Assessing Properties of Explanations for Anti-Money Laundering Machine Learning Models with Domain Experts

Albert Qi
Harvard College
Cambridge, Massachusetts, USA
albertqi@college.harvard.edu

Steve Dalla
Harvard College
Cambridge, Massachusetts, USA
stevedalla@college.harvard.edu

ABSTRACT

When a bank receives a transaction, how can they be certain whether or not that transaction is legitimate? We want to examine machine learning (ML) models for anti-money laundering (AML) that detect these fraudulent transactions. In such a high-stakes area, it is paramount that humans are able to understand and interpret the predictions from these models via explanations from interpretability tools. However, how should we navigate the trade-offs between certain properties of such explanations? Specifically, what properties of explanations do AML domain experts value the most for AML machine learning models?

To answer these questions, we first train an XGBoost model on the Bank Account Fraud (BAF) dataset, resulting in a TPR of 0.51, an FPR of 0.05, and an AUC of 0.88. Then, we generate initial explanations for our model via SHAP and analyze their fidelity, robustness, compactness, and homogeneity. We then meet with AML domain experts in order to determine what they prioritize throughout the AML process and how that relates to our explanation properties. After both adding regularization to our model and utilizing smoothed SHAP for our explanations, we see a slight decrease in model performance but a drastic improvement to our metrics across the board.

All source code can be accessed at https://github.com/albertqi/aml-xai.

1 INTRODUCTION

Money laundering is inevitable in a financial system. There will always exist someone who tries to take advantage of the system in order to hide money obtained through illegal activities such as drugs, theft, or human trafficking. The process follows three stages: placement, layering, and integration. Placement is the act of injecting illegitimate funds into the financial system. This is often done by breaking down a large sum of money into smaller portions, each of which is then deposited into some bank. Layering is obscuring the origin of the illegal funds by making transfers across multiple accounts or across jurisdictions in order to add as much confusion as possible. Finally, integration is the attempt to take this money and re-enter it into the economy through purchasing legitimate assets such as investments in real estate. Each stage successively obscures the origins of the illegal money, making detection of fraud increasingly difficult.

To combat money laundering, many corporations are obligated to implement some anti-money laundering (AML) practices. Effective AML practices not only protect financial institutions from fraud but also disrupt crimes like drug trafficking and terrorism, which are usually funded fraudulently. Regulatory bodies like the Financial Crimes Enforcement Network (FinCEN) enforce compliance standards to ensure that institutions can identify and report suspicious activities, thus preserving economic stability and public trust.

But, what exactly does the AML process look like? While it may vary from company to company, a common day-to-day life of an AML domain expert may look like the following:

(1) A transaction (e.g., money transfer or purchase with credit card) enters the system.
(2) The transaction is passed into an ML model, and the model predicts whether the transaction is fraudulent or not.

(3) Domain experts analyze any flagged transactions, looking for patterns such as an abnormally high-frequency of transactions or transactions that are unusually high for a customer's stated income.
(4) If an expert deems a transaction to be fraudulent, then they must file a suspicious activity report (SAR) to FinCEN or other authorities.

Explanations for these ML models generated by interpretability tools significantly enhance this workflow by clarifying the reasons behind flagged transactions. They help experts pinpoint which factors might contribute to a specific flagged instance, allowing them to streamline their analysis and accurately assess risk. This is especially valuable in prioritizing cases; clear, interpretable explanations allow AML professionals to focus on high-risk activities while minimizing time spent on potential false positives. As a result, explanations facilitate more efficient, targeted investigations and help AML experts respond promptly to regulatory demands, ensuring that each flagged activity is assessed with precision and accountability.

In this paper, we hope to optimize these explanations by determining which properties are the most important specifically for AML. For example, how should we balance the fidelity and the robustness of an explanation? What properties (i.e., fidelity, robustness, compactness, homogeneity) of explanations do AML domain experts value the most for AML machine learning models? These are some important questions to ask, and answering them should greatly help domain experts throughout the AML process.

To address these issues, we first look at the Bank Account Fraud (BAF) dataset [1], which contains a total of 1 million transactions; each transaction is either fraudulent or legitimate. We then train an XGBoost model on the BAF dataset, reaching a true positive rate (TPR) of 0.51, a false positive rate (FPR) of 0.05, and an area under the curve (AUC) of 0.88. Then, we use Shapley additive explanations (SHAP) [2] in order to generate explanations for our model and analyze them through metrics of fidelity, robustness, compactness, and homogeneity. This gives us an initial baseline of how good or bad our AML explanations may be. To improve our explanations, we then meet with three AML domain experts and determine what exactly they prioritize in their day-to-day life and how we can tailor our explanations towards them. With this knowledge, we add regularization and apply smoothing techniques, ultimately improving our explanations across every single metric.

Finally, while we have used the two terms rather loosely up until now, note that fraud and money laundering have separate meanings. Fraud involves deception or exploitation for financial gain, whereas money laundering involves concealing illegally obtained funds. In many cases, fraud might correspond to a single transaction, while AML involves building a legal case for an individual across multiple transactions. However, fraud and money laundering almost always go hand-in-hand (e.g., a fraudulent transaction can be an indicator of money laundering), so we will often relate the two together.

Overall, we make the following contributions within this paper:

(1) We highlight the need to optimize ML explanations in the field of anti-money laundering.
(2) We analyze the quality of AML explanations generated through SHAP with no optimizations applied at all.
(3) We meet with domain experts in order to determine what they prioritize in the AML process.
(4) We utilize both regularization and smoothing techniques in order to tailor our explanations to the requirements of the domain experts.

2 BACKGROUND

We begin by describing the model and the interpretability methods that we use as well as some of the existing literature around AML.

2.1 XGBoost

Data for anti-money laundering is very imbalanced. AML experts are typically sifting through hundreds if not thousands of entries in search of just one fraudulent case. Due to this imbalanced nature of AML data, we decide to use Extreme Gradient Boosting (XGBoost) as our model for predicting instances of fraud. XGBoost utilizes an ensemble of decision trees in making its predictions, where each tree focuses on correcting the mistakes made by the previous trees and the final prediction is a weighted combination of all the trees. Furthermore, XGBoost adds a regularization term to its objective function, ensuring that our model will not overfit and will properly weigh the instances of legitimate transactions and fraudulent transactions. Specifically, our objective function is as follows:

$$\mathcal{L} = Loss + Regularization$$

Moreover, we use log loss for our loss function:

$$Loss = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)]$$

Here, n represents the number of transactions, y_i represents the true label of transaction i (i.e., fraudulent or legitimate), and $\hat{y}_i$ represents the predicted probability of transaction i being fraudulent. For each iteration, the model attempts to adjust itself in order to reduce this loss.

Then, for our regularization function, we use the following equation:

$$Regularization = \gamma T + \frac{1}{2}\lambda\sum_{j} w_j^2$$

Here, T represents the number of leaves in a tree, w_j represents the weight assigned to each leaf, γ represents the penalty for adding more leaves, and λ represents the penalty for large leaf weights. In the context of this paper, this means that when our model identifies a pattern signifying fraud in the training data, it avoids creating an overly specific rule set that might apply only to the training data. This regularization thus encourages the model to create more generalized rules for patterns that it detects so that they are more likely to apply to future unseen data. Additionally, the regularization helps with explainability as it simplifies the model and allows it to focus on meaningful patterns in the data.

2.2 Shapley Additive Explanations

To generate explanations for our model's predictions, we use Shapley additive explanations (SHAP). SHAP is a framework for interpreting machine learning models that is based on the Shapley value formula; in the context of game theory, this calculates how much each "player" in a group contributes to the final result. In the context of this paper, the "players" represent our input features. SHAP will then assign a value to each feature that explains its impact on the model's prediction. The formula used is as follows:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!}\left(v(S \cup \{i\}) - v(S)\right)$$

Here, ϕ_i represents the contribution of feature i, variable S is a subset of features not including i, and $v(S)$ is the expected value of the model prediction based on the subset S, averaging over the possible values of other features. We utilize the `TreeExplainer` implementation of SHAP, which is specifically optimized for tree-based machine learning models (such as gradient-boosted decision trees like XGBoost).

While many other interpretability tools exist, we use SHAP in particular because it provides us with clear and interpretable explanations for attributing our model's predictions to specific input features. Unlike LIME [3], for example, SHAP also accounts for feature interactions and dependencies, and in datasets for AML, this is crucial as there are many features which may be connected to each other (e.g., transaction frequency and amount). Additionally, SHAP offers us properties such as consistency and local accuracy, ensuring reliable explanations that capture feature relationships across all predictions.

2.3 Interpretable ML for AML

It is worth noting that there already exists some work in the literature that uses ML models and interpretability methods to generate explanations for AML. Tertychnyi et al. propose a monitoring system that generates alerts for suspicious transactions, accompanied by interpretations [4]. They generate their explanations through Shapley values and ultimately calculate and evaluate metrics of recall, precision, and the like.

Ultimately, we want to build off of this paper by not only generating explanations for AML machine learning models but by also analyzing the properties of such explanations and determining which are valued the most by domain experts.

3 DESIGN

We begin by performing exploratory data analysis on the BAF dataset and then train an XGBoost model on said dataset. Next, we use SHAP to generate explanations for our model, analyzing certain metrics in order to get a baseline of performance. Afterward, we meet with domain experts to figure out what exactly is most important in the field of AML, and we use these insights to ultimately improve upon our initial explanations.

3.1 Dataset

We use the Bank Account Fraud (BAF) dataset, which is a synthesized dataset containing 1 million instances and 30 realistic features. Each transaction is labeled as either legitimate or fraudulent, and certain techniques (e.g., noise addition) are applied to preserve the privacy of potential applicants. We standardize the dataset and one-hot encode our categorical features, also imputing missing values with the median.

3.2 Model

We train an XGBoost model on the BAF dataset in order to handle the imbalanced nature of the dataset. Because the data is so imbalanced (i.e., we have many more negative labels than positive ones), we mandate that the FPR is at most 0.05. We also use the default L2 regularization value of 1.

3.3 Initial Explanations

We utilize SHAP to generate explanations for our model and calculate their fidelity, robustness, compactness, and homogeneity.

3.3.1 *Fidelity.*

We utilize loss-based-fidelity for fidelity.

$$\frac{1}{|\mathcal{X}|}\sum_{\mathbf{x}\in\mathcal{X}}\mathcal{L}\left(f(\mathbf{x}), E(f)(\mathbf{x})\right)$$

For our purposes, we arbitrarily choose to use the mean squared error as our loss function.

3.3.2 *Robustness.*

We utilize local-stability for robustness.

$$\max_{||\mathbf{x}'-\mathbf{x}||\leq r}\frac{||E(f,\mathbf{x}')-E(f,\mathbf{x})||}{||\mathbf{x}'-\mathbf{x}||}$$

To find the maximum ratio, we generate 10000 random perturbations $\mathbf{x}'$ that are at most a distance of $r = 0.1$ from our instance $\mathbf{x}$. Additionally, we mandate that the perturbation size is at least some $\epsilon = 0.0001$ or else it is too easy for the ratio to blow up in size.

3.3.3 *Compactness.*

We utilize entropy-complexity for compactness.

$$-\sum_{i=1}^{K} p_E(\mathbf{x})_i \ln\left(p_E(\mathbf{x})_i\right)$$

In order to calculate the overall compactness for the entire dataset, we simply take the mean of the entropy-complexity for all instances $\mathbf{x}$.

3.3.4 *Homogeneity.*

We utilize faithfulness-loss-per-group for homogeneity.

$$\frac{1}{|\mathcal{X}_g|}\sum_{\mathbf{x}\in\mathcal{X}_g}\mathcal{L}(f(\mathbf{x}), E(f,\mathbf{x}))$$

We reuse the fidelity metric in Section 3.3.1 to actually calculate the fidelity for each group.

3.4 Discussion with Domain Experts

We talk with three AML domain experts who have all worked in the finance space for 5-10 years. From our discussions, we get a lot of insight into their approach to AML and what they tend to look for in particular.

Some questions on which we base our discussions include the following:

(1) What does the day-to-day AML process look like for you?
(2) What factors do you consider when identifying whether or not a transaction is legitimate?
(3) Are there any features that often tend to raise suspicion, indicating a transaction might be fraudulent?
(4) What role does human expertise play in detecting money laundering? What is the current interaction between humans and machines in the field of AML?

With these initial set of questions, we are able to spark a meaningful conversation with our domain experts and learn more about AML.

3.5 Improving Our Explanations

After our discussion with AML domain experts, we update our model by adding an L1 regularization of 1 and decreasing the learning rate from 0.3 to 0.01. Additionally, we utilize smoothed SHAP by adding small random noise to the input and averaging the resulting SHAP values a total of 10 times. We then re-evaluate our explanations via the same metrics mentioned in Section 3.3 in order to see how much our explanations have improved.

4 EVALUATION

We begin by evaluating our XGBoost model on the BAF dataset. Then, we calculate the properties of our SHAP explanations for said model. Finally, we outline

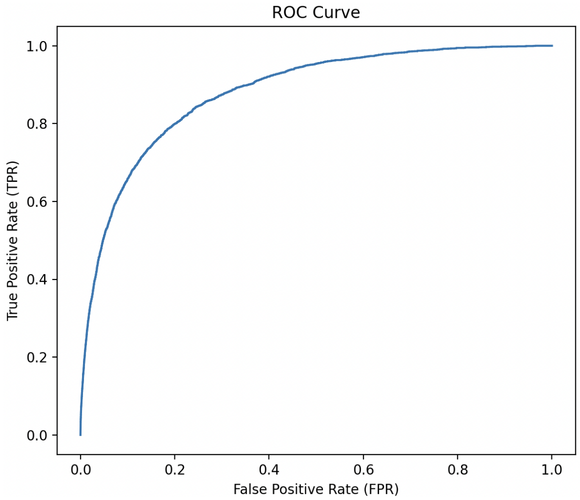

Figure 1: ROC Curve

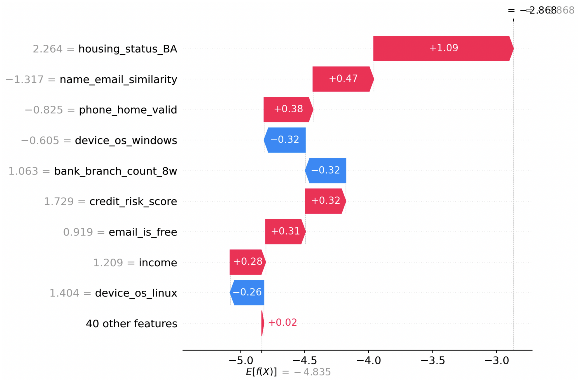

Figure 2: Initial SHAP Explanation

our takeaways from our discussion with AML domain experts and evaluate our updated explanations.

4.1 Experimental Setup

We run all of our tests with excess memory on an 8-core GPU that is integrated into the Apple M1 chip. Training and testing data are split at a 3:1 ratio.

The primary evaluation metrics that we use for the model are TPR, FPR, and AUC. For our explanations, we use the metrics of fidelity, robustness, compactness, and homogeneity that were previously mentioned in Section 3.3.

4.2 Model

We train an XGBoost model on the BAF dataset, resulting in a TPR of 0.51, an FPR of 0.05, and an AUC of 0.88. This is displayed in Table 1. Again, recall that we require the FPR to be at most 0.05. Ultimately, we see that the TPR is decent at above 0.5, while the FPR and AUC are both pretty good (i.e., low and high, respectively).

Figure 1 displays the ROC curve for the XGBoost model.

4.3 Initial Explanations

Using SHAP to generate our explanations, we are able to see what features seem to lead to a positive or negative prediction. Figure 2 illustrates the features that were most impactful for an example of a positive fraud prediction. We can see that a dissimilar name and email as well as an invalid home phone number are some factors that led to a positive prediction. Additionally, we see that housing status seems rather important, but unfortunately, the anonymized nature of the dataset means that we do not know exactly to what type of housing status BA refers. Nonetheless, we hypothesize that BA refers to some form of unstable housing that cannot be easily tracked or verified (and thus cannot be easily linked to an individual).

Moreover, we evaluate our explanations via the metrics outlined in Section 3.3. As shown in Table 2, we get a loss-based fidelity of 4.79, a local stability of 9197.87 on a legitimate transaction, a local stability of 14736.22 on a fraudulent transaction, and an entropy-complexity of 3.07. Additionally, we evaluate homogeneity of faithfulness for income, customer age, and housing status. These results are shown in Tables 3, 4, and 5, respectively.

We can see that fidelity, compactness, and homogeneity seem decent; it is mostly robustness that is alarming. First, the stability ratio is so high (i.e., around the tens of thousands) because our explanation difference $||E(f, \mathbf{x}') - E(f, \mathbf{x})||$ is very high, even via the tiniest perturbations $||\mathbf{x}' - \mathbf{x}||$. This is likely due to the complexity of AML, where every single factor needs to be considered and multiple different patterns can lead to the same prediction. Furthermore, we can see that the stability ratio is almost twice as large for a fraudulent transaction than it is for a legitimate transaction; this makes sense since the dataset is imbalanced and there are so few fraudulent transactions in the first place.

4.4 Discussion with Domain Experts

From our discussion with domain experts, we are able to gather a few insights into what they value throughout their AML process and what that means for our explanations.

First, domain experts need their own assessments of whether a transaction is fraudulent or legitimate to be as accurate as possible. This is unsurprising, as it is paramount that a fraudulent transaction is properly reported and that a legitimate transaction is not falsely reported. From this, we can infer that it is also paramount that our explanations are faithful. Explanations that accurately capture a model's underlying decision process should greatly help domain experts draw accurate conclusions, so as a result, fidelity seems to be the most important property for AML explanations.

Additionally, domain experts aim to catch every instance of fraud, not just the most common cases. This tells us that homogeneity seems to be at least somewhat important. Take debit card count, for example. Transactions from users with many different debit cards (e.g., over 20) are more likely to be fraudulent than transactions from users with only one or two debit cards. However, if our explanation is very heterogeneous, then it may only be faithful in the former case where the debit card count is very high, even though we want high fidelity in both scenarios. Another way to put this is as follows: we know that fidelity is important, and we want fidelity to be high for everyone, not just one subgroup; thus, a homogeneous explanation is preferred over a heterogeneous one.

Moreover, AML domain experts tend to look for consistent patterns that they can identify and use to classify a transaction as fraudulent. For example, FinCEN declares that a barrage of food-related transactions that all occur close to the U.S. border in a short period of time likely indicates human trafficking. Domain experts need to be able to recognize this pattern consistently. Thus, it seems that robustness in our explanations can be helpful, as robust explanations would allow us to generalize an explanation for one input to other similar inputs. Additionally, we know that users tend to trust robust explanations more, and this trust can be crucial in a high-stakes area like AML.

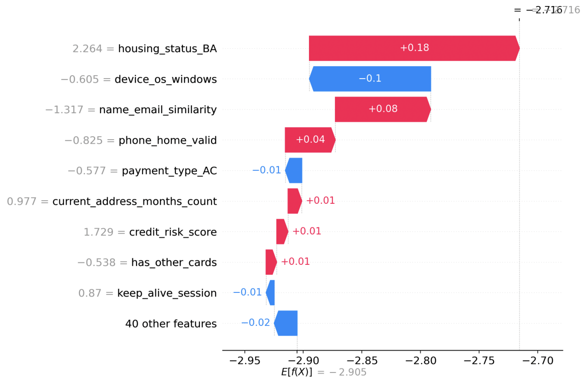

Figure 3: Smoothed SHAP Explanation

Table 1: Evaluation of XGBoost Model

Metric	Original	Regularized
TPR	0.51	0.45
FPR	0.05	0.05
AUC	0.88	0.84

Table 2: Evaluation of Explanations

Metric	SHAP	Smoothed SHAP
Loss-Based Fidelity	4.79	0.012
Local Stability (Legit)	9197.87	85.38
Local Stability (Fraud)	14736.22	555.21
Entropy-Complexity	3.07	2.03

Finally, when looking at whether a transaction is fraudulent or not, domain experts want to use as many resources as possible to get the most accurate result that they can. This indicates to us that compactness is not their highest priority. Of course, explanations still need to be comprehensible to some degree, but ultimately, domain experts seem to be willing to take more time and use more cognitive effort if it results in a more accurate conclusion. Cutting out features may lower our fidelity and not give experts as complete a picture as they require.

4.5 Improving Our Explanations

Our discussion with domain experts indicates to us that fidelity, homogeneity, and robustness are the most important properties. However, when looking at Table 2, we can see that robustness is a very large issue. Thus, in order to try and increase the robustness of our explanations, we add regularization to our model and use smoothed SHAP for our explanation; this helps

Table 3: Homogeneity of Fidelity for Income

Quartile	SHAP Fidelity	Smoothed SHAP Fidelity
1	5.78	0.0099
2	5.35	0.010
3	5.05	0.011
4	4.07	0.013

Table 4: Homogeneity of Fidelity for Customer Age

Quartile	SHAP Fidelity	Smoothed SHAP Fidelity
1	5.31	0.0010
2	4.17	0.013
3	3.76	0.022
4	4.21	0.026

Table 5: Homogeneity of Fidelity for Housing Status

Status	SHAP Fidelity	Smoothed SHAP Fidelity
BA	3.12	0.024
BB	4.67	0.0086
BC	5.51	0.0089
BD	3.94	0.0089
BE	5.79	0.0080
BF	5.39	0.0089
BG	5.24	0.0096

improve the generalizability and consistency of our explanations.

After applying the techniques, we get an XGBoost model with a TPR of 0.45, an FPR of 0.05, and an AUC of 0.84. This is just a slight decrease in performance from our original model. However, this trade-off comes with the added benefit that our explanations perform significantly better across all metrics. This can be seen in Table 2, and an updated explanation can be seen in Figure 3.

Loss-based fidelity decreases by a factor of 399 from 4.79 to 0.012, local stability for a legitimate transaction decreases by a factor of 108 from 9197.87 to 85.38, local stability for a fraudulent transaction decreases by a factor of 27 from 14736.22 to 555.21, and entropy-complexity decreases by a factor of 1.5 from 3.07 to 2.03. Homogeneity of fidelity is also very good, as our explanations seem to be similarly faithful for all subgroups. Thus, we can conclude that while regularization and smoothing techniques may have slightly decreased model performance, the added improvement in the explanations is well worth the trade-off.

5 FUTURE WORK

In the future, we would like to improve stability even further if possible. We noted stability as one of the important features from our talks with experts, but while we are able to reduce our stability metric significantly through methods such as regularization and smoothing, it is still not as low as it could be. Our belief is that the complexity of AML and the imbalanced nature of the BAF dataset are some of the main causes of this. Conducting further experiments to pinpoint the root causes of the high stability metric and identifying effective strategies to address the issue could provide valuable insights for improving the model's performance in future research.

Furthermore, we would like to perform tests on other datasets as well. The BAF dataset that we use is synthesized and anonymized, and while this dataset does have 30 realistic features, real-world AML experts are often dealing with many more. Testing our model on a real dataset that is not anonymized would make for an interesting comparison and could potentially show the usability of our model in a more real-world context. We would also like to see whether certain features are more or less important from what we have currently observed.

Finally, we want to look at how other models or explanation methods might perform as well. While we feel that our model and explanation choices are sound and rooted in attacking AML specifically, it would be interesting to see how other models compare and whether they improve performance or explainability. Looking into random forests or various ensemble methods could provide us with some interesting insights. In regards to using different explanation methods, it would be interesting to analyze the contrast between SHAP and LIME and see how our optimizations might change if our explanation method changes as well.

6 CONCLUSION

Money laundering remains a challenge within financial systems today, obscuring the origins of illicit funds. Anti-money laundering practices have become increasingly important in identifying and mitigating such activities, and with money laundering practices becoming more advanced by the year, the aid of ML models has become essential in staying ahead of fraudulent schemes. These models allow for the identification of suspicious transactions by analyzing vast amounts of data and detecting patterns indicative of money laundering activities. As these models grow in complexity, the need for interpretability becomes paramount to ensure that their predictions are both understandable and actionable by domain experts.

In this paper, we aim to optimize ML explanations for AML applications through tailoring properties of fidelity, robustness, complexity, and homogeneity to the preferences and needs of domain experts. Using the BAF dataset, we train an XGBoost model and generate SHAP explanations. We then hold discussions with domain experts, whose insights give us a prioritization of fidelity, robustness, and homogeneity. Finally, we refine our explanations based on their feedback, optimizing for these three properties through the use of regularization techniques and smoothed SHAP explanations.

Following our process, our initial XGBoost model produces a TPR of 0.51, an FPR of 0.05, and an AUC of 0.88. Our initial explanations give us a loss-based fidelity of 4.79, a local stability of 9197.87 on a legitimate transaction, a local stability of 14736.22 on a fraudulent transaction, an entropy-complexity of 3.07, and a homogeneity of fidelity that displays similar faithfulness across all subgroups. After our discussions and optimizations, we significantly improve our explanation metrics at a slight cost to our XGBoost performance. Our new TPR, FPR and AUC end up at 0.45, 0.05, and 0.84, respectively; our new loss-based fidelity, local stability on legitimate transactions, local stability on fraudulent transactions, and entropy-complexity end up at 0.012, 85.38, 555.21, and 2.03. The homogeneity of fidelity also sees an improvement across the board.

These results demonstrate that tailoring ML explanations to the preferences and priorities of domain experts can lead to significantly improved interpretability metrics, even if it entails a marginal trade-off in model performance. By aligning explanation properties with expert needs, our approach enhances the practical utility of AML models, making them more actionable and trustworthy in real-world applications.

REFERENCES

[1] Sérgio Jesus, José Pombal, Duarte Alves, André Cruz, Pedro Saleiro, Rita P. Ribeiro, João Gama, and Pedro Bizarro. 2022. Turning the Tables: Biased, Imbalanced, Dynamic Tabular Datasets for ML Evaluation. *Advances in Neural Information Processing Systems* (2022).

[2] Scott M. Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. *CoRR* abs/1705.07874 (2017). arXiv:1705.07874 http://arxiv.org/abs/1705.07874

[3] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *CoRR* abs/1602.04938 (2016). arXiv:1602.04938 http://arxiv.org/abs/1602.04938

[4] Pavlo Tertychnyi, Mariia Godgildieva, Marlon Dumas, and Madis Ollikainen. 2022. Time-aware and interpretable predictive monitoring system for Anti-Money Laundering. *Machine Learning with Applications* 8 (2022), 100306. https://doi.org/10.1016/j.mlwa.2022.100306