ISSN: 2664-9640

State-Aware Anomaly Detection for Railway Predictive Maintenance: A Reproducible Study on the MetroPT-3 Dataset

Ying Shi

Guangzhou Railway Polytechnic, Guangzhou 511300, China.

Abstract

Railway air-production units operate under changing compressor and train-motion conditions, which can make a single global anomaly model sensitive to normal operating transitions. This paper presents a reproducible state-aware anomaly-detection study on the MetroPT dataset. The downloaded dataset_train.csv contains 10,773,588 one-second records and 21 columns covering pressure, temperature, current, digital control signals, and GPS variables. We aggregate the stream into 60-second windows, identify compressor and motion states, and compare a global robust Z-score baseline, a global Isolation Forest, an RBF kernel baseline, a state-conditioned Isolation Forest, and a temporally smoothed event detector. Time-ordered splits use healthy data for training, later healthy windows for validation, and three documented maintenance failures for testing. The state-conditioned model reaches window-level precision 1.000 and recall 0.0328, while the smoothed detector reaches precision 0.9259 and recall 0.0282. Both methods detect two of the three failure events at the documented failure intervals; smoothing changes the distribution of alarms and recovers the short second event, but it does not improve aggregate window-level F1 in this initial configuration. These results show why event-level and window-level evaluation should be reported together and provide a reproducible baseline for future railway predictive-maintenance studies.

Keywords

Railway Predictive Maintenance; Anomaly Detection; Operating-State Conditioning; Event-Level Evaluation.

1. Introduction

Predictive-maintenance studies for railway and industrial equipment commonly face two coupled difficulties: failures are rare, and the distribution of healthy measurements changes with load, control commands, and machine operating mode. The MetroPT dataset was introduced as a real-world benchmark for this setting, with synchronized analog, digital, and GPS signals from an air-production unit. Its maintenance reports provide useful event descriptions, but they do not provide a continuously annotated failure signal. Consequently, an anomaly detector must be trained mainly from healthy observations and evaluated with event intervals defined after the fact [1], [2].

Unsupervised detectors are attractive in this setting because they do not require a large collection of labeled failures. Isolation Forest isolates unusual observations through random partitioning and is computationally practical for a large feature matrix [3]. Distance- and density-based methods, including standardized scores and one-class kernel methods, provide complementary baselines because they respond differently to changes in scale and local density [4]. However, a single global model implicitly assumes that one healthy distribution is adequate for all compressor and train-motion conditions. That assumption is questionable when a sensor value is normal during loading but unusual during shutdown, or when a change in GPS speed coincides with a compressor transition.

ISSN: 2664-9640

State-conditioned detection addresses this issue by estimating normality within operating regimes. Temporal smoothing and event aggregation address a second issue: a maintenance event can generate several weak, adjacent abnormal windows rather than one isolated extreme point. The present study combines both ideas in a reproducible, time-ordered comparison and reports window-level and event-level results separately [5]. This design makes the trade-off visible: a method can obtain high precision on a small number of alarm windows while still missing most windows inside a long failure interval.

Unexpected failures of an air-production unit can remove a metro vehicle from service. Sensor streams from the compressor contain pressure, temperature, current, digital control, and GPS information, but the same sensor value can have different meanings when the compressor is off, unloaded, loaded, parked, or moving. A global anomaly score can therefore confuse an operating transition with a fault.

This paper studies the following question: does conditioning an unsupervised anomaly detector on the operating state provide more useful failure-event detection than a global detector under a time-ordered evaluation protocol? The contributions are:

a chunk-compatible and reproducible preprocessing pipeline for the large MetroPT CSV;

a state-conditioned Isolation Forest using compressor and train-motion states;

a comparison of global and state-conditioned detectors with temporal smoothing; and

a joint window-level and event-level evaluation on three maintenance-report failure intervals.

The conclusions are limited to this MetroPT file and its three documented failures. The work does not claim universal railway-equipment performance.

2. Method

2.1. Dataset, Features, and Operating States

The section headings are in boldface capital and lowercase letters. Second level headings are typed as part of the succeeding paragraph (like the subsection heading of this paragraph). All manuscripts must be in English, also the table and figure texts, otherwise we cannot publish your paper. Please keep a second copy of your manuscript in your office. When receiving the paper, we assume that the corresponding authors grant us the copyright to use the paper for the book or journal in question. When receiving the paper, we assume that the corresponding authors grant us the copyright to use the paper for the book or journal in question. When receiving the paper, we assume that the corresponding authors grant us the copyright to use.

MetroPT records an air-production unit at 1 Hz. It contains 10,773,588 records from 2022-01-01 06:00:00 to 2022-06-02 15:49:53. No missing values or duplicate timestamps were found. Most adjacent records are one second apart, but 147 gaps of approximately four hours and several other long gaps occur. Windows crossing a gap greater than five minutes are therefore cut. The maintenance report defines three failure intervals, which are listed in Table 1.

Table 1. Documented failure intervals used for post-hoc evaluation.

ID	Type and component	Start	End
F1	Air leak, clients	2022-02-28 21:53	2022-03-01 02:00
F2	Air leak, air dryer	2022-03-23 14:54	2022-03-23 15:24
F3	Oil leak, compressor	2022-05-30 12:00	2022-06-02 06:18

The analyzed dataset contains 14,821, 1,801, and 202,683 one-second records in F1, F2, and F3, respectively. The intervals are used as post-hoc labels; the raw CSV itself does not include a failure-label column.

ISSN: 2664-9640

The 21 fields are grouped into analog process variables, digital control variables, and GPS-related variables. This grouping is used only to define feature operations: analog channels receive distribution statistics, digital channels receive one-valued proportions, and GPS speed is retained as a motion feature. A binary control signal is therefore not interpreted as a continuous measurement with a meaningful standard deviation in the same way as pressure or temperature.

The file-level checks found no missing values or duplicate timestamps, but the time axis is not perfectly continuous. The 147 approximately four-hour gaps are treated as acquisition gaps rather than as long stationary periods. A 60-second window is cut when it crosses a gap longer than five minutes. This rule prevents a feature mean from combining observations separated by an unobserved interval.

The stream is aggregated into 60-second windows. For eight analog variables, the pipeline computes mean, standard deviation, minimum, and maximum. For eight digital variables, it computes the fraction of samples equal to one. It also computes mean GPS speed. This aggregation produces 179,709 windows.

States are assigned from motor current and GPS speed: compressor-off, compressor-unloaded, or compressor-loaded, each with a moving or stopped suffix. The rule is deterministic and is applied before model fitting, validation, and testing. The state label is therefore available at inference time and does not use the maintenance label.

2.2. Experimental Setup and Evaluation

The first failure is held out together with its preceding two-hour context. Healthy windows before that context form the training set; later non-test windows form validation data. F1, F2, and F3 each form a separate test split. The final split contains 70,474 training windows, 105,334 validation windows, and 368, 151, and 3,382 windows for F1, F2, and F3 tests. All feature means and standard deviations are estimated from training windows only.

M0 is a global standardized Z-score, using the maximum absolute standardized value across its selected window features. M1 is a global Isolation Forest with 100 trees, a subsample size of 256, maximum depth 8, and a fixed random seed. M2 is an RBF kernel novelty baseline using 512 healthy prototypes and a validation-derived lower-tail threshold. M3 trains 80-tree Isolation Forests separately for each observed operating state. M4 applies a five-window moving average to the M3 score and merges alarms separated by no more than five windows into an event. For each method, the alarm threshold is selected from healthy validation scores.

Window-level labels are positive when a window overlaps a documented failure interval. We report precision, recall, and F1. Event-level recall reports the fraction of the three documented failures for which at least one alarm overlaps the failure interval. Alarms in the two-hour context are retained for lead-time analysis but are not counted as true positives for the interval-overlap window metrics.

The pipeline has four stages. The CSV is read in chunks and timestamps are sorted within the resulting stream. Gap-aware 60-second windows are then converted to a shared feature matrix. The chronological split is applied next, with normalization statistics fitted only on the training windows. Finally, each detector produces a score, healthy validation scores determine the threshold, and held-out windows are evaluated using that fixed threshold. The complete processing workflow is summarized in Fig. 1.

ISSN: 2664-9640

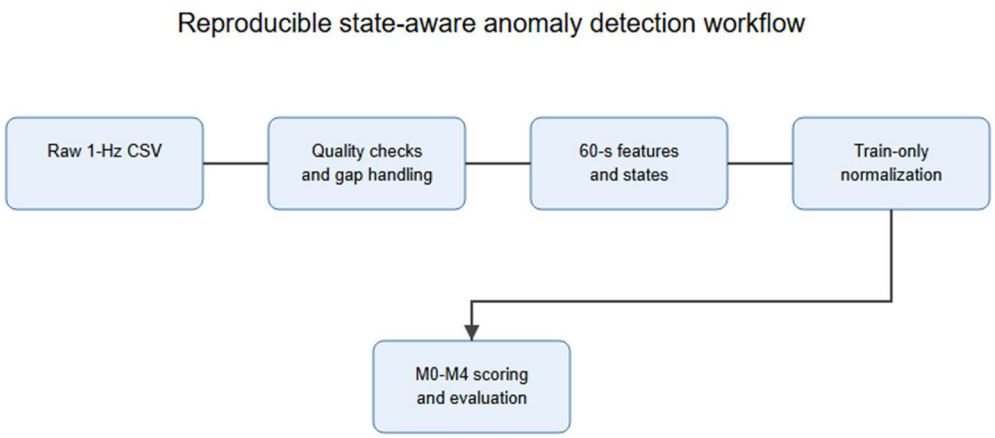


Fig. 1 Reproducible processing workflow

The fixed seed and parameter settings define the experimental configuration. They do not imply that the parameters are optimal; the threshold and event merge rule determine the precision–recall and event-grouping trade-offs. In particular, the validation threshold controls the precision–recall trade-off, whereas the event merge rule controls whether neighboring weak alarms are interpreted as one event. Table 2 summarizes the feature, model, aggregation, and threshold settings used in the experiments.

Table 2. Experimental settings and detector configuration.

Component	Current setting
Input sampling	1 Hz; chunk-compatible CSV reading
Feature window	60 s; cut across gaps longer than 5 min
Training normalization	Mean and standard deviation from healthy training windows only
Global Isolation Forest	100 trees, subsample 256, maximum depth 8
State-conditioned Isolation Forest	80 trees per observed state
Kernel baseline	512 healthy prototypes, fixed random seed
Temporal aggregation	Five-window moving average; merge gap at most five windows
Threshold	Healthy validation score quantile

The split also separates model selection from failure assessment. The threshold is selected from later healthy windows rather than from F1–F3, and no failure interval is used to choose a feature scale or an operating state. This prevents the reported event hits from being obtained by tuning directly on the test failures. It also means that a failure with a different operating context can expose a genuine generalization error, as seen in the contextual alarms around F3.

3. Results and Discussion

3.1. Baseline and State-Conditioned Results

M0 detects F1 but not F2 or F3 at its validation-derived threshold. M1 and the M2 RBF kernel baseline produce at least one alarm in all three held-out event windows. The global models are therefore sensitive at the event level, but this observation does not by itself establish a better false-alarm or window-level performance.

When the stricter interval-overlap rule is applied, M0 detects F1, M1 detects F1–F3, and M2 detects F1 and F3. The difference for M2 comes from an alarm in the F2 test context that does not overlap the documented F2 interval.

ISSN: 2664-9640

Table 3. Comparison of the five detectors. M0–M2 use the same chronological split, training normalization, and validation-derived thresholds. Event hits use the strict interval-overlap rule, so a contextual alarm outside a documented failure interval is not counted.

Method	Event hits (of 3)	Precision	Recall	F1	PR-AUC
M0 global robust Z-score	1	0.9948	0.0537	0.1018	0.9330
M1 global Isolation Forest	3	0.9954	0.0607	0.1145	0.9375
M2 RBF kernel baseline	2	0.9862	0.0604	0.1139	0.9527
M3 state-conditioned IF	2	1.0000	0.0328	0.0634	0.8932
M4 smoothing + aggregation	2	0.9259	0.0282	0.0548	0.8820

Table 3 reports the pooled window-level and event-level results for all five detectors. Figure 2 compares the window-level precision, recall, and F1 of M3 and M4. M3 has precision 1.000 and recall 0.0328 over the pooled test windows. M4 has precision 0.9259 and recall 0.0282. Both have event recall 2/3. M4 produces alarms overlapping F1 and F2, while its F3 alarms in this run are mainly in the preceding context rather than inside the documented interval.

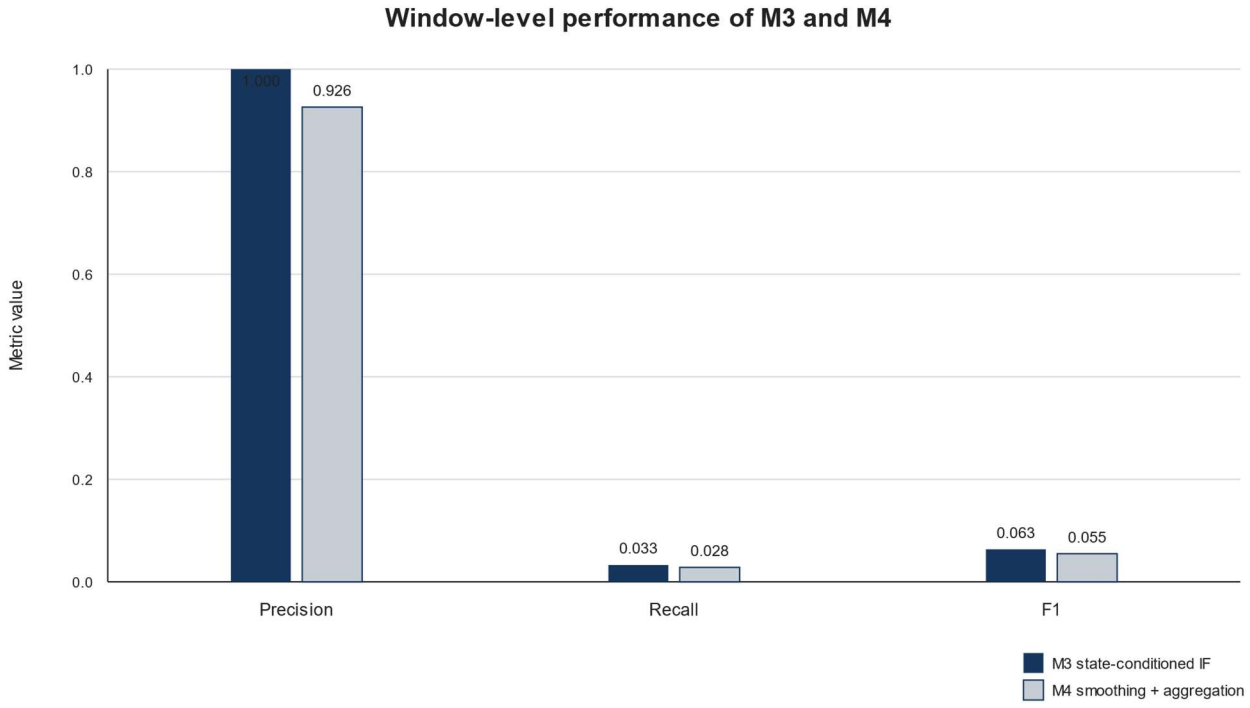


Fig. 2 Window-level metrics of M3 and M4

The event-level view explains why the pooled window metrics should not be read in isolation. M0 reaches only F1, whereas M1 produces an in-interval alarm for each of F1–F3. M2 produces alarms in all three held-out test segments, but its F2 alarm is confined to the preceding context, so its strict interval-overlap hits are F1 and F3. The state-conditioned detector M3 reaches F1 and F3, while M4 reaches F1 and F2. Thus, the two methods have identical event recall in this run while differing in which failure intervals they detect and in their window-level precision and recall.

The pooled ranking PR-AUC is 0.8932 for M3 and 0.8820 for M4. This ranking measure is reported separately because it summarizes ranking performance across score thresholds, whereas precision, recall, and F1 use one validation-selected operating threshold.

For the M0 baseline, the 99.5th-percentile validation threshold corresponds to approximatel

ISSN: 2664-9640

y 5.64 alarms per day. The rate is reported to show the trade-off between high precision and sparse window recall.

3.2. Interpretation and Limitations

The results support two cautious observations. First, separating compressor and motion states changes the score distribution and can reduce alarms caused by ordinary operating transitions. Second, temporal aggregation changes event behavior: it can connect weak neighboring signals and recover a short event, but it may also move alarms into the pre-failure context and reduce strict interval-overlap recall. The high imbalance between healthy and failure windows makes event-level reporting essential.

The study has several limitations. Only three maintenance events are available. The failure intervals are post-hoc reports rather than online labels, and the downloaded file has a different total row count from the descriptive count in the dataset paper. The method uses fixed state rules and one window length. The conclusions should therefore be read as a reproducible dataset-level method comparison, not as evidence of universal railway deployment performance.

From an operational perspective, the detector should be treated as an event-screening aid rather than an automatic maintenance decision rule. A high-precision alarm can prioritize inspection, but the low window-level recall means that the absence of an alarm cannot certify that the unit is healthy. The two-hour context is also operationally meaningful: a pre-failure alarm may be useful for planning even when strict interval-overlap scoring classifies it as a false positive. Future evaluation should therefore report both strict interval metrics and a pre-failure warning metric with a fixed, pre-registered tolerance.

The external validity of the comparison is limited in three ways. The file contains one asset stream and only three documented failures, so the observed event recall has high uncertainty. The maintenance intervals may include a reporting delay or a gradual degradation period, so their boundaries should not be interpreted as exact physical onset times. Finally, the state rules use current and GPS speed as proxies for operating mode; another vehicle or another sensor configuration may require different thresholds. These limitations motivate repeated failure holdout and cross-asset evaluation before any deployment claim.

4. Conclusion

This paper provides a reproducible state-aware anomaly-detection pipeline for MetroPT. The pipeline performs chunk-compatible quality checks, gap-aware 60-second feature extraction, training-only normalization, state-conditioned modeling, and separate window and event evaluation. In this evaluation, M3 and M4 detect two of three documented failures at the event level, with different trade-offs between precision and window recall. Future work should add repeated leave-one-failure-out training, calibrated lead-time metrics, sensitivity analysis for state rules and window lengths, and an independently implemented One-Class SVM baseline.

Acknowledgments

This research work is supported by the scientific research start-up project for newly introduced talents (GTXYR2210) of Guangzhou railway polytechnic.

References

[1] Veloso, B., Ribeiro, R. P., Gama, J., & Pereira, P. M. (2022). The MetroPT dataset for predictive maintenance. Scientific Data, 9, 764. https://doi.org/10.1038/s41597-022-01877-3

ISSN: 2664-9640

[2] Veloso, B., Gama, J., Ribeiro, R. P., & Pereira, P. M. (2022). MetroPT: A benchmark dataset for predictive maintenance. Zenodo. https://doi.org/10.5281/zenodo.6854240

[3] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In Proceedings of the 8th IEEE International Conference on Data Mining (pp. 413–422). https://doi.org/10.1109/ICDM.2008.17

[4] Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. Neural Computation, 13(7), 1443–1471. https://doi.org/10.1162/089976601750264965

[5] Jardine, A. K. S., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing, 20(7), 1483–1510. https://doi.org/10.1016/j.ymssp.2005.09.012