Irving Fisher Committee on
Central Bank Statistics

BIS

12th biennial IFC Conference: "Statistics and beyond: new data for decision making in central banks"

22-23 August 2024

A scalable, explainable machine learning approach for granular-level credit dataset's quality assurance[1]

Anak Yodpinyanee, Peranut Nimitsurachat,
Nontawit Cheewaruangroj and Supachai Saengthong,
Bank of Thailand

[1] This contribution was prepared for the conference. The views expressed in this publication are those of the authors and do not necessarily represent the official views of the Committee, its members, or the BIS.

A Scalable, Explainable Machine Learning Approach for Granular-Level Credit Dataset's Quality Assurance

Anak Yodpinyanee, Peranut Nimitsurachat, Nontawit Cheewaruangroj, Supachai Saengthong[1]

To enhance ongoing supervision, Bank of Thailand initiated the Regulatory Data Transformation (RDT) project, establishing a new granular-level regulatory reporting standard for credit data. Data quality assurance is crucial for supporting policymakers with reliable information. Testing rules, such as format/range validation and consistency check across related reports, are manually created and applied to ensure fulfillment of basic requirements. These rules, however, fail to capture unknown sophisticated errors in multivariate relationships. Thus, we evaluated the efficiency and interpretability of state-of-the-art outlier detection models including LODA (Lightweight On-line Detector of Anomalies), Isolation Forest with DIFFI (Depth-based Isolation Forest Feature Importance) and AWS (Assist-Based Weighting Scheme), and ECOD (Empirical-Cumulative-distribution-based Outlier Detection). We place strong emphasis on model's explainability as potential errors must be reviewed and communicated back to data providers. Scaling capability is also essential because RDT requires detailed fields with over 25 million records per month. Introducing the concept of random projections prior to ECOD leads to the best performance in overall detection recalls and, for explainability, the top-3 explained features cover more than 70% of the anomaly interpretation as reported by analysts. This top-performing model has been tested for scalability and will be implemented to production environments.

Keywords: microdata, anomaly detection, ECOD, LODA, isolation forest, feature importance, explainable machine learning

JEL classification: C14, C52, C55, C81

[1] The opinions expressed in this discussion paper are those of the authors and should not be attributed to the Bank of Thailand.

Contents

1. Introduction and Motivation

Bank of Thailand processes a large volume of data, from financial institution data to economic indicators. One of the most complex datasets comes from the Regulatory Data Transformation (RDT) program, which is a new granular-level regulatory reporting standard for credit data. Numerous records of data (25 million records per month) are sent from financial institutions regularly. These records need to satisfy the criteria set by Bank of Thailand. Validation rules are applied to ensure compliance with the requirements. However, these rules fail to capture unknown sophisticated errors in multivariate relationships. Thus, machine learning methods are employed to automatically and flexibly detect these anomalies.

Anomaly detection is a well-studied subject and is extensively used across many business domains to handle fraud and data quality issues. Existing methods for anomaly detection vary in their approaches and capabilities. For instance, K-Nearest Neighbours (KNN) (Ramaswamy, Rastogi, & Shim, 2000), Local Outlier Factor (LOF) (Kröger, Ng, & Sander, 2000), Isolation Forest (Liu, Ting, & Zhou, 2008), Half-Space Trees (Tan, Ting, & Liu, 2011), Histogram-Based Outlier Score (HBOS) (Goldstein & Dengel, 2012), Lightweight on-line detector of anomalies (LODA) (Pevný, 2016), xStream (Manzoor, Lamba, & Akoglu, 2018) and Empirical Cumulative Distribution-based Outlier Detection (ECOD) (Li, et al., 2023) are some of the notable algorithms employed in this domain. These techniques are unsupervised, not requiring any labels to train the models. While some of these methods, like Isolation Forest and Half-Space Trees, are known for their scalability and efficiency in handling large datasets, others such as ECOD and LODA offer greater explainability, which is crucial for understanding the reasoning behind the detected anomalies. Additionally, certain methods such as KNN, Isolation Forest and LODA are adept at dealing with multivariate outliers, which is essential for comprehensive analysis in real-world applications. The abilities to scale well, provide explainability, and handle multivariate data are important properties that determine the practical utility of anomaly detection methods in real scenarios. These properties ensure that the methods are not only theoretically sound but also applicable and effective in practical financial contexts.

Many central banks have employed both of supervised and unsupervised anomaly detection methods to detect fraud or abnormalities in their economics dataset or reports from financial institutions. Bank of Canada, for example, proposed a flexible machine learning framework that monitor the transaction data in high-value payment system (HVPS) (Desai, Kosse, & Sharples, 2024). This approach consists of two separate layers: Payments Classifier and Anomaly Detection on the Subset of Misclassified Payments. In the first layer, supervised machine learning models, including Logistic Regression, Random Forest, and Gradient Boosting, are trained to predict a target variable that indicates whether a payment is a morning or afternoon payment. The payment whose model prediction does not match the actual prediction time (e.g. a payment that is predicted to be a morning one but submitted in the afternoon) is flagged as an anomaly. These flagged unusual payments are passed on to the second layer where unsupervised techniques are used to detect anomalies among these unusual payments. The efficacy of this system was evaluated using an artificially manipulated dataset. The authors report impressive results, with the first layer achieving a 93% detection rate. In the second layer, transactions flagged as suspicious received outlier scores nearly double those of the original transactions.

These findings suggest that the two-layer approach is highly effective in identifying potential anomalies in HVPS transactions.

Another study by Bank of Italy stacked supervised and unsupervised machine learning models to detect anomalies in granular AnaCredit datasets reported by banks (Continanza, et al., 2022). First, they use robust regression and autoencoder with other related datasets to enrich the data with the divergence between the series. The outputs are then used as features to train supervised learning models including logistic regression, k-nearest neighbour, random forest, and support vector machine. The labels are artificially simulated using cases reported by their analysts and from banks.

These studies are similar to our research since both approaches aim to detect anomalies in large datasets where no labels indicating the anomaly cases exist, making it challenging to rely solely on supervised algorithms. Since anomalies are rare by nature, synthetic data is widely used to provide labels for the machine learning algorithms. However, using synthetic data to train the model may not guarantee its effectiveness in real-world scenarios. To tackle this challenge, our study focuses on utilizing unsupervised anomaly detection models. We manually select five common anomalous themes identified by analysts in the dataset and use them as a test set instead of artificially manipulated data.

For the interpretability of the anomalies, some studies use surrogative methods such as Local Interpretable Model-agnostic Explanations (LIME) (Accornero & Boscariol, 2021) and Shapley Additive explanations (SHAP) (Desai, Kosse, & Sharples, 2024) to explain the anomalies predicted from the black-box models. These methods are effective but may not scale well with large datasets. Hence, we aim to investigate anomaly detection methods that are interpretable with scalability suitable for large datasets using real dataset with real labels.

In our study, we select anomaly detection methods that have the following three properties: explainability, scalability and multivariate. Isolation Forest, Lightweight on-line detector of anomalies (LODA) and Empirical Cumulative Distribution-based Outlier Detection (ECOD) are among the state-of-the-art algorithms that can satisfy the required criteria. The ECOD algorithm is originally designed to capture only univariate outliers. We apply random projections to the dataset as in LODA to integrate the multivariate relationship of the anomalies into new features prior to applying ECOD. For the Isolation Forest algorithm, we employ two methods to calculate feature importance scores of flagged anomalies: Depth-based Isolation Forest Feature Importance (DIFFI) (Carletti, Terzi, & Susto, 2023) and Assist-Based Weighting Scheme (AWS) (Vercruyssen, Gautrais, & Sobha Kartha, 2021). In the case of LODA and ECOD, we perform hypothesis testing on the negative log-likelihood metric to determine the feature importance scores for each feature. We apply these algorithms to the granular RDT Credit data and evaluate the anomaly detection and feature explanations with the cases identified by analysts.

The paper is organized as follows. The rest of Section 1 gives an overview of the RDT Credit data, the anomaly detection methods applied in this study and the evaluation approach. Section 2 presents the data description and preparation, including description of the selected anomalous cases as well as the implementation of the anomaly detection models. Section 3 evaluates the results of the methods, emphasizing the performance of anomaly detection on real cases. Section 4 discusses the interpretability of the methods, comparing the feature importances from the model with the real cases explanation. Section 5 summarizes the study and proposes future work.

Background Knowledge

RDT Credit data

The Bank of Thailand launched the Regulatory Data Transformation (RDT) project in 2020 to improve the financial data reporting standards of financial institutions for supervisory and policy-making purposes. The project was a transformative redesign over the existing Data Management System (DMS), which largely consisted of fixed reports, limiting the Bank of Thailand from thoroughly analyzing data from different angles. The RDT project intended to overcome these challenges by introducing a new granular-level regulatory reporting standard for credit data, allowing more detailed analysis and comparison of contracts across various dimensions, such as underwriting standards for different borrower groups. The credit data in the RDT also includes detailed information such as credit lines, interest rate plans, monthly outstanding amounts, and collaterals.

Given its unprecedented level of granularity and scope, the RDT Credit dataset opens up a trove of new possibilities for high-impact policy applications, notably in the areas of supervision, financial stability, and monetary policy. For example, with borrowers' income data along with profiles and loan amounts, the dataset allows for micro-level analysis of each borrower's debt serviceability and possible contagion effects. In addition, the information on loan-level interest plans allows economists to measure the pass-through of monetary policy that could be hidden in the contractual terms and details.

To ensure the quality and reliability of the data collected, validation process is a vital component of the RDT project. The validation process involves applying a series of checks and rules to the data to verify its accuracy and consistency. This includes format and range validation, and consistency checks across related reports[2]. However, these validation rules might fail to capture errors that are more sophisticated or multivariate in nature, thus a need for anomaly detection algorithms to serve as the second line of defense.

Anomaly Detection Algorithms

We consider three state-of-the-art anomaly detection algorithms based on their explainability, scalability, and multivariate prediction. For more details of the algorithms and calculations of the anomaly scores and feature importance, see Appendix.

Isolation Forest

Isolation Forest algorithm separates samples by creating isolation trees that randomly choose features and split values. It assesses normality based on the average path length from the root node to a leaf node in a forest of random trees. Samples that have shorter paths are marked as anomalies (Liu, Ting, & Zhou, 2008). Isolation Forest is also known to be robust especially in high-dimensional datasets as well as having great scalability compared to other approach such as KNN. For interpretability, Depth-based Isolation Forest Feature Importance (DIFFI) can be used in find the feature importance for individual predictions by analyzing the path of a data point through the Isolation Trees and distributing the importance measure, which is roughly

[2] See www.bot.or.th for details on RDT Project and RDT Credit data.

inversely proportional to path lengths, to all contributing features (Carletti, Terzi, & Susto, 2023). Another method is Assist-Based Weighting Scheme (AWS), in which each split awards the associated feature a score based on its contribution in shortening the path length, proxied by the split's imbalance (Vercruyssen, Gautrais, & Sobha Kartha, 2021).

Lightweight On-line Detector of Anomalies (LODA)

LODA uses multiple sparse random projections and builds one-dimensional histograms for each projection. These histograms form an ensemble, where each histogram rates samples as anomalies based on the relative density of their respective bins. The sparse random projections enable LODA to find outliers in multiple dimensions, while being light weighted. The anomalies can also be explained via statistical tests, namely *t* test, to compare the average anomaly score contributions from the projections with and without each feature, allowing us to calculate feature importance with *t* score (Pevný, 2016).

Empirical Cumulative distribution-based Outlier Detection (ECOD)

ECOD is a novel anomaly detection model that addresses multiple challenges in current frameworks such as curse of dimensionality, limited interpretability of model, and time-consuming hyperparameter tuning. ECOD estimates the underlying distribution of each feature in the dataset as an Empirical Cumulative Distribution Function (ECDF), which is then used to calculate the probability of observing a data point as extreme (in the tails) for that dimension. Assuming independence among features, ECOD aggregates these tails probabilities in three ways—left-only, right-only, and skewness-based—and assign the maximum of these values as the data point's outlier score (Li, et al., 2023). To account for multivariate relationship among variables, we apply Sparse Random Projection to the data before ECOD in the same fashion as LODA. ECOD's Feature importance is computed via hypothesis testing similarly to LODA.

Model Evaluation

Recall measures the proportion of the actual anomalies that were correctly identified by the model. It is an essential metric for evaluating the performance of anomaly detection systems. Mathematically, recall is defined as:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \tag{1}$$

where True Positives (TP) represent the number of correctly identified anomalies and False Negatives (FN) represent the number of actual anomalies that were missed by the model. Recall values range from 0 to 1, where a recall of 1 means the algorithm successfully identifies all the labelled anomalies in the system.

Mean Reciprocal Rank (MRR) is a ranking quality metric commonly used to evaluate recommendation and information retrieval systems. It provides insight into how quickly a ranking system can present the first relevant item in the top-*k* results. In our study, we use MRR to evaluate feature importance of the anomalies for the explainability of the models. The mean reciprocal rank can be calculated as the average of the reciprocal ranks of results:

$$\text{MRR} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\text{rank}_i}, \tag{2}$$

where N is the number of records in the evaluated dataset and rank_i is the position of the first relevant item for record i in the top-k results. MRR values range from 0 to 1, where 1 indicates that the first relevant item is always at the top. Higher MRR signifies better performance.

2. Methodology

Dataset Preparation

In this experiment, we construct our dataset focusing on five cases of multi-variable anomalies occurring in the RDT Credit data identified by analysts, including inconsistencies between loan stages and their respective probability of default (PD) and days past due (DPD), large loans granted to small firms, overutilization, and overpledging[3]. As we aim to evaluate the performance of anomaly detection algorithms on actual data, we leave our data points as intact as possible[4] rather than artificially introducing errors or synthesizing data points. More specifically, we carefully subsample from the actual RDT Credit data, thereby preserving multi-variate relationship between features.

We create our dataset with a mild 5% contamination rate contributed by the five cases of anomalies (with overlap), specified in Table 1. While our approach is simply to downsample the data points without any anomaly conditions, we take precautions to ensure that our anomalies are reasonably discoverable by detection algorithms. Observe that the anomaly criteria impose thresholds on numerical variables—these cut-offs are motivated by business but may hardly be present in the (single-variate) distributions of these variables. Hence, without knowing the fraction of errors a priori, detection algorithms can probably still identify some anomaly cases, but performing actual classification with good accuracy would be unlikely. To remedy this problem, we introduce a normality condition for each anomaly counterpart, creating a gap within the valid range of their respective numerical feature. We exclude any data points that match a scope condition but falls into the respective gap. This solution leaves a visible gap in the conditional marginal distribution for each case, allowing detection algorithms to discover these thresholds.

Our dataset is subsampled from the RDT Credit dataset from 2024Q1 prepared for each loan account in monthly frequency. Under the described procedure, we reduce 71.2M total data points into a dataset with 13.4M data points containing 5% total anomalies among five cases. We apply standard feature engineering techniques to 17 selected features, resulting in the final dataset with 13 normalized numerical features, 2 normalized ordinal features, and 63 one-hot encoding Boolean features. (See Table 3 in the Appendix for descriptions of the selected features.) Finally, 2.5M data points are sampled for our experiment.

---

[3] Overpledging refers to the case where the pledged amount exceeds the collateral valuation amount.

[4] Aside from normalization and encoding, we add utilization ratio as a feature, so that Case 4 (overutilization) also involves exactly two features, to ensure comparability of feature importance evaluation among all cases.

Five selected anomaly cases in the constructed dataset Table 1

Case	Scope Condition	Anomaly Condition	Normality Condition	Fraction
1	Stage = Non-Performing	PD ≤ 50%	PD ≥ 80%	1.770%
2(a)	Stage = Performing	DPD ≥ 40	DPD ≤ 30	0.967%
2(b)	Stage = Under-Performing	DPD ≥ 100	DPD ≤ 90	0.411%
3	Firm Size = Small or Micro	Credit Line ≥ 300M THB	Credit Line ≤ 100M THB	0.208%
4	Revolving Flag = False	Utilization Ratio ≥ 1.5	Utilization Ratio ≤ 1.2	0.520%
5	Any	Pledge / Valuation ≥ 2	Pledge / Valuation ≤ 1	1.136%

Anomaly Detection

We consider three anomaly detection algorithms that offer explainability, scalability and multivariate prediction.

- **Isolation Forest.** We apply Isolation Forest with 1,024 trees, each of which has access to all 78 features and 1,024 random samples. Null entries for normalized and one-hot features are replaced by centers 0 and 0.5, respectively.
- **LODA.** We apply LODA using a random projection with 1,024 target dimensions. Null entries are replaced by 0 to remove their contribution to the anomaly score.
- **ECOD.** As ECOD inherently scores tail bounds of individual variables, we introduce multivariate relationship by applying scikit-learn's SparseRandomProjection with 1,024 target dimensions and replace null entries by 0, similarly to LODA.

As our dataset is essentially a subsample of the actual RDT Credit data with five anomaly cases amplified, other types of anomalies are likely still present and should be classified as anomalies along with the desired cases. So, for benchmarking, we let each algorithm classify up to 10% of the data points as anomalies and choose the recall rate as our evaluation metric.

Feature Importance

Based on the algorithm employed, we apply different feature importance technique(s) to score and rank features contributing to the anomaly score. We compute feature importance scores for all 78 post-engineered features and assign the importances of the 17 selected (original) features from the maximum score among those of their respective derived features.

- **Isolation Forest.** For each sample, we traverse the classification path of each tree, and employ both DIFFI and AWS methods for accounting and combining anomaly scores of features occurring on the decision nodes. Results of these two methods are evaluated separately.
- **Projection-Based Methods (LODA & ECOD).** For each sample, we apply the one-tailed two-sample t test to compare the anomaly scores (negative log-likelihood) between projected dimensions that use a feature and those that do not, to rank how much each feature contributes to the overall anomaly score.

Because we only know for certain the anomalous features for the five selected cases, we evaluate feature importance results only on these samples (among the 10% selected by each algorithm). As we expect analysts to investigate up to five proposed

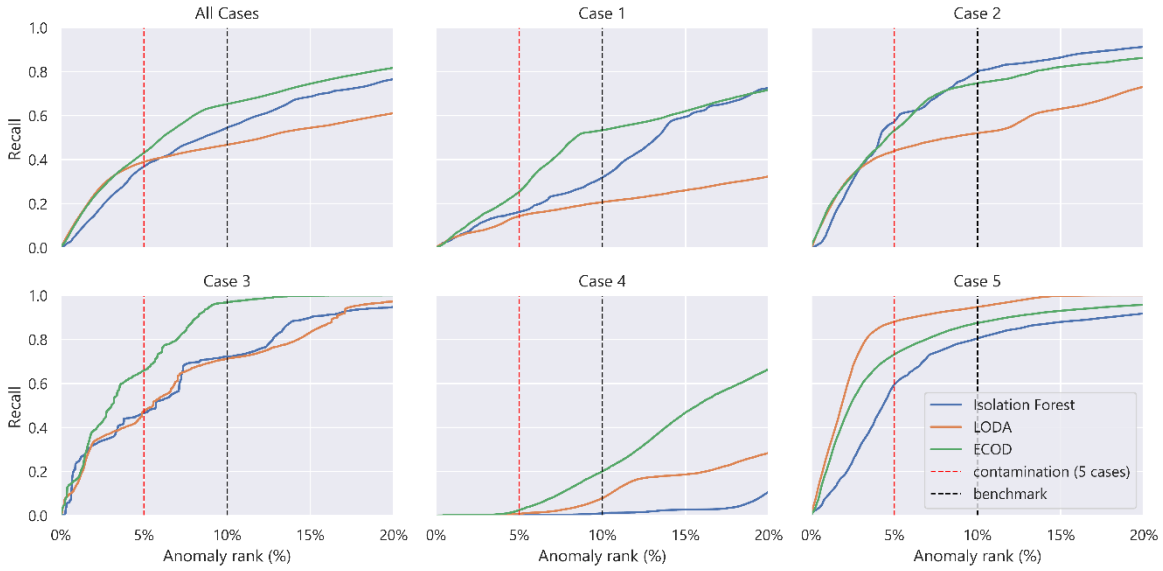
features in the actual application of RDT Credit data quality assurance, in lieu of recommendation metrics, we determine the rate of occurrence of an anomalous feature among the top-*k* ranked features for *k* up to 5, as well as the (truncated) mean reciprocal rank (MRR). Observe from Table 1 that, by design, every case involves exactly two (original) features: the higher rank of the two features is chosen for calculation in our metrics.

3. Anomaly Detection

Overall Performance

Consider the total recall rates for all five cases (combined), shown in the first figure of Figure 1. When algorithms are allowed to report up to 5% of the dataset as anomalies, LODA and ECOD exhibit comparable recall rates and outperform Isolation Forest. LODA's performance, however, drops below that of Isolation Forest after 6%, with no further changes in ranking even after the 10% benchmark rate. While ECOD excels for general purposes, Isolation Forest and LODA perform better at our benchmark on Case 2 and Case 5, respectively. However, Case 4 remains a challenge for all methods.

Recall Rates for Anomaly Detection — Figure 1

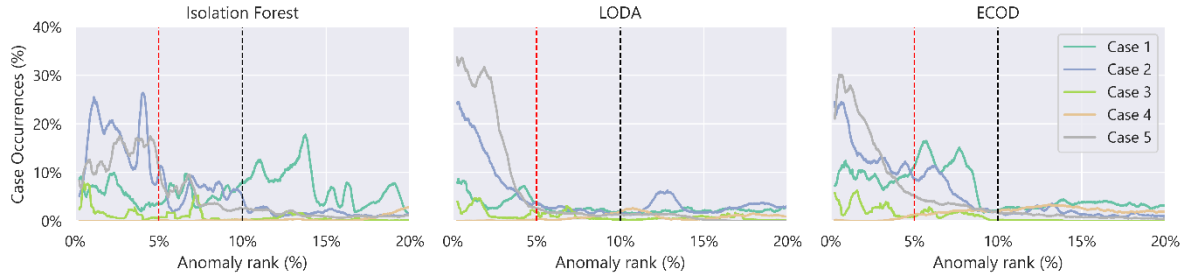


This chart shows the recall rates of anomaly detection algorithms, for all five cases and for each individual case. The x-axis represents the fraction of total data points reported, and the y-axis represents the fraction of anomalous samples uncovered by the reported samples.

Comparison of Detected Cases

To examine how each algorithm scores the data points, we calculate the fraction of reported samples belonging each case as the assigned anomaly rank varies, shown in Figure 2. Observe that the matched cases of Isolation Forest keep changing rapidly, whereas LODA exhibits quite a monotonic trend that turns flat at roughly 5%. ECOD seems to provide a mixture of the previous two.

Fraction of Reported Samples belonging to Anomaly Cases Figure 2



This chart shows the breakdown of reported samples into anomaly cases, at moving window of width 12,500 samples (0.5% of the total dataset). The y-axis shows the fraction of samples in the window that also belong to each anomaly case.

- **Isolation Forest.** The frequent peaks indicate that anomalous samples of the same cases are discovered in small batches, reflecting the partitioning behavior of this tree-based method. Among these algorithms, Isolation Forest is capable of recognizing conjunctive conditions on categorical values or specific ranges of numerical values, similar to those in Table 1. This property allows Isolation Forest to outshine other methods on Case 2 (which contains two subcases) but struggle on Case 5 (where the involved variables are linearly-related numerical variables). Further, in our setting, it fails to uncover the planted cut-offs and to assign a correct anomaly ranking, such as in Cases 1 and 4, where some peaks emerge well after our 10% benchmark. We note that the practice of coarsening tree splits may exacerbate performance degradation, leading to a potential neglect of low-frequency cases. In particular, we test an Isolation Forest with depth 8 (256 samples per tree) in the same setup: this hyperparameter places higher priority on Case 1 while hurting the recall for the sparser Cases 2 and 3.
- **LODA.** LODA uses histograms as density estimators on linear projections. Unlike other methods that rely on rankings/percentiles of feature values or projections thereof, LODA leverages the distributions of their actual values. This aspect resonates with the linear relationship between Pledge Amount and Valuation Amount in Case 5, where LODA impressively recalls 94.8% of the anomalies at our 10% benchmark. On the other hand, LODA's performance drops as the anomalous variables become more discrete: the recall rates are lower in Cases 1-3 when involving ordinal variables (Stage and Firm Size), and lowest for Case 4 with Boolean flag (Revolving).
- **ECOD.** Overall, our implementation of ECOD performs comparatively well for this dataset. Due to the random projections of the dataset, our ECOD implementation is very similar to LODA: the difference stems from the scoring function that relies on tail bounds instead of density estimations. For this dataset, the distribution of numerical variables are mostly well-behaved, with largely continuous densities and few modes, rendering density estimation less crucial. Further, conditions of the five selected cases are inherently inequalities, which can be conveniently scored via tail bounds. The random projections allows conjunction of inequalities to be approximately represented in certain resulting dimensions, preserving the expressivity of Isolation Forest when a small number of variables are involved in the anomaly conditions. The tail bounds from ECDFs also aggregate into a more continuous scoring function without explicitly partitioning the sample. These speculations are apparent in Figure 2; ECOD's detection rates exhibit peaks (representing conjunctions) similar to Isolation Forest's albeit smoother, while

continuing to output anomalies well after 5% (due to its robust ECDF-based scoring function) unlike LODA's. Hence, ECOD with random projection brings together the good aspects of LODA and Isolation Forest that allow it to excel, for this sampled dataset.

A Note on Case 4

Upon closer inspection, we realize that Case 4's normality condition generally holds when its scope condition fails[5]; that is, the utilization ratio alone almost suffices for identifying the anomaly. The three detection algorithms here tend to prioritize data points with anomalies co-occurring in multiple features, compounding their anomaly scores. The ratio between revolving and non-revolving loans in our data points are roughly 6:4, so Revolving Flag hardly has an impact on the resulting scores, resulting in Case 4's anomalous data points being reported after the 10% benchmark.

4. Feature Importance

Mean Reciprocal Rank (MRR)

We measure the performance of feature importance methods from the true positive anomalies: samples ranked within the top 10% by each algorithm that also belong to some selected cases. Consider the mean reciprocal rank (MRR) shown in Table 2, based on the higher rank among the two features involved in each case. Similar to the anomaly detection step, applying *t* test on the anomalies from ECOD with random projection achieves competitive or higher MRRs. Our explainability results surpass the benchmark MRR of 0.249, namely the expected MRR of a random feature ranking, by at least 80% on all selected cases.

Mean Reciprocal Ranks and Frequent Higher-Ranked Anomalous Features — Table 2

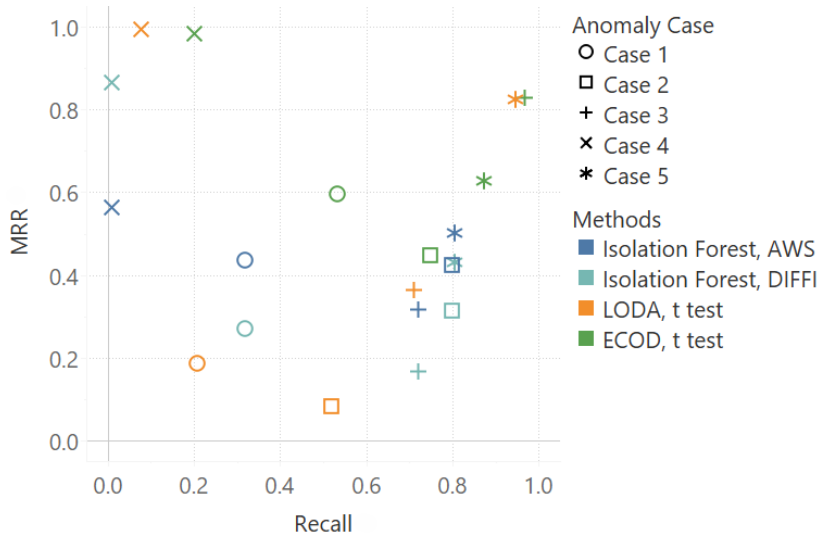
Case	Isolation Forest		LODA	ECOD
	AWS	**DIFFI**	***t* test**	
1	0.436 Stage (100.0%)	0.271 Stage (94.9%)	0.188 Stage (91.7%)	**0.597** Stage (99.7%)
2	0.425 DPD (98.3%)	0.313 DPD (97.4%)	0.082 DPD (85.0%)	**0.448** DPD (92.6%)
3	0.315 Credit Line (100.0%)	0.166 Credit Line (100.0%)	0.363 Credit Line (100.0%)	**0.827** Credit Line (100.0%)
4	0.564 Utilization Ratio (100.0%)	0.865 Utilization Ratio (100.0%)	**0.993** Utilization Ratio (100.0%)	0.984 Utilization Ratio (100.0%)
5	0.502 Valuation (66.1%)	0.430 Pledge (52.7%)	**0.825** Valuation (92.4%)	0.626 Valuation (89.3%)

For each cell, MRR (truncated at k = 5 features) is reported in the first line. The second line contains the feature that appears at a higher rank more frequently (between the two involved in each anomaly case); the respective frequency is given in parenthesis.

[5] Utilization Ratio > 1 is not unusual for revolving loans (hence Case 4 is chosen), but this scenario seldom occurs in our dataset. We are investigating for business explanation or reporting errors.

We further observe that the relative orderings of recalls and MRRs are similar for each case, as shown in Figure 3. For instance, LODA significantly outperforms ECOD only in Case 5, and so do their *t* tests. Hence, the superior feature importance of LODA is likely attributed to its high accuracy in the anomaly detection step.

Scatter Plot between Anomaly Detection Recall and Feature Importance MRR — Figure 3



This scatter plot shows the relationship between our chosen metrics in the anomaly detection stage (combined recall from all five cases, at 10% benchmark) and financial importance stage (MRR truncated at $k = 5$ features), grouped by anomaly cases and methods.

As for Isolation Forest, we test AWS and DIFFI methods, and AWS provides better MRRs on all cases aside from the irregular Case 4. DIFFI aims to account for interacting variables and distributes importance scores to all variables along decision tree paths. Presented with 78 features in this dataset, the expected number of paths precisely containing both involved features is at most $1{,}024 \cdot C(\log_2(1{,}024), 2) / C(78, 2) \approx 15.3$, which may not suffice for the top-5 feature ranking to converge. Since AWS attributes its score to the imbalance of each split in a more univariate fashion, it can reliably uncover one of the involved features. As a result, AWS reports higher-ranked features more decisively than DIFFI does as reflected by the higher frequencies (in parentheses) in Table 2.

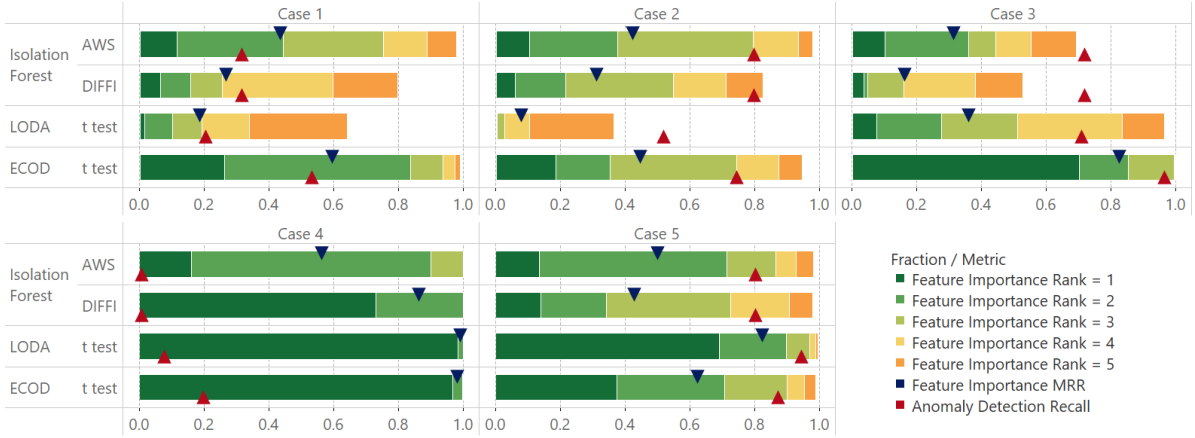
Distribution of Anomalous Features' Ranks

We now explore the actual ranks of the anomalous features. Figure 4 shows the fraction of detected anomalous samples where the higher-ranked anomalous feature is ranked among the top-5 features; the exact ranks are indicated by different color shades. Assuming the ECOD (*t* test) method is chosen, a correct anomalous feature is reported within the top 3 ranks (shades of green), for more than 70% anomalous samples of every case—this coverage should suffice for assisting analysts in further verification during data quality assurance. Adding the 4th and 5th ranks (shades of orange), however, only help explain additional 4.5% and 2.4% of the detected samples (on average among the five cases), respectively.

The distinctive drop in ECOD's performance for Case 2 is likely caused by Case 2 consisting of two subcases, 2(a) and 2(b), conditional on loan stage (Table 1), with vastly different characteristics. To support this claim, we draw an analogy from our

argument that the dual condition of Case 2 resonates well with tree-based methods, allowing Isolation Forest to outperform ECOD for anomaly detection. Analogously, AWS fully utilizes the tree structure, and achieves slightly higher (cumulative) recall rates at $k \geq 2$ than those of ECOD's t test. In particular, Stage 2 of Case 2(b) is fairly scarce, and by occupying the middle value (as an ordinal encoding), it obscures itself from being uncovered via ECOD's ECDF-based scoring function, thereby bypassing detection via t test.

Occurrences of Anomaly Features by Feature Importance Rank — Figure 4



This stacked bar chart shows the rank of the first correct (anomalous) feature reported by each method and each case. The width of each bar represents the fraction of correctly detected anomalies at each rank; hence, the x-axis shows the (cumulative) percentage of correctly explained anomalies. Additionally, Metrics MRR (for feature importance) and recall (for anomaly detection) are marked by triangles.

5. Conclusion and Further Work

Conclusion

In this paper, the efficiency and interpretability of three state-of-the-art anomaly detection models on the granular RDT Credit data are evaluated. Isolation Forest, LODA, and ECOD are selected as candidates, exploring their effectiveness in detecting anomalies; these three models are also equipped with techniques for computing importance scores such as AWS, DIFFI, and hypothesis testing. A test dataset with five types of anomalies is constructed, and the anomaly detection recalls, and the feature importance MRRs on the detected cases are measured.

Our results show that LODA performs well on numerical variables, especially for recognizing linear relationships, but struggles with discrete variables. Isolation Forest excels at dealing with conjunctions and categorical variables, but offers mediocre performance otherwise. ECOD is the best of both worlds: achieving good result for all considered cases of anomalies considered, even when mixed variable types are present. Still, these anomaly scoring methods tend to emphasize outlying behavior of multiple individual dimensions rather than their interactions.

For model interpretability, the performance generally follows from the quality of anomaly detection, with ECOD's t test performs competitively in all cases. In particular, top-3 features are sufficient for explaining more than 70% of the anomalies in every

case. For Isolation Forest, AWS outperforms DIFFI as it assigns more accurate and decisive scores to the relevant features according to our experiment. All methods are generally more effective for numerical features than for categorical features, as the latter involve more complex encoding schemes.

We conclude that the methods we tested are promising for anomaly detection in the RDT Credit data, especially ECOD with random projections. The models can detect various types of anomalies and provide interpretable explanations for the detected cases.

Further Work

Our eventual goal is to deploy ECOD for the full RDT Credit data in our production environment, implemented on-premises under Apache platform (Spark or Hadoop). We demonstrate the scalability of the proposed algorithm by computing the ECOD anomaly scores on the constructed dataset (13.5M rows and 78 features) projected to 1,024 dimensions. Our SQL query script completes the ECOD process via Impala in approximately 5 hours and requires auxiliary storage only proportional to the number of data points. We further test against the entire 2024Q1 data (71.2M rows and 100 selected features), and the running time scales roughly linearly with the data size—training and inferencing based on 3-month backward data is achievable, albeit requiring more than a day to complete. The bottleneck of this implementation is the computation of ECDFs that requires sorting data points in each of the projected dimension; if approximate scores are acceptable, then distributed computing should allow for a much faster running time via random sampling and approximate percentile computations. Overall, these methods require more research and experimentation on their scalability before we deploy the complete pipeline to the full dataset, as the enormous volume of the data may still pose some challenges.

Considering the limitations faced by distribution-based detection methods, we are currently exploring alternatives. One promising approach so far is to automatically generate validation rules given templates, such as "X is not null", "$X \leq Y$" and "if $C = C_1$ then <another predicate>". These are rarely correct rules, but we filter for plausible ones via heuristics based on the dataset, then have them confirmed by analysts. This method discovers Cases 1-2 among many other errors, but the precision is low—most rules are rejected for their lack of business interpretation. This method can be useful for periodic revision of validation rules, which can be applied during data acquisition with relatively low computation cost.

In addition to these computational challenges, there are also some limitations in computing feature importance scores for categorical variables and some extremely rare anomalies. We would like to alleviate these issues by refining our encoding schema, evaluating our method on future occurrences of these rare anomalies, or exploring new ways to interpret the flagged anomalies more effectively. It is also worth noting that RDT is still a relatively new dataset with a lot of room for exploration. We are interested in investigating how our method can be applied and adapted to the new variables and schemas of RDT or other comparable datasets.

References

Accornero, M., & Boscariol, G. (2021). Machine learning for anomaly detection in datasets. *IFC-Bank of Italy Workshop on "Machine learning in central banking"*. Rome.

Birgé, L., & Rozenholc, Y. (2006). How many bins should be put in a regular histogram. *ESAIM: Probability and Statistics, 10*, 93-104.

Carletti, M., Terzi, M., & Susto, G. A. (2023). Interpretable Anomaly Detection with DIFFI: Depth-based feature importance of Isolation Forest. *Engineering Applications of Artificial Intelligence, 119*, 105730.

Continanza, D. N., del Monaco, A., di Lucido, M., Figoli, D., Maddaloni, P., Filippo, Q., & Giuseppe, T. (2022). Stacking machine-learning models for anomaly detection. *IFC-Bank of Italy Workshop on "Data Science in Central Banking: Applications and tools"*.

Desai, A., Kosse, A., & Sharples, J. (2024). Finding a needle in a haystack: a machine learning framework for anomaly detection in payment systems. *BIS Working Papers, 1188*.

Freedman, D., & Diaconis, P. (1981). On the histogram as a density estimator: L2 theory. *Probability Theory and Related Fields., 57*(4), 453–476.

Goldstein, M., & Dengel, A. (2012). Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm. *KI-2012*.

Kröger, P., Ng, R. T., & Sander, J. (2000). LOF: Identifying Density-Based Local Outliers. *roc. ACM SIGMOD 2000 Int. Conf. On Management of Data*, (pp. 93-104).

Li, Z., Zhao, Y., Hu, X., Botta, N., Ionescu, C., & Chen, G. H. (2023). ECOD: Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions. *IEEE Transactions on Knowledge and Data Engineering*, 12181-12193.

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. *Eighth IEEE International Conference on Data Mining*. Pisa, Italy: IEEE.

Maddaloni, P., Continanza, D. N., Del Monaco, A., Figoli, D., Di Lucido, M., Quarta, F., & Turturiello, G. (2022). Stacking machine-learning models for anomaly detection: comparing AnaCredit to other banking datasets. *Bank of Italy Occasional Paper, 689*.

Manzoor, E., Lamba, H., & Akoglu, L. (2018). xStream: Outlier Detection in Feature-Evolving Data Streams. *KDD '18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (pp. 1963 - 1972).

Pevný, T. (2016). Loda: Lightweight on-line detector of anomalies. *Machine Learning*(102), 275–304.

Ramaswamy, S., Rastogi, R., & Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. *ACM SIGMOD Record, 29*(2), 427 - 438.

Scott, D. (2009). Sturges' rule. *WIREs Computational Statistics., 1*(3), 303-306.

Tan, S. C., Ting, K. M., & Liu, T. (2011). Fast anomaly detection for streaming data. *IJCAI Proceedings - International Joint Conference on Artificial Intelligence*, (pp. 1511–1516).

Vercruyssen, V., Gautrais, C., & Sobha Kartha, N. (2021). Why Are You Weird? Infusing Interpretability in Isolation Forest for Anomaly Detection. *Proceedings for the Explainable Agency in AI Workshop at the 35th AAAI Conference on Artificial Intelligence*, (pp. 51 - 57).

Appendix

Anomaly Scores and Feature Importance

Isolation Forest

Isolation Forest algorithm isolates observations by randomly selecting features and split values. It measures normality based on the average path length from the root node to a terminal node in a forest of random trees. Anomalies are indicated by specific samples that have shorter paths (Liu, Ting, & Zhou, 2008).

Isolation Trees. Multiple instances of random trees are created to isolate data points by recursively partitioning the data.

Anomaly Scores. The anomaly score for the data point x is calculated from the path length $h(x)$ with

$$\text{Anomaly Score: } s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}, \tag{3}$$

where $E(h(x))$ is the average of $h(x)$ from a collection of trees, n is the sample size and $c(n)$ is the average path length of unsuccessful binary tree searches.

Depth-based Isolation Forest Feature Importance (DIFFI) calculates feature importance for individual predictions using the following steps. For each tree in the Isolation Forest, traverse the path from the root to the leaf node where the predicted outlier ends up. For each internal node in the path, increase the feature counter and cumulative importance for each related feature:

$$\text{Feature counter: } C_O^{loc}(f) \leftarrow C_O^{loc}(f) + 1,$$

$$\text{Cumulative importance: } I_O^{loc}(f) \leftarrow I_O^{loc}(f) + \frac{1}{h_t(x_O)} - \frac{1}{h_{\max}},$$

where $h_t(x_O)$ denotes the depth of the leaf node in tree t and $h_{\max}$ is the maximum depth. The subscript O here refers to the outliers and f refers to the feature that the node splits. After evaluating all trees, the feature importance for each sample can be calculated by computing Local Feature Importance (LFI) for each j^{th} feature:

$$\text{DIFFI Feature Importance: } t_j(x) = LFI(f_j) = \frac{I_O^{loc}(f_j)}{C_O^{loc}(f_j)}. \tag{4}$$

This method ensures that the importance of each feature is evaluated based on its contribution to isolating the outlier in the specific path taken (Carletti, Terzi, & Susto, 2023).

Assist-Based Weighting Scheme (AWS) calculates the weight of each attribute contributing to the anomalies by considering how much it shortens the path length of an example in the isolation tree, how imbalance the splits are and the node size that splits occur (Vercruyssen, Gautrais, & Sobha Kartha, 2021). The weight for each attribute is computed using the formula for the score of each relevant node:

$$score = \log_2\left(\frac{|node(x).parent|}{|node(x)|}\right) - 1, \tag{5}$$

where $|node(x)|$ is the size of the node example x ends up in after the split while $|node(x).parent|$ refers to the size of that node's parent. The larger score indicates more effective splits. The feature importance is then computed by summing the scores from all the node related to the j^{th} feature:

$$\text{AWS Feature Importance: } t_j(x) \leftarrow t_j(x) + score.$$

LODA

LODA's approach involves generating multiple sparse random projections and constructing one-dimensional histograms for each projection. These histograms approximate the joint probability distribution of the data, enabling the detection of anomalies by evaluating deviations from this distribution.

Random projection. LODA's projections consist of sparse vectors with $\sqrt{d}$ non-zero features. The non-zero items are randomly selected from $N(0,1)$. This choice allows the L_2 distances between points to be approximately preserved (Pevný, 2016).

Histogram. The projected data from each projection is then fit to a histogram of equal width. The number of histogram bins may be chosen with Birgé and Rozenholc's method (Birgé & Rozenholc, 2006) as in the original study. Alternatively, Freedman-Diaconis Estimator (Freedman & Diaconis, 1981) or Sturge's rule (Scott, 2009) may be used for good all-around performance[6].

Anomaly Scores. The anomaly scores can be calculated for each data point x using the formula:

$$\text{Anomaly Score: } s(x) = -\frac{1}{m}\sum_{i=1}^{m} \log \hat{p}_i(x^{\mathrm{T}} w_i), \tag{6}$$

where m is the number of projections, w_i is i^{th} projection vector and $\hat{p}_i(x^{\mathrm{T}} w_i)$ denotes the probability estimated by the i^{th} histogram. The less likely the data point is an anomaly, the larger the anomaly score.

Feature importance. To explain the cause of an anomaly, we can perform a *t* test between the projections that use and do not use each feature to determine to which degree each feature contributes to the anomaly score. The feature importance can be calculated as

$$\text{Feature Importance: } t_j(x) = \frac{\mu_j - \bar{\mu}_j}{\sqrt{\frac{\sigma_j^2}{|I_j|} + \frac{\bar{\sigma}_j^2}{|\bar{I}_j|}}}, \tag{7}$$

where $I_j/\bar{I}_j$ denote sets of projections that use/do not use j^{th} feature, $\mu_j/\bar{\mu}_j$ is the mean and $\sigma_j^2/\bar{\sigma}_j^2$ is the variance of $-\log \hat{p}_i$ calculated with $i \in I_j$ / $i \in \bar{I}_j$, respectively. The higher feature importance t_j the larger contribution to the anomaly score from the j^{th} feature.

ECOD

Empirical Cumulative Distribution-based Outlier Detection (ECOD) is a novel anomaly detection model that addresses multiple challenges in current frameworks

[6] In our study, we use the minimum bin width between the Sturges' rule and Freedman-Diaconis Estimator as recommended by the NumPy package.

such as curse of dimensionality, limited interpretability of model, and time-consuming hyperparameter tuning. ECOD involves estimating the underlying distribution of each feature in the dataset using Empirical Cumulative Distribution Functions (ECDFs). These underlying distributions are used to calculate the probability of observing a data point as extreme (in the tails) for each dimension. Given the value z of feature j, we can calculate the left- and right-tail probabilities of z as shown in the following equations.

$$F^{(j)}{}_{left}(z) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\left\{X_i^{(j)} \leq z\right\} \quad \forall z \in \mathbb{R} \text{ : left tail ECDF}$$

$$F^{(j)}{}_{right}(z) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\left\{X_i^{(j)} \geq z\right\} \quad \forall z \in \mathbb{R} \text{ : right tail ECDF}$$

Assuming independence between each dimension, ECOD aggregates these tails probability in three ways—left-only, right-only, and auto (skewness-based)—and set outlier score of datapoint X_i (O_i) to be

$$\text{Anomaly Score: } s(x) = \max\left\{O_{left-only}(X_i), O_{right-only}(X_i), O_{auto}(X_i)\right\}, \tag{8}$$

where $O_{left-only}(X_i)$, $O_{right-only}(X_i)$, $O_{auto}(X_i)$ are defined as following

$$O_{left-only}(X_i) = -\sum_{j=1}^{d} \log\left(F_{left}^{(j)}\left(X_i^{(j)}\right)\right),$$

$$O_{right-only}(X_i) = -\sum_{j=1}^{d} \log\left(F_{right}^{(j)}\left(X_i^{(j)}\right)\right),$$

$$O_{auto}(X_i) = -\sum_{j=1}^{d} \mathbf{1}\{\gamma_j < 0\}\log\left(F_{left}^{(j)}\left(X_i^{(j)}\right)\right) + \mathbf{1}\{\gamma_j \geq 0\}\log\left(F_{right}^{(j)}\left(X_i^{(j)}\right)\right).$$

Here γ_j is the sample skewness coefficient for the j^{th} feature's distribution defined as following

$$\gamma_j = \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i^{(j)} - \bar{X}^{(j)})^3}{\left[\frac{1}{n-1}\sum_{i=1}^{n}\left(X_i^{(j)} - \bar{X}^{(j)}\right)^2\right]^{\frac{3}{2}}}, \tag{9}$$

where $\bar{X}^{(j)}$ is the sample mean of j^{th} feature.

Random projection. Sparse Random Projection is applied to the data before ECOD in the same fashion as LODA, since ECOD focuses on calculating anomaly score of each feature independently before summing the score of each feature. This approach may limit the ability to capture complex relationships between variables. Random Project is, therefore, used to project the original features into higher dimensional space (1,024 dimensions) to introduce a new set of features that capture these multivariate relationships more effectively.

Feature Importance. To determine the features that cause anomalies, we employ hypothesis testing as described in LODA. However, instead of using mean and variance of $-\log \hat{p}_i$, we use mean and variance of $\max\{O_{left-only}(X_i),\ O_{right-only}(X_i),\ O_{auto}(X_i)\}$ with the same formula as LODA.

RDT Credit Data Features Selected for our Experiment

Selected Features — Table 3

Original Feature	Abbrev	Cases	Type	Transformation		
				Numerical	Ordinal	One-Hot
Loan and Contingent Type	-		Categorical	-	-	27
Asset and Contingent Class	Stage	1 & 2		-	1 *Z*-score	3
Business Firm Size	Firm Size	3				2
Credit Line Amount	Credit Line	3	Currency (THB)	1 *Z*-score of log	-	2
Contract Amount	-					
Outstanding Amount	-					
Total Pledge Amount	Pledge	5				
Total Collateral Valuation Amount	Valuation	5				
Days past Due	DPD	2	Integer	1 *Z*-score	-	2
Contract Date	-		Date			
Effective Date	-					
Maturity Date	-					
Probability of Default	PD	1	Float (in percent)			
Original Effective Interest Rate	-					
Current Effective Interest Rate	-					
Utilization Ratio	-	4	Float (actual value)	-	1 *Z*-score	4
Revolving Loan Flag	Revolving Flag	4	Boolean	-	-	3

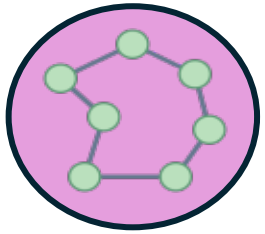
The list of features from the original dataset, along with their roles in the experiment, and the transformation during feature engineering. One-hot features are mostly "is numerical/ordinal", "is null", and actual category/flag values. Asset and Contingent Class, which employs ordinal encoding, has an extra "is miscellaneous" one-hot variable. Utilization Ratio is an exception: it is not one of the original features, but is computed as Outstanding Amount / Credit Line Amount; its one-hot variables additionally include "is infinite" (Credit Line Amount = 0 but Outstanding Amount > 0) and "is undefined" (both Amounts are 0).

A Scalable, Explainable Machine Learning Approach for Granular-Level Credit Dataset's Quality Assurance

Anak Yodpinyanee, Peranut Nimitsurachat,
Nontawit Cheewaruangroj, Supachai Saengthong

Regulatory Data Transformation (RDT)

A transformative redesign from static Data Management System to **granular-level** regulatory reporting standard for **credit data**



Include variables across many dimensions of credit data such as credit lines, interest rate plan, and outstanding amount

Multi-faceted data architecture with specialized cubes for diverse and thorough credit data analysis

Traditional validation framework ensuring quality through format/range check, referential integrity check, and data consistency check

This project aims to enhance RDT validation framework with **scalable** model that will **accurately** detect **subtle anomalies** and **interpret results** effectively

Model Selection

Three requirements

Accuracy

Accurately identify anomalies of any types among all existing anomalies (high recall)

Scalability

Scalable on our production stack (Spark) without consuming excessive computational resources

Explanability

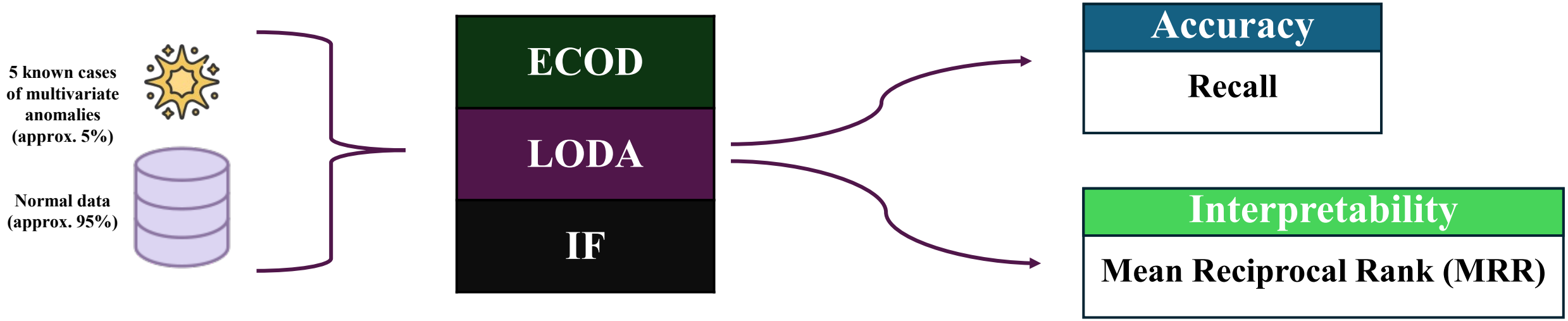
Accurately identify and rank the top fields that cause anomalies

Model Selection

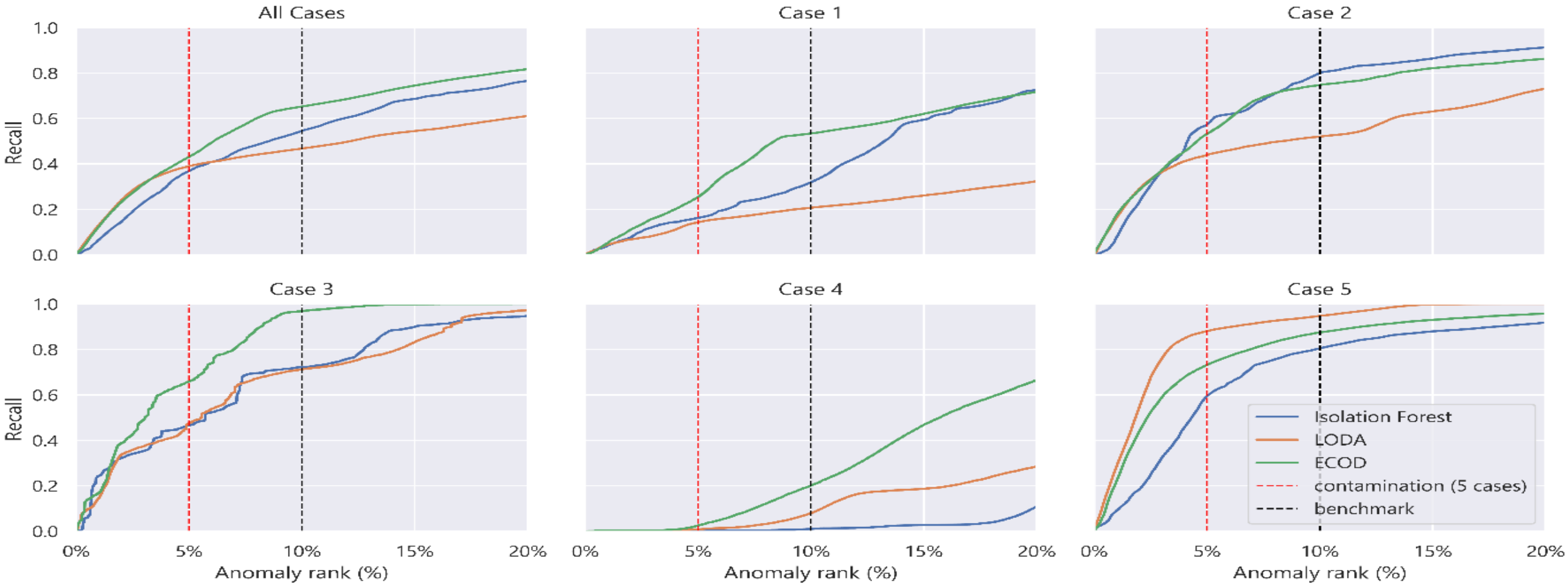
	Model Summary	Strength	Weakness
ECOD	• Estimate tail probabilities for each dimension using empirical cumulative distribution • Compute anomaly score by aggregating probabilities from all dimensions	• Interpretability • Simplicity of the algorithm • Fast computation time • High accuracy compared to other SOTA models	• Designed for linear relationship among variables • Feature independence assumption
LODA	• Use random projection and histrogramming to detect anomalies • Anomaly score is computed from the bin's sparsity across multiple projections	• Interpretability • Simplicity of the algorithm • Fast computation time	• Designed for linear relationship among variables
Isolation Forest	• Detect anomalies by randomly selecting features and splitting values and construct binary trees • Compute scores by considering average path length from root to anomalies	• More robust to non-linear relationships among variables • Relatively fast compared to other ML anomaly detection models	• Scalability issue and slower runtime than ECOD and LODA • Interpretability

Model Evaluation

Case	Scope Condition	Anomaly Condition	Normality Condition	Fraction
1	Stage = Non-Performing	PD ≤ 50%	PD ≥ 80%	1.770%
2(a)	Stage = Performing	DPD ≥ 40	DPD ≤ 30	0.967%
2(b)	Stage = Under-Performing	DPD ≥ 100	DPD ≤ 90	0.411%
3	Firm Size = Small or Micro	Credit Line ≥ 300M THB	Credit Line ≤ 100M THB	0.208%
4	Revolving Flag = False	Utilization Ratio ≥ 1.5	Utilization Ratio ≤ 1.2	0.520%
5	Any	Pledge / Valuation ≥ 2	Pledge / Valuation ≤ 1	1.136%



Recall Rates for Anomaly Detection



This chart shows the recall rates of anomaly detection algorithms, for all five cases and for each individual case. The x-axis represents the fraction of total data points reported, and the y-axis represents the fraction of anomalous samples uncovered by the reported samples.

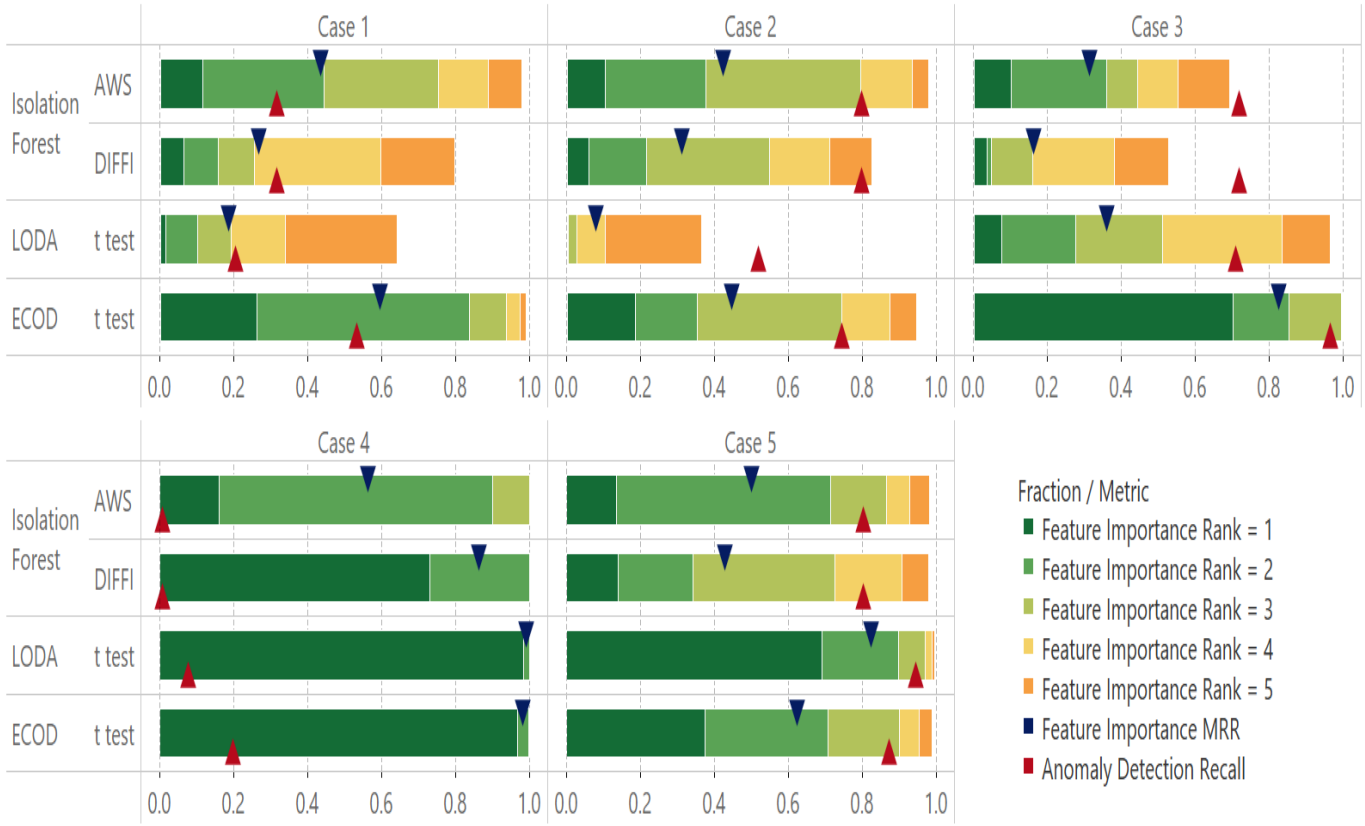
Interpretability & Feature Importance

Mean Reciprocal Ranks

Case	Isolation Forest		LODA	ECOD
	AWS	DIFFI	t test	
1	0.436	0.271	0.188	**0.597**
2	0.425	0.313	0.082	**0.448**
3	0.315	0.166	0.363	**0.827**
4	0.564	0.865	**0.993**	0.984
5	0.502	0.430	**0.825**	0.626

For each cell, MRR (truncated at k = 5 features) is reported.

Occurrences of Anomaly Features by Feature Importance Rank



This stacked bar chart displays the rank of the first correct anomaly detected by each method and case. Bar width indicates the fraction of correctly detected anomalies at each rank; x-axis shows the cumulative percentage of correctly explained anomalies. Metrics MRR and recall are marked by triangles.

Discussion & Conclusion

Model Performance

- LODA excels at recognizing linear patterns in numerical data but has difficulty with discrete variables
- Isolation Forest excels at dealing with conjunctions and categorical variables, but offers mediocre performance otherwise.
- ECOD performs well across diverse anomaly types and mixed variable types.

Model Interpretability

- ECOD's t-test offers competitive model interpretability, aligning with its strong anomaly detection performance.
- Top-3 features explain over 70% of anomalies in all cases. For Isolation Forest, AWS outperforms DIFFI in feature scoring accuracy.
- All methods are more effective for numerical features than categorical ones, due to simpler encoding.