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

Anonymous Authors¹

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 inter-

leaved 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 y_t together with the queried prediction \hat{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

¹Anonymous 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.

055 and multi-task settings and applied in real systems (Mao
 056 et al., 2024c; Strong et al., 2024; Palomba et al., 2025;
 057 Montreuil et al., 2025b;c). Missing expert predictions have
 058 been studied in offline/batch learning (Nguyen et al., 2025).
 059 Sequential L2D has been studied in a different setting: Joshi
 060 et al. (2021) formulate deferral in a non-stationary MDP
 061 and learn a *pre-emptive* deferral policy by comparing the
 062 long-term value of deferring now versus delaying deferral.
 063 In contrast, we study time-series expert routing where the
 064 router selects among available experts *online* under censored
 065 (bandit-style) feedback, with potentially non-stationary data
 066 and a time-varying expert pool. We are not aware of existing
 067 L2D formulations that jointly address non-stationarity,
 068 censored observations, and dynamic expert availability.

3. Background

3.1. Offline Learning-to-Defer

074 We briefly recall the standard *offline* learning-to-defer (L2D)
 075 setup (Madras et al., 2018; Mozannar & Sontag, 2020;
 076 Narasimhan et al., 2022; Mao et al., 2024c). In its sim-
 077 plest form, one observes i.i.d. samples $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$, where
 078 $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ and $\mathbf{y} \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$. There is a fixed registry
 079 $\mathcal{K} = \{1, \dots, K\}$ of experts (or predictors), each providing
 080 a prediction $\hat{\mathbf{y}}_k(\mathbf{x}) \in \mathcal{Y}$ when queried. Given a per-expert
 081 consultation fee $\beta_k \geq 0$ and a loss on the prediction error
 082 $\psi : \mathbb{R}^{d_y} \rightarrow \mathbb{R}_{\geq 0}$, the incurred cost of routing (\mathbf{x}, \mathbf{y}) to
 083 expert k is

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

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

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

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

$$k^*(\mathbf{x}) \in \arg \min_{k \in \mathcal{K}} \mathbb{E}[C_k(\mathbf{x}, \mathbf{y}) | \mathbf{x}]. \quad (3)$$

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

094 Most prior works learn π from a fixed dataset by empirical
 095 risk minimization on a dedicated surrogate loss (Mozan-
 096 nar & Sontag, 2020), often assuming access to all ex-
 097 perts' predictions $(\hat{\mathbf{y}}_k(\mathbf{x}_i))_{k=1}^K$ (or equivalently all costs
 098 $(C_k(\mathbf{x}_i, \mathbf{y}_i))_{k=1}^K$) for each training sample. Practical algo-
 099 rithms parameterize π with a model (e.g., a neural network)

100 and may use surrogates or relaxations to handle discrete rout-
 101 ing decisions and to obtain statistical guarantees (Mozannar
 102 & Sontag, 2020; Verma et al., 2022; Mao et al., 2024c).

3.2. Non-Stationary Time Series and SSMs

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

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

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

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

117 and the Kalman filter (Kalman, 1960; Welch et al., 1995)
 118 yields tractable online posteriors and predictive uncertain-
 119 ties. Switching linear dynamical systems (SLDSs) (Bengio
 120 & Frasconi, 1994; Ghahramani & Hinton, 2000; Fox
 121 et al., 2008; Hu et al., 2024; Geadah et al., 2024) en-
 122 rich this model with a discrete regime variable $z_t \in$
 123 $\{1, \dots, M\}$ selecting among multiple linear-Gaussian dy-
 124 namics; conditioned on $z_t = m$, (A, Q, C, R) are replaced
 125 by (A_m, Q_m, C_m, R_m) .

4. Context-Aware Routing in Non-Stationary Environments

4.1. Problem Formulation

126 Building on the offline learning-to-defer setup in Section 3.1,
 127 we study *sequential* expert routing in *non-stationary* time
 128 series under *censored feedback* (Neu et al., 2010; Dani et al.,
 129 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 support-
 130 ing all random variables. At each round t , the environment
 131 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
 132 $d_y \geq 1$, and a non-empty finite set of available expert
 133 identities \mathcal{E}_t . We allow \mathcal{E}_t to vary with t , capturing both tem-
 134 porary unavailability and newly arriving experts. The router
 135 maintains a time-varying *expert registry* \mathcal{K}_t , containing the
 136 experts for which it stores per-expert state, with $\mathcal{E}_t \subseteq \mathcal{K}_t$ at
 137 decision time. For scalability, \mathcal{K}_t may discard stale experts
 138 and reinitialize them upon re-entry (details in Section 4.2.4).

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

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

$$e_{t,k} := \hat{\mathbf{y}}_{t,k} - \mathbf{y}_t. \quad (6)$$

116 When $I_t = k$ is queried, the realized observation is
 117 $e_t := e_{t,I_t}$. We model residuals (rather than the nonnegative
 118 loss $\psi(e_{t,k})$) because the state-space emission model
 119 is defined on \mathbb{R}^{d_y} , preserving signed deviations (over- vs.
 120 under-prediction) that would be lost after applying ψ . The
 121 corresponding (potential) routing cost is
 122

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

123 where $\psi : \mathbb{R}^{d_y} \rightarrow \mathbb{R}_{\geq 0}$ is a known convex loss (e.g., squared
 124 error for $d_y = 1$ or squared norm $\psi(e) = \|e\|_2^2$ in general)
 125 and $\beta_k \geq 0$ is a known, expert-specific query fee. When
 126 $I_t = k$ is queried, the realized cost is $C_t := C_{t,I_t}$.

127 **Observation model (censoring).** At each round, the
 128 router selects an expert index $I_t \in \mathcal{E}_t$. Due to bandit-
 129 style feedback, it observes only the queried prediction
 130 $\hat{\mathbf{y}}_{t,I_t}$ (and hence only e_{t,I_t} and C_{t,I_t}); for $k \in \mathcal{E}_t \setminus \{I_t\}$,
 131 $(\hat{\mathbf{y}}_{t,k}, e_{t,k}, C_{t,k})$ remain unobserved. We denote the post-
 132 action feedback tuple by $O_t := (I_t, \hat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$.

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

137 A policy $\pi = (\pi_t)_{t=1}^T$ is a sequence of decision rules where
 138 $\pi_t(\cdot | \mathcal{F}_t)$ is an \mathcal{F}_t -measurable distribution over \mathcal{E}_t . The
 139 action is sampled as $I_t \sim \pi_t(\cdot | \mathcal{F}_t)$, so that $I_t \in \mathcal{E}_t$ almost
 140 surely.

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

1. **Decision-time revelation:** the environment reveals
 143 $(\mathbf{x}_t, \mathcal{E}_t)$, thereby determining \mathcal{F}_t .
2. **Action:** the router samples $I_t \sim \pi_t(\cdot | \mathcal{F}_t)$.
3. **Censored feedback:** the router observes $O_t =$
 144 $(I_t, \hat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$ and can evaluate the realized residual
 145 e_{t,I_t} and cost C_{t,I_t} .

146 **Non-stationarity and exogeneity.** We do not assume i.i.d.
 147 data: the joint law of $(\mathbf{x}_t, \mathcal{E}_t, \mathbf{y}_t)$ may drift over time (Section
 148 3.2). Concretely, we allow a sequence of time-varying
 149 conditional laws $\{\mathcal{D}_t\}_{t \geq 1}$ such that

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

150 This captures non-stationarity (e.g., concept drift or regime
 151 shifts). We additionally assume *exogeneity*: past routing

152 actions affect which expert predictions are observed, but do
 153 not influence the data-generating process. Equivalently,
 154 $(\mathbf{x}_t, \mathcal{E}_t, \mathbf{y}_t)$ is conditionally independent of past actions
 155 $I_{1:t-1}$ given $\sigma((\mathbf{x}_\tau, \mathcal{E}_\tau, \mathbf{y}_\tau)_{\tau < t})$.

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

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

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

$$k_t^* \in \arg \min_{k \in \mathcal{E}_t} \mathbb{E}[C_{t,k} | \mathcal{F}_t]. \quad (9)$$

159 Under full feedback, (9) is directly evaluable from con-
 160 temporaneous observations of all experts' costs. Under
 161 censoring, however, $C_{t,k}$ is observed only for the queried
 162 expert (Neu et al., 2010), so (9) is not directly computable.
 163 Since β_k is known, evaluating (9) reduces to forecasting
 164 $\mathbb{E}[\psi(e_{t,k}) | \mathcal{F}_t]$ for unqueried experts. In subsequent
 165 sections, we introduce a latent-state model that yields tractable
 166 one-step-ahead predictive beliefs $p(e_{t,k} | \mathcal{F}_t)$.

4.2. Generative Model: Factorized Switching LDS

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

170 We therefore model the *potential residuals* $e_{t,k} = \hat{\mathbf{y}}_{t,k} - \mathbf{y}_t$ from Section 4.1 as emissions of a **factorized switching**
 171 **linear dynamical system** (Bengio & Frasconi, 1994; Linderman et al., 2016; Hu et al., 2024). The central bottleneck
 172 is censoring: at round t we observe only the queried residual
 173 $e_t := e_{t,I_t}$, while $(e_{t,k})_{k \neq I_t}$ remain counterfactual. We address this by combining (i) a *switching* latent regime z_t
 174 to capture abrupt changes, (ii) a *shared* global factor g_t that
 175 couples experts and enables information transfer, and (iii)
 176 *idiosyncratic* expert-specific dynamics $u_{t,k}$. For scalability
 177 under a growing registry, our inference later maintains per-
 178 expert marginals via a factorized filtering approximation.
 179 The resulting linear-Gaussian structure yields Kalman-style
 180 updates and closed-form information quantities used in our
 181 routing rule.

4.2.1. LATENT STATE HIERARCHY

182 We represent non-stationarity via a two-level hierarchy sep-
 183 arating systemic shifts from expert-specific drifts. The hi-
 184 erarchy is designed so that a single queried residual can
 185 update a *shared* latent factor g_t , which immediately refines
 186 predictions for all experts. Expert-specific states $u_{t,k}$ then
 187 capture persistent idiosyncratic deviations that cannot be
 188 explained by global conditions alone.

165 **Context-dependent regime switching.** A discrete regime
 166 $z_t \in \{1, \dots, M\}$ selects the active dynamical law (e.g.,
 167 “bull” vs. “crisis”). While classical SLDSs often use a
 168 time-homogeneous transition matrix, we allow transition
 169 probabilities to depend on the observed context \mathbf{x}_t (input-
 170 driven switching; e.g., [Bengio & Frasconi \(1994\)](#)). Let
 171 $\Pi_\theta(\mathbf{x}_t) \in [0, 1]^{M \times M}$ be a row-stochastic matrix with
 172

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

173 This lets the filter incorporate exogenous signals in \mathbf{x}_t to update
 174 its regime belief before observing the queried residual
 175 e_t . Contexts that shift mass toward regime m favor experts
 176 with low mode- m predicted cost, yielding an interpretable
 177 link between \mathbf{x}_t , regimes, and expert specialization.
 178

179 We parameterize the logits of $\Pi_\theta(\mathbf{x}_t)$ with a low-rank scaled-
 180 attention form to control statistical and computational com-
 181 plexity ([Vaswani et al., 2017](#); [Kossaen et al., 2021](#); [Mehta et al., 2022](#)). Specifically, for a chosen bottleneck dimension
 182 d_{attn} , we compute $Q_\theta(\mathbf{x}_t), K_\theta(\mathbf{x}_t) \in \mathbb{R}^{M \times d_{\text{attn}}}$ and set
 183

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

184 so that $\text{rank}(S(\mathbf{x}_t)) \leq d_{\text{attn}}$. Applying a row-wise softmax
 185 yields the transition matrix:
 186

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

187 **Global factor dynamics.** Under censored feedback, the
 188 only way to learn about *unqueried* experts is to exploit struc-
 189 ture that couples them (see [Proposition 2](#)). We therefore
 190 introduce a continuous *shared* latent state $\mathbf{g}_t \in \mathbb{R}^{d_g}$ rep-
 191 resenting system-wide conditions (e.g., overall difficulty,
 192 market volatility, sensor drift) that affect many experts si-
 193 multaneously. Because \mathbf{g}_t appears in every expert’s residual
 194 model, updating \mathbf{g}_t from the single observed residual
 195 $e_t = e_{t, I_t}$ tightens the predictive beliefs for other experts
 196 $k \neq I_t$, providing the cross-expert information transfer
 197 needed for routing.
 198

199 Conditioned on $z_t = m$, we model \mathbf{g}_t with linear-Gaussian
 200 dynamics to retain Kalman-style updates and closed-form
 201 predictive quantities used later for exploration:
 202

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

203 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
 204 $(\mathbf{w}_t^{(g)})_t$ are independent across time and independent of
 205 all other process and emission noise terms.
 206

207 **Expert-specific dynamics.** Not all variation is shared: ex-
 208 perts can drift due to recalibration, local overfitting, model
 209 updates, or intermittent failures. We capture these *idiosyn-
 210 cratic* effects with a per-expert latent state $\mathbf{u}_{t,k} \in \mathbb{R}^{d_\alpha}$.
 211

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
 212 across experts and time. To maintain statistical strength
 213 under sparse feedback, we share the dynamics parameters
 214 $(\mathbf{A}_m^{(u)}, \mathbf{Q}_m^{(u)})$ across experts.
 215

Reliability composition and residual emission. Expert het-
 216 erogeneity is then expressed through (i) the expert-specific
 217 state realization $\mathbf{u}_{t,k}$ and (ii) expert-specific loadings \mathbf{B}_k ,
 218 which determine how each expert responds to the shared
 219 factor \mathbf{g}_t .
 220

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

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

Given regime $z_t = m$, context \mathbf{x}_t , and latent states
 224 $(\mathbf{g}_t, \mathbf{u}_{t,k})$, the signed residual $e_{t,k} = \hat{\mathbf{y}}_{t,k} - \mathbf{y}_t$ is gener-
 225 ated by the linear-Gaussian emission

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

where $\mathbf{R}_{m,k} \in \mathbb{S}_{++}^{d_y}$ is an expert- and regime-specific noise covariance.
 226

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.
 227

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.
 228

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.
 229

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 $(e_{t,j}^{\text{pred}}, e_{t,I_t}^{\text{pred}})$ denote the one-step-ahead predictive residuals under $p(e_{t,\cdot} | \mathcal{F}_t, z_t = m)$. Assume that this predictive pair is jointly Gaussian conditional on \mathcal{G}_t and that $\text{Cov}(e_{t,I_t}^{\text{pred}} | \mathcal{G}_t)$ is non-singular (e.g., $\mathbf{R}_{m,I_t} \succ \mathbf{0}$). Then

$$\begin{aligned}\mathbb{E}[e_{t,j}^{\text{pred}} | e_t, \mathcal{G}_t] &= \mathbb{E}[e_{t,j}^{\text{pred}} | \mathcal{G}_t] \\ \iff \text{Cov}(e_{t,j}^{\text{pred}}, e_{t,I_t}^{\text{pred}} | \mathcal{G}_t) &= \mathbf{0}.\end{aligned}$$

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

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 $(\mathcal{F}_t, z_t = m)$, $\text{Cov}(e_{t,j}^{\text{pred}}, e_{t,i}^{\text{pred}})$ contains the shared-factor term

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

where $\Sigma_{g,t|t-1}^{(m)}$ 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 $e_{t,k}^{\text{pred}} \sim p(e_{t,k} | \mathcal{F}_t)$ and predicted cost

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

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

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

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:

$$\text{IG}_t(k) := \mathcal{I}((z_t, \mathbf{g}_t); e_{t,k}^{\text{pred}} | \mathcal{F}_t). \quad (16)$$

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

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

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

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 $\tau_{\text{last}}(k) \in \{0, 1, \dots, t-1\}$ 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 Δ_{\max} steps, where $\Delta_{\max} \geq 1$ is a user-chosen staleness horizon:

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

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

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

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

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

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 $\alpha_{t,\ell}$ 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; Δ_{\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

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

Applying the regime-conditioned time update (12), the corresponding predictive prior for $\mathbf{u}_{t,j}$ is

$$\mathbf{u}_{t,j} \mid (z_t = m) \sim \mathcal{N}\left(\mathbf{A}_m^{(u)} \mu_{\text{init},j}^{(m)}, \mathbf{A}_m^{(u)} \Sigma_{\text{init},j}^{(m)} (\mathbf{A}_m^{(u)})^\top + \mathbf{Q}_m^{(u)}\right). \quad (21)$$

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 $\alpha_{t,j} = \mathbf{B}_j \mathbf{g}_t + \mathbf{u}_{t,j}$.

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 $\text{Cov}(\mathbf{g}_t, \mathbf{u}_{t,k} \mid \cdot) = \mathbf{0}$ and $\text{Cov}(\mathbf{u}_{t,i}, \mathbf{u}_{t,j} \mid \cdot) = \mathbf{0}$ for $i \neq j$), for any experts $j \neq k$,*

$$\text{Cov}(\alpha_{t,j}, \alpha_{t,k} \mid \mathcal{F}_t, z_t = m) = \mathbf{B}_j \Sigma_{g,t|t-1}^{(m)} \mathbf{B}_k^\top,$$

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 $\alpha_{t,j}$ and $\alpha_{t,k}$ are not independent and hence $\mathcal{I}(\alpha_{t,j}; \alpha_{t,k} \mid \mathcal{F}_t, z_t = m) > 0$.

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 \hat{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 separate 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 $z_t := 1 + \lfloor \frac{t-1}{L} \rfloor \bmod 2$. The target follows a regime-dependent AR(1), and the

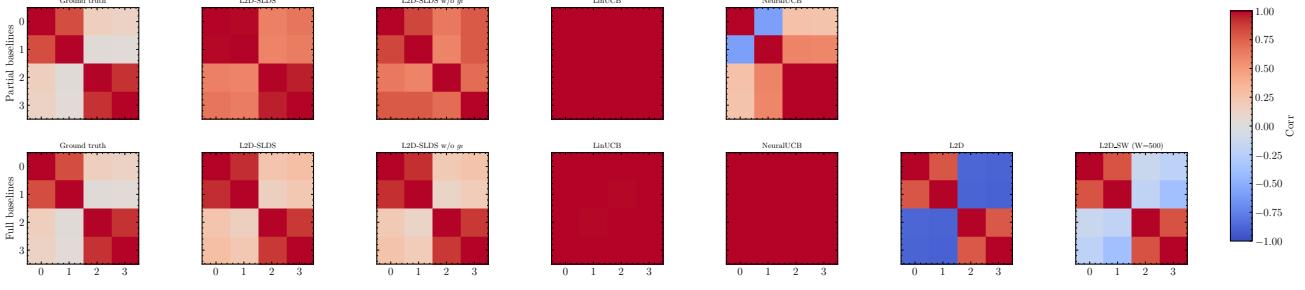


Figure 1. Regime-0 expert dependence in the synthetic transfer experiment. Each heatmap shows the pairwise Pearson correlation (color: $[-1, 1]$) between experts’ per-round losses (experts indexed 0–3). Top row: partial feedback (only queried losses observed); bottom row: full feedback. Columns (left-to-right) 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.

context is the one-step lag:

$$y_t = 0.8 y_{t-1} + d_{z_t} + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \sigma_y^2). \quad (22)$$

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. 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:

$$\hat{y}_{t,k} := 0.8 y_{t-1} + b_k + \varepsilon_{t,k}. \quad (23)$$

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

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

with independent components $s_{t,1}, s_{t,2}, (\tilde{\varepsilon}_{t,k})_k$ and regime-dependent variances $s_{t,1} \sim \mathcal{N}(0, \sigma_{z_t,1}^2), s_{t,2} \sim \mathcal{N}(0, \sigma_{z_t,2}^2), \tilde{\varepsilon}_{t,k} \sim \mathcal{N}(0, \sigma_{id}^2)$, where $(\sigma_{1,1}^2, \sigma_{1,2}^2) = (\sigma_{hi}^2, \sigma_{lo}^2)$ and $(\sigma_{2,1}^2, \sigma_{2,2}^2) = (\sigma_{lo}^2, \sigma_{hi}^2)$ with $\sigma_{hi}^2 \gg \sigma_{lo}^2$. 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.

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 | 13.58 ± 0.07 | 10.17 ± 0.01 |
| L2D-SLDS w/o 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 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 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, 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.
- 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.
- Sample efficient learning of predictors that complement humans, 2022.

- 440 Joshi, S., Parbhoo, S., and Doshi-Velez, F. Learning-to-defer
441 for sequential medical decision-making under uncertainty.
442 *arXiv preprint arXiv:2109.06312*, 2021.
- 443 Kalman, R. E. A new approach to linear filtering and pre-
444 diction problems. 1960.
- 445 Kossaifi, J., Band, N., Lyle, C., Gomez, A. N., Rainforth,
446 T., and Gal, Y. Self-attention between datapoints: Going
447 beyond individual input-output pairs in deep learning.
448 *Advances in Neural Information Processing Systems*, 34:
449 28742–28756, 2021.
- 450 Li, L., Chu, W., Langford, J., and Schapire, R. E. A
451 contextual-bandit approach to personalized news article
452 recommendation. In *Proceedings of the 19th international
453 conference on World wide web*, pp. 661–670, 2010.
- 454 Linderman, S. W., Miller, A. C., Adams, R. P., Blei, D. M.,
455 Paninski, L., and Johnson, M. J. Recurrent switching lin-
456 ear dynamical systems. *arXiv preprint arXiv:1610.08466*,
457 2016.
- 458 Madras, D., Pitassi, T., and Zemel, R. Predict responsibly:
459 improving fairness and accuracy by learning to defer.
460 *Advances in neural information processing systems*, 31,
461 2018.
- 462 Mao, A., Mohri, C., Mohri, M., and Zhong, Y. Two-stage
463 learning to defer with multiple experts. In *Thirty-seventh
464 Conference on Neural Information Processing Systems*,
465 2023. URL <https://openreview.net/forum?id=GIlsH0T4b2>.
- 466 Mao, A., Mohri, M., and Zhong, Y. Principled approaches
467 for learning to defer with multiple experts. In *ISAIM*,
468 2024a.
- 469 Mao, A., Mohri, M., and Zhong, Y. Realizable h -consistent
470 and bayes-consistent loss functions for learning to defer.
471 *Advances in neural information processing systems*, 37:
472 73638–73671, 2024b.
- 473 Mao, A., Mohri, M., and Zhong, Y. Regression with multi-
474 expert deferral. In *Proceedings of the 41st International
475 Conference on Machine Learning*, ICML’24. JMLR.org,
476 2024c.
- 477 Mao, A., Mohri, M., and Zhong, Y. Mastering multiple-
478 expert routing: Realizable $\$h\$$ -consistency and strong
479 guarantees for learning to defer. In *Forty-second Interna-
480 tional Conference on Machine Learning*, 2025.
- 481 Mehta, S., Székely, É., Beskow, J., and Henter, G. E. Neural
482 hmms are all you need (for high-quality attention-free tts).
483 In *ICASSP 2022-2022 IEEE International Conference on
484 Acoustics, Speech and Signal Processing (ICASSP)*, pp.
485 7457–7461. IEEE, 2022.
- 486 Montreuil, Y., Carlier, A., Ng, L. X., and Ooi, W. T. Why
487 ask one when you can ask k ? two-stage learning-to-defer
488 to the top- k experts. *arXiv preprint arXiv:2504.12988*,
489 2025a.
- 490 Montreuil, Y., Heng, Y. S., Carlier, A., Ng, L. X., and Ooi,
491 W. T. A two-stage learning-to-defer approach for multi-
492 task learning. In *Forty-second International Conference
493 on Machine Learning*, 2025b.
- 494 Montreuil, Y., Yeo, S. H., Carlier, A., Ng, L. X., and Ooi,
495 W. T. Optimal query allocation in extractive qa with llms:
496 A learning-to-defer framework with theoretical guaran-
497 tees. *arXiv preprint arXiv:2410.15761*, 2025c.
- 498 Mozannar, H. and Sontag, D. Consistent estimators for learn-
499 ing to defer to an expert. In *Proceedings of the 37th Inter-
500 national Conference on Machine Learning*, ICML’20. JMLR.org, 2020.
- 501 Mozannar, H., Lang, H., Wei, D., Sattigeri, P., Das, S.,
502 and Sontag, D. A. Who should predict? exact algo-
503 rithms for learning to defer to humans. In *Inter-
504 national Conference on Artificial Intelligence and Statistics*,
505 2023. URL <https://api.semanticscholar.org/CorpusID:255941521>.
- 506 Narasimhan, H., Jitkrittum, W., Menon, A. K., Rawat,
507 A., and Kumar, S. Post-hoc estimators for learning
508 to defer to an expert. In Koyejo, S., Mohamed, S.,
509 Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.),
510 *Advances in Neural Information Processing Systems*, volume 35, pp. 29292–29304. Curran Associates, Inc.,
511 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/bc8f76d9caadd48f77025b1c889d2e2d-Paper-Conference.pdf.
- 512 Neu, G., Antos, A., György, A., and Szepesvári, C. On-
513 line markov decision processes under bandit feedback.
514 *Advances in Neural Information Processing Systems*, 23,
515 2010.
- 516 Nguyen, C. C., Do, T.-T., and Carneiro, G. Probabilistic
517 learning to defer: Handling missing expert annotations
518 and controlling workload distribution. In *The Thirteenth
519 International Conference on Learning Representations*,
520 2025.
- 521 Palomba, F., Pugnana, A., Alvarez, J. M., and Ruggieri, S. A
522 causal framework for evaluating deferring systems. In *The
523 28th International Conference on Artificial Intelligence
524 and Statistics*, 2025. URL <https://openreview.net/forum?id=mkkFubLdNW>.
- 525 Rabiner, L. and Juang, B. An introduction to hidden markov
526 models. *ieee assp magazine*, 3(1):4–16, 2003.

- 495 Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning
496 internal representations by error propagation. Technical report,
497 1985.
- 498 Russo, D. and Van Roy, B. Learning to optimize via
499 information-directed sampling. *Advances in neural infor-*
500 *mation processing systems*, 27, 2014.
- 502 Sezer, O. B., Gudelek, M. U., and Ozbayoglu, A. M. Fi-
503 nancial time series forecasting with deep learning: A
504 systematic literature review: 2005–2019. *Applied soft*
505 *computing*, 90:106181, 2020.
- 506
- 507 Shumway, R. H. Time series analysis and its applications:
508 With r examples, 2006.
- 509
- 510 Strong, J., Men, Q., and Noble, A. Towards human-AI
511 collaboration in healthcare: Guided deferral systems with
512 large language models. In *ICML 2024 Workshop on LLMs*
513 *and Cognition*, 2024. URL <https://openreview.net/forum?id=4c5rg9y4me>.
- 514
- 515 Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones,
516 L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I.
517 Attention is all you need. In Guyon, I., Luxburg, U. V.,
518 Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S.,
519 and Garnett, R. (eds.), *Advances in Neural Information*
520 *Processing Systems*, volume 30. Curran Associates, Inc.,
521 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf.
- 522
- 523
- 524
- 525
- 526 Verma, R., Barrejon, D., and Nalisnick, E. Learning to defer
527 to multiple experts: Consistent surrogate losses, confi-
528 dence calibration, and conformal ensembles. In *Inter-*
529 *national Conference on Artificial Intelligence and Statistics*,
530 2022. URL <https://api.semanticscholar.org/CorpusID:253237048>.
- 531
- 532 Wei, Z., Cao, Y., and Feng, L. Exploiting human-ai depen-
533 dence for learning to defer. In *Forty-first International*
534 *Conference on Machine Learning*, 2024.
- 535
- 536 Welch, G., Bishop, G., et al. An introduction to the kalman
537 filter. 1995.
- 538
- 539 Zhou, D., Li, L., and Gu, Q. Neural contextual bandits with
540 ucb-based exploration, 2020. URL <https://arxiv.org/abs/1911.04462>.
- 541
- 542
- 543
- 544
- 545
- 546
- 547
- 548
- 549

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

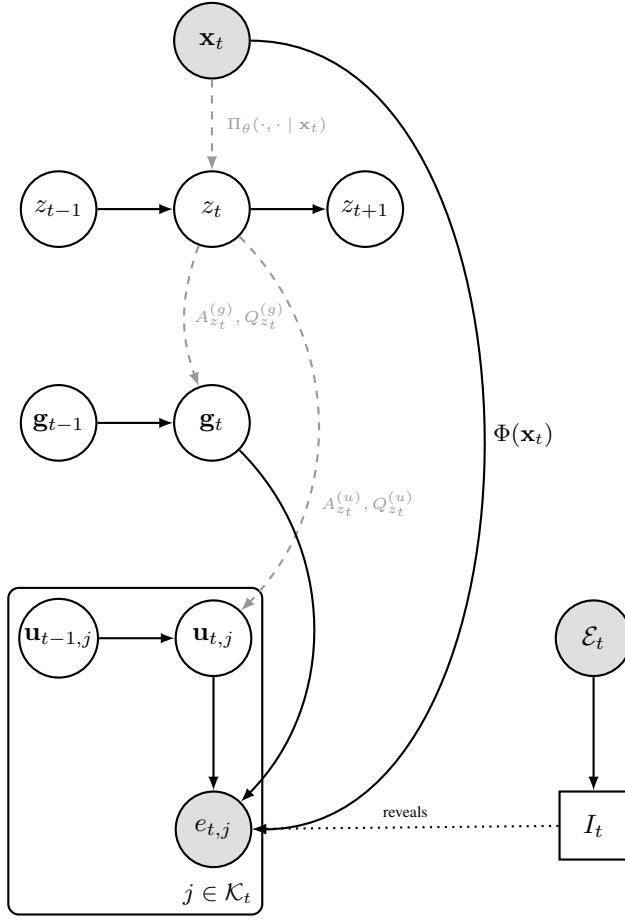
| Symbol | Meaning |
|---|--|
| Time, data, and actions | |
| $t \in [T]$ | Round index; finite horizon T . |
| $\mathbf{x}_t \in \mathbb{R}^d$ | Observed context at round t . |
| $\mathbf{y}_t \in \mathbb{R}^{d_y}$ | Target/label at round t . |
| \mathcal{E}_t | Set of available experts at round t (may vary with time). |
| $I_t \in \mathcal{E}_t$ | Queried expert at round t . |
| $\hat{\mathbf{y}}_{t,k} \in \mathbb{R}^{d_y}$ | Prediction of expert k at round t . |
| $O_t = (I_t, \hat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$ | 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 | |
| $e_{t,k} = \hat{\mathbf{y}}_{t,k} - \mathbf{y}_t$ | 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^* | 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. |
| $\alpha_{t,k} = \mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k}$ | Latent “reliability” vector of expert k at time t . |

| 605 606 607 608 609 610 611 | 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 | | |
| 612 | $\bar{w}_t^{(m)} = \mathbb{P}(z_t = m \mathcal{F}_t)$ | Predictive (pre-observation) regime weight. |
| 613 | $w_t^{(m)} = \mathbb{P}(z_t = m \mathcal{F}_t, I_t, e_t)$ | Filtering (post-observation) regime weight. |
| 614 | $\gamma_t^{(m)}$ | Posterior regime responsibility used in (Monte Carlo) EM. |
| 615 | $\xi_{t-1}^{(\ell,m)}$ | Posterior transition responsibility used in (Monte Carlo) EM. |
| 616 | $e_{t,k}^{\text{pred}}$ | One-step-ahead predictive residual random variable for expert k . |
| 617 | $\bar{C}_{t,k}^{\text{pred}}$ | Predicted cost: $\mathbb{E}[\psi(e_{t,k}^{\text{pred}}) \mathcal{F}_t] + \beta_k$. |
| 618 | k_t^{pred} | Myopic predicted-cost minimizer in \mathcal{E}_t . |
| 619 | $\Delta_t(k)$ | Predicted cost gap relative to k_t^{pred} . |
| 620 | $\text{IG}_t(k)$ | Information gain: $\mathcal{I}((z_t, \mathbf{g}_t); e_{t,k}^{\text{pred}} \mathcal{F}_t)$. |
| 621 | ϵ_w | Mixing floor for predictive mode weights $\bar{w}_t^{(m)}$ in IMM updates. |
| 622 | ϵ_{IG} | Information-gain floor used in IDS (avoids division by zero and clamps Monte Carlo noise). |
| 623 | S | Monte Carlo sample size used to estimate the mode-identification term in $\text{IG}_t(k)$. |
| Dynamic registry management | | |
| 624 | \mathcal{K}_t | Maintained expert registry: experts for which per-expert filtering marginals (hence $\mathbf{u}_{t,k}$) are stored. |
| 625 | $\mathcal{E}_t^{\text{init}} = \mathcal{E}_t \setminus \mathcal{K}_{t-1}$ | Entering experts at round t (new or re-entering after pruning). |
| 626 | $\tau_{\text{last}}(k)$ | Last round at which expert k was queried. |
| 627 | Δ_{\max} | Staleness horizon controlling pruning. |
| 628 | $\mathcal{K}_t^{\text{stale}}$ | 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.

660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688



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

691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709

D. Algorithms

710

D.1. Router and Filtering Recursion

711

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).

712
713
714

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

716 1: **Input:** horizon T ; parameters Θ ; feature map Φ ; loss ψ ; fees $(\beta_k)_k$; default entry priors $(\mu_{\text{init},\text{def}}^{(m)}, \Sigma_{\text{init},\text{def}}^{(m)})_{m=1}^M$;
 717 staleness Δ_{\max} ; floors $(\epsilon_w, \epsilon_{\text{IG}})$; Monte Carlo budget S for $\text{IG}_t(k)$ (Appendix E).

718 2: **Initialize:** $w_0^{(m)} \leftarrow \mathbb{P}(z_1 = m); (\mu_{g,0|0}^{(m)}, \Sigma_{g,0|0}^{(m)})_{m=1}^M; \mathcal{K}_0 \leftarrow \emptyset; \tau_{\text{last}}(k) \leftarrow 0$ for all k .

719 3: **for** $t = 1$ to T **do**

720 4: Observe $(\mathbf{x}_t, \mathcal{E}_t)$.

721 5: **Registry:** $\mathcal{E}_t^{\text{init}} \leftarrow \mathcal{E}_t \setminus \mathcal{K}_{t-1}$.

722 6: **Registry:** $\mathcal{K}_t^{\text{stale}} \leftarrow \{k \in \mathcal{K}_{t-1} \setminus \mathcal{E}_t : t - \tau_{\text{last}}(k) > \Delta_{\max}\}$.

723 7: **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}

724 8: For each $k \in \mathcal{E}_t^{\text{init}}$, set $(\mu_{\text{init},k}^{(m)}, \Sigma_{\text{init},k}^{(m)})_{m=1}^M$ (default: $(\mu_{\text{init},\text{def}}^{(m)}, \Sigma_{\text{init},\text{def}}^{(m)})$).

725 9: {**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
 726 moment-match mixed priors at time $t - 1$.}

727 10: {**Time update:** apply Eqs. 11, 12, and 21 to obtain $(\mu_{g,t|t-1}^{(m)}, \Sigma_{g,t|t-1}^{(m)})$ and $(\mu_{u,k,t|t-1}^{(m)}, \Sigma_{u,k,t|t-1}^{(m)})$ for $k \in \mathcal{K}_t$.}

728 11: For each $m \in [M]$ and $k \in \mathcal{E}_t$, compute $(\bar{e}_{t,k}^{\text{pred},(m)}, \Sigma_{t,k}^{\text{pred},(m)})$ from Eq. 14.

729 12: For each $k \in \mathcal{E}_t$, set $\bar{C}_{t,k}^{\text{pred}} \leftarrow \sum_{m=1}^M \bar{w}_t^{(m)} (\mathbb{E}_{e \sim \mathcal{N}(\bar{e}_{t,k}^{\text{pred},(m)}, \Sigma_{t,k}^{\text{pred},(m)})} [\psi(e)] + \beta_k)$.

730 13: $k_t^{\text{pred}} \in \arg \min_{k \in \mathcal{E}_t} \bar{C}_{t,k}^{\text{pred}}$; $\Delta_t(k) \leftarrow \bar{C}_{t,k}^{\text{pred}} - \bar{C}_{t,k_t^{\text{pred}}}^{\text{pred}}$ for all $k \in \mathcal{E}_t$.

731 14: Compute $\text{IG}_t(k) = \mathcal{I}((z_t, \mathbf{g}_t); e_{t,k}^{\text{pred}} \mid \mathcal{F}_t)$ as in Appendix E; clamp $\text{IG}_t(k) \leftarrow \max(\text{IG}_t(k), \epsilon_{\text{IG}})$.

732 15: Choose $I_t \in \arg \min_{k \in \mathcal{E}_t} \Delta_t(k)^2 / \text{IG}_t(k)$.

733 16: Observe $(\hat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$, set $e_t \leftarrow \hat{\mathbf{y}}_{t,I_t} - \mathbf{y}_t$, and update $\tau_{\text{last}}(I_t) \leftarrow t$.

734 17: Run Algorithm 2 to obtain w_t and updated posteriors for \mathbf{g}_t and $(\mathbf{u}_{t,k})_{k \in \mathcal{K}_t}$.

735 18: Optional: update Θ via Algorithm 4.

736 19: **end for**

740
741
742
743
744
745
746

747 **Algorithm 2** CORRECT: Queried Kalman Update and Mode Posterior

748 1: **Input:** \mathbf{x}_t , queried residual e_t , queried expert I_t ; predictive weights \bar{w}_t ; predictive states $\{\mu_{g,t|t-1}^{(m)}, \Sigma_{g,t|t-1}^{(m)}\}_{m=1}^M$ and
 749 $\{\mu_{u,k,t|t-1}^{(m)}, \Sigma_{u,k,t|t-1}^{(m)}\}_{m \in [M], k \in \mathcal{K}_t}$; parameters $(\mathbf{B}_{I_t}, (\mathbf{R}_{m,I_t})_{m=1}^M)$.

750 2: $H_t \leftarrow [\Phi(\mathbf{x}_t)^\top \mathbf{B}_{I_t} \Phi(\mathbf{x}_t)^\top]$.

751 3: **for** $m = 1$ to M **do**

752 4: $\mu_{s,t|t-1}^{(m)} \leftarrow [(\mu_{g,t|t-1}^{(m)})^\top (\mu_{u,I_t,t|t-1}^{(m)})^\top]^\top$.

753 5: $\Sigma_{s,t|t-1}^{(m)} \leftarrow \text{diag}(\Sigma_{g,t|t-1}^{(m)}, \Sigma_{u,I_t,t|t-1}^{(m)})$.

754 6: $\bar{e}_{t,I_t}^{\text{pred},(m)} \leftarrow H_t \mu_{s,t|t-1}^{(m)}$, $\Sigma_{t,I_t}^{\text{pred},(m)} \leftarrow H_t \Sigma_{s,t|t-1}^{(m)} H_t^\top + \mathbf{R}_{m,I_t}$.

755 7: $K_t^{(m)} \leftarrow \Sigma_{s,t|t-1}^{(m)} H_t^\top (\Sigma_{t,I_t}^{\text{pred},(m)})^{-1}$.

756 8: $\mu_{s,t|t}^{(m)} \leftarrow \mu_{s,t|t-1}^{(m)} + K_t^{(m)} (e_t - \bar{e}_{t,I_t}^{\text{pred},(m)})$.

757 9: $\Sigma_{s,t|t}^{(m)} \leftarrow \Sigma_{s,t|t-1}^{(m)} - K_t^{(m)} \Sigma_{t,I_t}^{\text{pred},(m)} (K_t^{(m)})^\top$.

758 10: Project to factorized marginals: keep only the diagonal blocks for \mathbf{g}_t and \mathbf{u}_{t,I_t} ; set $(\mu_{u,k,t|t}^{(m)}, \Sigma_{u,k,t|t}^{(m)}) \leftarrow$
 759 $(\mu_{u,k,t|t-1}^{(m)}, \Sigma_{u,k,t|t-1}^{(m)})$ for $k \neq I_t$.

760 11: $\mathcal{L}_t^{(m)} \leftarrow \mathcal{N}(e_t; \bar{e}_{t,I_t}^{\text{pred},(m)}, \Sigma_{t,I_t}^{\text{pred},(m)})$.

761 12: **end for**

762 13: $w_t^{(m)} \leftarrow \frac{\mathcal{L}_t^{(m)} \bar{w}_t^{(m)}}{\sum_{\ell=1}^M \mathcal{L}_t^{(\ell)} \bar{w}_t^{(\ell)}}$ for all $m \in [M]$.

763 14: **Return:** w_t and updated posteriors $\{\mu_{g,t|t}^{(m)}, \Sigma_{g,t|t}^{(m)}\}_{m=1}^M, \{\mu_{u,k,t|t}^{(m)}, \Sigma_{u,k,t|t}^{(m)}\}_{m \in [M], k \in \mathcal{K}_t}$.

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)

```

775 1: Input: window  $\mathcal{T} = \{t_a, \dots, t_b\}$ ; stream  $(\mathbf{x}_t, I_t, e_t)_{t \in \mathcal{T}}$  with  $e_t = \hat{\mathbf{y}}_{t, I_t} - \mathbf{y}_t$ ; feature map  $\Phi$ ; EM iterations  $N_{\text{EM}}$ ;
776   MCMC settings ( $N_{\text{samp}}, N_{\text{burn}}$ ); occupancy floor  $\epsilon_N > 0$ ; (optional) regularization  $(\lambda_\theta, \lambda_B)$  for  $(\Pi_\theta, \mathbf{B})$ .
777 2:  $\mathcal{K}_{\mathcal{T}}^{\text{qry}} \leftarrow \{I_t : t \in \mathcal{T}\}$ . {Experts queried in the window}
778 3: Initialize: parameters  $\Theta^{(0)}$  and priors for  $z_{t_a}, \mathbf{g}_{t_a}$ , and  $\{\mathbf{u}_{t_a, k}\}_{k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}}$ .
779 4: for iteration  $i = 1$  to  $N_{\text{EM}}$  do
780   5: // E-step: Monte Carlo posterior (blocked Gibbs)
781   6: Draw samples from  $p(z_{t_a:t_b}, \mathbf{g}_{t_a:t_b}, (\mathbf{u}_{t_a:t_b, k})_{k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}} | (\mathbf{x}_t, I_t, e_t)_{t \in \mathcal{T}}, \Theta^{(i-1)})$  by alternating:
782     1) sample  $z_{t_a:t_b}$  via FFBS from the conditional HMM given  $\mathbf{g}_{t_a:t_b}$  and  $(\mathbf{u}_{t_a:t_b, k})_k$ ;
783     2) sample  $\mathbf{g}_{t_a:t_b}$  via Kalman smoothing given  $z_{t_a:t_b}$  and  $(\mathbf{u}_{t_a:t_b, k})_k$ ;
784     3) for each  $k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}$ , sample  $\mathbf{u}_{t_a:t_b, k}$  via Kalman smoothing using only  $\{(t, e_t) : I_t = k\}$ .
785 10: From post-burn-in samples, estimate  $\gamma_t^{(m)} \approx \mathbb{P}(z_t = m | \cdot)$ ,  $\xi_{t-1}^{(\ell, m)} \approx \mathbb{P}(z_{t-1} = \ell, z_t = m | \cdot)$ , and the moments
786   used in the M-step.
787 11: // M-step: MAP / regularized updates (factorized moments)
788 12: Update  $(\mathbf{A}_m^{(g)}, \mathbf{Q}_m^{(g)})_{m=1}^M$  and  $(\mathbf{A}_m^{(u)}, \mathbf{Q}_m^{(u)})_{m=1}^M$  using weighted least-squares/covariance matching (skip updates
789   when the effective count is  $\leq \epsilon_N$ ; see below).
790 13: Update  $(\mathbf{B}_k)_{k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}}$  and  $(\mathbf{R}_{m, k})_{m \in [M], k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}}$  via weighted linear-Gaussian regression (skip updates when the
791   effective count is  $\leq \epsilon_N$ ; see below).
792 14: Update  $\theta$  by maximizing  $\sum_{t \in \mathcal{T} \setminus \{t_a\}} \sum_{\ell, m} \xi_{t-1}^{(\ell, m)} \log \Pi_\theta(\mathbf{x}_t)_{\ell m} - \frac{\lambda_\theta}{2} \|\theta\|_2^2$ .
793 15: end for
794 16: Return:  $\Theta^{(N_{\text{EM}})}$ 

```

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 $(\mathbf{u}_{t_a:t_b, k})_k$, 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

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

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

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

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

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

825 If $N_m^{(u)} > \epsilon_N$, set $\mathbf{A}_m^{(u)} \leftarrow S_{uu^-}^{(m)} \left(S_{u^-u^-}^{(m)} \right)^{-1}$ and
 826

827
$$\mathbf{Q}_m^{(u)} \leftarrow \frac{1}{N_m^{(u)}} \sum_{t=t_a+1}^{t_b} \sum_{k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}} \left\langle \mathbf{1}\{z_t = m\} \left(\mathbf{u}_{t,k} - \mathbf{A}_m^{(u)} \mathbf{u}_{t-1,k} \right) \left(\mathbf{u}_{t,k} - \mathbf{A}_m^{(u)} \mathbf{u}_{t-1,k} \right)^{\top} \right\rangle.$$

 829

830 **Emission parameters** ($\mathbf{B}_k, \mathbf{R}_{m,k}$). Fix an expert $k \in \mathcal{K}_{\mathcal{T}}^{\text{qry}}$ and denote $\Phi_t := \Phi(\mathbf{x}_t)$. For each $t \in \mathcal{T}$ with $I_t = k$, define
 831 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 $X_t := (\mathbf{g}_t^{\top} \otimes \Phi_t^{\top}) \in$
 832 $\mathbb{R}^{d_y \times (d_g d_{\alpha})}$, so that $y_t = X_t \text{vec}(\mathbf{B}_k) + v_t$ with $v_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{z_t,k})$. Here \otimes is the Kronecker product and $\text{vec}(\cdot)$ stacks
 833 matrix columns. Given current $(\mathbf{R}_{m,k})_{m=1}^M$, a (ridge) generalized least-squares update is
 834

835
$$\text{vec}(\mathbf{B}_k) \leftarrow \left(\sum_{t \in \mathcal{T}: I_t=k} \sum_{m=1}^M \left\langle \mathbf{1}\{z_t = m\} X_t^{\top} \mathbf{R}_{m,k}^{-1} X_t \right\rangle + \lambda_B \mathbf{I} \right)^{-1} \left(\sum_{t \in \mathcal{T}: I_t=k} \sum_{m=1}^M \left\langle \mathbf{1}\{z_t = m\} X_t^{\top} \mathbf{R}_{m,k}^{-1} y_t \right\rangle \right).$$

 836

837 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
 838 weighted covariance matching:

839
$$\mathbf{R}_{m,k} \leftarrow \frac{1}{N_{m,k}} \sum_{t \in \mathcal{T}: I_t=k} \left\langle \mathbf{1}\{z_t = m\} r_{t,k} r_{t,k}^{\top} \right\rangle, \quad r_{t,k} := e_t - \Phi_t^{\top} (\mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k}).$$

 840

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

- 842 1: **Input:** current time t ; stream $(\mathbf{x}_{\tau}, \mathcal{E}_{\tau}, I_{\tau}, e_{\tau})_{\tau \leq t}$; current parameters $\Theta^{(t-1)}$; window length W ; update period K ; EM
 843 iterations $N_{\text{EM}}^{\text{win}}$; MCMC settings; occupancy floor ϵ_N ; hyperparameters as in Algorithm 3.
 844 2: $\tau_0 \leftarrow t - W + 1$.
 845 3: **if** $t < W$ **or** $t \bmod K \neq 0$ **then**
 846 4: $\Theta^{(t)} \leftarrow \Theta^{(t-1)}$ and **return**.
 847 5: **end if**
 848 6: Define window $\mathcal{T}_t \leftarrow \{\tau_0, \dots, t\}$ and $\mathcal{K}_{\mathcal{T}_t}^{\text{qry}} \leftarrow \{I_{\tau} : \tau \in \mathcal{T}_t\}$.
 849 7: Initialize priors for z_{τ_0} , \mathbf{g}_{τ_0} , and $\{\mathbf{u}_{\tau_0,k}\}_{k \in \mathcal{K}_{\mathcal{T}_t}^{\text{qry}}}$ from the stored filtering belief at time $\tau_0 - 1$ (plus one time-update); if
 850 unavailable, use conservative default priors.
 851 8: Run Algorithm 3 on \mathcal{T}_t with initialization $\Theta^{(t-1)}$, floor ϵ_N , and $N_{\text{EM}}^{\text{win}}$ iterations.
 852 9: 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).
 853 10: **Return:** updated parameters $\Theta^{(t)}$.
-

860 **D.3. Cross-Covariance in the Exact Update**

861 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
 862 $\mathbf{u}_t := \mathbf{u}_{t,I_t}$ and write $\mathbf{s}_t = (\mathbf{g}_t, \mathbf{u}_t)$. Even if the predictive covariance is block-diagonal (our factorized predictive belief),
 863 the *exact* posterior covariance after conditioning on the queried residual e_t generally has non-zero off-diagonal blocks:
 864

865
$$\Sigma_{s,t|t}^{(m)} = \begin{bmatrix} \Sigma_{g,t|t}^{(m)} & \Sigma_{gu,t|t}^{(m)} \\ (\Sigma_{gu,t|t}^{(m)})^{\top} & \Sigma_{u,t|t}^{(m)} \end{bmatrix}, \quad \Sigma_{gu,t|t}^{(m)} \neq \mathbf{0} \text{ in general.}$$

 866

867 These cross terms arise because the observation matrix $H_t = [\Phi(\mathbf{x}_t)^{\top} \mathbf{B}_{I_t} \quad \Phi(\mathbf{x}_t)^{\top}]$ couples \mathbf{g}_t and \mathbf{u}_{t,I_t} . Retaining $\Sigma_{gu,t|t}^{(m)}$
 868 would propagate correlation into subsequent steps and into cross-expert predictive covariances.
 869

870 **Closed-form cross-covariance.** Write the Kalman gain in block form $K_t^{(m)} = [(K_{g,t}^{(m)})^{\top} (K_{u,t}^{(m)})^{\top}]^{\top}$, and let $\Sigma_{t,I_t}^{\text{pred},(m)}$
 871 denote the innovation covariance of the queried residual as in Algorithm 2: $\Sigma_{t,I_t}^{\text{pred},(m)} = H_t \Sigma_{s,t|t-1}^{(m)} H_t^{\top} + \mathbf{R}_{m,I_t}$. Then
 872 the covariance update can be written as $\Sigma_{s,t|t}^{(m)} = \Sigma_{s,t|t-1}^{(m)} - K_t^{(m)} \Sigma_{t,I_t}^{\text{pred},(m)} (K_t^{(m)})^{\top}$. If the predictive covariance is
 873 block-diagonal, then the off-diagonal block is
 874

875
$$\Sigma_{gu,t|t}^{(m)} = -K_{g,t}^{(m)} \Sigma_{t,I_t}^{\text{pred},(m)} (K_{u,t}^{(m)})^{\top} = -\Sigma_{g,t|t-1}^{(m)} H_{g,t}^{\top} (\Sigma_{t,I_t}^{\text{pred},(m)})^{-1} H_{u,t} \Sigma_{u,t|t-1}^{(m)},$$

 876

880 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
 881 is non-zero.
 882

883 **Why we discard it.** Keeping $\Sigma_{gu,t|t}^{(m)}$ is exact but undermines the factorized SLDS approximation that enables scalable
 884 inference under a growing expert registry. Once \mathbf{g}_t becomes correlated with \mathbf{u}_{t,I_t} , future prediction steps introduce non-zero
 885 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
 886 dependence across many experts. This breaks the stored-marginal structure, increases both compute and memory (scaling
 887 with the full registry size), and complicates closed-form quantities used in Section 4.2.3 (e.g., the Gaussian channel form and
 888 information gain). For these reasons, we project back to a factorized belief after each update and retain only the diagonal
 889 blocks as in Algorithm 2.
 890

891 E. Information Gain for Exploration

893 **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
 894 rather than conditioning only on \mathbf{g}_t . By the chain rule for mutual information:
 895

$$896 \quad 897 \quad 898 \quad 899 \quad 900 \quad 901 \quad 902 \quad 903 \quad 904 \quad 905 \quad 906 \quad 907 \quad 908 \quad 909 \quad 910 \quad 911 \quad 912 \quad 913 \quad 914 \quad 915 \quad 916 \quad 917 \quad 918 \quad 919 \quad 920 \quad 921 \quad 922 \quad 923 \quad 924 \quad 925 \quad 926 \quad 927 \quad 928 \quad 929 \quad 930 \quad 931 \quad 932 \quad 933 \quad 934 \quad \mathcal{I}\left((z_t, \mathbf{g}_t); e_{t,k}^{\text{pred}} \mid \mathcal{F}_t\right) = \underbrace{\mathcal{I}\left(z_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t\right)}_{\text{mode-identification}} + \underbrace{\mathcal{I}\left(\mathbf{g}_t; e_{t,k}^{\text{pred}} \mid z_t, \mathcal{F}_t\right)}_{\text{shared-factor refinement}}. \quad (24)$$

The first term measures how much observing the residual $e_{t,k}^{\text{pred}}$ 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 $(\mathbf{A}_m^{(g)}, \mathbf{Q}_m^{(g)}, \mathbf{A}_m^{(u)}, \mathbf{Q}_m^{(u)}, \mathbf{R}_{m,k})$.

Computational note: The mode-identification term requires Monte Carlo estimation because the predictive distribution $p(e_{t,k}^{\text{pred}} \mid \mathcal{F}_t)$ 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 , $\bar{w}_t^{(m)} = \mathbb{P}(z_t = m \mid \mathcal{F}_t)$, and the predictive moments $(\mu_{g,t|t-1}^{(m)}, \Sigma_{g,t|t-1}^{(m)})$, $(\mu_{u,k,t|t-1}^{(m)}, \Sigma_{u,k,t|t-1}^{(m)})$, and $\mathbf{R}_{m,k}$. For Monte Carlo, we use $\tilde{\cdot}$ to denote sampled quantities and write $\tilde{z} \sim \text{Cat}((\tilde{w}_t^{(m)})_{m=1}^M)$ 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

$$e_{t,k}^{\text{pred}} \sim p(e_{t,k} \mid \mathcal{F}_t). \quad (25)$$

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

935
936 **Per-mode linear-Gaussian predictive parametrization (IMM outputs).** Fix a regime $z_t = m$. The IMM predictive step
937 yields a Gaussian predictive prior for the shared factor:

$$938 \quad 939 \quad \mathbf{g}_t | (\mathcal{F}_t, z_t = m) \sim \mathcal{N}\left(\mu_{g,t|t-1}^{(m)}, \Sigma_{g,t|t-1}^{(m)}\right). \quad (26)$$

940 Under the factorized predictive belief, querying expert k induces the linear-Gaussian observation channel
941

$$942 \quad 943 \quad e_{t,k}^{\text{pred}} | (\mathbf{g}_t, \mathcal{F}_t, z_t = m) \sim \mathcal{N}(\mathbf{H}_{t,k}\mathbf{g}_t + \mathbf{b}_{t,k}^{(m)}, \mathbf{S}_{t,k}^{(m)}), \quad (27)$$

944 with mode-specific quantities

$$\begin{aligned} 945 \quad 946 \quad \mathbf{H}_{t,k} &:= \Phi(\mathbf{x}_t)^{\top} \mathbf{B}_k \in \mathbb{R}^{d_y \times d_g}, \\ 947 \quad 948 \quad \mathbf{b}_{t,k}^{(m)} &:= \Phi(\mathbf{x}_t)^{\top} \mu_{u,k,t|t-1}^{(m)} \in \mathbb{R}^{d_y}, \\ 949 \quad 950 \quad \mathbf{S}_{t,k}^{(m)} &:= \Phi(\mathbf{x}_t)^{\top} \Sigma_{u,k,t|t-1}^{(m)} \Phi(\mathbf{x}_t) + \mathbf{R}_{m,k} \in \mathbb{S}_{++}^{d_y}. \end{aligned} \quad (28)$$

951 **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
952 query fee. In practice, we use squared loss,

$$953 \quad 954 \quad \psi(u) = \|u\|_2^2, \quad (29)$$

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

$$957 \quad 958 \quad C_{t,k}^{\text{pred}} := \psi(e_{t,k}^{\text{pred}}) + \beta_k, \quad k \in \mathcal{E}_t, \quad (30)$$

959 with conditional mean

$$960 \quad 961 \quad \bar{C}_{t,k}^{\text{pred}} := \mathbb{E}[C_{t,k}^{\text{pred}} | \mathcal{F}_t] = \mathbb{E}[\psi(e_{t,k}^{\text{pred}}) | \mathcal{F}_t] + \beta_k. \quad (31)$$

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

$$963 \quad 964 \quad \Delta_t(k) := \bar{C}_{t,k}^{\text{pred}} - \bar{C}_{t,k_t^{\text{pred}}}^{\text{pred}} \geq 0. \quad (32)$$

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

$$970 \quad 971 \quad \bar{e}_{t,k}^{\text{pred},(m)} := \mathbb{E}[e_{t,k}^{\text{pred}} | \mathcal{F}_t, z_t = m] = \mathbf{H}_{t,k} \mu_{g,t|t-1}^{(m)} + \mathbf{b}_{t,k}^{(m)} \in \mathbb{R}^{d_y}, \quad (33)$$

$$972 \quad 973 \quad \Sigma_{t,k}^{\text{pred},(m)} := \text{Cov}(e_{t,k}^{\text{pred}} | \mathcal{F}_t, z_t = m) = \mathbf{H}_{t,k} \Sigma_{g,t|t-1}^{(m)} \mathbf{H}_{t,k}^{\top} + \mathbf{S}_{t,k}^{(m)} \in \mathbb{S}_{++}^{d_y}. \quad (34)$$

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

$$977 \quad 978 \quad \mathbb{E}[\psi(e_{t,k}^{\text{pred}}) | \mathcal{F}_t] = \sum_{m=1}^M \bar{w}_t^{(m)} \mathbb{E}[\psi(E)]_{E \sim \mathcal{N}(\bar{e}_{t,k}^{\text{pred},(m)}, \Sigma_{t,k}^{\text{pred},(m)})}. \quad (35)$$

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

$$982 \quad 983 \quad \bar{C}_{t,k}^{\text{pred}} = \left(\sum_{m=1}^M \bar{w}_t^{(m)} (\text{tr}(\Sigma_{t,k}^{\text{pred},(m)}) + \|\bar{e}_{t,k}^{\text{pred},(m)}\|_2^2) \right) + \beta_k. \quad (36)$$

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

$$986 \quad 987 \quad \text{IG}_t(k) := \mathcal{I}((z_t, \mathbf{g}_t); e_{t,k}^{\text{pred}} | \mathcal{F}_t). \quad (37)$$

990 By the chain rule,

$$\text{IG}_t(k) = \mathcal{I}(z_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t) + \mathcal{I}(\mathbf{g}_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t, z_t) \quad (38)$$

$$= \underbrace{\mathcal{I}(z_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t)}_{\text{mode-identification}} + \underbrace{\sum_{m=1}^M \bar{w}_t^{(m)} \mathcal{I}(\mathbf{g}_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t, z_t = m)}_{\text{shared-factor refinement}}. \quad (39)$$

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

1000 **Closed form:** $\mathcal{I}(\mathbf{g}_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t, z_t = m)$. Fix $z_t = m$. Let $G := \mathbf{g}_t$ and $Y := e_{t,k}^{\text{pred}}$. Equation (27) implies the affine
 1001 Gaussian channel $Y = \mathbf{H}_{t,k}G + \mathbf{b}_{t,k}^{(m)} + \varepsilon$ with $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{S}_{t,k}^{(m)})$ independent of G . Then

$$\mathcal{I}(\mathbf{g}_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t, z_t = m) = \frac{1}{2} \log \det(\mathbf{I}_{d_y} + \mathbf{H}_{t,k} \Sigma_{g,t|t-1}^{(m)} \mathbf{H}_{t,k}^\top (\mathbf{S}_{t,k}^{(m)})^{-1}). \quad (40)$$

1002 **Monte Carlo:** $\mathcal{I}(z_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t)$ for a Gaussian mixture. Let $p_m(e) := p(e_{t,k}^{\text{pred}} = e \mid \mathcal{F}_t, z_t = m) =$
 1003 $\mathcal{N}(e; \bar{e}_{t,k}^{\text{pred},(m)}, \Sigma_{t,k}^{\text{pred},(m)})$ and $p_{\text{mix}}(e) := \sum_{m=1}^M \bar{w}_t^{(m)} p_m(e)$. Then

$$\mathcal{I}(z_t; e_{t,k}^{\text{pred}} \mid \mathcal{F}_t) = \sum_{m=1}^M \bar{w}_t^{(m)} \text{KL}(p_m \parallel p_{\text{mix}}) \quad (41)$$

$$= \sum_{m=1}^M \bar{w}_t^{(m)} \mathbb{E}_{E \sim p_m} [\log p_m(E) - \log p_{\text{mix}}(E)]. \quad (42)$$

1004 This suggests the estimator (with S samples per mode):

$$\widehat{\mathcal{I}}_t^{(z)}(k) := \sum_{m=1}^M \bar{w}_t^{(m)} \left(\frac{1}{S} \sum_{s=1}^S [\log p_m(E_{m,s}) - \log p_{\text{mix}}(E_{m,s})] \right), \quad E_{m,s} \stackrel{\text{iid}}{\sim} \mathcal{N}(\bar{e}_{t,k}^{\text{pred},(m)}, \Sigma_{t,k}^{\text{pred},(m)}). \quad (43)$$

1005 **Stable evaluation of $\log p_{\text{mix}}(e)$.** Compute Gaussian log-densities via

$$\log \mathcal{N}(e; \mu, \Sigma) = -\frac{1}{2} (d_y \log(2\pi) + \log \det(\Sigma) + (e - \mu)^\top \Sigma^{-1} (e - \mu)). \quad (44)$$

1006 Define $\ell_m(e) := \log \bar{w}_t^{(m)} + \log \mathcal{N}(e; \bar{e}_{t,k}^{\text{pred},(m)}, \Sigma_{t,k}^{\text{pred},(m)})$. Then compute $\log p_{\text{mix}}(e)$ by a stable log-sum-exp:

$$\log p_{\text{mix}}(e) = \log \left(\sum_{m=1}^M e^{\ell_m(e)} \right) = a(e) + \log \left(\sum_{m=1}^M e^{\ell_m(e) - a(e)} \right), \quad a(e) := \max_{m \in \{1, \dots, M\}} \ell_m(e). \quad (45)$$

1007 **Final (z_t, \mathbf{g}_t) -information gain.** Combine (38), (40), and (43):

$$\widehat{\text{IG}}_t(k) := \widehat{\mathcal{I}}_t^{(z)}(k) + \sum_{m=1}^M \bar{w}_t^{(m)} \frac{1}{2} \log \det(\mathbf{I}_{d_y} + \mathbf{H}_{t,k} \Sigma_{g,t|t-1}^{(m)} \mathbf{H}_{t,k}^\top (\mathbf{S}_{t,k}^{(m)})^{-1}). \quad (46)$$

1008 In Algorithm 1, we use $\text{IG}_t(k)$ as a shorthand for this computable estimate $\widehat{\text{IG}}_t(k)$.

F. Proofs

F.1. Proof of Proposition 2

1009 **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 $j \neq I_t$
 1010 and let $(e_{t,j}^{\text{pred}}, e_{t,I_t}^{\text{pred}})$ denote the one-step-ahead predictive residuals under $p(e_{t,.} \mid \mathcal{F}_t, z_t = m)$. Assume that this predictive
 1011

pair is jointly Gaussian conditional on \mathcal{G}_t and that $\text{Cov}(e_{t,I_t}^{\text{pred}} \mid \mathcal{G}_t)$ is non-singular (e.g., $\mathbf{R}_{m,I_t} \succ \mathbf{0}$). Then

$$\begin{aligned}\mathbb{E}[e_{t,j}^{\text{pred}} \mid e_t, \mathcal{G}_t] &= \mathbb{E}[e_{t,j}^{\text{pred}} \mid \mathcal{G}_t] \\ \iff \text{Cov}(e_{t,j}^{\text{pred}}, e_{t,I_t}^{\text{pred}} \mid \mathcal{G}_t) &= \mathbf{0}.\end{aligned}$$

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

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 $(e_{t,j}^{\text{pred}}, e_{t,I_t}^{\text{pred}}) \mid \mathcal{G}_t$ 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 $e_t \mid \mathcal{G}_t \stackrel{d}{=} e_{t,I_t}^{\text{pred}} \mid \mathcal{G}_t$ (since e_{t,I_t}^{pred} is exactly the one-step predictive residual that generates e_{t,I_t}). Let

$$\boldsymbol{\mu}_j := \mathbb{E}[e_{t,j}^{\text{pred}} \mid \mathcal{G}_t], \quad \boldsymbol{\mu}_I := \mathbb{E}[e_{t,I_t}^{\text{pred}} \mid \mathcal{G}_t],$$

and define the predictive covariance and cross-covariance matrices

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

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

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

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

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

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

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 $\alpha_{t,\ell}$ and the one-step predictive law of $e_{t,\ell}^{\text{pred}}$ are identical before and after pruning, for every retained $\ell \notin P_t$.

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:

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

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

$$q_{t-1|t-1}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}) = \int q_{t-1|t-1}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1,\mathcal{K}'}, (\mathbf{u}_{t-1,k})_{k \in P_t}) \prod_{k \in P_t} d\mathbf{u}_{t-1,k}. \quad (47)$$

1100 On the other hand, the post-pruning belief $q_{t-1|t-1}^{\text{pr}(P_t)}$ is defined by exactly the same integral:
 1101
 1102
 1103
$$q_{t-1|t-1}^{\text{pr}(P_t)}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1, \mathcal{K}'}) := \int q_{t-1|t-1}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1, \mathcal{K}'}, (\mathbf{u}_{t-1, k})_{k \in P_t}) \prod_{k \in P_t} d\mathbf{u}_{t-1, k}.$$

 1104
 1105 Comparing with (47) yields
 1106
$$q_{t-1|t-1}^{\text{pr}(P_t)}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1, \mathcal{K}'}) = q_{t-1|t-1}(\mathbf{g}_{t-1}, \mathbf{u}_{t-1, \mathcal{K}}),$$

 1107 which proves that pruning P_t leaves the joint belief over all retained variables unchanged.
 1108
 1109 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
 1110 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
 1111 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
 1112 of $(\mathbf{g}_t, \mathbf{u}_{t, \ell})$ and $e_{t, \ell}^{\text{pred}}$ follows the emission model given these states, the predictive distributions of $\boldsymbol{\alpha}_{t, \ell}$ and $e_{t, \ell}^{\text{pred}}$ are
 1113 unchanged by pruning. ■
 1114
 1115
 1116

F.3. Proof of Proposition 4

1117 **Proposition 4** (Coupling at birth through the shared factor). *Fix time t and condition on $(\mathcal{F}_t, z_t = m)$. Under the Factorized
 1118 SLDS one-step predictive belief (i.e., with $\text{Cov}(\mathbf{g}_t, \mathbf{u}_{t, k} | \cdot) = \mathbf{0}$ and $\text{Cov}(\mathbf{u}_{t, i}, \mathbf{u}_{t, j} | \cdot) = \mathbf{0}$ for $i \neq j$), for any experts
 1119 $j \neq k$,*

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

1120 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
 1121 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 $\mathcal{I}(\boldsymbol{\alpha}_{t, j}; \boldsymbol{\alpha}_{t, k} | \mathcal{F}_t, z_t = m) > 0$.
 1122
 1123

1124 **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
 1125 marginal factorization
 1126

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

1127 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,
 1128

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

1129 where $\Sigma_{g, t|t-1}^{(m)} := \text{Cov}(\mathbf{g}_t | \mathcal{F}_t, z_t = m)$. If $\mathbf{B}_j \Sigma_{g, t|t-1}^{(m)} \mathbf{B}_k^\top \neq \mathbf{0}$ and the joint predictive law of $(\boldsymbol{\alpha}_{t, j}, \boldsymbol{\alpha}_{t, k})$ is Gaussian,
 1130 then the pair is not independent, hence $\mathcal{I}(\boldsymbol{\alpha}_{t, j}; \boldsymbol{\alpha}_{t, k} | \mathcal{F}_t, z_t = m) > 0$. ■
 1131
 1132

G. Experiments Details

1133 We provide additional details on the experiments of Section 5, including experimental setup, hyperparameters, and
 1134 implementation details.
 1135

G.1. Baselines

1136 **Feedback regimes (partial vs. full).** At round t , the router observes $(\mathbf{x}_t, \mathcal{E}_t)$, chooses $I_t \in \mathcal{E}_t$, and then observes $(\hat{\mathbf{y}}_{t, I_t}, \mathbf{y}_t)$,
 1137 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 $e_{t, k} = \hat{\mathbf{y}}_{t, k} - \mathbf{y}_t$
 1138

(Appendix B). *Partial feedback* means only $(\hat{\mathbf{y}}_{t,I_t}, \mathbf{y}_t)$ is observed after acting. *Full feedback* means $\{(\hat{\mathbf{y}}_{t,k}, \mathbf{y}_t)\}_{k \in \mathcal{E}_t}$ is observed after acting, hence all $\{C_{t,k}\}_{k \in \mathcal{E}_t}$ are available for evaluation and parameter updates. Importantly, in our experiments this additional information is revealed *after* selecting I_t , so it does not change the decision-time information \mathcal{F}_t ; it only changes what supervision is available to update a baseline.

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: $\alpha_{t,k} = \mathbf{B}_k \mathbf{g}_t + \mathbf{u}_{t,k}$ and $e_{t,k} \mid (z_t = m, \mathbf{g}_t, \mathbf{u}_{t,k}, \mathbf{x}_t) \sim \mathcal{N}(\Phi(\mathbf{x}_t)^\top \alpha_{t,k}, \mathbf{R}_{m,k})$ ((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 $\alpha_{t,k} = \mathbf{u}_{t,k}$ 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 $\varphi : \mathbb{R}^d \rightarrow \mathbb{R}^p$ (in our experiments, either raw \mathbf{x}_t or an RNN embedding). Assume a linear model for the conditional mean cost of each expert: $\mathbb{E}[C_{t,k} \mid \mathbf{x}_t] \approx \varphi(\mathbf{x}_t)^\top \theta_k$. Maintain ridge statistics per expert k , with ridge parameter $\lambda > 0$. Under *partial feedback*:

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

where $C_s = C_{s,I_s}$ is the realized (queried) cost at round s . Under *full feedback*:

$$\mathbf{V}_{t,k} := \lambda \mathbf{I}_p + \sum_{s < t: k \in \mathcal{E}_s} \varphi(\mathbf{x}_s) \varphi(\mathbf{x}_s)^\top, \quad \mathbf{b}_{t,k} := \sum_{s < t: k \in \mathcal{E}_s} \varphi(\mathbf{x}_s) C_{s,k}, \quad \hat{\theta}_{t,k} := \mathbf{V}_{t,k}^{-1} \mathbf{b}_{t,k}.$$

At time t , set $\hat{C}_{t,k} := \varphi(\mathbf{x}_t)^\top \hat{\theta}_{t,k}$ 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

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

Partial feedback LinUCB updates only the chosen arm I_t (so only $C_{t,I_t} = C_t$ is observed). *Full feedback LinUCB* updates every $k \in \mathcal{E}_t$ at each round; we set $\alpha_t = 0$ (no exploration), hence $I_t \in \arg \min_{k \in \mathcal{E}_t} \hat{C}_{t,k}$.

NeuralUCB. Let $f_\omega(\mathbf{x}, k)$ 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) $\mathbf{h}_{t,k} := \nabla_\omega f_\omega(\mathbf{x}_t, k) \in \mathbb{R}^{p_\omega}$. Maintain a (regularized) Gram matrix. Under *partial feedback*:

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

Under *full feedback*:

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

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

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

The network is trained online by stochastic gradient steps on squared error. *Partial feedback NeuralUCB* uses the loss $(f_\omega(\mathbf{x}_t, I_t) - C_t)^2$ (only C_t observed). *Full feedback NeuralUCB* uses $\sum_{k \in \mathcal{E}_t} (f_\omega(\mathbf{x}_t, k) - C_{t,k})^2$; we set $\alpha_t = 0$ (no exploration), hence $I_t \in \arg \min_{k \in \mathcal{E}_t} \hat{C}_{t,k}$.

1210 Oracle baseline. The (per-round) oracle chooses the best available expert in hindsight:
 1211
 1212
 1213

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

1214 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
 1215 achievable cumulative cost.
 1216

1217 Learning-to-Defer baselines (full feedback). The Learning-to-Defer baselines assume access to full-feedback costs
 1218 $\{C_{t,k}\}_{k \in \mathcal{E}_t}$ at each round and are therefore reported only as full-feedback methods in our tables. In our implementation,
 1219 L2D trains a parametric router score function $r_\phi : \mathcal{X} \rightarrow \mathbb{R}^K$ (Mao et al., 2024c; Montreuil et al., 2025b). At round t , it
 1220 induces a distribution over the available experts via a masked softmax:
 1221

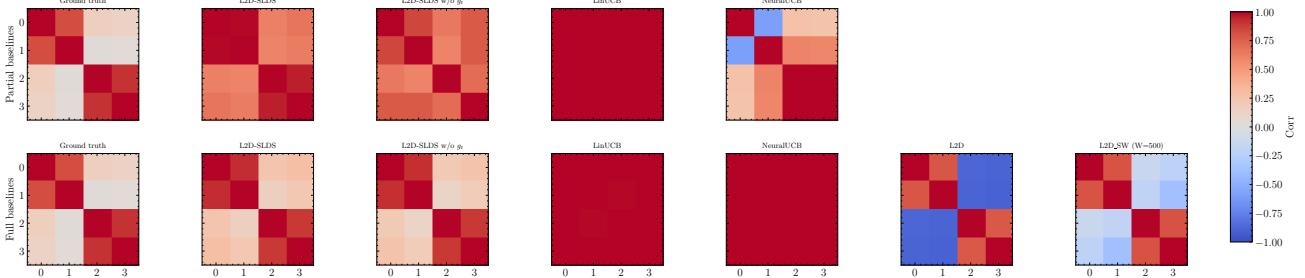
$$\pi_\phi(k | \mathbf{x}_t) := \frac{\exp(r_\phi(\mathbf{x}_t)_k)}{\sum_{j \in \mathcal{E}_t} \exp(r_\phi(\mathbf{x}_t)_j)}, \quad k \in \mathcal{E}_t. \quad (48)$$

1222 With full-feedback supervision $\{C_{t,k}\}_{k \in \mathcal{E}_t}$, we train the router using a cost-sensitive log-softmax loss:
 1223
 1224

$$\mathcal{L}_t^{\text{L2D}}(\phi) := - \sum_{k \in \mathcal{E}_t} w_{t,k} \log \pi_\phi(k | \mathbf{x}_t), \quad w_{t,k} := \sum_{\substack{i \in \mathcal{E}_t \\ i \neq k}} C_{t,i}. \quad (49)$$

1225 Equivalently, $w_{t,k} = \sum_{i \in \mathcal{E}_t} C_{t,i} - C_{t,k}$, so experts with smaller cost $C_{t,k}$ receive larger weight. The sliding-window variant
 1226 L2D-SW minimizes $\sum_{t=T-W+1}^T \mathcal{L}_t^{\text{L2D}}(\phi)$ using only the most recent W rounds.
 1227

1228 G.2. Synthetic: Regime-Dependent Correlation and Information Transfer



1229 Figure 3. Regime-0 expert dependence in the synthetic transfer experiment. Each heatmap shows the pairwise Pearson correlation (color:
 1230 $[-1, 1]$) between experts' per-round losses (experts indexed 0–3). Top row: partial feedback (only queried losses observed); bottom
 1231 row: full feedback. Columns (left-to-right) show the ground-truth correlation implied by the simulator and the correlations estimated by
 1232 each method. L2D-SLDS best recovers the block-structured correlations (experts {0, 1} vs. {2, 3}), highlighting the benefit of modeling
 1233 shared latent factors for cross-expert information transfer under censoring.
 1234

1235 Design goal. We construct a controlled routing instance in which (i) experts are *correlated* in a regime-dependent way,
 1236 so that observing one expert should update beliefs about others (information transfer; Proposition 2); and (ii) one expert
 1237 temporarily disappears and re-enters, so that the maintained registry \mathcal{K}_t matters (see Appendix).
 1238

1239 Environment (regimes, target, context). We use $M = 2$ regimes and deterministic switching in blocks of length $L = 150$
 1240 over horizon $T = 3000$ such as $z_t := 1 + \lfloor \frac{t-1}{L} \rfloor \bmod 2$. The target follows a regime-dependent AR(1), and the context is
 1241 the one-step lag:
 1242

$$y_t = 0.8 y_{t-1} + d_{z_t} + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \sigma_y^2). \quad (50)$$

1243 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)
 1244 and the single queried prediction \hat{y}_{t,I_t} (after acting).
 1245

1265 **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
 1266 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
 1267 slope and expert-specific intercept plus noise:

$$\hat{y}_{t,k} := 0.8 y_{t-1} + b_k + \varepsilon_{t,k}. \quad (51)$$

1269 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
 1270 well-calibrated in regime $z_t = 2$.

1273 To induce *regime-dependent correlation* under bandit feedback, we generate the expert noises as

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

1275 with independent components $s_{t,1}, s_{t,2}, (\tilde{\varepsilon}_{t,k})_k$ and regime-dependent variances $s_{t,1} \sim \mathcal{N}(0, \sigma_{z_t,1}^2), s_{t,2} \sim$
 1276 $\mathcal{N}(0, \sigma_{z_t,2}^2), \tilde{\varepsilon}_{t,k} \sim \mathcal{N}(0, \sigma_{\text{id}}^2)$, where $(\sigma_{1,1}^2, \sigma_{1,2}^2) = (\sigma_{\text{hi}}^2, \sigma_{\text{lo}}^2)$ and $(\sigma_{2,1}^2, \sigma_{2,2}^2) = (\sigma_{\text{lo}}^2, \sigma_{\text{hi}}^2)$ with $\sigma_{\text{hi}}^2 \gg \sigma_{\text{lo}}^2$. This makes
 1277 experts $\{0, 1\}$ strongly correlated in regime 1 and experts $\{2, 3\}$ strongly correlated in regime 2. We report the MSE of each
 1278 expert in Table 1.

1281 **Compared methods.** We compare our **L2D-SLDS** router under bandit feedback to the following baselines. (i) *Ablation*:
 1282 L2D-SLDS without the shared global factor (set $d_g = 0$). (ii) *Contextual bandits*: LinUCB (Li et al., 2010) and NeuralUCB
 1283 (Zhou et al., 2020). (iii) *Full-feedback*: a full-feedback variant of L2D-SLDS and online Learning-to-Defer baselines (Mao
 1284 et al., 2024c; Narasimhan et al., 2022) that assume access to all experts' predictions each round (hence are not feasible
 1285 under censoring): standard L2D (Narasimhan et al., 2022; Mao et al., 2024c), and a sliding-window L2D (L2D-SW) with
 1286 $W = 500$ taking the most recent data to handle non-stationarity. We use an RNN encoder (Rumelhart et al., 1985) as a
 1287 drop-in context representation for methods that require learned features.

1289 *Table 3.* 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 | 13.58 ± 0.07 | 10.17 ± 0.01 |
| L2D-SLDS w/o 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 |

1304 **Correlation recovery.** Figure 1 compares the regime-0 loss correlation structure. The ground truth exhibits a clear block
 1305 structure: experts $\{0, 1\}$ form one correlated group while experts $\{2, 3\}$ form another. Under partial feedback, L2D-SLDS is
 1306 the only method that reliably recovers this clustering from partial observations, whereas removing the shared factor g_t blurs
 1307 the separation and inflates cross-group correlations, consistent with losing cross-expert information transfer. In contrast,
 1308 LinUCB/NeuralUCB yield near-degenerate correlation estimates (e.g., overly uniform or unstable patterns), reflecting that
 1309 purely discriminative bandit updates do not maintain a coherent joint belief over experts' latent error processes. Under
 1310 full feedback, the gap between L2D-SLDS and its ablation largely closes, as observing all experts makes explicit transfer
 1311 less critical; however, the remaining baselines can still exhibit spurious structure, highlighting that modeling regime-wise
 1312 coupling is beneficial beyond simply having access to more feedback.

1315 **Results.** Table 1 shows that **L2D-SLDS** achieves the lowest routing cost under partial feedback (13.58 ± 0.07), improving
 1316 over LinUCB/NeuralUCB by a wide margin and also outperforming the best fixed expert. Crucially, it also beats the ablation
 1317 that removes the shared factor g_t (14.68 ± 0.01), a $\approx 7.5\%$ reduction, which directly supports our central claim: under
 1318 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.

1328

1329

1330

1331

1332

1333

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

1350

1351

1352

1353

1354

1355

1356

1357

1358

1359

1360

1361

1362

1363

1364

1365

1366

1367

1368

1369

1370

1371

1372

1373

1374