Churn Prediction & Data Drift

Offline MLOps Project — Methodology & Results Report

Dataset: data_tp_churn.csv | 5,000 observations | 24 months | February 2026

TCHEMAKO KEN-ANDREW MEKUIGNE STELLA ALY SALL

1. Executive Summary

This report documents the methodology, analytical findings, and design decisions for a customer churn prediction system built under realistic production constraints. The central challenge is data drift: customer behaviour changes significantly across a 24-month observation window, making any static once-trained model progressively unreliable. Using 5,000 customer records, we implement a leakage-free pipeline, compare two retraining strategies, automatically select the optimal retraining window, and calibrate the decision threshold to real business costs. The recommended configuration reduces total business cost by 64.9% versus the default threshold setting.

2. Data Overview and Drift Analysis

2.1 Dataset and Measured Drift

Figure 1 captures two simultaneous forms of drift. Panel (a) shows target drift: the monthly churn rate rose steadily from 19.3% in Month 1 to 56.0% in Month 24, a near-tripling at a linear rate of approximately +1.5 percentage points per month. A model trained on early data inherently underestimates churn risk in later periods, since it learned patterns from a regime where churn was below 25% but must then predict in a regime exceeding 50%. This finding alone disqualifies any static, never-retrained model from production use.

Panel (b) shows missingness drift: the proportion of records with missing billing amounts (Monthly_Amount) grew from approximately 8% in Month 1 to over 22% by Month 24. A Missing Not At Random (MNAR) analysis confirmed that this is not random noise; customers with missing billing records churned at 55.7% versus 33.4% for those with complete records, a 22-percentage-point gap. This finding motivated the inclusion of explicit MNAR indicator features in the preprocessing pipeline.

Figure 1 — Data Drift Analysis

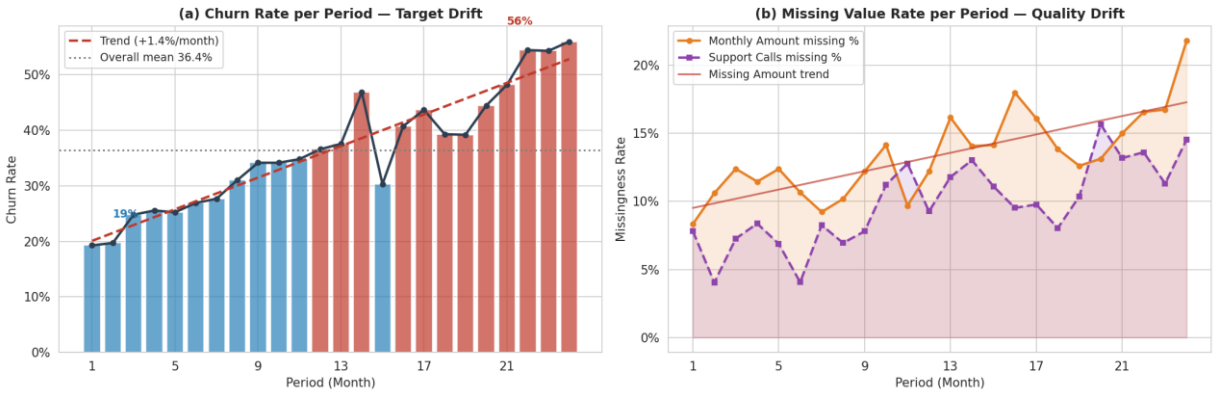

Figure 1 — Drift Analysis: (a) churn rate rising from 19.3% to 56.0% over 24 months (target drift); (b) missing billing record rate growing from 8% to 22% (data quality drift).

3. Methodology

3.1 Preprocessing Pipeline

All data preparation is implemented as a scikit-learn Pipeline - a transformation chain fitted exclusively on training data and applied to test data, ensuring no future information leaks into the training process. For numeric columns (Tenure, Monthly_Amount, Support_Calls, and the two MNAR indicators), the pipeline applies median imputation followed by StandardScaler normalisation. Median imputation was chosen over mean imputation because Monthly_Amount spans EUR 20-131, and extreme values would distort a mean-based fill. For the categorical Contract column, constant imputation and One-Hot Encoding produce three binary features.

Two binary MNAR indicator columns (Monthly_Amount_missing, Support_Calls_missing) are engineered before imputation, preserving the predictive signal that missingness itself carries. Without these indicators, the downstream model would treat imputed customers identically to those with genuine billing records, losing the 22-pp churn-rate gap observed in Figure 1. The final feature vector contains 8 inputs: 5 numeric features plus 3 one-hot contract dummies.

3.2 Temporal Evaluation Strategy

Walk-forward (time-series) cross-validation is used throughout. At each evaluation step, the model is trained on all data available up to period i and tested exclusively on period i+1, correctly simulating production deployment. Two retraining policies are compared: the Fixed (expanding) policy trains on all history from Month 1 to the current period, maximizing data volume but risking dilution by pre-drift patterns; the Rolling window policy trains on only the most recent W months, ensuring freshness at the cost of smaller training sets.

4. Model Performance

4.1 Logistic Regression Baseline

Logistic Regression (regularization C = 1.0) was selected as the baseline for its interpretability and well-calibrated probabilities. Trained on Months 1-16 and evaluated on Months 17-24, it achieved AUC = 0.8486. Figure 2b shows AUC per test period remaining stable between 0.83 and 0.86, with a slight downward trend in the furthest months reflecting concept drift, the model degrades as the gap between training conditions and deployment conditions widens.

4.2 Random Forest and Feature Importances

A Random Forest (150 trees, max depth 6, minimum 10 samples per leaf) improved AUC to 0.8590. The ROC curves in Figure 2a show the Random Forest consistently higher, particularly at low false-positive rates , the operating region relevant to targeted retention campaigns. Feature importances (Figure 2c) confirmed Monthly_Amount and Tenure as the strongest predictors, followed by Support_Calls. The MNAR missingness indicators achieved non-trivial importance scores, validating the engineering decision to include them explicitly.

Figure 2 — Model Performance Analysis

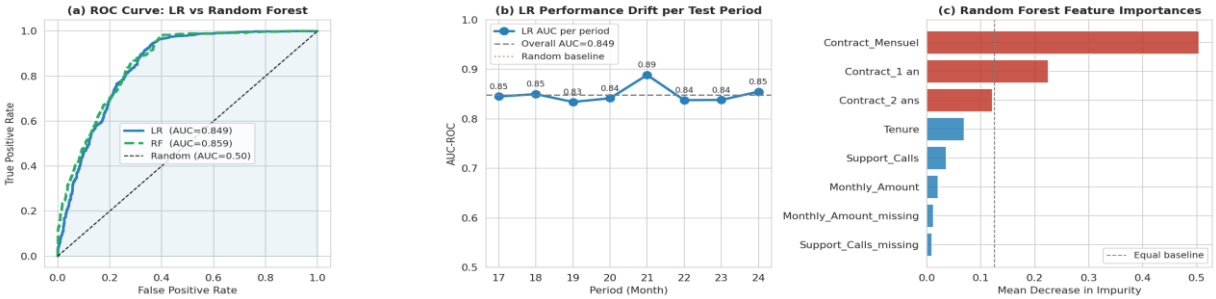

Figure 2 — Model Performance: (a) ROC curves — RF (AUC=0.859) outperforms LR (AUC=0.849); (b) LR AUC per test period showing mild drift-induced decline; (c) RF feature importances.

5. Retraining Strategy and Window Selection

5.1 Automatic Window Selection

Rather than selecting the rolling window size subjectively, we implemented offline backtesting across eight candidates: 3, 4, 5, 6, 7, 8, 10, and 12 months. For each candidate W, walk-forward evaluation was run across all available historical periods and mean AUC across test months served as the selection criterion. This treats window size as a data-driven hyperparameter decision, removing look-ahead bias from the process.

Figure 3a shows that W=3 and W=5 tie on mean AUC (both 0.8654), but W=5 was selected because it has a lower standard deviation (0.0259 vs 0.0269) and a better worst-case performance (min AUC 0.8240 vs 0.8166). The selection criterion used is Mean− 0.5×Std, which balances performance and stability — giving a score of 0.8524 for W=5 versus 0.8519 for W=3. Performance declined for windows of 8 months or more, confirming that incorporating pre-drift data (Months 1-10, churn below 30%) into a model deployed in a high-churn period (40-55%) actively harms prediction. Figure 3b compares fixed and rolling policies period-by-period; the rolling window shows a consistent advantage in later, higher-drift months, confirming that data freshness outweighs data volume under strong concept drift.

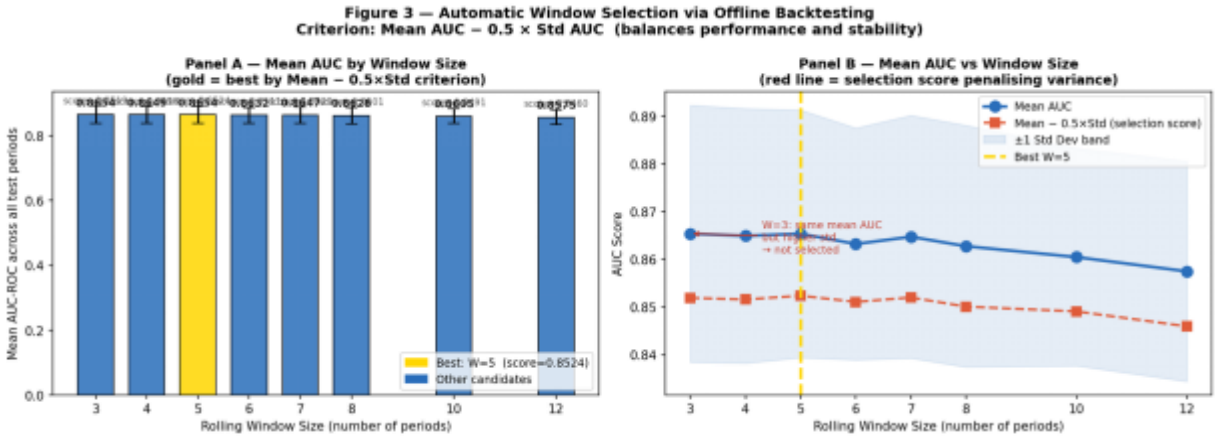

Figure 3 — Window Selection: (a) mean AUC per candidate size — 5 months selected (gold bar); (b) Fixed vs. Rolling AUC per test period — rolling has consistent advantage under strong drift.

6. Business Optimisation : Cost-Sensitive Threshold Tuning

6.1 The Asymmetric Cost Problem

Standard classification optimizes for balanced accuracy, treating false positives and false negatives as equally costly. In churn prediction, this assumption is incorrect. A False Negative (a churner predicted to stay) means no retention action is taken, and the customer is lost, estimated at EUR 50 in forfeited revenue. A False Positive (a loyal customer incorrectly flagged) triggers an unnecessary retention offer costing EUR 5. This 10:1 asymmetry means the conventional threshold of 0.50, which minimises balanced error, is economically suboptimal and leaves significant savings uncaptured.

6.2 Optimal Threshold and Results

Total business cost (False Negatives x EUR 50 + False Positives x EUR 5) was evaluated across 500 candidate thresholds. Figure 4a shows the resulting U-shaped cost curve: very low thresholds create excessive false alarms; very high thresholds miss most churners; the curve minimum identifies the optimal setting. The optimal threshold was found at 0.094, flagging any customer with predicted churn probability above 9.4%. Figure 4b confirms that at this threshold recall is high while precision is moderate, the correct operating point when missing a churner costs 10x more than a false alarm. Figure 4c shows the concrete outcome shift: substantially fewer missed churners at the cost of more false alarms, but with a dramatically lower total cost.

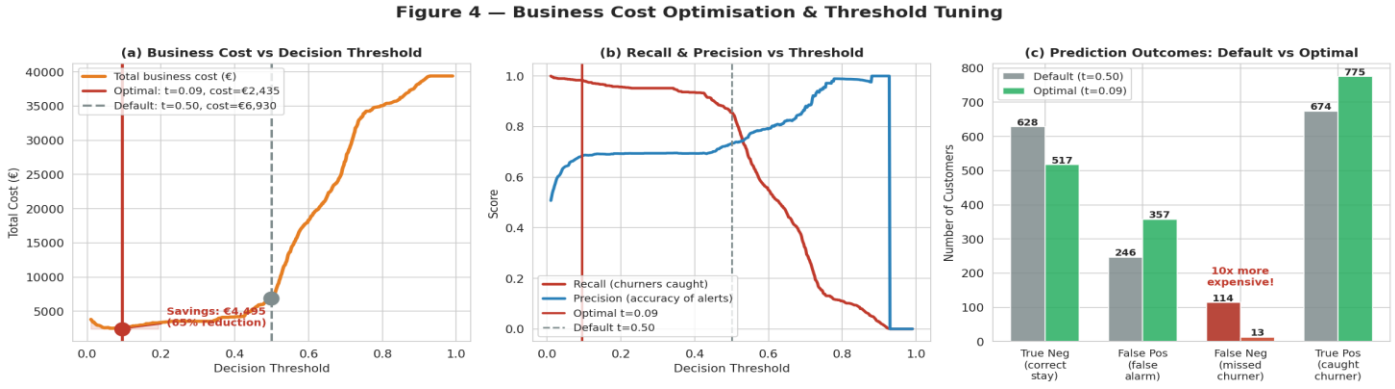

Figure 4 — Business Optimisation: (a) cost curve minimised at threshold 0.094; (b) recall and precision vs. threshold; (c) prediction outcomes — default vs. optimal threshold.

7. Key Design Decisions and Justifications

Parameter / Finding	Value / Result
Temporal split (no random split)	Random splits violate temporal ordering, allowing the model to see future data during training. All splits enforce chronological order: train on the past, test on the future.
Median imputation (not mean)	Monthly_Amount spans EUR 20-131. An outlier would distort the mean-based fill. The median is robust and produces a representative imputed value regardless of skew.
MNAR indicator features	Missing billing data is not random , it predicts churn (55.7% vs. 33.4%). Imputing without indicators discards this signal. Binary flags preserve it after the numeric value is filled.
Rolling over fixed window	With +1.5% churn per month, data from 12 months ago reflects a fundamentally different customer behaviour regime. Data freshness matters more than data volume under strong drift.
5-month window (not 3 or 12)	Selected via offline backtesting using Mean−0.5×Std criterion. W=3 and W=5 tie on mean AUC (0.8654), but W=5 has lower std (0.0259 vs 0.0269) and a better worst-case floor — making it more reliable in production.
Threshold = 0.094 (not 0.50)	A missed churner (False Negative) costs EUR 50 vs. EUR 5 for a false alarm. Lowering the threshold to 0.094 catches far more churners at an acceptable false-alarm rate, minimising total cost.
Random Forest in addition to LR	Captures non-linear feature interactions (e.g., short tenure AND high billing AND monthly contract) that a linear decision boundary cannot express without manual feature engineering. **Best Model & AUC:** Random Forest : AUC = 0.8590 on held-out test set (Months 17-24)