Automated Feature Drift Detection and Analysis Framework for Robust Machine Learning Deployment

---

Abstract

Deployed machine learning models are vulnerable to feature drift-changes in the statistical properties of input features over time-which can lead to degraded performance and unreliable predictions. This project proposes an automated framework that detects, quantifies, and visualizes feature drift in real time. The framework leverages statistical tests (e.g., Kolmogorov–Smirnov) and divergence measures (e.g., Jensen–Shannon divergence) to monitor drift across multiple datasets. In addition, the framework evaluates the impact of drift on model performance using metrics such as accuracy, precision, recall, and F1-score. We compare our approach against a baseline naive drift detection method. Experimental evaluations are conducted on at least four diverse datasets, demonstrating that our tool not only reliably detects drift but also correlates drift magnitude with performance degradation. This framework aims to provide data scientists with an early-warning system, facilitating timely model recalibration and maintenance, thereby ensuring continued effectiveness in production environments.

Problem Description

What Element Are We Improving?

We target the **monitoring phase** of the data science pipeline—specifically, the detection and analysis of **feature drift** in deployed machine learning models.

Why Improve This Element?

Feature drift occurs when the underlying distribution of input features changes from the training data. For instance, in a credit risk model, if the applicant income distribution shifts upward over time, predictions may become inaccurate. Manual detection is both time-consuming and error-prone. An automated, robust method is essential to:

- Prevent unexpected drops in model performance.
- Enable timely model updates.
- Reduce the operational overhead in model monitoring.

Solution Overview

Proposed Framework

Our solution is a tool that continuously monitors deployed models by:

1. **Detecting Drift**: Automatically computing statistical tests (KS test) and divergence metrics (Jensen–Shannon divergence) to detect deviations between the training and real-time data distributions.
2. **Visualizing Drift**: Providing intuitive visualizations (e.g., overlayed histograms, time series plots) that highlight the extent and nature of drift for each feature.
3. **Evaluating Impact**: Measuring how drift correlates with changes in model performance using key metrics (accuracy, precision, recall, F1-score).
4.

Measurement and Baseline Comparison

- **Metrics**: The framework quantifies drift and its impact on model outputs.
- **Baselines**: Our approach is compared to a naive method (e.g., simple threshold-based alerting on mean/variance changes) to demonstrate its improved sensitivity and robustness.

Integration with Class Tools

The proposed solution leverages statistical methods and visualization techniques discussed in class, extending them to an automated monitoring system.

Experimental Evaluation

Datasets

The framework will be evaluated on at least four diverse datasets, chosen to represent various domains (e.g., financial credit risk, healthcare metrics, e-commerce, and sensor data). Each dataset will either naturally exhibit drift or be synthetically manipulated to simulate drift.

Experiment Plan

1. **Drift Simulation/Identification**: For each dataset, we will monitor selected features over time. Synthetic drift (e.g., shifting the mean or variance) will be introduced in controlled experiments.
2. **Performance Analysis**: We will track changes in model performance as drift occurs, using evaluation metrics (accuracy, precision, recall, F1-score).
3. **Comparison with Baselines**: The drift detection results and their impact on model performance will be compared with a baseline approach.
4. **Visualization & Reporting**: Graphs and tables will summarize the drift detection metrics and performance correlations, demonstrating the advantages of our framework.

Conclusion

This project introduces an automated framework for detecting and analyzing feature drift in deployed machine learning models. By combining robust statistical methods, clear visualizations, and comprehensive performance metrics, our solution provides an effective early-warning system for model degradation. Through experiments on multiple datasets and comparison with baseline methods, we demonstrate that timely drift detection leads to actionable insights for model maintenance and retraining. This work not only extends methodologies covered in class but also highlights the critical need for continuous monitoring in real-world data science applications.