Building a Production-Ready ML Pipeline for Loan Default Prediction

Created by: Justin Ng

Date: Nov 25

High-Level Architecture: The 3 DAG System

Training DAG

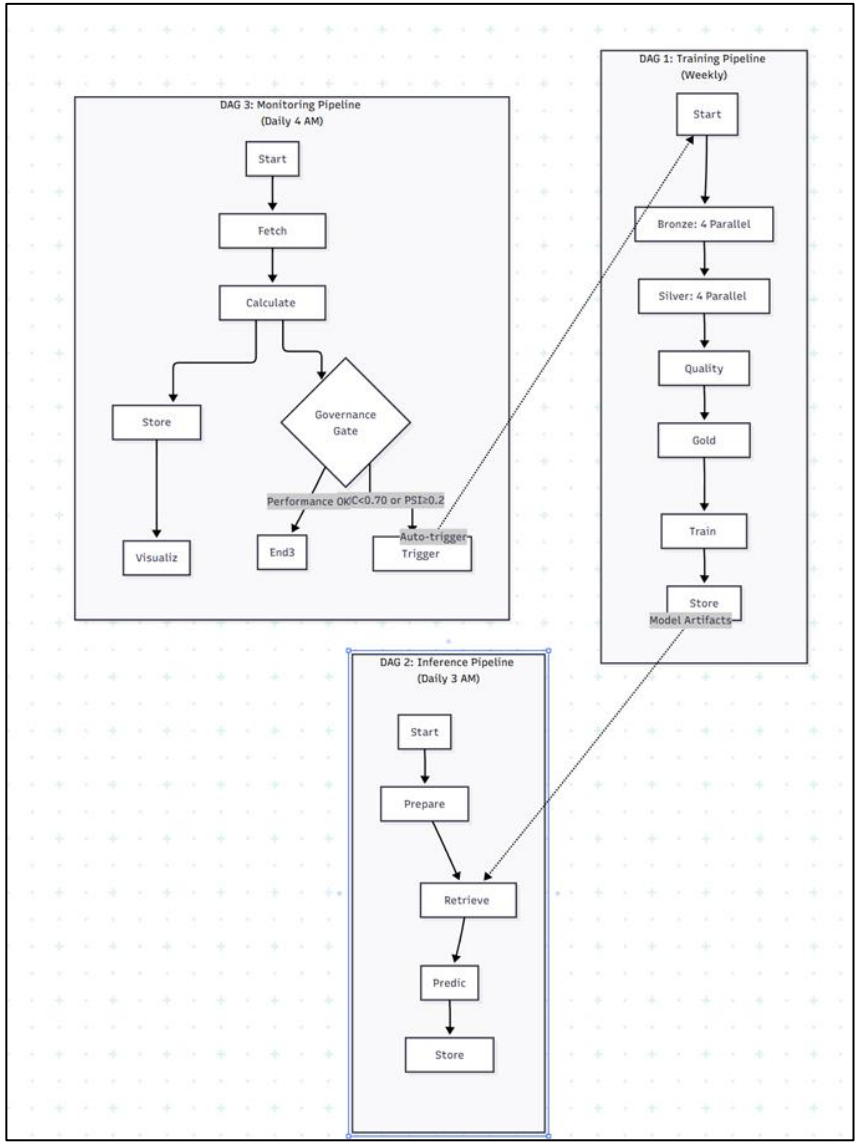
Builds and validates the best model from raw data. Runs weekly or on-demand.

Inference DAG

Loads the production model to generate batch predictions. Runs daily.

Monitoring DAG

Tracks performance and data drift, triggering retraining via the Governance Gate.



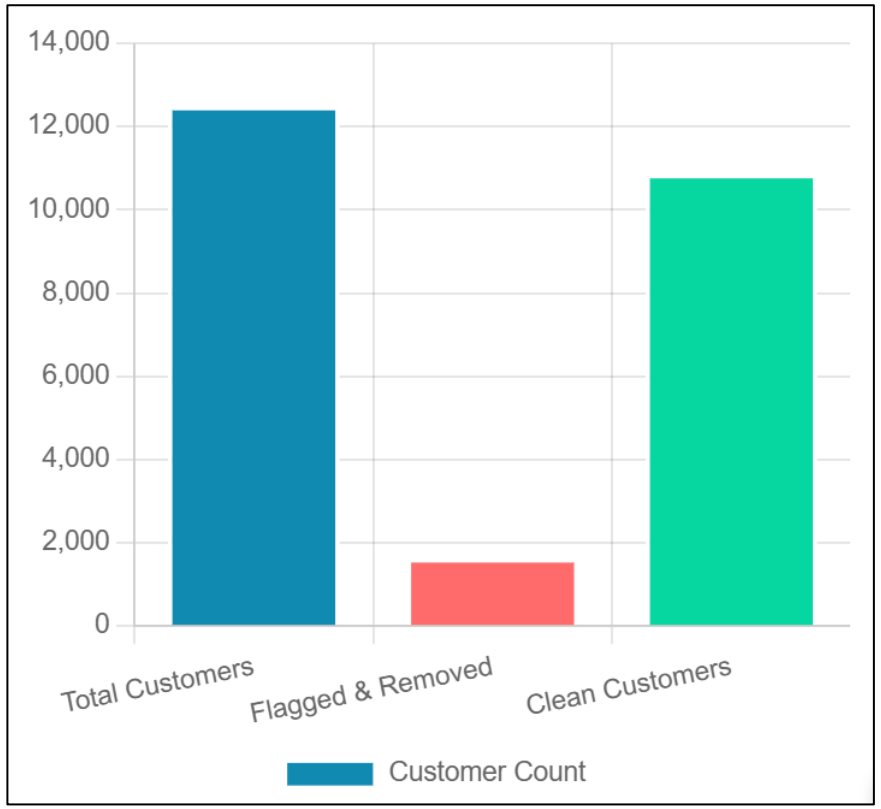
DAG 1: The Training Pipeline

Part 1: Bronze & Silver Data Cleaning

Key Silver Cleaning Logic (PySpark)

- Attributes Table:
  - Validates SSN (e.g., rlike(r'^\d{3}-\d{2}-\d{4}$')) and cleans Age (e.g., regexp_replace(r'[^0-9]', ''))
- Financials Table:
  - Cleans 9 float and 6 integer columns using mass regex and removes placeholders like '_', 'NM', and '!@9#%8'
- Clickstream Table:
  - Cleans all 20 fe features, correctly handling negative numbers with regexp_replace(r'[^0-9-]', '')

Data Quality Check Gate



The Quality Check Gate identifies and removes customers with invalid data (e.g., bad SSNs, negative income) from all 4 tables before the Gold layer.

Part 2: Gold Feature Engineering

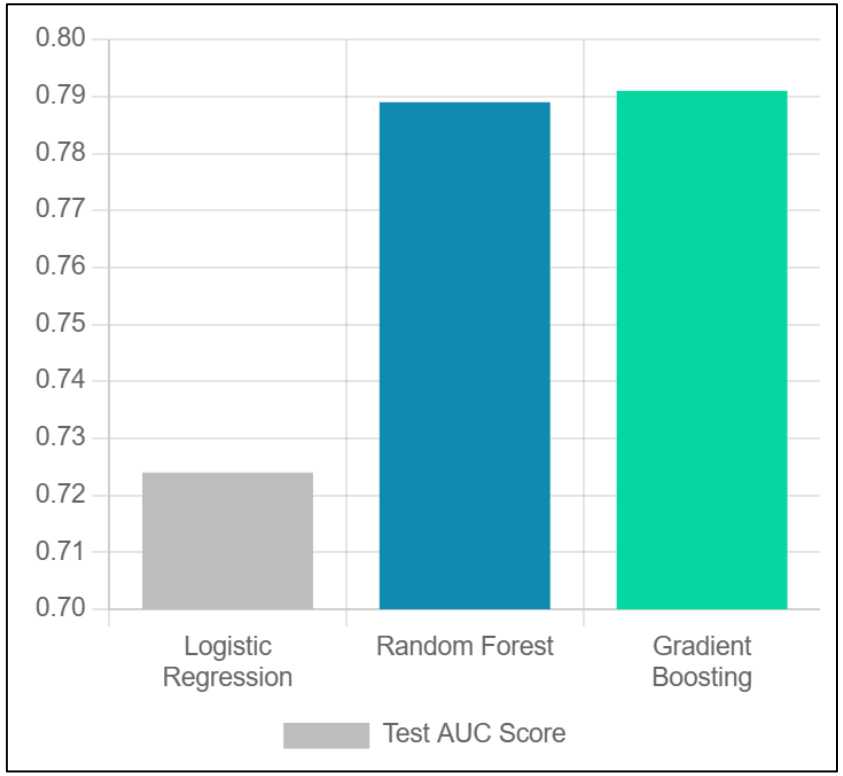
Key Temporal Logic (PySpark)

- Point-in-Time Snapshots: Prevents data leakage by joining prediction_date against attributes and financials tables to get the latest snapshot of customer data on or before that date.
- Windowed Aggregates: Clickstream features are aggregated only from data before loan_start_date. Loan history is aggregated only from installments before Month 3.

Key Engineered Features

- Credit_History_Months: Parses "7 Years 3 Months"
- DTI (Debt-to-Income): Total_EMI_per_month / Monthly_Inhand_Salary
- Savings_Ratio: Amount_invested_monthly / Monthly_Inhand_Salary
- hist_Loan_Payment_Ratio: hist_total_paid / hist_total_due

Model Competition: Test AUC



Three models compete. The best model (highest Test AUC) is automatically selected and versioned with its metadata and encoders for production use.

DAG 2: The Inference Pipeline

Part 1: Daily Inference Pipeline

Retrieve Model

Load best_model.pkl and label_encoders.pkl from Model Store.

Prepare Data

Apply the same Gold Layer feature engineering to new data.

Generate Predictions

Score the new features and generate prediction_proba.

Store Results

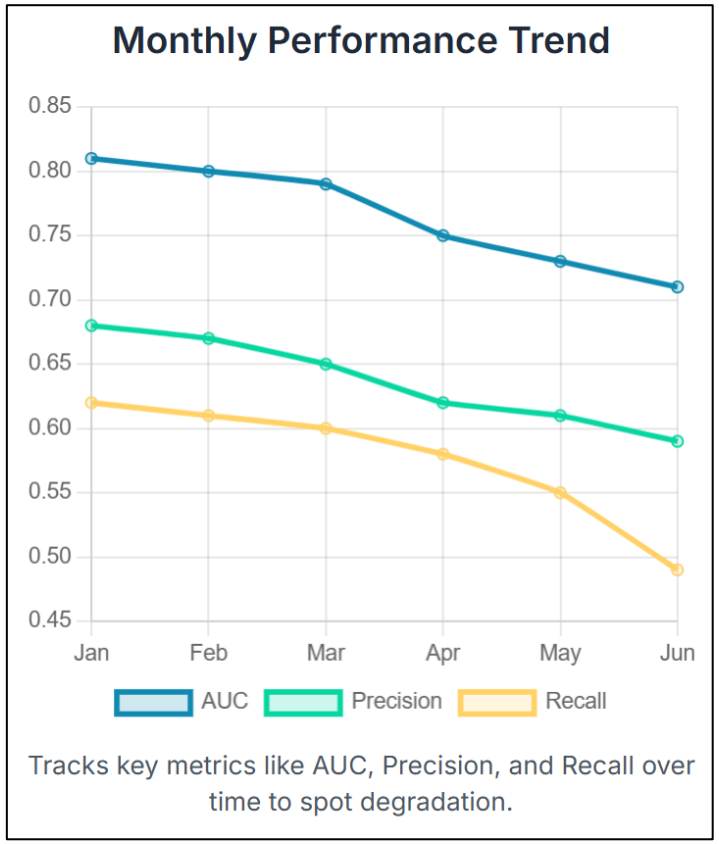
Save predictions to a Gold datamart table for the Monitoring DAG.

The daily inference pipeline is simple and fast. It loads the versioned production model and applies it to new data to generate daily predictions.

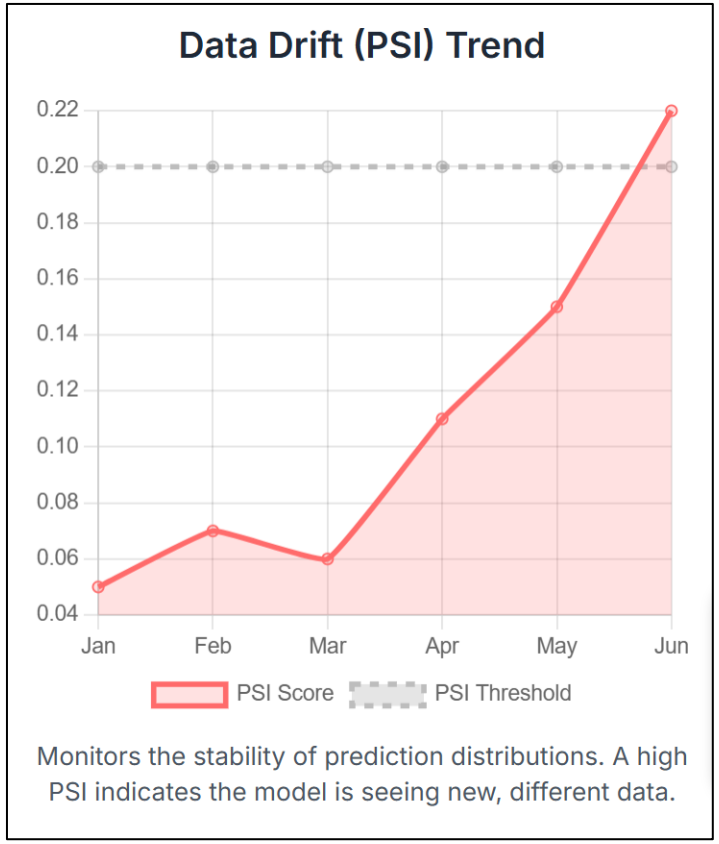
DAG 3: Monitoring & Governance

Part 1: Performance & Drift Tracking

The Monitoring DAG runs daily to calculate performance against actuals and detect data drift by tracking the Population Stability Index (PSI).



Tracks key metrics like AUC, Precision, and Recall over time to spot degradation.



Monitors the stability of prediction distributions. A high PSI indicates the model is seeing new, different data.

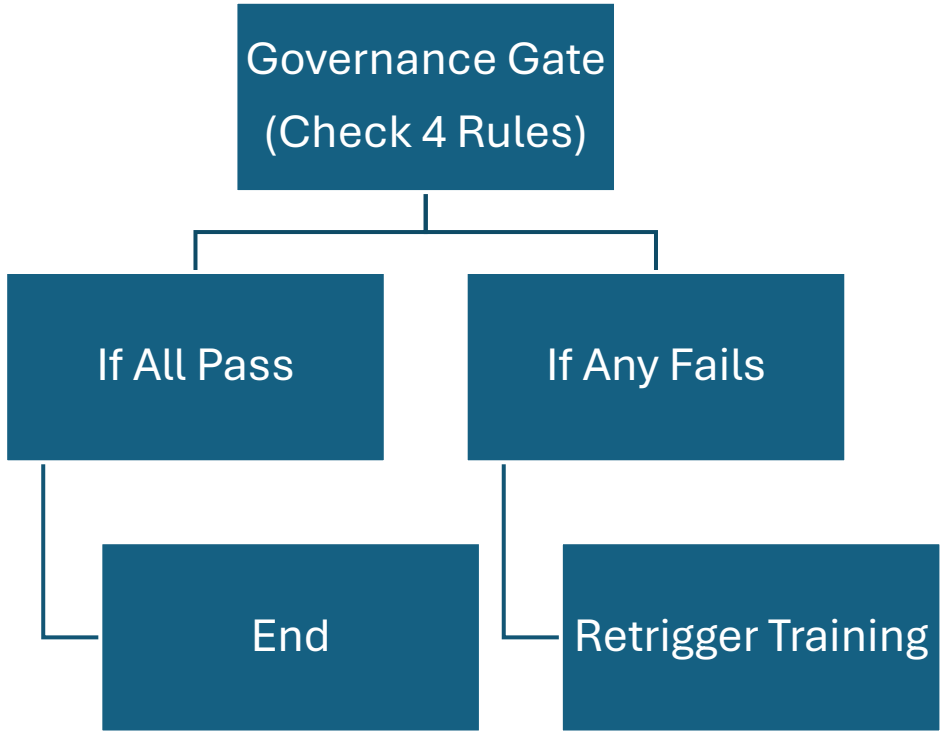
Part 2: The Automated Governance Gate

The Airflow BranchPythonOperator checks a robust set of 4 rules. If any rule is breached, it automatically triggers a new run of the entire Training Pipeline, creating a self-healing system.

Governance Triggers

- AUC < 0.70
  - Model performance is poor.
- Precision < 0.60
  - Too many false positives.
- Recall < 0.50
  - Missing too many real defaults.
- PSI >= 0.2
  - Significant data drift detected.

Automated Action



Key Insights & Lessons Learnt

Business Impact

- Faster Response
  - Automated governance provides a rapid response to changing data patterns.
- Improved Reliability
  - Stakeholder trust improved due to automated governance and drift detection.
- Enhanced Compliance
  - Creates a transparent, auditable trail for regulatory requirements.

Technical Lessons

- Modularity is Key
  - Modular, testable code (like the separate Python scripts) is critical for maintainability.
- Clear Thresholds
  - Defining explicit metrics (AUC < 0.7, PSI >= 0.2) simplifies governance.
- Data Lineage
  - Robust storage (Bronze/Silver/Gold) is essential for debugging and compliance.

Process Lessons

- Collaboration:
  - Close partnership between Data Science and Engineering accelerates delivery.
- Automation Frees Capacity
  - Automation reduces manual errors and frees up the team to focus on new problems.
- Monitoring is Non-Negotiable
  - Continuous monitoring is the key to long-term model success.