Short Tech Report — Predictive Maintenance

Carlos Torres Sánchez
2025-09-03

Objective

Predict near-term machine failures using the Microsoft Azure Predictive Maintenance dataset.

Data & Label

14-day aggregated base per machine.
Shape: 876100 rows, 12 columns.
Target distribution (14-day horizon): 73.05% no-failure, 26.95% failure.
Business framing: Reduce missed failures (higher recall) while controlling false alarms to keep operational load manageable.

Key Decisions

- **Prediction horizon & aggregation:** We compared 1d, 7d, and 14d targets. A 1d target was too sparse (extreme imbalance) and 7d remained imbalanced ($\approx 86/14$). The **14-day** horizon provided a healthier balance ($\sim 73/27$) and more stable evaluation, so it was selected.
  Why not a longer horizon? Longer windows (>14d) tend to dilute the signal and reduce timing precision, yielding more false positives and less actionable alerts. The 14-day window balances class distribution with near-term actionability.
- **Baseline model choice: Logistic Regression** with **L1 + liblinear + class_weight=balanced** inside a scikit-learn `Pipeline` for reproducibility and auditability.
- **Threshold strategy:** Operating threshold set at **0.393** from **max KS = 0.422** to prioritize **recall**. (See *Results* for full metrics.)
- **Collinearity policy:** Variance Inflation Factor (VIF, threshold ≈ 5) and low-variance pruning were applied to limit redundancy, stabilize coefficients, and reduce noise.
- **Metric focus:** Emphasis on **Recall**, **KS**, and **PR-AUC** over raw accuracy, aligned with the business cost of missed failures.

Methodology

We engineered a 14-day feature window by joining telemetry, events, and asset metadata per machine. This produced 33 candidate variables including recent error and maintenance counts, operational ranges (voltage, rotation, pressure, vibration), and historical failures. The target variable (`fail_next_14d`) indicated whether a failure occurred within the following 14 days.

To ensure robustness, the pipeline enforced temporal validation (training on past, evaluating on future) and encapsulated preprocessing and classification in a single reproducible framework. Feature selection followed two stages: first, multicollinearity control via variance inflation factors

(VIF) to prune redundant signals; second, variance-based filtering to remove near-constant or low-dispersion features. This dual cleanup eliminated unstable predictors and left a compact set of 11 informative features, mainly capturing the statistical behavior of voltage, rotation, pressure, and vibration.

Modeling was performed with logistic regression, tuned through grid search. The operating threshold was calibrated using the Kolmogorov–Smirnov statistic to emphasize recall while maintaining reasonable precision.

Decisions & Trade-offs

- **Parsimony vs. coverage.** Feature reduction (33 → 11 features) through VIF filtering and low-variance pruning improved coefficient stability, mitigated multicollinearity, simplified monitoring, and kept explanations crisp; the trade-off is the potential loss of subtle interactions among correlated signals.
- **Interpretability vs. raw lift.** A regularized Logistic Regression (`L1`, `liblinear`, `class_weight=balanced`) was favored for transparency, calibration, and ease of explanation. More complex ensembles were deferred to future iterations to avoid additional complexity and preserve straightforward attributions.
- **Imbalance handling.** A 2:1 undersampling strategy increased recall but materially raised false positives. Final decision: retain the natural class ratio and adjust the operating threshold (KS-based), maintaining probability stability while making the recall/precision trade-off explicit.
- **Explainability alignment.** SHAP global importance and logistic coefficients consistently highlighted the same key drivers (voltage, rotation, pressure, vibration), reinforcing domain intuition and ensuring auditability.

Results (Test Set) — Logistic Regression Baseline

Model Evaluation — Logistic Regression Baseline.
The baseline logistic regression model (**L1** penalty, **liblinear** solver, **class-weight balanced**) was evaluated on the 14-day prediction target.

Performance summary The baseline logistic regression model shows solid discriminative ability and high recall, at the cost of lower precision. The detailed metrics are summarized in Table 1.

Metric	Value	Notes
ROC AUC	0.788	Discriminative ability
KS (max)	0.422	@ threshold = 0.393
Recall	84.1%	Failures correctly flagged
Precision	42.6%	Share of true failures among alerts
Accuracy	65.1%	Overall correct predictions
Specificity	58.1%	True negative rate

Table 1: Baseline Logistic Regression (L1, liblinear, class-weight balanced) — 14-day prediction target.

Best Model Configuration The final model is a Logistic Regression inside a scikit-learn `Pipeline` with the following hyperparameters:

- Regularization strength: `C = 0.001` (strong L1 penalty)
- Penalty: `l1`, Solver: `liblinear`
- Class weights: `balanced`
- Tolerance: `1e-5`, Max iterations: `3000`
- Intercept included (`fit_intercept = True`)

This configuration enforces sparsity, stabilizes coefficients under collinearity, and improves recall on the minority (failure) class while maintaining probability calibration.

Interpretation This baseline model is simple, interpretable, and stable. It prioritizes **recall**, which is crucial in predictive maintenance.
Main drivers: *voltage (mean, rstd)*, *rotation (mean, std)*, *pressure (mean, range)*, *vibration (mean, rstd)*.

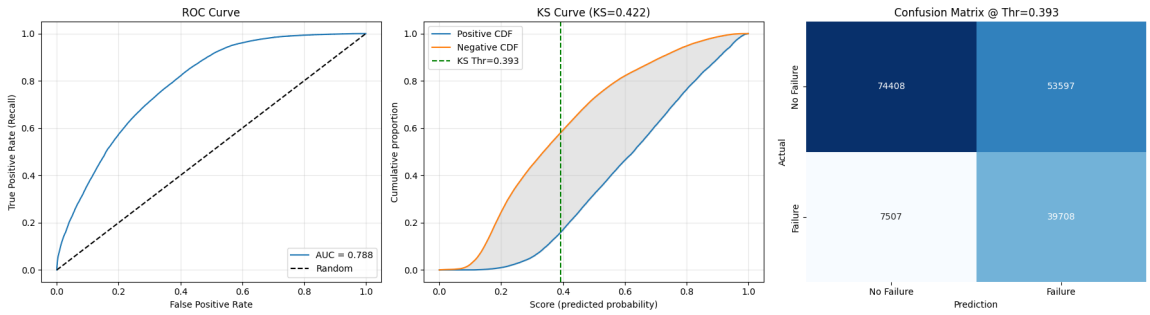

Figure 1: ROC (AUC = 0.788), KS curve (max KS = 0.422, thr = 0.393) and Confusion Matrix at thr = 0.393.

Explainability (SHAP)

We applied SHAP values to interpret the logistic regression model. Global importance confirms that the failure risk signal is dominated by voltage and rotation behavior, followed by vibration and pressure features. The table below summarizes the mean absolute SHAP value per feature.

Feature	Mean |SHAP|
volt_mean_14d	0.366
rotate_std_14d	0.347
rotate_mean_14d	0.321
vibration_mean_14d	0.273
pressure_mean_14d	0.265
pressure_range_14d	0.185
vibration_rstd_14d	0.133
volt_rstd_14d	0.128
vibration_range_14d	0.053
volt_range_14d	0.051
rotate_range_14d	0.000

Table 2: Global SHAP importance (test set).

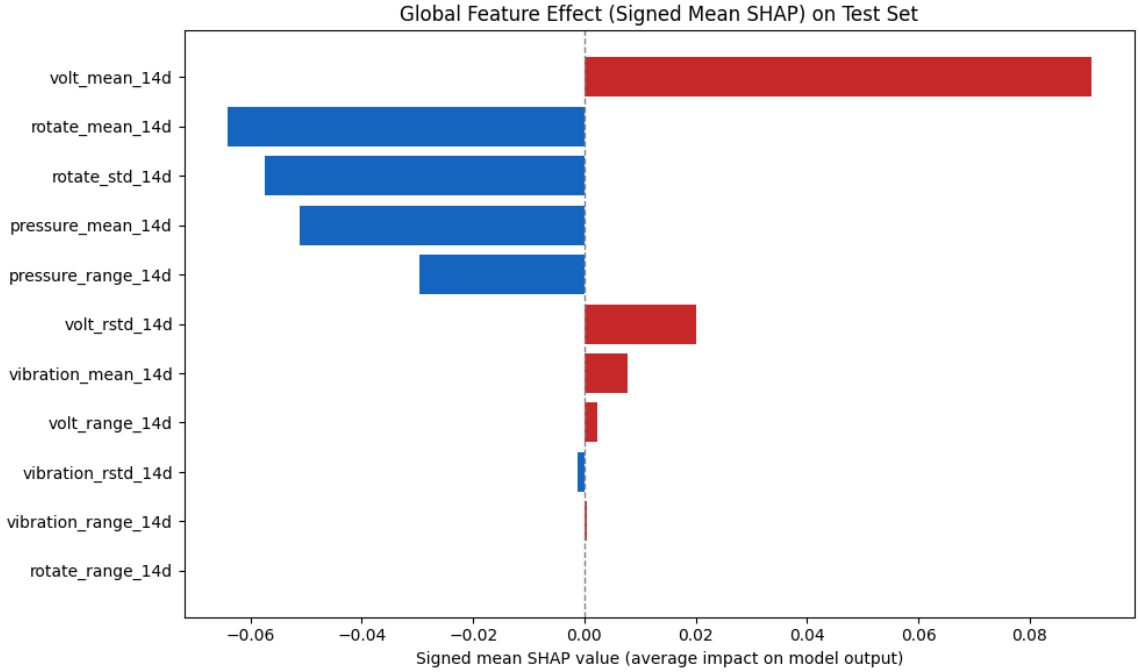

Figure 2: Global SHAP summary plot (bar chart with positive and negative contributions).

Interpretation The SHAP analysis shows that *voltage mean*, *rotation variability (mean and std)*, and *vibration mean* are the strongest predictors of failures, with *pressure signals* also contributing meaningfully. This ranking is consistent with the coefficient-based analysis, reinforcing interpretability and ensuring auditability of the model decisions.

Best Model Summary

Model: Logistic Regression in a scikit-learn Pipeline.
Best hyperparameters (refit=roc_auc): {logreg__C: 0.001, logreg__class_weight: balanced, logreg__fit_intercept: True, logreg__penalty: l1, logreg__solver: liblinear, logreg__tol: 1e-05}.
Best CV AUC: **0.7885**.
CV KS (mean ± std): **0.4247** ± **0.0024**.

Lessons Learned

- **Well-calibrated simplicity beats rushed complexity:** Regularized logistic regression with a carefully chosen threshold provided solid, defensible performance.
- **Threshold rules:** Adjusting the *decision threshold* based on FN/FP costs gave better operational control than aggressive resampling.
- **Dominant drivers:** *voltage (mean, rstd)*, *rotation (mean, std)*, *pressure (mean, range)*, and *vibration (mean, rstd)* consistently emerged as top predictors; SHAP and coefficients aligned, reinforcing interpretability.
- **14-day window:** Delivered better class balance (73/27) and stability than shorter horizons, while preserving actionable near-term signals.

Next Steps

- **Cost-sensitive optimization:** Define explicit FN/FP costs and optimize threshold or the loss accordingly.
- **Calibration:** Validate that predicted probabilities match observed outcomes using reliability diagrams.
- **Feature engineering:** Enrich with trends, derivatives, lags, and component-level signals; explore interaction effects.
- **Benchmarks:** Compare against *LightGBM/XGBoost* (e.g., with monotonic constraints) to explore precision gains while preserving interpretability and controlling false positives.

Appendix — Glossary of Key Features

To support interpretability, the table below defines the engineered features used in the model. This glossary is intended to help non-technical readers connect statistical variables with their operational meaning. All features are computed from telemetry signals aggregated over the past 14 days.

Feature	Description
volt_mean_14d	Average voltage measured over the past 14 days. Captures the central level of machine voltage.
volt_range_14d	Range (max–min) of voltage observed over 14 days, indicating operational spread.
volt_rstd_14d	Relative standard deviation of voltage (std/mean), capturing normalized variability.
rotate_mean_14d	Average rotation speed over the past 14 days.
rotate_std_14d	Standard deviation of rotation speed over 14 days, measuring volatility of shaft/motor activity.
rotate_range_14d	Range (max–min) of rotation speed, showing extremes in operational load.
pressure_mean_14d	Average pressure levels over the past 14 days.
pressure_range_14d	Range (max–min) of pressure observed, highlighting fluctuations in system load.
vibration_mean_14d	Average vibration amplitude measured across 14 days.
vibration_range_14d	Range (max–min) of vibration amplitude, showing extreme oscillations.
vibration_rstd_14d	Relative standard deviation of vibration (std/mean), indicating stability vs. instability of components.

Table 3: Glossary of engineered features (14-day window).