Financial ML Model Monitoring and Drift Audit: Theoretical Foundation and Proven Implementation

July 2, 2025

Motivation

In regulated financial environments, machine learning (ML) models are widely used for credit scoring, fraud detection, risk assessment, and trading. These models must comply with strict regulatory standards (e.g., SR 11-7, GDPR) and ensure fairness, transparency, and stability over time.

Continuous monitoring of model performance and input data drift is critical to maintain compliance, mitigate financial risk, and uphold trust.

Need for Drift Detection

Data Drift: Change in the statistical properties of input data over time, potentially degrading performance.

Concept Drift: Change in the relationship between features and target variables (e.g., macroeconomic shifts).

Failure to detect drift leads to financial losses, compliance violations, and reputational harm.

Formal Hypotheses

For each feature X_i:

$$H_0 : F_0 = F_1 \quad \text{(distribution unchanged)}$$

$$H_A : F_0 \neq F_1 \quad \text{(distribution drifted)}$$

where F_0 and F_1 denote empirical cumulative distributions under the reference and new batches, respectively.

Drift Metrics

- **Kolmogorov-Smirnov (KS) Statistic**: Measures maximum difference between empirical CDFs.
- **Population Stability Index (PSI)**: Measures bin-wise distribution shift; converges to KL-divergence with fine binning.
- **Wasserstein Distance**: Measures transport cost to align distributions; rooted in optimal transport theory.

Statistical Corrections

To control the family-wise error rate across multiple feature tests, we apply Bonferroni correction:

$$\alpha' = \frac{\alpha}{m}$$

where m = number of features.

Empirical KS Threshold

Empirical thresholds are estimated via permutation bootstrapping. We repeatedly resample from the reference distribution and compute:

$$KS^* = \sup_x |F_0^*(x) - F_1^*(x)|$$

We define the threshold as the $1 - \alpha$ quantile of $\{KS_b^*\}_{b=1}^B$, providing a data-driven and rigorous cutoff.

Bootstrap Confidence Intervals

For metrics such as accuracy and AUC, we use bootstrap resampling to compute:

$$CI_{1-\alpha} = [\hat{\theta}_{\alpha/2}, \hat{\theta}_{1-\alpha/2}]$$

ensuring rigorous uncertainty quantification.

Algorithm

Algorithm: Drift Monitoring with Statistical Guarantees

1. Train model M on initial data D_0.
2. For each incoming batch B_t:

(a) Compute predictions, accuracy, AUC, log loss.

(b) For each feature X_i:

i. Compute KS, PSI, and Wasserstein metrics.

ii. Calculate p-values from KS tests.

iii. Apply Bonferroni correction.

(c) Compare KS to empirical threshold.

(d) If any condition is met (KS ¿ threshold, p-value ¡ α', PSI ¿ 0.1, Wasserstein ¿ 0.1), flag as DRIFT.

(e) Log metrics and predictions to audit database.

(f) Compute SHAP values to attribute feature drift contributions.

3. Update plots and audit reports.

Proof of Validity

- **KS Test**: Non-parametric, distribution-free under H_0. Empirical thresholds control type I error even in finite samples.
- **Bonferroni Correction**: Maintains family-wise error rate at α, ensuring valid multi-feature testing.
- **PSI and Wasserstein**: Consistent for detecting distributional changes; Wasserstein grounded in optimal transport, PSI approximates KL-divergence asymptotically.
- **Bootstrap CIs**: Asymptotically valid, provide consistent uncertainty intervals under i.i.d. assumptions.

Code Implementation

The final Python code exactly follows this theoretical design:

- Empirical KS threshold via permutation bootstrapping.
- Multi-feature corrected hypothesis tests.
- Audit logging to SQLite for reproducibility.
- SHAP-based interpretability for drift contributions.
- Generation of plots and LaTeX-ready audit reports.

Conclusion

Our system unifies rigorous hypothesis testing, empirical drift thresholds, bootstrap confidence intervals, and interpretability. It ensures compliance with financial regulations and supports robust academic or regulatory review.

Repository-ready: The code is fully reproducible and prepared for GitHub deployment with audit logs, reports, and plots.