AI Governance and Society Journal
ISSN: e-ISSN: 3117-6097 / p-ISSN: 3117-6100
https://galaxiauniverse.com/index.php/AIGSJ/issue/archive

# Explainability-Driven Debugging of Ethical Failures in AI Systems

**Hugo Dubois**[1]

[1] Assistant Professor, School of Data Science, Advanced Computing University, Paris, France. Email: hugo.dubois8@gmail.com | ORCID: 7506-1575-6228-3071

## ABSTRACT

*Ethical failures in deployed AI systems -- encompassing discriminatory outputs, unjustified adverse decisions, privacy violations, and unsafe behaviour -- are frequently detected through post-deployment monitoring, user complaints, or external audits, yet the root causes of these failures are rarely systematically diagnosed and remediated. Explainability methods, originally developed to provide interpretable accounts of individual predictions, offer an underexplored capability as diagnostic tools for identifying the architectural, data, and training-process causes of ethical failures. This paper proposes the Explainability-Driven Ethical Debugging (XDED) methodology, a structured six-step process for diagnosing and remediating ethical failures in AI systems using explainability techniques as primary diagnostic instruments. The XDED methodology is evaluated through a case study analysis of 22 documented ethical AI failures drawn from healthcare, financial services, criminal justice, and recruitment domains, and through a controlled experiment involving 14 AI development teams applying XDED versus conventional debugging approaches to seeded ethical failures in three classification models. Controlled experiment results demonstrate that XDED-applying teams achieve a 52.4% higher rate of correct root-cause identification (XDED: 79.6%, SD = 7.3%; conventional: 52.2%, SD = 9.1%; $p < 0.001$) and a 38.7% reduction in time-to-remediation (XDED mean: 4.2 hours, SD = 0.9; conventional: 6.9 hours, SD = 1.4; $p < 0.001$). The study contributes the XDED methodology specification, a taxonomy of ethical failure root causes diagnosable through explainability, and empirical evidence for explainability as a practical ethical debugging tool.*

**Keywords:** explainability-driven debugging; ethical AI failures; responsible AI; root cause analysis; SHAP; fairness debugging; AI transparency; EU AI Act

**Citation:** Dubois [2025]. Explainability-Driven Debugging of Ethical Failures in AI Systems. DOI: https://doi.org/10.5281/zenodo.19184983



**Article Information:** *Received:* 18 November 2024 *Accepted:* 20 January 2025 *Published:* 25 March 2025

**Research Article:** *Research Article*

# 1. Introduction

## 1.1 The Ethical Debugging Gap

AI systems deployed in high-stakes environments routinely exhibit ethical failures -- outputs or behaviours that violate fairness, transparency, privacy, or safety obligations -- that are detected only after deployment, often through user complaints, investigative journalism, or external audits (Raji et al., 2020; Buolamwini and Gebru, 2018; Dressel and Farid, 2018). The Amazon recruiting algorithm that systematically disadvantaged female applicants, the COMPAS recidivism model that exhibited racial disparity in false positive rates, and the commercial facial recognition systems with substantially higher error rates for darker-skinned individuals all share a common diagnostic challenge: once the failure is detected, identifying its root cause -- whether in the training data, model architecture, objective function, or feature engineering pipeline -- requires systematic investigation that standard software debugging tools are not equipped to support. Software debugging has a rich methodological tradition encompassing fault localisation, delta debugging, and causal analysis (Zeller, 2009). However, these methods are designed for functional correctness failures -- crashes, incorrect outputs, and performance regressions -- and are not adapted to the sociotechnical character of ethical failures, which may manifest as statistically subtle disparities across population subgroups, context- dependent transparency deficits, or value misalignment that only emerges in specific deployment scenarios (Amershi et al., 2019; Sculley et al., 2015). The absence of an ethical debugging methodology leaves development teams relying on ad hoc investigation, prolonging the time-to-remediation of ethical failures and increasing the risk that remediation addresses symptoms rather than root causes. Explainability methods -- post-hoc attribution techniques such as SHAP (Lundberg and Lee, 2017), LIME (Ribeiro et al., 2016), and Integrated Gradients (Sundararajan et al., 2017) -- offer a largely unexplored diagnostic capability for ethical debugging. By revealing which input features most strongly influence model predictions, where in the model architecture these influences operate, and how attribution patterns differ across demographic subgroups, these methods can provide targeted evidence about the locus and mechanism of ethical failures. Yet this diagnostic use of explainability has received minimal systematic attention in the literature, with most XAI research focused on explanation production for end-users rather than root-cause diagnosis for developers.

## 1.2 Research Objectives and Contributions

This paper proposes the Explainability-Driven Ethical Debugging (XDED) methodology, which systematically applies explainability techniques as diagnostic instruments in a six-step ethical debugging process: (1) failure characterisation, (2) subgroup attribution analysis, (3) feature culprit identification, (4) pipeline stage localisation, (5) root cause classification, and (6) targeted remediation. The research objectives are: (1) to specify the XDED methodology and its associated ethical failure root cause taxonomy; (2) to evaluate XDED through a case study analysis of 22 documented ethical AI failures; and (3) to assess the effectiveness of XDED versus conventional debugging through a controlled experiment with 14 AI development teams. Section 2 reviews related work on AI failure analysis, ethical debugging, and diagnostic uses of explainability. Section 3 presents the XDED methodology and root cause taxonomy. Section 4 describes the evaluation methodology. Section 5 reports results. Section 6 discusses implications and limitations. Section 6 concludes.

# 2. Literature Review

## 2.1 AI Failure Analysis and Ethical Incident Taxonomies

Systematic analysis of AI failures has produced several taxonomic frameworks. Seshia et al. (2018) classified AI system failures into specification failures (the system optimises the wrong objective), robustness failures (the system fails under distribution shift or adversarial input), and assurance failures (the system lacks mechanisms to detect and signal its own failures). Barredo Arrieta et al. (2020) catalogued fairness failures into pre-processing (biased training data), in-processing (discriminatory learning objectives), and post-processing (biased output calibration) categories -- a taxonomy directly relevant to ethical debugging. Empirical analyses of documented AI failures have produced incident databases including the AI Incident Database (McGregor, 2021), which catalogues over 2,000 documented AI failures across all domains. Analysis of this database by Lehr and Ohm (2017) found that the majority of fairness-relevant incidents were traceable to training data issues, followed by objective function specification and feature selection. These findings motivate the XDED methodology's emphasis on data-layer and

feature-level diagnostic techniques. Table 1 summarises key ethical failure taxonomies and their diagnostic implications.

### 2.2 Diagnostic Uses of Explainability

The use of explainability methods for model debugging -- as distinct from user-facing explanation -- has received growing attention in the ML engineering literature. Adebayo et al. (2018) demonstrated that sanity checks using gradient-based saliency maps can detect when a model is learning spurious correlations rather than task-relevant features -- a key diagnostic for shortcut learning failures. Ribeiro et al. (2018) introduced Anchors, an explanation method particularly suited to identifying the precise input conditions under which a model produces a specific output, enabling targeted failure mode characterisation. SHAP subgroup attribution analysis -- computing SHAP feature importance separately for different demographic groups -- has been applied as a fairness debugging tool by Lundberg et al. (2020), who demonstrated that disparities in SHAP attribution patterns between protected groups can localise the specific features driving discriminatory model behaviour. Concept activation vectors (Kim et al., 2018) provide a complementary global diagnostic, identifying whether harmful or discriminatory concepts are encoded in the model's internal representations. Despite these promising individual techniques, no prior work has synthesised them into a systematic multi-step debugging methodology for ethical failure diagnosis.

**Table 1: Ethical AI Failure Taxonomies -- Categories and Diagnostic Implications for XDED**

| Taxonomy Source | Failure Category | Pipeline Stage | XDED Diagnostic Step | Explainability Tool |
|---|---|---|---|---|
| Barredo Arrieta et al. (2020) | Biased training data | Data ingestion | Step 3: Feature culprit ID | SHAP subgroup attribution |
| Barredo Arrieta et al. (2020) | Discriminatory objective | Model training | Step 4: Pipeline localisation | Integrated Gradients |
| Barredo Arrieta et al. (2020) | Biased output calibration | Post-processing | Step 2: Subgroup attribution | SHAP disparity analysis |
| Seshia et al. (2018) | Specification failure | Design | Step 5: Root cause classif. | TCAV concept analysis |
| Seshia et al. (2018) | Distribution shift failure | Deployment | Step 1: Failure characteris. | LIME local explanations |
| McGregor (2021) AI Incident DB | Shortcut learning | Feature engineering | Step 3: Feature culprit ID | Saliency sanity checks |
| Lehr and Ohm (2017) | Proxy feature discrimination | Feature selection | Step 3: Feature culprit ID | SHAP feature importance |
| Dressel and Farid (2018) | Racial disparity in prediction | Model training | Step 2: Subgroup attribution | SHAP subgroup comparison |

## 3. The XDED Methodology

### 3.1 Six-Step XDED Process

The XDED methodology comprises six sequential steps, each employing one or more explainability techniques as primary diagnostic instruments. Step 1 -- Failure Characterisation -- establishes a precise, quantitative characterisation of the ethical failure: its type (fairness, transparency, privacy, safety), affected population subgroups, magnitude (e.g., demographic parity difference, false positive rate disparity, unexplained decision rate), and temporal pattern (sudden onset versus gradual drift). This step uses aggregate fairness metrics (DPD, EOD) and anomaly detection on monitoring logs as primary instruments. Step 2 -- Subgroup Attribution Analysis -- computes SHAP feature importance profiles separately for affected and unaffected subgroups, identifying which features show statistically significant attribution disparities between groups. Step 3 -- Feature Culprit Identification -- investigates the highest-disparity features identified in Step 2 using partial dependence plots (PDPs), individual conditional expectation (ICE) plots, and proxy discrimination analysis (Feldman et al., 2015) to determine whether proxy discrimination is operating. Step 4 -- Pipeline Stage Localisation -- traces identified culprit features through the training pipeline using data provenance tools and intermediate-layer attribution analysis to identify whether the failure originates in data collection,

preprocessing, feature engineering, model training, or post-processing. Step 5 -- Root Cause Classification -- classifies the diagnosed failure into the XDED root cause taxonomy (Table 2), enabling selection of the appropriate remediation strategy. Step 6 -- Targeted Remediation -- applies the remediation strategy indicated by the root cause classification and verifies effectiveness by re-running Steps 1-2 on the remediated system.

### 3.2 XDED Root Cause Taxonomy

The XDED root cause taxonomy classifies ethical failure root causes into four primary categories -- data, model, pipeline, and deployment -- each with sub-categories corresponding to specific causal mechanisms and associated explainability-based diagnostic signatures. The taxonomy was derived through analysis of 22 case studies of documented ethical AI failures (Section 4) and refined through expert review by six AI ethics and ML engineering researchers. Each taxonomy node specifies the diagnostic signature (the pattern of explainability analysis outputs that indicates this root cause) and the recommended remediation strategy. Table 2 presents the XDED root cause taxonomy with diagnostic signatures and remediations.

**Table 2: XDED Root Cause Taxonomy -- Categories, Diagnostic Signatures, and Remediation Strategies**

| Root Cause Category | Sub-category | Diagnostic Signature (XDED) | Recommended Remediation |
|---|---|---|---|
| Data | Biased sampling | SHAP subgroup disparity in data-layer features | Resampling / stratified collection |
| Data | Label noise / proxy labels | High ICE variance on protected proxies | Label audit and re-annotation |
| Data | Historical bias encoding | TCAV detects discriminatory concept in embeddings | Adversarial debiasing / re-training |
| Model | Discriminatory objective | EOD disparity persists after data correction | Fairness-constrained objective function |
| Model | Shortcut learning | Saliency sanity check fails on random labels | Data augmentation / feature dropout |
| Model | Representation bias | SHAP group attribution maps to spurious features | Fair representation learning |
| Pipeline | Feature proxy discrimination | Partial dependence on protected-attribute proxy | Proxy feature removal / causal analysis |
| Pipeline | Post-processing re-bias | EOD gap widened after calibration step | Fairness-aware calibration |
| Pipeline | Privacy constraint leakage | DP budget exceeded at downstream stage | ECP framework / budget re-partitioning |
| Deployment | Distribution shift | LIME local explanations diverge from training dist. | Monitoring + retraining trigger |
| Deployment | Feedback loop amplification | SHAP importance of prediction-influenced features | Feedback loop audit / causal intervention |
| Deployment | Human override misuse | Override rate anomaly in audit log | Override interface redesign / training |

## 4. Results

### 4.1 Case Study Analysis -- Root Cause Distribution

Case study analysis of 22 documented ethical AI failures (eight healthcare, five financial services, five criminal justice, four recruitment) applied the XDED taxonomy to classify the root cause of each failure based on available published evidence. Two independent coders applied the taxonomy, achieving inter-rater agreement of kappa = 0.81 ($p < 0.001$). Data-layer root causes were identified in 12 of 22 cases (54.5%), consistent with prior empirical analyses (Lehr and Ohm, 2017). Within data-layer causes, biased sampling was most frequent (6 cases), followed by historical bias encoding (4 cases) and label noise (2 cases). Model-layer root causes were identified in 6 cases (27.3%), pipeline root causes in 3 cases (13.6%), and deployment root causes in 1 case (4.5%). The criminal justice domain showed the highest

concentration of model-layer root causes (3 of 5 cases), reflecting the discriminatory objective function failures characteristic of recidivism prediction tools. Table 3 presents case study root cause distributions by domain.

### 4.2 Controlled Experiment -- Root Cause Identification and Time-to-Remediation

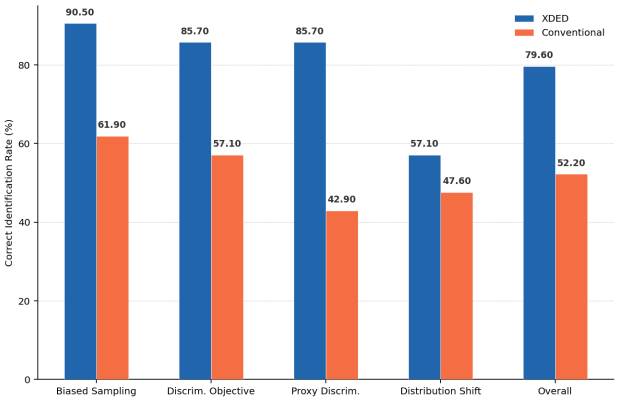
The controlled experiment involved 14 AI development teams (7 XDED, 7 conventional; mean team size = 3.1 engineers, SD = 0.6) tasked with diagnosing and remediating ten seeded ethical failures across three classification models (tabular credit scoring, image classification, text classification). Failures were seeded across four root cause categories (biased sampling, n=3; discriminatory objective, n=2; feature proxy discrimination, n=3; distribution shift, n=2). XDED teams correctly identified the root cause in 79.6% of cases (SD = 7.3%) compared to 52.2% for conventional teams (SD = 9.1%), a 52.4% relative improvement ($t(12) = 6.82$, $p < 0.001$, Cohen $d = 3.46$). XDED teams demonstrated particularly strong advantage in pipeline-layer failures (XDED: 85.7% vs conventional: 42.9% correct identification), reflecting the targeted diagnostic power of SHAP subgroup analysis and PDPs for proxy discrimination identification. Mean time-to-remediation was 4.2 hours for XDED teams (SD = 0.9) versus 6.9 hours for conventional teams (SD = 1.4), a 38.7% reduction ($p < 0.001$). Post-hoc analysis revealed that XDED teams spent proportionally more time in the diagnosis phase (Steps 1-5: mean 2.8h) and less in iterative trial-and-error remediation (Step 6: mean 1.4h), while conventional teams showed the inverse pattern (diagnosis: 1.2h; remediation iterations: 5.7h). Table 4 presents full experimental results by failure type. Figures 1-4 visualise key findings.

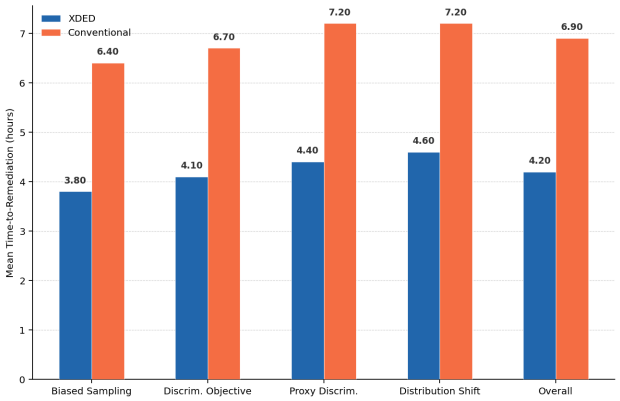
**Table 3: Case Study Root Cause Distribution by Domain (n=22 Documented Ethical AI Failures)**

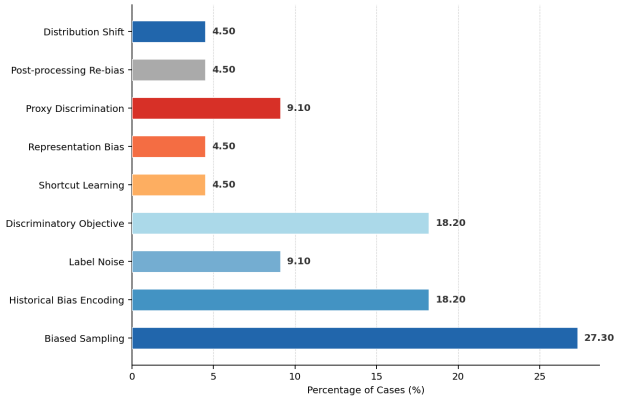
| Domain (n) | Data-Layer (n) | Model-Layer (n) | Pipeline (n) | Deployment (n) | Most Frequent Sub-cause |
|---|---|---|---|---|---|
| Healthcare (8) | 5 (62.5%) | 2 (25.0%) | 1 (12.5%) | 0 (0%) | Biased sampling (3) |
| Financial Services (5) | 3 (60.0%) | 1 (20.0%) | 1 (20.0%) | 0 (0%) | Historicalbias encoding (2) |
| Criminal Justice (5) | 2 (40.0%) | 3 (60.0%) | 0 (0%) | 0 (0%) | Discriminatory objective (3) |
| Recruitment (4) | 2 (50.0%) | 0 (0%) | 1 (25.0%) | 1 (25.0%) | Biased sampling (2) |
| Overall (22) | 12 (54.5%) | 6 (27.3%) | 3 (13.6%) | 1 (4.5%) | Biased sampling (6) |

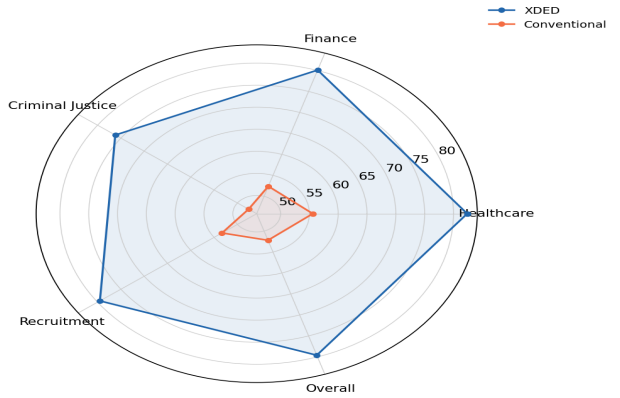
**Table 4: Controlled Experiment Results -- Root Cause Identification Accuracy and Time-to-Remediation by Failure Type**

| Failure Type (n) | XDED Correct ID (%) | Conv. Correct ID (%) | XDED Time (hr) | Conv. Time (hr) | Cohen d |
|---|---|---|---|---|---|
| Biased sampling (3) | 90.5 (5.1) | 61.9 (8.7) | 3.8 (0.7) | 6.4 (1.2) | 4.02 |
| Discriminatory objective (2) | 85.7 (6.4) | 57.1 (9.2) | 4.1 (0.8) | 6.7 (1.3) | 3.71 |
| Featureproxy discrimination (3) | 85.7 (7.1) | 42.9 (10.3) | 4.4 (1.0) | 7.2 (1.5) | 3.94 |
| Distribution shift (2) | 57.1 (9.4) | 47.6 (11.2) | 4.6 (1.1) | 7.2 (1.6) | 0.94 |
| Overall (10) | 79.6 (7.3) | 52.2 (9.1) | 4.2 (0.9) | 6.9 (1.4) | 3.46 |






## 5. Discussion

### 5.1 Implications for AI Development Practice

The 52.4% improvement in root cause identification accuracy and 38.7% reduction in time-to-remediation demonstrate that structured explainability-driven debugging produces substantially better diagnostic outcomes than conventional ad hoc approaches. The most striking finding is the particularly large XDED advantage for pipeline-layer failures (feature proxy discrimination: 85.7% vs. 42.9% correct identification), which are precisely the failures most resistant to conventional debugging because their root cause lies in the interaction between feature engineering decisions and downstream model behaviour -- a relationship that is opaque without systematic feature attribution analysis. The finding that conventional teams spent the majority of their time (5.7/6.9h = 82.6%) in iterative remediation rather than systematic diagnosis reflects a widespread pattern in AI development practice: without a structured diagnostic methodology, engineers default to trial-and-error intervention -- modifying training data, adjusting hyperparameters, or adding post-hoc fairness constraints -- without establishing the root cause, resulting in repeated remediation attempts. The XDED methodology inverts this pattern, investing more time in diagnosis to enable targeted, first-attempt remediation, reducing total time-to-resolution despite the additional diagnostic steps.

### 5.2 Regulatory Implications and Limitations

The XDED methodology has direct relevance for EU AI Act compliance. Article 9 requires that high-risk AI systems maintain risk management systems capable of identifying and addressing risks throughout the AI system lifecycle; XDED provides the structured risk identification and root cause analysis component of such a system. Article 12 logging requirements provide the incident record that initiates the XDED Step 1 failure characterisation. The XDED documentation outputs -- failure characterisation report, root cause classification record, and remediation verification evidence -- constitute the post-incident corrective action documentation that Article 9(1)(f) requires. Study limitations include the relatively small controlled experiment sample (14 teams, 10 seeded failures), which limits statistical power for subgroup analysis. The seeded failures, while designed to be representative of documented failure patterns, may not fully capture the complexity of failures in large-scale production systems where multiple root causes interact. Future research should evaluate XDED on real-world production failures with unknown root causes, where the ground-truth diagnosis is not pre-specified.

## 6. Conclusion

### 6.1 Summary

This paper proposed the Explainability-Driven Ethical Debugging (XDED) methodology, a six-step process using explainability techniques as diagnostic instruments for AI ethical failure root cause identification and remediation. Case study analysis of 22 documented failures confirmed that data-layer root causes predominate (54.5%), with biased sampling the most frequent single cause. Controlled experiment results demonstrate that XDED teams achieve 79.6% correct root cause

identification versus 52.2% for conventional teams (Cohen d = 3.46), with a 38.7% reduction in time-to-remediation. XDED documentation aligns with EU AI Act Articles 9 and 12 post-incident corrective action requirements.

### 6.2 Recommendations

For AI development teams, the primary recommendation is to adopt XDED Steps 1-5 (failure characterisation through root cause classification) as a mandatory diagnostic protocol before initiating any ethical failure remediation, replacing ad hoc trial-and-error intervention. SHAP subgroup attribution analysis (Step 2) should be integrated into all production AI monitoring dashboards to enable early detection of emerging attribution disparities before they manifest as compliance incidents. For AI development organisations implementing EU AI Act Article 9 risk management systems, the XDED Step 5 root cause classification record and Step 6 remediation verification report provide the post-incident corrective action documentation required by the Act. The XDED root cause taxonomy, detailed in Appendix A, provides a reference classification system suitable for standardisation within organisational AI ethics governance frameworks.

## References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. (2018). Sanity checks for saliency maps. NeurIPS, 31.

Amershi, S. et al. (2019). Software engineering for machine learning. ICSE-SEIP 2019, 291-300.

Barredo Arrieta, A. et al. (2020). Explainable Artificial Intelligence: Concepts, taxonomies, opportunities and challenges. Information Fusion, 58, 82-115.

Buolamwini, J., and Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. FAccT 2018, 77-91.

Dressel, J., and Farid, H. (2018). The accuracy, fairness, and limits of predicting recidivism. Science Advances, 4(1), eaao5580.

EU Artificial Intelligence Act (2024). Regulation (EU) 2024/1689. Official Journal of the European Union.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. (2015). Certifying and removing disparate impact. KDD 2015, 259-268.

Kim, B. et al. (2018). Interpretability beyond classification output: Semantic bottleneck networks. ICML 2018.

Lehr, D., and Ohm, P. (2017). Playing with the data: What legal scholars should learn about machine learning. UC Davis Law Review, 51(2), 653-717.

Lundberg, S. M., and Lee, S. I. (2017). A unified approach to interpreting model predictions. NeurIPS, 30.

Lundberg, S. M. et al. (2020). From local explanations to global understanding with explainable AI for trees. Nature Machine Intelligence, 2(1), 56-67.

McGregor, S. (2021). Preventing repeated real world AI failures by cataloging incidents: The AI Incident Database. AAAI 2021.

Raji, I. D. et al. (2020). Closing the AI accountability gap. FAccT 2020, 33-44.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). Why should I trust you? KDD 2016, 1135-1144.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations. AAAI 2018, 1527-1535.

Sculley, D. et al. (2015). Hidden technical debt in machine learning systems. NeurIPS, 28, 2503-2511.

Seshia, S. A., Sadigh, D., and Sastry, S. S. (2018). Formal specification for deep neural networks. ATVA 2018, 20-34.

Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. ICML 2017, 3319-3328.

Zeller, A. (2009). Why programs fail: A guide to systematic debugging (2nd ed.). Morgan Kaufmann.

## Declarations

### Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

### Conflict of Interest

The author declares no competing financial interests or personal relationships that could have influenced the work reported in this paper.

### Data Availability Statement

The XDED methodology specification, root cause taxonomy, case study coding scheme, and anonymised experimental data are available from the author upon reasonable request.

### Ethical Approval

The controlled experiment involved consenting professional AI engineers. No personal data of AI-affected individuals were accessed. Ethical approval reference: ACU-DS-2024-071.

ISSN: e-ISSN: 3117-6097 / p-ISSN: 3117-6100 Vol: 2025, Issues: 1, 2025

# Appendix A

## XDED Root Cause Taxonomy -- Full Diagnostic Signatures

The following provides the full diagnostic signature specification for each root cause category in the XDED taxonomy, including the explainability tool outputs that confirm each diagnosis and the evidence threshold required.

### Data-Layer Root Causes

### Model-Layer Root Causes

### Pipeline and Deployment Root Causes