0.1 Tri-Cycle Correlation Experiment

Goal. We design a synthetic experiment to isolate the two properties emphasized by the factorized SLDS: (i) correlation-mediated information transfer across experts, and (ii) predictive scheduling under regime-dependent availability.

Regime dynamics. We use $M = 3$ regimes and a deterministic cycle $z_t \in \{1, 2, 3\}$ with pattern $1 \to 2 \to 3 \to 2 \to 3 \to 1$ repeated in blocks of fixed length. The observed target follows an AR(1) with regime-dependent drift,

$$y_t = 0.8\, y_{t-1} + \mu_{z_t} + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2). \tag{1}$$

The context is the lagged target, $x_t = y_{t-1}$.

Experts and correlation structure. We define $K = 5$ experts with linear predictors $\widehat{y}_{t,k} = a_{z_t,k}\, x_t + b_{z_t,k} + \eta_{t,k}$. To encode regime-specific correlations, we inject a shared perturbation into specific expert pairs within each regime:

$$\begin{aligned}
&\text{Regime 1:} \quad (k = 1, 2) \text{ share a common noise component,} && (2)\\
&\text{Regime 2:} \quad (k = 3, 4) \text{ share a common noise component,} && (3)\\
&\text{Regime 3:} \quad (k = 5, 1) \text{ share a common noise component.} && (4)
\end{aligned}$$

The remaining experts receive independent noise. This makes the residuals $\{e_{t,k}\}$ correlated within the specified pair for the active regime, while correlations change as regimes shift.

Regime-dependent availability. We further impose known availability constraints: expert 5 is feasible only in regimes 2 and 3, and expert 4 is feasible only in regimes 1 and 2. Formally, for each lookahead h, the feasible set $\mathcal{E}^{\text{feas}}_{t+h|t}$ is computed by intersecting the working registry with these regime-dependent availability rules.

Why this setting is diagnostic. This construction stresses the core claims of the factorized model: (1) correlations change with regime, so information transfer should be contextual rather than static; (2) partial feedback is informative because correlated experts should be updated even when not queried; and (3) predictive scheduling must adapt to both regime uncertainty and feasibility constraints.

What to observe. We recommend reporting:

1. **Correlation recovery:** estimated cross-expert correlations (or shared-factor loadings) should track the true pairings across regimes.
2. **Partial-feedback advantage:** in partial feedback, the factorized model should still update unqueried correlated experts, yielding lower loss than baselines that treat experts as independent.
3. **Regime transitions:** posterior regime weights should react quickly to the cycle and the correlated pair should switch accordingly.
4. **Scheduling structure:** the active sets $\widehat{\mathcal{S}}_{t,h}(\delta)$ should (i) respect feasibility and (ii) expand near regime transitions when scenario uncertainty spreads mass across multiple correlated candidates.

Implementation details. The experiment is reproducible via the config file `config/config_tri_cycle_corr.yaml`. We set `environment.setting = tri_cycle_corr` with `num_experts = 5`, `num_regimes = 3`, and total horizon T. The regime cycle is controlled by `environment.tri_cycle.regime_pattern` and `environment.tri_cycle.regime_block_len`. Regime drifts μ_{z_t} are set by `environment.tri_cycle.drift_levels`, while observation noise uses `environment.noise_scale`. Expert correlations are induced by shared noise with scales `shared_noise_scale` and `indiv_noise_scale`. Regime-dependent expert accuracy is encoded through `biases_by_regime` (and optionally `slopes_by_regime`). Note that the implementation uses 0-based expert indices, so paper experts 1–5 correspond to code indices 0–4. Availability constraints are enforced at the environment level: expert 4 (index 3) is available only in regimes 1 and 2, and expert 5 (index 4) is available only in regimes 2 and 3.

If you want, I can also add a short paragraph on the Monte Carlo scheduling setup for this experiment (i.e., the specific `horizon`$_{p}lanningknobsandhowtheyreflectSection$ **??**