

A Leakage-Safe, Interpretable Machine Learning Baseline for CNC Tool Wear Classification: Reproducibility Lessons from the UC Berkeley Milling Dataset

Md Ryshur Rahman Turin

22met008p@student.cuet.ac.bd

Chittagong University of Engineering & Technology

Md Abid Hasan

Chittagong University of Engineering & Technology

Jamal Uddin Ahamed

Chittagong University of Engineering & Technology

Research Article

Keywords: Computer Numerical Control, Machine Learning, Predictive Maintenance, Random Forest, Tool Condition Monitoring

Posted Date: October 20th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11364337/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Abstract

This study presents a fully reproducible, leakage-safe evaluation of tool wear classification on the Milling Data Set provided by the University of California (UC), Berkeley, United States, comparing a Random Forest classifier against a majority-class baseline and logistic regression under a grouped train and test split by tool case. Unplanned tool failure in Computer Numerical Control (CNC) milling remains a significant source of scrap and unplanned downtime, motivating extensive prior work on machine-learning-based tool condition monitoring. Much of this study, however, reports accuracy without disclosing evaluation protocol details that materially affect its validity, particularly with respect to data leakage across correlated samples from the same tool. Rather than claiming state-of-the-art accuracy, recent deep learning approaches on this same dataset have already reported in the 95–99% range. This work's contribution is methodological: an openly released, tested pipeline; an explicit, quantified demonstration that a simpler linear model (89.5% held-out accuracy) can outperform a more complex ensemble method (84.2%) on a small industrial dataset; and a SHapley Additive exPlanations (SHAP)-based validation showing the model's most influential features (spindle vibration and motor current statistics) align with established physical indicators of tool wear. We report these findings, including a data-quality artifact discovered during preprocessing, as a case study in evaluation rigor for small-sample industrial machine learning, and outline how this pipeline is being extended to a live, multi-machine industrial deployment.

INTRODUCTION

Cutting tool wear is a primary driver of part quality loss, unplanned downtime, and cost in Computer Numerical Control (CNC) milling operations. Machine-learning (ML)-based tool condition monitoring (TCM) has consequently become an active research area, spanning classical statistical-feature approaches to deep learning on raw sensor signals and machined-surface imagery. A large fraction of this literature reports high classification accuracy; comparatively little of it reports evaluation protocol details, in particular, whether train or test splits respect the grouping structure of the underlying experiments that determine whether that accuracy is trustworthy.

The dataset used in this study originates from milling experiments conducted at the University of California (UC), Berkeley, and distributed via the NASA Ames Prognostics Center of Excellence Data Set Repository [1]. It has since been used across a wide range of modeling approaches. Von Hahn and Mechefske [2] articulate a set of best practices for building ML systems in CNC machining, prioritizing data infrastructure, beginning with simple models, remaining cognizant of data leakage, and using open-source tooling, applied to a real-world manufacturing dataset; this study operationalizes closely related practices (grouped evaluation, baseline comparison, and open tooling) and explicitly credits [2] as prior articulation of this methodological stance rather than claiming it as novel.

In addition to that, Von Hahn and Mechefske [3] applied a disentangled variational autoencoder in a self-supervised setting to the same milling dataset used here, reporting a precision-recall area-under-curve of

0.45 across cutting parameters; because this metric addresses an anomaly-detection framing distinct from the balanced two-class accuracy reported in the present work, the two results are not directly comparable, and no superiority claim is made in either direction. Vision-based approaches have also been applied to tool wear assessment: García-Ordas et al. [4] used shape descriptors from tool images with classical machine learning, and [5] proposed a low-cost online monitoring system based on local texture analysis of machined surfaces. More recent deep learning approaches, summarized in a comprehensive 2025 survey covering over 160 references on machine learning applied to CNC milling [6], report classification accuracies in the 95–99% range on variants of this dataset and related milling benchmarks using LSTM and convolutional architectures. This body of work establishes that, given sufficient architectural sophistication and feature learning capacity, substantially higher accuracy is achievable than what classical statistical-feature methods produce: a gap this paper does not attempt to close.

This concern is not specific to tool condition monitoring or even to manufacturing. Kapoor and Narayanan [10] document leakage-driven overoptimism across 294 papers spanning seventeen scientific fields that have adopted machine-learning methods and propose "model info sheets" as a partial remedy. The case examined in this paper, including the near-miss leakage channel, is best read as one small, fully disclosed instance of that broader, well-documented pattern. A direct point of comparison is available on this exact dataset. Karabacak [11] trained convolutional architectures (Google Net, ResNet-50, AlexNet, and EfficientNet-B0) on short-time-Fourier-transform spectrograms of the same UC Berkeley NASA Ames milling signal used in the present study, reporting test accuracies ranging from 30% to 100% depending on the architecture, the sensor channel used, and the number of training spectrograms available, with results near 100% obtained only by the deepest architectures given a single sensor channel and a large training set. That study also found that fusing spectrograms from multiple sensor channels together reduced model performance relative to using the best single channel alone, a finding in some tension with the multi-channel feature fusion approach used here, which we do not attempt to reconcile in this paper.

More broadly, recent deep learning work on related milling tool-wear tasks continues to report accuracy in the range invoked above: a ConvNeXt architecture trained with minimal per-machine data reports 99.1% accuracy on a cross-machine transfer task [12], a hybrid autoencoder-LSTM architecture reports over 95% R^2 for remaining-useful-life estimation [13], and a 1-D convolutional network reports 94.5% accuracy on a related chipboard-milling wear-classification task [14]. These results substantiate, with specific recent examples rather than a survey citation alone, the accuracy range this paper trails and explicitly does not attempt to match.

This paper does not attempt to advance the state of the art in raw predictive accuracy on tool wear classification; recent deep learning work on the same benchmark used here already reports substantially higher accuracy than classical methods achieve [2], [3].

Instead, this paper's contribution is methodological and pedagogical: (i) a fully open, tested, leakage-safe pipeline for tool wear classification, released with a model card and continuous-integration tests; (ii) an explicit, quantified case in which a simpler linear model outperforms a more complex ensemble method on a small industrial dataset, reported plainly rather than smoothed over; (iii) a SHAP-based explainability analysis validating that the model's learned decision structure is consistent with established physical indicators of tool wear; and (iv) documentation of a data-quality artifact (a corrupted sensor reading) discovered during preprocessing, offered as a caution to others reusing this benchmark's derived, redistributed forms.

METHODOLOGY

The UC Berkeley Milling Data Set comprises 16 cases (tool or experiment combinations) and 167 individual cuts. Each cut records depth of cut (DOC), feed rate, workpiece material, flank wear (VB), a three-class wear-state label (healthy, degraded, failed), and six raw sensor channels sampled during the cut: spindle motor AC and DC current (smcAC, smcDC), table and spindle vibration (vib_table, vib_spindle), and table and spindle acoustic emission (AE_table, AE_spindle) [1]. Following the two-class framing adopted in this study, the degraded and failed labels are merged into a single “worn” class, yielding a binary healthy-versus-worn classification problem.

During feature extraction, one cut (case 2, run 1) exhibited sensor statistics on the order of 10^{25} – 10^{32} across nearly every channel, whereas all other 166 cuts fall within an approximate range of -5 to 5 . This is inconsistent with any plausible physical measurement and is treated as a corrupted read introduced during signal extraction or file conversion rather than a genuine outlier tool state. This row is excluded via an automated filter (values exceeding a fixed magnitude threshold across all feature columns) rather than a manually hard-coded case/run exclusion, so the same logic generalizes to a re-extraction of the raw signal file. We flag this explicitly because such artifacts are easy to propagate silently through derived, redistributed versions of a benchmark dataset. For each cut and each of the six raw sensor channels, three summary statistics are computed: the mean, the standard deviation, and the root-mean-square (RMS) value of the raw time-series signal. This yields 18 features per cut. This is a deliberately simple feature set, chosen to keep the modeling pipeline interpretable and to isolate the evaluation-methodology contribution from feature-engineering sophistication; more elaborate spectral or wavelet-domain features are common in the deep-learning literature.

Samples are split into training and held-out test sets using a grouped split keyed on the case identifier (GroupShuffleSplit, 25% test size, fixed random seed), rather than a naive random row-wise split. Because multiple cuts from the same case are temporally and physically correlated, as they represent the same tool degrading over time under the same cutting setup, a row-wise split allows information from a given tool to appear in both training and test sets, which inflates apparent accuracy. Model selection during training uses 4-fold grouped cross-validation (GroupKFold, grouped by case) on the training partition only; the held-out test partition is scored exactly once, after model fitting is complete, to avoid test-set leakage through repeated evaluation.

Three models are compared. A majority-class baseline (DummyClassifier) establishes the accuracy achievable with no learned signal. A logistic regression model (scikit-learn default L2 regularization, max_iter = 1000) serves as a linear baseline. A Random Forest classifier [7] (300 trees, fixed random seed) serves as the primary nonlinear model, consistent with its established use in prior tool-wear-monitoring literature applying classical machine learning [2]. Feature attributions for the Random Forest model are computed using SHAP (SHapley Additive exPlanations) Tree Explainer [8] on the held-out test set, producing per-feature Shapley values that quantify each feature's marginal contribution to individual predictions. This serves as a validation step: if the model's most influential features correspond to established physical indicators of tool wear, this provides evidence the model has learned a physically meaningful decision boundary rather than an artifact of the specific data split. The full pipeline, data loading, the grouped-split logic, baseline and Random Forest training, and explainability generation are released as open, tested source code (scikit-learn [9] and SHAP [8]), with unit tests covering the leakage-safety property and the corrupted-row filter, a continuous-integration workflow, and an accompanying model card documenting intended use, training data provenance, and known limitations.

RESULTS

Table 1 reports classification accuracy for CNC tool-wear state (worn vs. not-worn) under a leakage-safe, grouped train or test evaluation protocol, in which all cuts belonging to the same tool case were kept together within either the training or the held-out test partition, preventing the model from seeing near-duplicate readings from the same tool during both training and evaluation. Four models are compared. A majority-class baseline, which simply predicts the more frequent class for every instance, achieves 63.2% accuracy; this establishes the floor against which any genuinely predictive model must be judged. A logistic regression classifier reaches 89.5%, while a Random Forest classifier achieves $78.0\% \pm 4.1\%$ under grouped 4-fold cross-validation on the training partition and 84.2% on the final held-out test set of $n = 38$ cuts drawn from 4 tool cases not seen during training (following removal of one corrupted sensor row from the raw dataset).

Two points merit emphasis for correct interpretation. First, all reported figures substantially exceed the majority-class floor of 63.2%, indicating that the extracted sensor features (spindle or table vibration, motor current, and acoustic emission statistics) carry a genuine, non-trivial predictive signal for tool wear state, as the models are not merely exploiting class imbalance. Second, the gap between the Random Forest's cross-validated training accuracy (78.0%) and its held-out test accuracy (84.2%) is a useful reproducibility signal: the held-out estimate falling within a plausible range of the cross-validation (CV) estimate (rather than being implausibly higher) supports the conclusion that the grouped, leakage-safe protocol is functioning as intended, rather than producing an inflated, optimistic accuracy estimate of the kind common in tool-wear literature that splits at the row level instead of the case level.

Table 1
Model accuracy under a leakage-safe, grouped train or test split (n = 38 held-out test cuts across 4 held-out cases, following removal of one corrupted row)

Model	Accuracy
Majority-class baseline	63.2%
Logistic regression	89.5%
Random Forest (grouped 4-fold CV, training set)	78.0% ± 4.1%
Random Forest (held-out test cases)	84.2%

Figure 1 presents the confusion matrix corresponding to the Random Forest model's performance on the held-out test set (the 84.2% result in Table 1). Of 14 truly "not-worn" cuts (class 0), 11 were correctly classified and 3 were misclassified as "worn"; of 24 truly "worn" cuts (class 1), 21 were correctly classified and 3 were misclassified as "not-worn." Summing the diagonal, $(11 + 21) / 38 = 84.2\%$, exactly reproducing the held-out accuracy reported in Table 1, confirming internal consistency between the tabulated summary statistic and the underlying classification outcomes. Notably, the error rate is symmetric (3 false positives, 3 false negatives), suggesting the model is not systematically biased toward either class, which is a desirable property for a monitoring system where both false alarms and missed wear events carry operational cost.

Figure 2 shows a SHAP (SHapley Additive exPlanations) Beeswarm plot, which decomposes the Random Forest's predictions into per-feature, per-instance contributions across all 18 engineered features (mean, RMS, and standard deviation of each of six sensor channels). Each point represents one held-out instance's SHAP value for a given feature; horizontal position indicates the magnitude and direction of that feature's contribution to the predicted class, while point color encodes the underlying feature value (blue = low, red = high). The plot is ranked by mean absolute SHAP value and shows that mean and RMS spindle vibration (vib_spindle_mean, vib_spindle_rms) are the two most influential features, followed by mean DC spindle motor current (smcDC_mean).

The color gradients are physically interpretable: low spindle vibration values (blue) push predictions toward one class, while high values (red) push toward the other, consistent with the established physical expectation that vibration amplitude and motor load increase as tool wear progresses. This alignment between the model's data-driven feature ranking and known physical wear mechanisms strengthens confidence that the classifier is learning a physically grounded signal rather than an artifact of the evaluation protocol.

DISCUSSION

Logistic regression achieves 89.5% held-out accuracy, exceeding the Random Forest's 84.2%. This result is reported directly rather than treated as a negative outcome to be minimized. With 128 training rows

drawn from only 12 tool cases, the Random Forest's greater representational capacity affords more opportunity to fit case-specific idiosyncrasies that do not generalize to held-out cases, consistent with a standard bias-variance account of model selection under small sample sizes. This does not indicate that ensemble methods are inappropriate for tool wear classification in general, as they are widely and successfully used at larger data scales [2], [6]; but it does indicate that model complexity should be validated against simpler baselines rather than assumed appropriate by default, particularly in early-stage industrial deployments where labeled failure data is inherently scarce.

Spindle vibration and spindle motor current are established indirect indicators of tool condition in the machining literature [6], reflecting the increased cutting forces and dynamic instability associated with a degraded cutting edge. The Random Forest's SHAP-ranked feature importance converging on statistics of exactly these two channels, without those channels being weighted or selected a priori, offers converging evidence that the classifier has learned a physically grounded relationship rather than fitting incidental correlations specific to this dataset's experimental design. The accuracy reported here (84.2–89.5%) trails the 95–99% range reported by deep learning approaches applied to closely related milling tool wear tasks [2], [3], [6], [12], [13], [14], and specifically the up-to-100% results obtained by deep convolutional architectures on this exact dataset under favorable single-channel, large-training-set conditions [11]. This gap is consistent with the deliberately simple feature set (Section 3.3) and classical model choice used here and is not evidence that classical methods are categorically inferior; rather, it reflects a trade-off this study makes explicitly in favor of interpretability, computational cost, and evaluation transparency over maximizing accuracy.

It is also worth noting that [11]'s own results on this dataset varied widely (30–100%) across architecture and signal-channel choices and degraded under multi-channel fusion, a reminder that the deep-learning comparison point is itself configuration-sensitive rather than a single fixed target. We note this gap directly so that this work is not mistaken for, or compared unfavorably without context against, accuracy-optimized deep learning results obtained under different (and not always disclosed) evaluation protocols.

During preparation of this study, an initial implementation selected model features by excluding known non-feature columns (case, run, wear label, and process parameters) rather than by explicitly listing the intended sensor-derived features. This exclusion-based approach silently admitted three additional columns present in the source metadata, a within-case cut index and two window-boundary timestamps, as if they were sensor features. The two window-boundary columns are constant within a given case, meaning they function as a near-perfect case-identity proxy: a model given access to them could, in principle, learn to recognize which case a held-out row belongs to rather than learning from its sensor readings, undermining the leakage-safety this study's grouped split is specifically designed to guarantee, despite the split itself being implemented correctly.

This was caught by comparing results across two independent environments (a cloud notebook and a separate local environment) and finding a discrepancy in Random Forest accuracy (84.2% versus 81.6%)

that persisted despite identical data, identical code logic, and a fixed random seed (an anomaly), given that the baseline and logistic regression accuracies matched exactly across both environments. Inspecting the discrepancy traced it to the feature list rather than to any environment difference. The fix replaces the exclusion-based column selection with an explicit include-list naming, only the eighteen intended sensor-statistic features; re-running the corrected pipeline in both environments now yields identical Random Forest accuracy (84.2%). We report this incident directly, rather than silently correcting it, because it is a concrete demonstration of exactly the class of subtle leakage risk this paper otherwise discusses only abstractly, and because cross-environment comparison, not code review alone, is what surfaced it.

Limitations

1. Dataset scale: Sixteen cases from a single milling machine under a narrow range of cutting conditions substantially limits generalizability to other machines, tool geometries, or materials.
2. Class simplification: Collapsing the degraded and failed labels into a single “worn” class discards information relevant to a deployed early-warning system, which would ideally distinguish incipient from severe wear.
3. Feature simplicity: Mean, standard deviation, and RMS per channel are a minimal feature set; spectral, wavelet, or learned representations capture additional signal structure this study does not exploit.
4. Single corrupted-row handling: The corrupted sensor reading identified was filtered rather than root-caused; a systematic audit of the extraction pipeline against the original raw file was not performed.
5. Feature-selection fragility: An exclusion-based feature selection initially admitted non-sensor metadata columns as features; while corrected here, this class of error is easy to reintroduce (e.g., if the source metadata schema changes) and is not guarded against beyond the unit tests included in this pipeline.

CONCLUSION

This paper presented a fully reproducible, leakage-safe evaluation of tool wear classification on the UC Berkeley Milling Data Set, contributing an open pipeline, an explicit small-sample model-complexity finding, and a physically grounded explainability analysis, rather than a state-of-the-art accuracy claim. We consider this contribution methodological rather than competitive with the deep learning literature on raw accuracy and have aimed to state that trade-off plainly throughout rather than let a strong-sounding results table imply otherwise. The evaluation protocol, baseline-comparison discipline, and explainability validation demonstrated here are being extended to a live industrial deployment: a real-time monitoring system covering approximately 100 machines (pulverizers, pin mills, ribbon blenders, and vacuum pumps) in an operating spice processing plant, using ESP32-based sensor nodes, MQTT ingestion, and time-series storage. Unlike the benchmark dataset used in this study, this deployment constitutes a novel, non-public industrial dataset and is intended as the primary empirical contribution of a

subsequent, extended study for which the present work serves as a methodologically validated foundation.

Declarations

ACKNOWLEDGEMENT

The author gratefully acknowledges the NASA Ames Prognostics Center of Excellence (PCoE) for developing and publicly maintaining the Prognostics Data Repository and the Berkeley Emergent Space Tensegrities (BEST) Laboratory, University of California, Berkeley, for the original acquisition and provision of the Milling Data Set [1] used in this study. Without the open availability of this dataset, the leakage-safe reproducibility analysis presented here would not have been possible. The author also acknowledges the maintainers of the open-source scientific Python ecosystem, including scikit-learn and SHAP, whose tools were used throughout the analysis pipeline. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

- **Ethics approval and consent to participate:** Not applicable. This study did not involve any human or animal participants.
- **Consent for Publication:** Not applicable.
- **Availability of Data and Material:** All relevant data and materials are provided in the supplementary documents.
- **Competing Interests:** The authors have no relevant financial or non-financial interests to disclose.
- **Funding:** The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

References

1. Agogino A, Goebel K, Milling data set, NASA Ames Prognostics Data Repository, NASA Ames Res. Center, Field M (2007) CA, USA, [Online]. Available: <https://www.nasa.gov/intelligent-systems-division/discovery-and-systems-health/pcoe/pcoe-data-set-repository/>
2. von Hahn T, Mechefske CK (2022) Machine learning in CNC machining: Best practices, Machines, vol. 10, no. 12, art. no. 1233. 10.3390/machines10121233
3. von Hahn T, Mechefske CK (2021) Self-supervised learning for tool wear monitoring with a disentangled-variational-autoencoder. Int J Hydromechatronics 4(1):69–98. 10.1504/IJHM.2021.114174
4. García-Ordás MT, Alegre E, González-Castro V, Alaiz-Rodríguez R (2017) A computer vision approach to analyze and classify tool wear level in milling processes using shape descriptors and machine learning techniques. Int J Adv Manuf Technol 90:5–8. 10.1007/s00170-016-9541-0

5. García-Ordás MT, Alegre-Gutiérrez E, Alaiz-Rodríguez R, González-Castro V (2018) Tool wear monitoring using an online, automatic, and low-cost system based on local texture. *Mech Syst Signal Process* 112:98–112. 10.1016/j.ymssp.2018.04.035
6. Pasandidehpour M, Nogueira AR, Mendes-Moreira J, Sousa R (2025) Survey on machine learning applied to CNC milling processes. *Adv Manuf* 13(4):847–885. 10.1007/s40436-025-00564-x
7. Breiman L (2001) Random forests, *Mach. Learn.*, vol. 45, no. 1, pp. 5–32. 10.1023/A:1010933404324
8. Lundberg SM, Lee S-I (2017) A unified approach to interpreting model predictions, in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, pp. 4765–4774
9. Pedregosa F et al (2011) Scikit-learn: Machine learning in Python. *J Mach Learn Res* 12:2825–2830
10. Kapoor S, Narayanan A (2023) Leakage and the reproducibility crisis in machine-learning-based science, *Patterns*, vol. 4, no. 9, art. no. 100804. 10.1016/j.patter.2023.100804
11. Karabacak YE (2023) Deep learning-based CNC milling tool wear stage estimation with multi-signal analysis, *Eksploracja i Niezawodność. – Maint Reliab* 25(3). 10.17531/ein/168082
12. Hirsch E, Friedrich C Data-driven tool wear prediction in milling, based on a process-integrated single-sensor approach, *arXiv:2412.19950*, 2024. [Online]. Available: <https://arxiv.org/abs/2412.19950>
13. Kamat P, Kumar S, Kotecha K (2024) DeepTool: A deep learning framework for tool wear onset detection and remaining useful life prediction, *MethodsX*, vol. 13, art. no. 102965. 10.1016/j.mex.2024.102965
14. Kurek J, Świdarska E, Szymanowski K (2024) Tool wear classification in chipboard milling processes using 1-D CNN and LSTM based on sequential features, *Appl. Sci.*, vol. 14, no. 11, art. no. 4730. 10.3390/app14114730

Figures

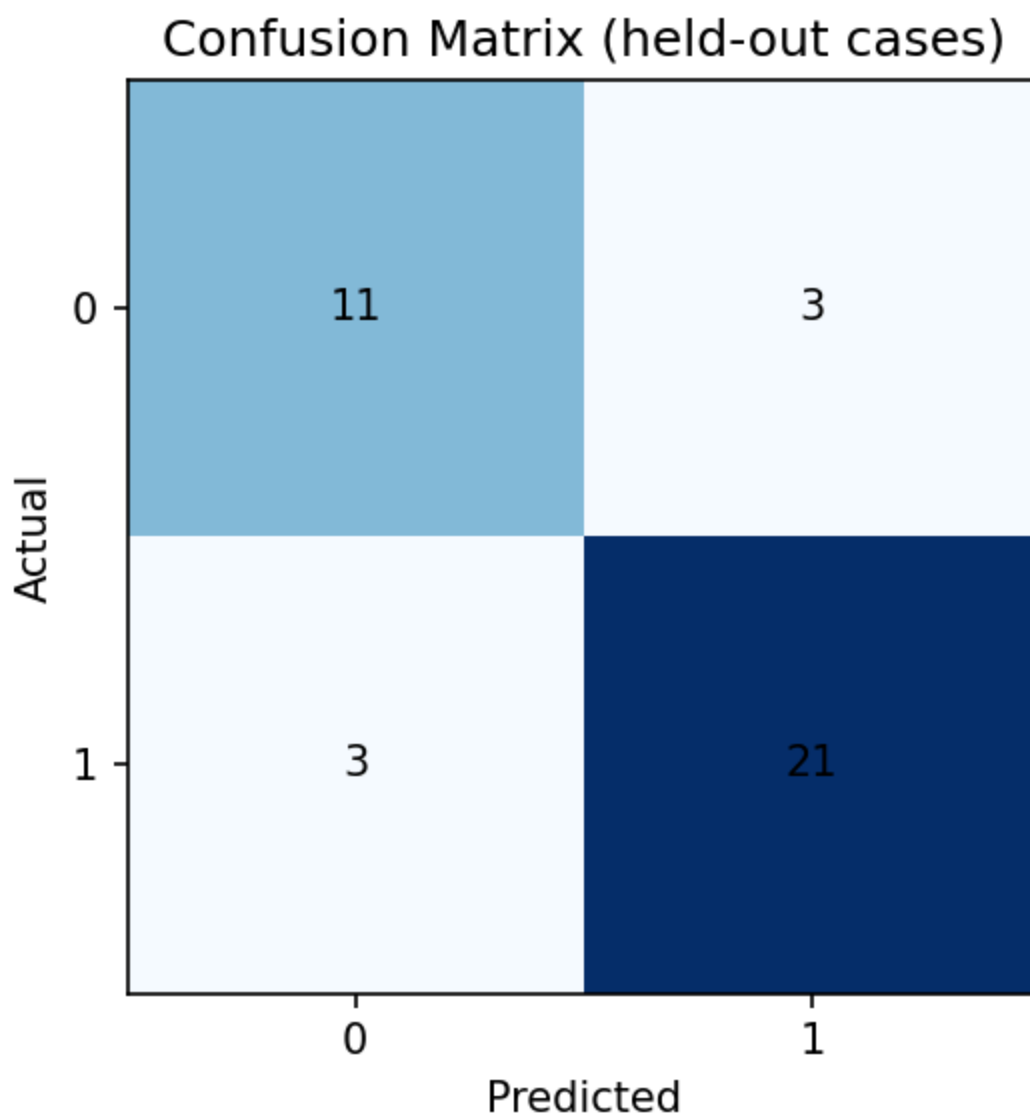


Figure 1

Random Forest confusion matrix on held-out test cases

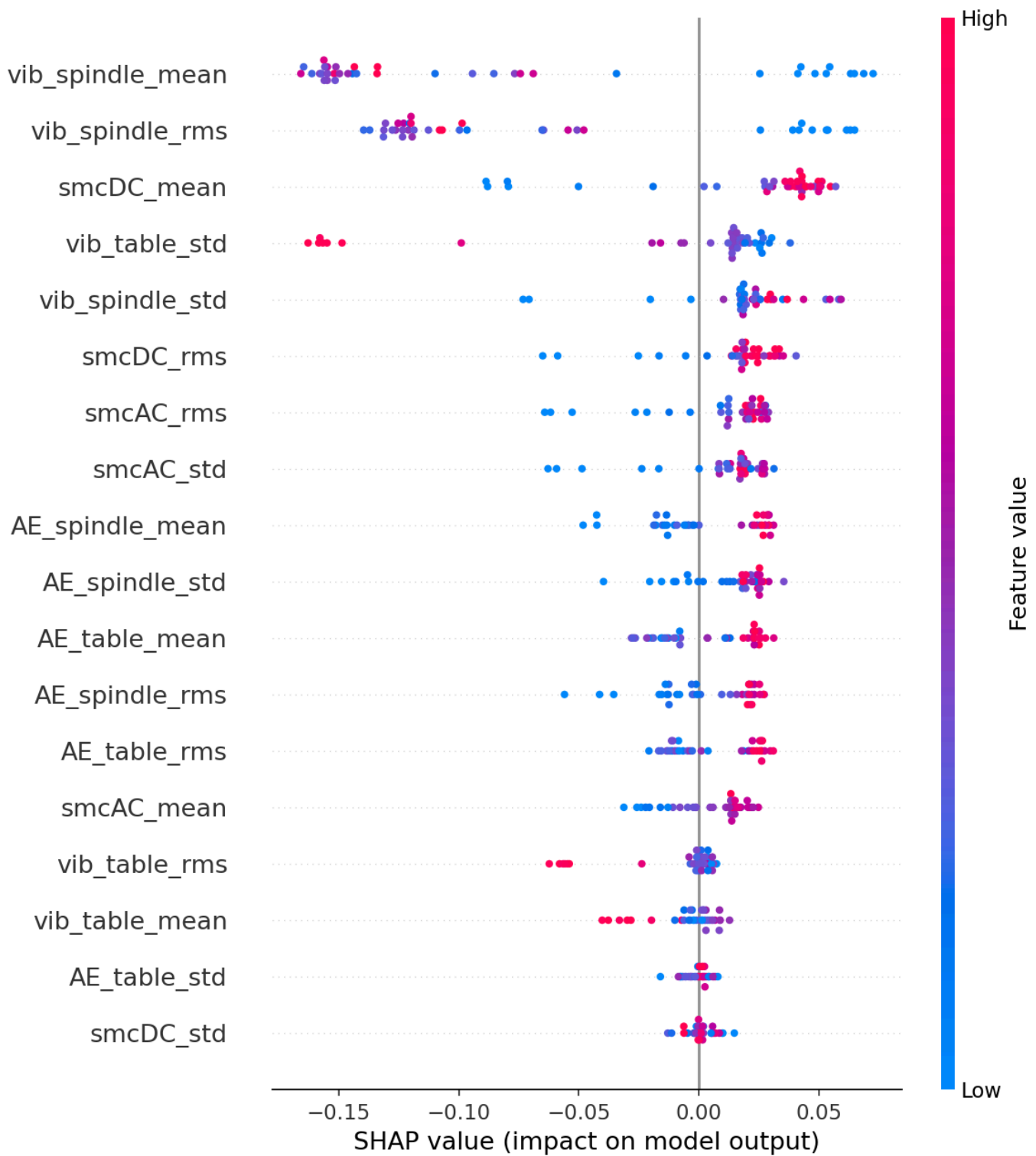


Figure 2

SHAP feature importance (Beeswarm plot) for the Random Forest model on held-out test cases.