}_t, I_t = k) \overset{d}{=} e^{\text{pred}}_{t,k} \mid \mathcal{F}_t$.

Per-mode linear-Gaussian predictive parametrization (IMM outputs). Fix a regime $z_t = m$. The IMM predictive step yields a Gaussian predictive prior for the shared factor:

$$\mathbf{g}_t \mid (\mathcal{F}_t, z_t = m) \sim \mathcal{N}\Big(\mu^{(m)}_{g,t|t-1},\, \Sigma^{(m)}_{g,t|t-1}\Big). \tag{25}$$

Under the factorized predictive belief, querying expert k induces the linear-Gaussian observation channel

$$e^{\text{pred}}_{t,k} \mid (\mathbf{g}_t, \mathcal{F}_t, z_t = m) \sim \mathcal{N}\big(\mathbf{H}_{t,k}\mathbf{g}_t + \mathbf{b}^{(m)}_{t,k},\, \mathbf{S}^{(m)}_{t,k}\big), \tag{26}$$

with mode-specific quantities

$$\begin{aligned}
\mathbf{H}_{t,k} &\coloneqq \Phi(\mathbf{x}_t)^\top \mathbf{B}_k \in \mathbb{R}^{d_y \times d_g}, \\
\mathbf{b}^{(m)}_{t,k} &\coloneqq \Phi(\mathbf{x}_t)^\top \mu^{(m)}_{u,k,t|t-1} \in \mathbb{R}^{d_y}, \\
\mathbf{S}^{(m)}_{t,k} &\coloneqq \Phi(\mathbf{x}_t)^\top \Sigma^{(m)}_{u,k,t|t-1} \Phi(\mathbf{x}_t) + \mathbf{R}_{m,k} \in \mathbb{S}^{d_y}_{++}.
\end{aligned} \tag{27}$$

Exploitation score: predictive cost and gap. Recall the realized cost $C_{t,k} = \psi(e_{t,k}) + \beta_k$, where $\beta_k \geq 0$ is the known query fee. In practice, we use squared loss,

$$\psi(u) = \|u\|_2^2, \tag{28}$$

and we will simplify expressions accordingly; nothing in the $(z_t, \mathbf{g}_t)$-information score depends on this choice. Define the predictive (virtual) cost random variable

$$C^{\text{pred}}_{t,k} \coloneqq \psi(e^{\text{pred}}_{t,k}) + \beta_k, \qquad k \in \mathcal{E}_t, \tag{29}$$

with conditional mean

$$\bar{C}^{\text{pred}}_{t,k} \coloneqq \mathbb{E}\Big[C^{\text{pred}}_{t,k} \,\Big|\, \mathcal{F}_t\Big] = \mathbb{E}\Big[\psi(e^{\text{pred}}_{t,k}) \,\Big|\, \mathcal{F}_t\Big] + \beta_k. \tag{30}$$

Let $k^{\text{pred}}_t \in \arg\min_{k \in \mathcal{E}_t} \bar{C}^{\text{pred}}_{t,k}$ and define the predictive gap

$$\Delta_t(k) \coloneqq \bar{C}^{\text{pred}}_{t,k} - \bar{C}^{\text{pred}}_{t,k^{\text{pred}}_t} \geq 0. \tag{31}$$

Computing $\bar{C}^{\text{pred}}_{t,k}$ from per-mode moments. From (26)–(27), the mode-conditioned predictive residual is Gaussian with

$$\bar{e}^{\text{pred},(m)}_{t,k} \coloneqq \mathbb{E}\Big[e^{\text{pred}}_{t,k} \,\Big|\, \mathcal{F}_t, z_t = m\Big] = \mathbf{H}_{t,k}\mu^{(m)}_{g,t|t-1} + \mathbf{b}^{(m)}_{t,k} \in \mathbb{R}^{d_y}, \tag{32}$$

$$\Sigma^{\text{pred},(m)}_{t,k} \coloneqq \mathrm{Cov}\Big(e^{\text{pred}}_{t,k} \,\Big|\, \mathcal{F}_t, z_t = m\Big) = \mathbf{H}_{t,k}\Sigma^{(m)}_{g,t|t-1}\mathbf{H}^\top_{t,k} + \mathbf{S}^{(m)}_{t,k} \in \mathbb{S}^{d_y}_{++}. \tag{33}$$

Let $\bar{w}^{(m)}_t = \mathbb{P}(z_t = m \mid \mathcal{F}_t)$. Then $p(e^{\text{pred}}_{t,k} \mid \mathcal{F}_t) = \sum_{m=1}^M \bar{w}^{(m)}_t \,\mathcal{N}(\bar{e}^{\text{pred},(m)}_{t,k}, \Sigma^{\text{pred},(m)}_{t,k})$. For general ψ,

$$\mathbb{E}\Big[\psi(e^{\text{pred}}_{t,k}) \,\Big|\, \mathcal{F}_t\Big] = \sum_{m=1}^M \bar{w}^{(m)}_t \,\mathbb{E}[\psi(E)]_{E \sim \mathcal{N}(\bar{e}^{\text{pred},(m)}_{t,k}, \Sigma^{\text{pred},(m)}_{t,k})}. \tag{34}$$

In the squared-loss case $\psi(e) = \|e\|_2^2$ from (29), we have $\mathbb{E}[\|E\|_2^2] = \mathrm{tr}(\Sigma) + \|\mu\|_2^2$, hence

$$\bar{C}^{\text{pred}}_{t,k} = \left(\sum_{m=1}^M \bar{w}^{(m)}_t \big(\mathrm{tr}(\Sigma^{\text{pred},(m)}_{t,k}) + \|\bar{e}^{\text{pred},(m)}_{t,k}\|_2^2\big)\right) + \beta_k. \tag{35}$$

Learning score: information about $(z_t, \mathbf{g}_t)$. Define the $(z_t, \mathbf{g}_t)$-information gain of querying expert k by

$$\mathrm{IG}_t(k) \coloneqq \mathcal{I}\Big((z_t, \mathbf{g}_t);\; e^{\text{pred}}_{t,k} \,\Big|\, \mathcal{F}_t\Big). \tag{36}$$

By the chain rule,

$$\mathrm{IG}_t(k) = \mathcal{I}\Big(z_t;\ e^{\mathrm{pred}}_{t,k} \,\Big|\, \mathcal{F}_t\Big) + \mathcal{I}\Big(\mathbf{g}_t;\ e^{\mathrm{pred}}_{t,k} \,\Big|\, \mathcal{F}_t, z_t\Big) \tag{37}$$

$$= \underbrace{\mathcal{I}\Big(z_t;\ e^{\mathrm{pred}}_{t,k} \,\Big|\, \mathcal{F}_t\Big)}_{\text{mode-identification}} + \underbrace{\sum_{m=1}^{M} \bar{w}_t^{(m)}\, \mathcal{I}\Big(\mathbf{g}_t;\ e^{\mathrm{pred}}_{t,k} \,\Big|\, \mathcal{F}_t, z_t = m\Big)}_{\text{shared-factor refinement}}. \tag{38}$$

The second term admits a closed form per mode; the first term is an information quantity for a d_y-dimensional Gaussian mixture that can be computed accurately with light Monte Carlo.

Closed form: $\mathcal{I}(\mathbf{g}_t; e^{\mathrm{pred}}_{t,k} \mid \mathcal{F}_t, z_t = m)$**.** Fix $z_t = m$. Let $G := \mathbf{g}_t$ and $Y := e^{\mathrm{pred}}_{t,k}$. Equation (27) implies the affine Gaussian channel $Y = \mathbf{H}_{t,k} G + \mathbf{b}^{(m)}_{t,k} + \varepsilon$ with $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{S}^{(m)}_{t,k})$ independent of G. Then

$$\mathcal{I}\Big(\mathbf{g}_t;\ e^{\mathrm{pred}}_{t,k} \,\Big|\, \mathcal{F}_t, z_t = m\Big) = \frac{1}{2}\log\det\Big(\mathbf{I}_{d_y} + \mathbf{H}_{t,k}\Sigma^{(m)}_{g,t|t-1}\mathbf{H}_{t,k}^\top (\mathbf{S}^{(m)}_{t,k})^{-1}\Big). \tag{39}$$

Monte Carlo: $\mathcal{I}(z_t; e^{\mathrm{pred}}_{t,k} \mid \mathcal{F}_t)$ **for a Gaussian mixture.** Let $p_m(e) := p(e^{\mathrm{pred}}_{t,k} = e \mid \mathcal{F}_t, z_t = m) = \mathcal{N}(e; \bar{e}^{\mathrm{pred},(m)}_{t,k}, \Sigma^{\mathrm{pred},(m)}_{t,k})$ and $p_{\mathrm{mix}}(e) := \sum_{m=1}^{M} \bar{w}_t^{(m)} p_m(e)$. Then

$$\mathcal{I}\Big(z_t;\ e^{\mathrm{pred}}_{t,k} \,\Big|\, \mathcal{F}_t\Big) = \sum_{m=1}^{M} \bar{w}_t^{(m)}\, \mathrm{KL}(p_m \,\|\, p_{\mathrm{mix}}) \tag{40}$$

$$= \sum_{m=1}^{M} \bar{w}_t^{(m)}\, \mathbb{E}_{E\sim p_m}[\log p_m(E) - \log p_{\mathrm{mix}}(E)]. \tag{41}$$

This suggests the estimator (with S samples per mode):

$$\widehat{\mathcal{I}}^{(z)}_t(k) := \sum_{m=1}^{M} \bar{w}_t^{(m)} \left(\frac{1}{S}\sum_{s=1}^{S}\Big[\log p_m(E_{m,s}) - \log p_{\mathrm{mix}}(E_{m,s})\Big]\right), \qquad E_{m,s} \overset{\mathrm{iid}}{\sim} \mathcal{N}(\bar{e}^{\mathrm{pred},(m)}_{t,k}, \Sigma^{\mathrm{pred},(m)}_{t,k}). \tag{42}$$

Stable evaluation of $\log p_{\mathrm{mix}}(e)$**.** Compute Gaussian log-densities via

$$\log\mathcal{N}(e;\mu,\Sigma) = -\frac{1}{2}\left(d_y\log(2\pi) + \log\det(\Sigma) + (e-\mu)^\top\Sigma^{-1}(e-\mu)\right). \tag{43}$$

Define $\ell_m(e) := \log \bar{w}_t^{(m)} + \log\mathcal{N}(e; \bar{e}^{\mathrm{pred},(m)}_{t,k}, \Sigma^{\mathrm{pred},(m)}_{t,k})$. Then compute $\log p_{\mathrm{mix}}(e)$ by a stable log-sum-exp:

$$\log p_{\mathrm{mix}}(e) = \log\left(\sum_{m=1}^{M} e^{\ell_m(e)}\right) = a(e) + \log\left(\sum_{m=1}^{M} e^{\ell_m(e)-a(e)}\right), \qquad a(e) := \max_{m\in\{1,\dots,M\}} \ell_m(e). \tag{44}$$

Final $(z_t, \mathbf{g}_t)$**-information gain.** Combine (38), (40), and (43):

$$\widehat{\mathrm{IG}}_t(k) := \widehat{\mathcal{I}}^{(z)}_t(k) + \sum_{m=1}^{M} \bar{w}_t^{(m)}\, \frac{1}{2}\log\det\Big(\mathbf{I}_{d_y} + \mathbf{H}_{t,k}\Sigma^{(m)}_{g,t|t-1}\mathbf{H}_{t,k}^\top(\mathbf{S}^{(m)}_{t,k})^{-1}\Big). \tag{45}$$

In Algorithm 1, we use $\mathrm{IG}_t(k)$ as a shorthand for this computable estimate $\widehat{\mathrm{IG}}_t(k)$.

F. Proofs

F.1. Proof of Proposition 2

Proposition 2 (Information transfer under a shared factor)**.** *Fix t and $z_t = m$, and let $\mathcal{G}_t := \sigma(\mathcal{F}_t, I_t, z_t = m)$. Let $j \neq I_t$ and let $(e^{\mathrm{pred}}_{t,j}, e^{\mathrm{pred}}_{t,I_t})$ denote the one-step-ahead predictive residuals under $p(e_{t,\cdot} \mid \mathcal{F}_t, z_t = m)$. Assume that this predictive* *pair is jointly Gaussian conditional on* $\mathcal{G}_t$ *and that* $\mathrm{Cov}(e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t)$ *is non-singular (e.g.,* $\mathbf{R}_{m,I_t} \succ \mathbf{0}$*). Then*

$$\begin{aligned} &\mathbb{E}\left[e^{\mathrm{pred}}_{t,j} \mid e_t, \mathcal{G}_t\right] = \mathbb{E}\left[e^{\mathrm{pred}}_{t,j} \mid \mathcal{G}_t\right] \\ \iff \quad &\mathrm{Cov}\left(e^{\mathrm{pred}}_{t,j}, e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t\right) = \mathbf{0}. \end{aligned}$$

In particular, if the covariance is non-zero, then observing $e_t = e_{t,I_t}$ *updates the posterior predictive mean of* $e^{\mathrm{pred}}_{t,j}$*.*

Proof Fix t and m, and let $\mathcal{G}_t := \sigma(\mathcal{F}_t, I_t, z_t = m)$. By assumption, the one-step-ahead predictive pair $(e^{\mathrm{pred}}_{t,j}, e^{\mathrm{pred}}_{t,I_t}) \mid \mathcal{G}_t$ is jointly Gaussian, where each term lies in $\mathbb{R}^{d_y}$. Under $\mathcal{G}_t$ the realized observation is $e_t = e_{t,I_t}$, and $e_t \mid \mathcal{G}_t \overset{d}{=} e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t$ (since $e^{\mathrm{pred}}_{t,I_t}$ is exactly the one-step predictive residual that generates e_{t,I_t}). Let

$$\boldsymbol{\mu}_j := \mathbb{E}[e^{\mathrm{pred}}_{t,j} \mid \mathcal{G}_t], \qquad \boldsymbol{\mu}_I := \mathbb{E}[e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t],$$

and define the predictive covariance and cross-covariance matrices

$$\Sigma_I := \mathrm{Cov}(e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t) \in \mathbb{S}^{d_y}_{++}, \qquad \Sigma_{jI} := \mathrm{Cov}(e^{\mathrm{pred}}_{t,j}, e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t) \in \mathbb{R}^{d_y \times d_y}.$$

Assume Σ_I is non-singular (e.g., due to additive observation noise with $\mathbf{R}_{m,I_t} \succ \mathbf{0}$). For jointly Gaussian vectors, the conditional expectation is given by the standard formula

$$\mathbb{E}[e^{\mathrm{pred}}_{t,j} \mid e^{\mathrm{pred}}_{t,I_t} = e_t, \mathcal{G}_t] = \boldsymbol{\mu}_j + \Sigma_{jI}\,\Sigma_I^{-1}\left(e_t - \boldsymbol{\mu}_I\right).$$

Therefore, $\mathbb{E}[e^{\mathrm{pred}}_{t,j} \mid e_t, \mathcal{G}_t] = \boldsymbol{\mu}_j$ for all values of e_t if and only if $\Sigma_{jI} = \mathbf{0}$, i.e., $\mathrm{Cov}(e^{\mathrm{pred}}_{t,j}, e^{\mathrm{pred}}_{t,I_t} \mid \mathcal{G}_t) = \mathbf{0}$. ■

F.2. Proof of Proposition 3

Proposition 3 (Pruning does not affect retained experts)**.** *Fix time* t *and let* $P_t \subseteq \mathcal{K}_{t-1}$ *be any set of experts to be pruned. Let* $q_{t-1|t-1}\big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell\in\mathcal{K}_{t-1}}\big)$ *denote the (exact or approximate) filtering belief at the end of round* $t-1$ *conditioned on the realized history. Define the pruned belief by marginalization:*

$$\begin{aligned} &q^{\mathrm{pr}(P_t)}_{t-1|t-1}\big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell\in\mathcal{K}_{t-1}\setminus P_t}\big) := \\ &\int q_{t-1|t-1}\big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell\in\mathcal{K}_{t-1}}\big) \prod_{k\in P_t} d\mathbf{u}_{t-1,k}. \end{aligned}$$

Then $q^{\mathrm{pr}(P_t)}_{t-1|t-1}$ *equals the marginal of* $q_{t-1|t-1}$ *on the retained variables. Consequently, after applying the standard SLDS time update to obtain the predictive belief at round* t*, the predictive distribution of* $\boldsymbol{\alpha}_{t,\ell}$ *and the one-step predictive law of* $e^{\mathrm{pred}}_{t,\ell}$ *are identical before and after pruning, for every retained* $\ell \notin P_t$*.*

Proof The statement is a direct consequence of the definition of marginalization.

Write the filtering belief at the end of round $t-1$ (conditioned on the realized history, which we omit from the notation) as a joint density over the shared factor and all idiosyncratic states:

$$q_{t-1|t-1}\Big(\mathbf{g}_{t-1}, (\mathbf{u}_{t-1,\ell})_{\ell\in\mathcal{K}_{t-1}}\Big).$$

Let $\mathcal{K}' := \mathcal{K}_{t-1} \setminus P_t$ denote the retained experts and denote $\mathbf{u}_{t-1,\mathcal{K}'} := (\mathbf{u}_{t-1,\ell})_{\ell\in\mathcal{K}'}$. By the definition of a marginal density, the joint marginal of the retained variables under $q_{t-1|t-1}$ is

$$q_{t-1|t-1}\big(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}\big) \;=\; \int q_{t-1|t-1}\big(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}, (\mathbf{u}_{t-1,k})_{k\in P_t}\big) \prod_{k\in P_t} d\mathbf{u}_{t-1,k}. \tag{46}$$

On the other hand, the post-pruning belief $q^{\mathrm{pr}(P_t)}_{t-1|t-1}$ is *defined* by exactly the same integral:

$$q^{\mathrm{pr}(P_t)}_{t-1|t-1}\big(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}\big) \;:=\; \int q_{t-1|t-1}\big(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}, (\mathbf{u}_{t-1,k})_{k\in P_t}\big) \prod_{k\in P_t} d\mathbf{u}_{t-1,k}.$$

Comparing with (47) yields

$$q^{\mathrm{pr}(P_t)}_{t-1|t-1}\big(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}\big) = q_{t-1|t-1}\big(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}\big),$$

which proves that pruning P_t leaves the joint belief over all retained variables unchanged.

For the stated consequences, let $\ell \notin P_t$. The SLDS time update propagates $(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\ell})$ to $(\mathbf{g}_t, \mathbf{u}_{t,\ell})$ using the same linear-Gaussian transition under both beliefs. Since the retained marginal $q_{t-1|t-1}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\ell})$ is identical before and after pruning, the predictive distribution of $(\mathbf{g}_t, \mathbf{u}_{t,\ell})$ is also identical. Because $\boldsymbol{\alpha}_{t,\ell} = \mathbf{B}_\ell \mathbf{g}_t + \mathbf{u}_{t,\ell}$ is a measurable function of $(\mathbf{g}_t, \mathbf{u}_{t,\ell})$ and $e^{\mathrm{pred}}_{t,\ell}$ follows the emission model given these states, the predictive distributions of $\boldsymbol{\alpha}_{t,\ell}$ and $e^{\mathrm{pred}}_{t,\ell}$ are unchanged by pruning. ■

F.3. Proof of Proposition 4

Proposition 4 (Coupling at birth through the shared factor)**.** *Fix time t and condition on $(\mathcal{F}_t, z_t = m)$. Under the Factorized SLDS one-step predictive belief (i.e., with $\mathrm{Cov}(\mathbf{g}_t, \mathbf{u}_{t,k} \mid \cdot) = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{u}_{t,i}, \mathbf{u}_{t,j} \mid \cdot) = \mathbf{0}$ for $i \neq j$), for any experts $j \neq k$,*

$$\mathrm{Cov}\left(\boldsymbol{\alpha}_{t,j}, \boldsymbol{\alpha}_{t,k} \mid \mathcal{F}_t, z_t = m\right) = \mathbf{B}_j \Sigma^{(m)}_{g,t|t-1} \mathbf{B}_k^\top,$$

where $\Sigma^{(m)}_{g,t|t-1}$ is the regime-m one-step predictive covariance of $\mathbf{g}_t$. In particular, if the joint predictive law is Gaussian and $\mathbf{B}_j \Sigma^{(m)}_{g,t|t-1} \mathbf{B}_k^\top \neq \mathbf{0}$, then $\boldsymbol{\alpha}_{t,j}$ and $\boldsymbol{\alpha}_{t,k}$ are not independent and hence $\mathcal{I}(\boldsymbol{\alpha}_{t,j}; \boldsymbol{\alpha}_{t,k} \mid \mathcal{F}_t, z_t = m) > 0$.

Proof Fix t and condition on $(\mathcal{F}_t, z_t = m)$. Under the factorized one-step predictive belief, for any $j \neq k$ we have the marginal factorization

$$q(\mathbf{g}_t, \mathbf{u}_{t,j}, \mathbf{u}_{t,k} \mid \mathcal{F}_t, z_t = m) = q(\mathbf{g}_t \mid \mathcal{F}_t, z_t = m)\, q(\mathbf{u}_{t,j} \mid \mathcal{F}_t, z_t = m)\, q(\mathbf{u}_{t,k} \mid \mathcal{F}_t, z_t = m),$$

so $\mathbf{g}_t \perp\!\!\!\perp \mathbf{u}_{t,\ell}$ for all ℓ and $\mathbf{u}_{t,j} \perp\!\!\!\perp \mathbf{u}_{t,k}$ for $j \neq k$. Recalling $\boldsymbol{\alpha}_{t,\ell} = \mathbf{B}_\ell \mathbf{g}_t + \mathbf{u}_{t,\ell}$ and using bilinearity of covariance,

$$\begin{aligned}
\mathrm{Cov}(\boldsymbol{\alpha}_{t,j}, \boldsymbol{\alpha}_{t,k} \mid \mathcal{F}_t, z_t = m) &= \mathrm{Cov}(\mathbf{B}_j \mathbf{g}_t + \mathbf{u}_{t,j},\, \mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k} \mid \mathcal{F}_t, z_t = m) \\
&= \mathrm{Cov}(\mathbf{B}_j \mathbf{g}_t,\, \mathbf{B}_k \mathbf{g}_t \mid \mathcal{F}_t, z_t = m) + \mathrm{Cov}(\mathbf{B}_j \mathbf{g}_t,\, \mathbf{u}_{t,k} \mid \mathcal{F}_t, z_t = m) \\
&\quad + \mathrm{Cov}(\mathbf{u}_{t,j},\, \mathbf{B}_k \mathbf{g}_t \mid \mathcal{F}_t, z_t = m) + \mathrm{Cov}(\mathbf{u}_{t,j},\, \mathbf{u}_{t,k} \mid \mathcal{F}_t, z_t = m) \\
&= \mathbf{B}_j \,\mathrm{Cov}(\mathbf{g}_t, \mathbf{g}_t \mid \mathcal{F}_t, z_t = m)\, \mathbf{B}_k^\top \\
&= \mathbf{B}_j \,\Sigma^{(m)}_{g,t|t-1}\, \mathbf{B}_k^\top,
\end{aligned}$$

where $\Sigma^{(m)}_{g,t|t-1} := \mathrm{Cov}(\mathbf{g}_t \mid \mathcal{F}_t, z_t = m)$. If $\mathbf{B}_j \Sigma^{(m)}_{g,t|t-1} \mathbf{B}_k^\top \neq 0$ and the joint predictive law of $(\boldsymbol{\alpha}_{t,j}, \boldsymbol{\alpha}_{t,k})$ is Gaussian, then the pair is not independent, hence $\mathcal{I}(\boldsymbol{\alpha}_{t,j}; \boldsymbol{\alpha}_{t,k} \mid \mathcal{F}_t, z_t = m) > 0$. ■

G. Experiments Details

We provide additional details on the experiments of Section 5, including experimental setup, hyperparameters, and implementation details.

Compared methods. We compare our **L2D-SLDS** router under bandit feedback to the following baselines. *(i) Ablation:* L2D-SLDS without the shared global factor (set $d_g = 0$). *(ii) Contextual bandits:* LinUCB (Li et al., 2010) and NeuralUCB (Zhou et al., 2020) with more details in **??**.

Metric. We report the time-averaged cumulative routing cost over horizon T (Eq. (8)). Concretely, we compute the estimate $\hat{J}(\pi) := \frac{1}{T}\sum_{t=1}^{T} C_{t,I_t}$, where C_{t,I_t} is the realized cost of deferring to the selected expert at round t. Lower is better.

G.1. Baselines

Feedback regimes. At round t, the router observes $(\mathbf{x}_t, \mathcal{E}_t)$, chooses $I_t \in \mathcal{E}_t$, and then observes $(\widehat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$, hence the realized residual $e_t = e_{t,I_t}$ and realized cost $C_t = C_{t,I_t}$, where $C_{t,k} := \psi(e_{t,k}) + \beta_k$ and $e_{t,k} = \widehat{\mathbf{y}}_{t,k} - \mathbf{y}_t$ (Appendix B). *Partial feedback* means only $(\widehat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$ is observed after acting.

L2D-SLDS and ablation without $\mathbf{g}_t$. Our method is the model-based router of Algorithm 1 under the generative residual model of Definition 1: $\boldsymbol{\alpha}_{t,k} = \mathbf{B}_k\mathbf{g}_t + \mathbf{u}_{t,k}$ and $e_{t,k} \mid (z_t = m, \mathbf{g}_t, \mathbf{u}_{t,k}, \mathbf{x}_t) \sim \mathcal{N}\big(\Phi(\mathbf{x}_t)^\top \boldsymbol{\alpha}_{t,k}, \mathbf{R}_{m,k}\big)$ ((13)–(14)). **L2D-SLDS w/o $\mathbf{g}_t$** is the ablation obtained by setting $d_g = 0$ (equivalently $\mathbf{B}_k\mathbf{g}_t \equiv \mathbf{0}$ for all k), so that $\boldsymbol{\alpha}_{t,k} = \mathbf{u}_{t,k}$ and the per-expert predictive residuals are conditionally independent across experts under the factorized belief (no cross-expert transfer through a shared factor).

Contextual bandits: LinUCB and NeuralUCB (partial and full feedback). Both methods operate on the per-round *cost* $C_{t,k}$ and are implemented as *lower* confidence bound (LCB) rules since we minimize cost. Under *full feedback*, the router observes $\{C_{t,k}\}_{k\in\mathcal{E}_t}$ regardless of which expert I_t was selected. Consequently, the usual exploration–exploitation trade-off disappears: the choice of I_t does not affect what data is available for learning, so the exploration bonus can be set to 0 (yielding greedy selection) without sacrificing statistical efficiency. We still state the LCB form below for a unified presentation.

LinUCB. Fix a feature map $\varphi : \mathbb{R}^d \to \mathbb{R}^p$ (in our experiments, either raw $\mathbf{x}_t$ or an RNN embedding). Assume a linear model for the conditional mean cost of each expert: $\mathbb{E}[C_{t,k} \mid \mathbf{x}_t] \approx \varphi(\mathbf{x}_t)^\top \boldsymbol{\theta}_k$. Maintain ridge statistics per expert k, with ridge parameter $\lambda > 0$. Under *partial feedback*:

$$\mathbf{V}_{t,k} := \lambda \mathbf{I}_p + \sum_{s<t:\ I_s=k} \varphi(\mathbf{x}_s)\varphi(\mathbf{x}_s)^\top, \qquad \mathbf{b}_{t,k} := \sum_{s<t:\ I_s=k} \varphi(\mathbf{x}_s) C_s, \qquad \widehat{\boldsymbol{\theta}}_{t,k} := \mathbf{V}_{t,k}^{-1}\mathbf{b}_{t,k}.$$

where $C_s = C_{s,I_s}$ is the realized (queried) cost at round s. At time t, set $\widehat{C}_{t,k} := \varphi(\mathbf{x}_t)^\top \widehat{\boldsymbol{\theta}}_{t,k}$ and exploration bonus $u_t(k) := \alpha_t \sqrt{\varphi(\mathbf{x}_t)^\top \mathbf{V}_{t,k}^{-1} \varphi(\mathbf{x}_t)}$. The decision rule is

$$I_t \in \arg\min_{k\in\mathcal{E}_t} \ \widehat{C}_{t,k} - u_t(k).$$

Partial feedback LinUCB updates only the chosen arm I_t (so only $C_{t,I_t} = C_t$ is observed).

NeuralUCB. Let $f_{\boldsymbol{\omega}}(\mathbf{x}, k)$ be a neural predictor of the conditional mean cost of expert k given $\mathbf{x}$ (we use a shared encoder with a per-expert head). Define a parameter-gradient feature (to avoid overloading the shared factor $\mathbf{g}_t$) $\mathbf{h}_{t,k} := \nabla_{\boldsymbol{\omega}} f_{\boldsymbol{\omega}}(\mathbf{x}_t, k) \in \mathbb{R}^{p_\omega}$. Maintain a (regularized) Gram matrix. Under *partial feedback*:

$$\mathbf{A}_t := \lambda \mathbf{I}_{p_\omega} + \sum_{s<t} \mathbf{h}_{s,I_s}\mathbf{h}_{s,I_s}^\top.$$

At time t, set $\widehat{C}_{t,k} := f_{\boldsymbol{\omega}}(\mathbf{x}_t, k)$ and $u_t(k) := \alpha_t \sqrt{\mathbf{h}_{t,k}^\top \mathbf{A}_t^{-1} \mathbf{h}_{t,k}}$. The decision rule is

$$I_t \in \arg\min_{k\in\mathcal{E}_t} \ \widehat{C}_{t,k} - u_t(k).$$

The network is trained online by stochastic gradient steps on squared error. *Partial feedback NeuralUCB* uses the loss $(f_{\boldsymbol{\omega}}(\mathbf{x}_t, I_t) - C_t)^2$ (only C_t observed).

Oracle baseline. The (per-round) oracle chooses the best available expert in hindsight:

$$I_t^{\text{oracle}} \in \arg\min_{k\in\mathcal{E}_t} C_{t,k}.$$

This is infeasible under partial feedback because $C_{t,k}$ is not observed for all k, but we report it as a lower bound on achievable cumulative cost.

G.2. Synthetic: Regime-Dependent Correlation and Information Transfer

Design goal. We construct a controlled routing instance in which (i) experts are *correlated* in a regime-dependent way, so that observing one expert should update beliefs about others (information transfer; Proposition 2); and (ii) one expert temporarily disappears and re-enters, so that the maintained registry $\mathcal{K}_t$ matters (see Appendix).

Environment (regimes, target, context). We use $M = 2$ regimes and deterministic switching in blocks of length $L = 150$ over horizon $T = 3000$ such as $z_t := 1 + \left\lfloor \frac{t-1}{L} \right\rfloor \bmod 2$. The target follows a regime-dependent AR(1), and the context is the one-step lag:

$$y_t = 0.8\, y_{t-1} + d_{z_t} + \eta_t, \qquad \eta_t \sim \mathcal{N}(0, \sigma_y^2). \tag{47}$$

We set the router's context to $x_t := y_{t-1}$. The regime z_t is latent to the router: the router observes only x_t (before acting) and the single queried prediction $\hat{y}_{t,I_t}$ (after acting).

Experts and availability. We use $K = 4$ experts indexed $k \in \{0, 1, 2, 3\}$. Expert $k = 1$ is removed from the available set $\mathcal{E}_t$ for a contiguous interval $t \in [2000, 2500]$ and then re-enters. Each expert is a one-step forecaster $\hat{y}_{t,k} = f_k(x_t)$ with a shared slope and expert-specific intercept plus noise:

$$\hat{y}_{t,k} := 0.8\, y_{t-1} + b_k + \varepsilon_{t,k}. \tag{48}$$

We set $(b_0, b_1, b_2, b_3) = (d_1, d_1, d_2, d_2)$, so experts $\{0, 1\}$ are well-calibrated in regime $z_t = 1$ and experts $\{2, 3\}$ are well-calibrated in regime $z_t = 2$.

To induce *regime-dependent correlation* under bandit feedback, we generate the expert noises as

$$\varepsilon_{t,k} := s_{t,g(k)} + \tilde{\varepsilon}_{t,k}, \qquad g(k) := 1 + \mathbf{1}\{k \in \{2, 3\}\},$$

with independent components $s_{t,1}, s_{t,2}, (\tilde{\varepsilon}_{t,k})_k$ and regime-dependent variances $s_{t,1} \sim \mathcal{N}(0, \sigma_{z_t,1}^2), s_{t,2} \sim \mathcal{N}(0, \sigma_{z_t,2}^2), \tilde{\varepsilon}_{t,k} \sim \mathcal{N}(0, \sigma_{\mathrm{id}}^2)$, where $(\sigma_{1,1}^2, \sigma_{1,2}^2) = (\sigma_{\mathrm{hi}}^2, \sigma_{\mathrm{lo}}^2)$ and $(\sigma_{2,1}^2, \sigma_{2,2}^2) = (\sigma_{\mathrm{lo}}^2, \sigma_{\mathrm{hi}}^2)$ with $\sigma_{\mathrm{hi}}^2 \gg \sigma_{\mathrm{lo}}^2$. This makes experts $\{0, 1\}$ strongly correlated in regime 1 and experts $\{2, 3\}$ strongly correlated in regime 2. We report the MSE of each expert in Table 1.

Table 3. Averaged cumulative cost (8) on experiment (Section 5.1). We report mean $\pm$ standard error across five runs. Lower is better.

Method	Averaged Cumulative Cost
L2D-SLDS	$\mathbf{13.58 \pm 0.07}$
L2D-SLDS w/o $\mathbf{g}_t$	14.68 ± 0.01
LinUCB	22.94 ± 0.01
NeuralUCB	21.92 ± 0.31
Random	26.13 ± 0.25
Always expert 0	23.07
Always expert 1	28.66
Always expert 2	23.05
Always expert 3	29.36
Oracle	9.04

Model Configuration. We use $M = 2$ regimes with shared factor dimension $d_g = 1$ and idiosyncratic dimension $d_\alpha = 1$. The staleness horizon for pruning is $\Delta_{\max} = 500$. We simply run a small warmup of 100 steps before running L2D-SLDS and UCBs.

Correlation recovery. Figure 1 compares the regime-0 loss correlation structure. The ground truth exhibits a clear block structure: experts $\{0, 1\}$ form one correlated group while experts $\{2, 3\}$ form another. Under partial feedback, L2D-SLDS is the only method that reliably recovers this clustering from partial observations, whereas removing the shared factor $\mathbf{g}_t$ blurs the separation and inflates cross-group correlations, consistent with losing cross-expert information transfer. In contrast, LinUCB/NeuralUCB yield near-degenerate correlation estimates (e.g., overly uniform or unstable patterns), reflecting that purely discriminative bandit updates do not maintain a coherent joint belief over experts' latent error processes.

Results and Analysis. Table 1 shows that **L2D-SLDS** achieves the lowest routing cost under partial feedback (13.58 ± 0.07), improving over LinUCB/NeuralUCB by a wide margin and also outperforming the best fixed expert. Crucially, it also beats the ablation that removes the shared factor $\mathbf{g}_t$ (14.68 ± 0.01), a $\approx 7.5\%$ reduction, which directly supports our central claim: under censoring, modeling a *global* latent component enables *cross-expert information transfer* from a single queried residual (see Proposition 4). Intuitively, $\mathbf{g}_t$ captures regime-dependent common shocks that couple experts; thus, querying one expert updates beliefs about unqueried experts in a way that contextual bandits (which treat arms largely independently) and independent per-expert dynamics cannot replicate.

In Appendix, we provide additional experiments that probe this regime-dependent setting in more depth, including detailed analyses of expert pruning and re-entry.

Figure 3. We report the selection frequency of each expert over time as a function of the underlying regime. The top figure corresponds to the oracle, while the bottom figure shows our approach evaluated against the baselines. By construction, experts 0 and 1 perform better in regime 1, whereas experts 2 and 3 perform better in regime 2. Accordingly, a well-adapted router should select experts 0 and 1 more frequently in regime 1 and experts 2 and 3 more frequently in regime 2. L2D-SLDS (with and without g_t) is the only method that captures this structure, closely matching the oracle's selection behavior. In contrast, LinUCB and NeuralUCB fail to adapt their selection frequencies to the regimes.

G.3. ETTh1

Environment. We evaluate L2D-SLDS on the ETTh1 electricity transformer temperature dataset (**?**), using the oil temperature (OT) channel as the target y_t. We run the router over the full horizon ($T = 17420$ hourly observations). Following the synthetic setup, the router uses a one-step lag as context, $x_t := y_{t-1}$ (with $x_0 = 0$). There is no observed regime annotation for ETTh1; the router observes only x_t before acting and, after selecting $I_t \in \mathcal{E}_t$, it observes the realized outcome y_t and the single queried prediction $\hat{y}_{t,I_t}$ (hence the queried residual e_{t,I_t}).

Experts and availability. We consider $K = 6$ fixed experts (Table **??**). To stress-test dynamic availability and our pruning/re-birth mechanism, we enforce time-varying expert sets: the strong multi-lag baseline (Expert 4) is available only on the interval $t \in [1000, 2000]$, while Expert 0 is unavailable on the same interval. This prevents degenerate "always-pick-the-best" policies and forces the router to handle both expert arrival/departure (Expert 4) and temporary unavailability with later return (Expert 0).

Table 4. Configuration of experts for ETTh1.

Index	Base	Modification
0	AR(1)	small variance
1	AR(1)	large variance
2	MLP	trained on early 2/3 of data
3	MLP	trained on late 2/3 of data
4	AR multi-lag baseline	using lags [1, 24, 168]
5	Constant	always predict 0

Model Configuration. We use $M = 5$ regimes with shared factor dimension $d_g = 2$ and idiosyncratic dimension $d_\alpha = 1$. The staleness horizon for pruning is $\Delta_{\max} = 250$. The exploration term considers information gain on both global factor g and regime z. Online EM adaptation is enabled with a sliding window of $W = 600$ and updates every 300 steps.

Results and analysis. Table **??** reports the averaged cumulative routing cost. Under *partial feedback*, L2D-SLDS achieves the lowest cost among adaptive methods that learn online from bandit feedback (0.80 ± 0.06), improving over LinUCB (0.84) and substantially outperforming NeuralUCB (1.09 ± 0.19). Most importantly, removing the shared factor $\mathbf{g}_t$ degrades performance (0.93 ± 0.08), a relative increase of $\approx 15\%$. This gap is consistent with the role of $\mathbf{g}_t$ under censoring: ETTh1 exhibits common shocks (e.g., global load/temperature patterns) that affect multiple experts similarly, so a shared latent component lets a single queried residual update beliefs about *unqueried* experts via the learned cross-expert dependence.

Table 5. Averaged cumulative cost (8) on ETTh1 (Section **??**). We report the mean $\pm$ standard error over five runs; lower is better. The averaged cumulative cost is computed both over the full time horizon and, for each expert, only over the periods during which that expert is available. This explains why Expert 4 attains a low cost despite being available for only a short duration. Consequently, it is expected that baseline methods exhibit higher averaged costs than Expert 4.

Method	Averaged Cumulative Cost
L2D-SLDS	$\mathbf{0.80 \pm 0.06}$
L2D-SLDS w/o $\mathbf{g}_t$	0.93 ± 0.08
LinUCB	0.84 ± 0.01
NeuralUCB	1.09 ± 0.19
Random	14.51 ± 0.73
Always expert 0	0.81
Always expert 1	1.19
Always expert 2	0.77
Always expert 3	1.21
Always expert 4	0.74
Always expert 5	166.65
Oracle	0.24 ± 0.01

G.4. FRED: Treasury Securities at 10-Year Constant Maturity

Environment. We evaluate on the FRED DGS10 series (10-year U.S. Treasury constant-maturity yield) (**?**), using the daily observations in `data/FRED_DGS10.csv` from 1990-01-02 through 2023-12-29 ($T = 8506$). The target is $y_t := \text{DGS10}_t$. The router uses a fixed context vector $x_t \in \mathbb{R}^{10}$ consisting of yield lags at $\{1, 5, 20, 60, 120, 250\}$ days and calendar features for day-of-week and month encoded as sine/cosine pairs. We z-score normalize each context dimension using the first 2520 observations. As in all partial-feedback experiments, at each round t the router observes $(x_t, \mathcal{E}_t)$, chooses I_t, and then observes $\hat{y}_{t,I_t}$ and y_t (hence the queried residual e_{t,I_t}).

Experts. We use $K = 4$ ridge-regularized linear autoregressive experts (AR) of the form $\hat{y}_{t,k} = w_k^\top x_t + b_k$, each trained offline on a disjoint historical date range and then deployed across the full evaluation horizon. To avoid a single expert becoming deterministically dominant, we add mild i.i.d. Gaussian prediction noise with standard deviation 0.03 to each expert's output. All experts are available at all times ($\mathcal{E}_t = \{0, 1, 2, 3\}$).

Table 6. Configuration of experts for the FRED DGS10 experiment.

Index	Model	Training window
0	AR (linear ridge on x_t)	1990-01-02–2000-12-31
1	AR (linear ridge on x_t)	2001-01-01–2007-12-31
2	AR (linear ridge on x_t)	2008-01-01–2015-12-31
3	AR (linear ridge on x_t)	2016-01-01–2023-12-31

Model Configuration. We use $M = 4$ regimes with shared factor dimension $d_g = 2$ and idiosyncratic dimension $d_\alpha = 10$ (matching the context dimension). The staleness horizon is $\Delta_{\max} = 4000$, measurement noise is set to $R = 0.01$, and exploration uses the information gain over both $\mathbf{g}_t$ and z_t (mode `g_z`). We disable EM adaptation in this experiment.

Results and analysis. Table **??** reports the averaged cumulative routing cost. Under partial feedback, **L2D-SLDS** achieves the lowest cost among adaptive methods (0.004327 ± 0.000003), improving over LinUCB, NeuralUCB, and random routing. Removing the shared factor slightly degrades performance (0.004411 ± 0.000011), consistent with shared latent structure providing additional cross-expert signal when only one residual is observed per round.

Table 7. Averaged cumulative cost (8) on the FRED (DGS10) experiment (Appendix ??). We report mean $\pm$ standard error across five runs; lower is better.

Method	Average Cumulative Cost
L2D-SLDS	$\mathbf{0.004327 \pm 0.000003}$
L2D-SLDS w/o $\mathbf{g}_t$	0.004411 ± 0.000011
LinUCB	0.004452 ± 0.000002
NeuralUCB	0.004424 ± 0.000023
Random	0.004455 ± 0.000009
Always expert 0	0.004411
Always expert 1	0.004567
Always expert 2	0.004505
Always expert 3	0.004329
Oracle	0.001754