










Towards Refined Classifications Driven by SHAP Explanations

Yusuf Arslan¹(✉) , Bertrand Lebichot¹ , Kevin Allix¹ , Lisa Veiber¹ ,
Clément Lefebvre², Andrey Boytsov² , Anne Goujon², Tegawendé F. Bissyandé¹ ,
and Jacques Klein¹ 

¹ SnT – University of Luxembourg, Esch-sur-Alzette, Luxembourg
yusuf.arslan@uni.lu

² BGL BNP Paribas, Luxembourg, Luxembourg

Abstract. Machine Learning (ML) models are inherently approximate; as a result, the predictions of an ML model can be wrong. In applications where errors can jeopardize a company’s reputation, human experts often have to manually check the alarms raised by the ML models by hand, as wrong or delayed decisions can have a significant business impact. These experts often use interpretable ML tools for the verification of predictions. However, post-prediction verification is also costly. In this paper, we hypothesize that the outputs of interpretable ML tools, such as SHAP explanations, can be exploited by machine learning techniques to improve classifier performance. By doing so, the cost of the post-prediction analysis can be reduced. To confirm our intuition, we conduct several experiments where we use SHAP explanations directly as new features. In particular, by considering nine datasets, we first compare the performance of these “SHAP features” against traditional “base features” on binary classification tasks. Then, we add a second-step classifier relying on SHAP features, with the goal of reducing false-positive and false-negative results of typical classifiers. We show that SHAP explanations used as SHAP features can help to improve classification performance, especially for false-negative reduction.

Keywords: Interpretable machine learning · SHAP Explanations · Second-step classification

1 Introduction

Machine Learning (ML) is being massively explored to automate a variety of prediction and decision-making processes in various domains. However, the predictions of an ML model can be wrong since ML models are inherently approximate [28]. In the finance sector, for example, when an ML model predicts a transaction is suspicious, it raises an alarm, which can be a true-positive or a false-positive. Such predictions are automatically queued for further manual inspection by financial experts [9]. The existence of these false alarms increases the cost of post-prediction analysis, and wrong or delayed decisions can have a significant business impact [23].

Figure 1 summarizes the key steps of an ML pipeline where a domain expert must intervene to triage ML predictions. In such a setting, reducing the number of false-positive (or false-negative) predictions upstream is paramount in order to reduce the workload of financial experts and increase the customers’ trust in companies. To do so, financial companies often rely on manual interventions by domain experts. It can decrease false-positive rates by several percentage points, but involving domain experts is also costly [35].

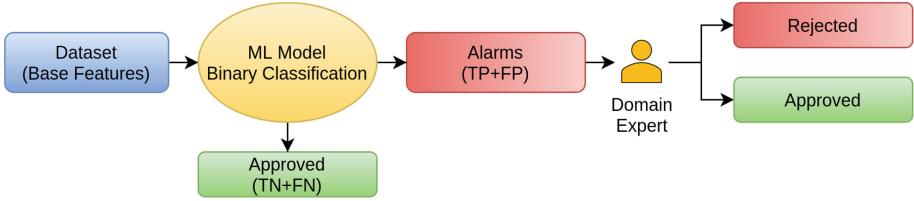


Fig. 1. Financial ML framework with human intervention (TP: true-positive, FP: false-positive, TN: true negative and FN: false negative)

Towards ensuring that domain experts are provided with relevant information for assessing model results, interpretable ML techniques and tools are increasingly leveraged. Among the state-of-the-art interpretable ML approaches, SHAP [20] is a popular technique that is widely used in the literature and by practitioners: its SHAP values, which are derived from SHAP explanations, are computed to evaluate the importance of the contributions of different features on the predictions, potentially enabling the identification of prediction errors as well as providing investigation directions.

Our hypothesis is that if SHAP can help humans to better understand a model, it could also help algorithms. Indeed, if humans are able to leverage information in SHAP explanations, such information may be automatically and systematically exploited in an automated setting. To confirm our intuition, we inspect SHAP values on binary classification tasks. This hypothesis is actually supported by recent works. For instance, [1, 36] show the usage of SHAP explanations by domain experts for the reasoning of case-based scenarios of frauds and anomalies.

To do the first step towards the automatization of the processing of the SHAP explanations, let us see SHAP as a transformation of the learning space. If n_f is the number of features and n_s is the number of samples for a given dataset, SHAP values can be seen as the result of a (nonlinear) transformation f of the learning space: $f : \mathbb{R}^{n_s \times n_f} \rightarrow \mathbb{R}^{n_s \times n_f}$. Indeed, each n_s sample will receive n_f SHAP values. SHAP, for each sample, provides a float per feature, reflecting its contribution to the prediction of that sample. The full set of SHAP features has the same size as the full set of base features. In the rest of this paper, the features obtained through SHAP explanations will be referred to as *SHAP features*, and the original features that were available for the classification will be referred to as *base features*.

The idea is to use this transformation to, hopefully, send the data to a **more separable space** (It seems to be the case for SHAP in practice, as shown in [2]). This

idea is one of the cornerstones of SVM and is widely used in many domains [4, 27, 31]. According to these hypotheses, SHAP values may hold information, to be exploited, for improving the performance of ML classifiers.

This Paper. We present an empirical investigation of our hypothesis by focusing on seven publicly available binary classification datasets and two proprietary datasets from the financial domain. In particular, we study the added value of the information encoded in SHAP explanations compared to the features that were available for the classification. More precisely, we first compare the performance (in terms of accuracy) of a classifier where SHAP explanations are used as features in comparison with a “traditional” classifier relying on base features. We then investigate the feasibility of building a pipeline of cascaded classifiers where the second classifier leverages SHAP explanations to filter out incorrectly classified samples after the initial classifier. In the end, we show that this strategy indeed increases classifier performance.

2 Background and Related Work

The Cost of Being Wrong

Financial institutions are wary of the “cost of being wrong” [3]. This cost is two-fold. First, *bad* decisions—made by a human being or by an automatic system—carry severe risks of financial loss, direct or indirect. Furthermore, in a line of business where *Trust* is of prime importance, any loss in reputation, through scandals or mere negative hearsay, can quickly lead to substantial financial consequences. Second, trying to prevent automatic systems from making *wrong* decisions itself incurs significant costs in the form of increased workload for expensive experts, lack of flexibility arising from the delays needed to have automated decisions vetted by experts, and the massive extra cost of designing systems and processes that provably mitigate the risks of *bad* decisions.

Counter-intuitively, the fear of *bad* decisions may lead some actors to forgo approaches that could be more accurate but do not help analysts vet the decisions. In particular, Deep-Learning—despite its documented prowess—is sometimes deemed inappropriate [34] because it brings nothing to help justify the decisions and no explanations to archive for auditing purposes.

Overall, these costs and risks call for more precise techniques that enable and ease manual inspection and leave exploitable audit trails. Interpretable ML techniques can decrease these costs and risks. [37] uses SHAP explanations for case-based reasoning tasks and reports that the similarity of SHAP explanations is more helpful than the similarity of feature values for domain experts, though finding the most appropriate distance function that shows similarity for a specific dataset is not a fully resolved question. [18] evaluates SHAP and Local Interpretable Model-agnostic Explanations (LIME)¹ [25] to obtain useful information for domain experts and facilitate the FP reduction task. [18] suggests eliminating FPs by employing an ML filter that uses SHAP features instead of base features. According to their findings, the performance of the ML filter using SHAP features is better than the ML model using base features and thus can be leveraged [18].

¹ <https://github.com/marcotcr/lime>.

In the interpretable ML domain, little research has been conducted about the use of explanations to enhance model performance [13]. In the context of this study, we aim at inspecting the effect of SHAP values in a two-step classification pipeline. Two-step cascaded classification has been used for financial applications, as reported by various studies [5, 7, 14, 33]. The idea behind two-step classification is to obtain SHAP features as SHAP values by using a first step classifier and then use a second step classifier with SHAP features to improve classification performance.

We inspect SHAP explanations, which are already investigated for various tasks, including but not limited to clustering [2], rule mining [18], case-based reasoning [37], and feature selection [15], from the classification point of view.

Shapley Values

Shapley Values [29], which guarantee a fair distribution of payout among players, derive from the cooperative game theory domain and have been quite influential in various domains for a long time [24, 30]. Among attribution methods that aim at distributing the prediction scores of a model for the specific input to its base features, the Shapley values method is the one that satisfies the properties of symmetry, dummy, and additivity [21]. These three properties, namely, symmetry (interchangeable players should receive same the pay-offs), dummy (dummy players should receive nothing), and additivity (if the game is separated, so do the pay-offs), can be considered a definition of a fair payout. Recently, Shapley values have been investigated on the interpretable ML domain to solve the fairness issue of feature contribution values [20]. In this study, features are considered as players, and predictions are regarded as pay-offs. This implementation, which is SHAP² [20], shows how to distribute the payout fairly among the features. Besides, SHAP explanations are suitable for the needs of finance actors [10].

Overall, Shapley values can be seen as providing another *representation* of the original data [17] and hence might help an ML algorithm to learn patterns it would not have been able to infer from the raw base features of the datasets.

3 Empirical Datasets

We perform our empirical evaluation by relying on seven publicly available binary classification datasets and two proprietary binary classification datasets from our industrial partner, a major national bank.

The details of the datasets are as follows:

1. The *Adult* dataset³, which is also known as “Census Income”, contains 32 561 samples with 12 categorical and numerical features. The prediction task of the dataset is to find out whether a person makes more than \$50K per year or not.
2. The *Bank Marketing* [22] dataset⁴ contains marketing campaigns of a banking institution. It has 44 581 samples with 16 categorical and numerical features. The prediction task of the dataset is to identify whether a customer will subscribe to a term deposit or not.

² <https://github.com/slundberg/shap>.

³ <https://archive.ics.uci.edu/ml/datasets/adult>.

⁴ <https://archive.ics.uci.edu/ml/datasets/Bank+Marketing>.

3. The *Credit Card Fraud* dataset⁵ contains credit card transactions. It has 284 807 samples with 30 numerical features. The prediction task is to identify whether a transaction is non-fraudulent or fraudulent.
4. The *Heloc* (Home equity line of credit) dataset⁶ comes from an explainable machine learning challenge of the FICO company⁷. It contains anonymized Heloc applications of real homeowners. It has 10 459 samples with 23 numerical features. The prediction task of the dataset is to classify the risk performance of an applicant as *good* or *bad*. *Good* means that an applicant made payments within a three-month period in the past two years. *Bad* means that an applicant did not make payments at least one time in the past two years.
5. The *Lending Club* dataset⁸ contains loans made through the Lending Club platform. It has 73 157 samples with 63 categorical and numerical features. The prediction task of the dataset is to identify whether a customer that is requesting a loan will be able to repay it or not.
6. The *Paysim* [19] dataset⁹ is a financial mobile money simulator. It has 6 362 260 samples with 7 categorical and numerical features. The prediction task of the dataset is to identify whether a transaction is non-fraudulent or fraudulent.
7. The *ULB Fraud* [16] dataset¹⁰ contains simulated transactions. It has 32 561 samples with 12 categorical and numerical features. The prediction task of the dataset is to identify fraudulent transactions.
8. *Proprietary-1*: The first proprietary dataset contains transaction records. It contains 29 200 samples with 10 categorical and numerical features. With this dataset, the goal is to classify a transaction as Type-A or Type-B (for confidentiality reasons, we cannot detail Type-A and Type-B). We will use *Proprietary-1* to name this dataset.
9. *Proprietary-2*: The second proprietary dataset contains financial requests. We will use *Proprietary-2* to name this dataset. It contains 389 451 samples with 87 categorical and numerical features. The prediction task is to classify financial requests as Type-A or Type-B (for confidentiality reasons, we cannot detail Type-A and Type-B). We will use *Proprietary-2* to name this dataset.

4 Experiment Setup

4.1 Research Questions

Our study takes form around the question of whether SHAP features (see Sect. 1), derived from SHAP explanations, could be useful to improve the classification performance or not. The intuition behind it is that just like SHAP can help humans, it may be able to help algorithms. More concretely, this study assesses whether the feature transformation induced by SHAP, i.e., the computation of the SHAP features, can be

⁵ <https://www.kaggle.com/mlg-ulb/creditcardfraud>.

⁶ <https://aix360.readthedocs.io/en/latest/datasets.html>.

⁷ <https://community.fico.com/s/explainable-machine-learning-challenge>.

⁸ <https://www.kaggle.com/wordsforthewise/lending-club>.

⁹ <https://www.kaggle.com/ealaxi/paysim1>.

¹⁰ <https://github.com/Fraud-Detection-Handbook>.

exploited by machine learning techniques. To that end, we answer two research questions.

RQ1: Can SHAP features outperform base features in a traditional one-step classification approach in terms of accuracy?

RQ2: If we compare a traditional (one-step) classification approach against a two-step classification approach where the second classifier uses either SHAP or base features, what is the best alternative in terms of accuracy? We will divide our answer in terms of false-positive reduction (RQ2.1) and false-negative reduction (RQ2.2). Through RQ2, we want to assess whether SHAP features used in a two-step approach could help domain experts quickly triage ML decisions, for instance, by reducing the number of false-positive decisions.

4.2 Experiment Process

Training step and Machine Learning Algorithms: The training step and the used algorithms are represented in Fig. 2-a.

- *Base Features, GBC, and MLP:* On each of our nine datasets, we first train two binary classifiers by using the base features that are proposed (cf. Sect. 3). For one classifier, we use Gradient Boosting Classifier (GBC). For the other one, we use Multi Layer Perceptron (MLP). There are two reasons for these choices: (1) GBC is the current state-of-the-art approach for tabular unbalanced classification problems, and MLP is better than tree-based approaches to capture additive structure [11], which is the case for SHAP explanations, (2) we tested various other classifiers (e.g., random forests (RF) and logistic regression (LR)) and GBC & MLP were the best on most of our datasets (results not reported here).
- *SHAP features:* The idea is to use the SHAP explanations as features to train a new classifier with these newly computed SHAP features. In practice, we compute the SHAP features by applying the SHAP explainer on the GBC classifier. Then, we use the obtained SHAP features as inputs of an MLP classifier. Note that we also consider RF, MLP, RL, and GBC, but the best results (not reported here) are achieved with MLP.

The training phase can be seen in Fig. 2-a.

Testing Steps: To answer our research questions, we implement three scenarios depicted in Fig. 2-b, Fig. 2-c, and Fig. 2-d respectively.

- *One Step Binary Classification (Fig. 2-b):* To answer RQ1, we compare our three classifiers - GBC with base features, MLP with base features, and MLP with SHAP features - in traditional binary classification tasks on the nine datasets and tasks described in Sect. 3. We compute the Precision-Recall curve and the ROC curve to compare the results for all decision thresholds.
- *Two-Step Classification – Positive (Fig. 2-c):* To answer RQ2.1, as a first step, we use a GBC classifier with base features. Then, as a second step, we apply two classifiers on the positively classified samples only: one by considering the base

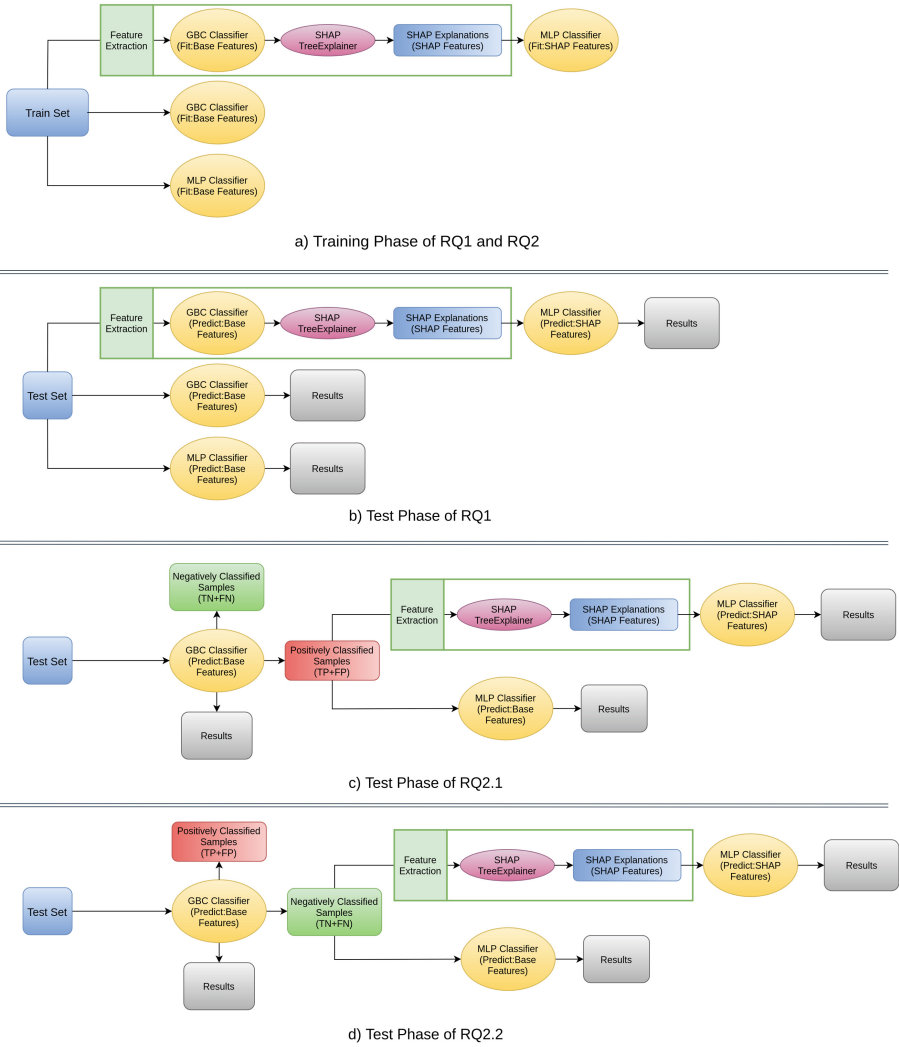


Fig. 2. Experiment process (TP: true-positive, FP: false-positive, TN: true negative and FN: false negative)

features and another one by considering the SHAP features. The number of positively classified samples depends on the decision threshold, which impacts the classification results. A second classification threshold is also considered for the second-step classifier. For each threshold, we test the values $[0.1, 0.2, 0.3, \dots, 0.9]$ to identify the best results. We report the best F1 score and balanced classification rate (BCR) for these thresholds.

- Two-Step Classification – Negative (Fig. 2-d): To answer RQ2.2, we use a similar process as the one for RQ2.1, but we focus on negatively classified samples.

Finally, all our experiments are performed using 5-Fold cross-validation, and are repeated 5 times. The averaged results are then reported.

4.3 Evaluation Metrics

In this paper, we are using the following metrics and tests for evaluation of the results:

Receiver operating characteristic (ROC) curve: True-positive Rate (TPR) is the number of true-positives over the number of true-positives and false-negatives. It shows the performance of models in the prediction of the positive class when the actual outcome is positive. False-positive Rate (FPR) is the number of false-positives over the number of false-positives and true-negatives. It shows the number of positive classifications while the actual outcome is negative. ROC Curve is a visual representation of the trade-off between TPR and FPR [6].

Precision recall (PR) curve: Precision is the number of true-positives over the number of true-positives plus the number of false-positives. Recall is the number of true-positives over the number of true-positives plus the number of false negatives. PR curve shows the trade-off between precision and recall for different thresholds. PR metric evaluates output quality of a classifier. It is used especially in case of class imbalance. High precision implies a low false-positive rate and a high recall implies low false-negative rate. ROC curves are suitable for balanced datasets, whereas PR curves are suitable for imbalanced datasets [26]. We choose these two metrics (ROC and PR curve) since they show the performance of classification models for all thresholds.

F1 score: It is the (balanced) harmonic mean of precision and recall. A high value of F1 score means high classification performance [32].

Balanced classification rate (BCR): All classifiers aim at increasing the sensitivity without sacrificing the specificity [32]. BCR combines sensitivity (TPR) and specificity (1-FPR).

$$\text{Balanced Classification Rate} = \frac{\frac{\text{True Positive}}{(\text{True Positive} + \text{False Negative})} + \frac{\text{True Negative}}{(\text{True Negative} + \text{False Positive})}}{2}$$

5 Results and Discussion

Answers of RQ1: We use ROC Curves and PR Curves to answer this research question. The ROC Curves for each dataset can be seen in Fig. 3, and the PR Curves can be seen in Fig. 4.

According to ROC Curves (and the Area Under Curve - AUC), SHAP features obtain better results than base features for 6 out of 9 datasets.

Similar to ROC Curves, according to PR Curves, SHAP features obtain better results than base features for 5 out of 9 datasets.

One of the interesting findings in our experiments is that the MLP classifier with base features (orange line) is less successful than the MLP classifier with SHAP features (green line). This result can be explained by the fact that SHAP features are “well-separable” as indicated in [36]. More specifically, the data is better separated in the

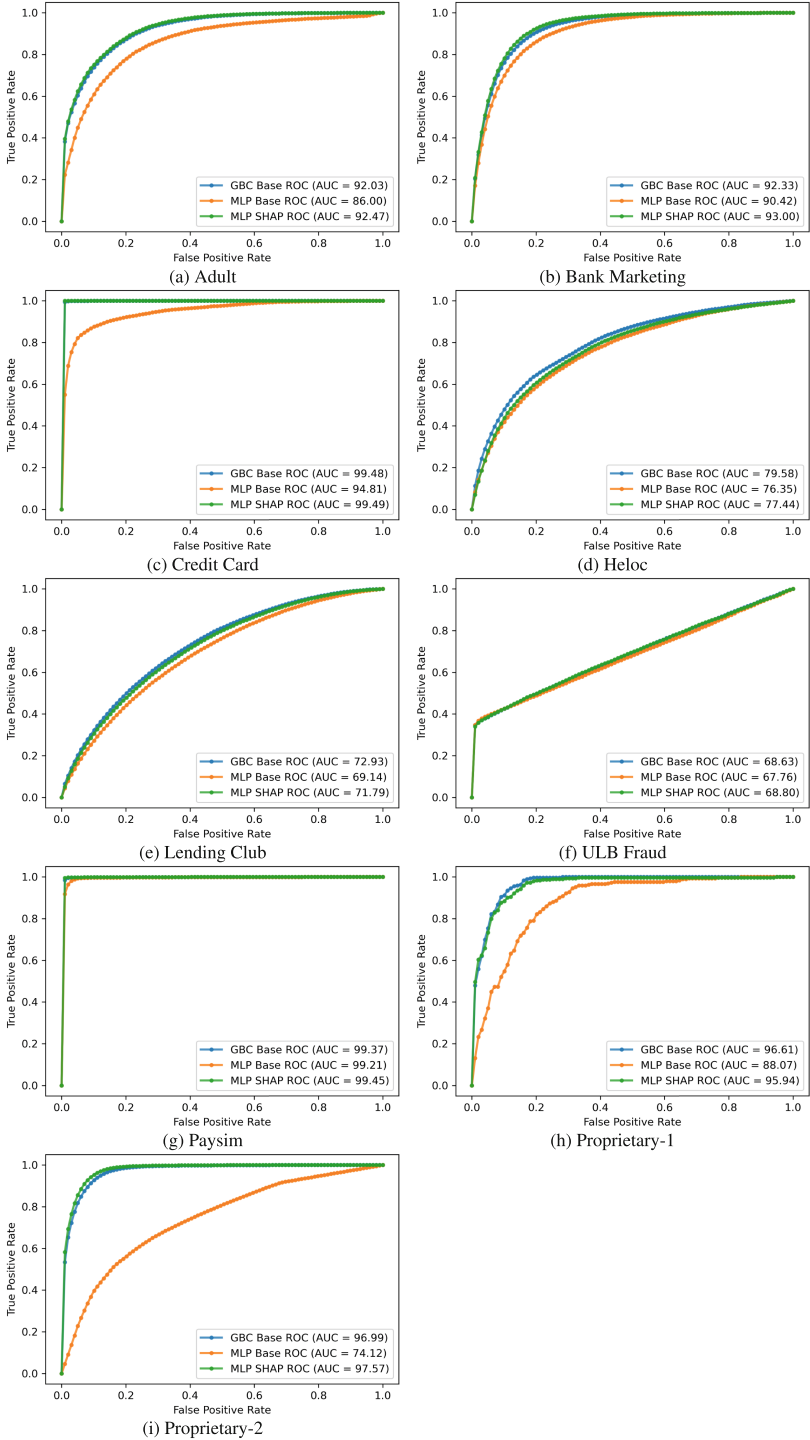


Fig. 3. ROC curves comparison of GBC with base features vs. MLP with base features vs. MLP with SHAP features (5-fold cross validation + repeated 5 times) (Color figure online)

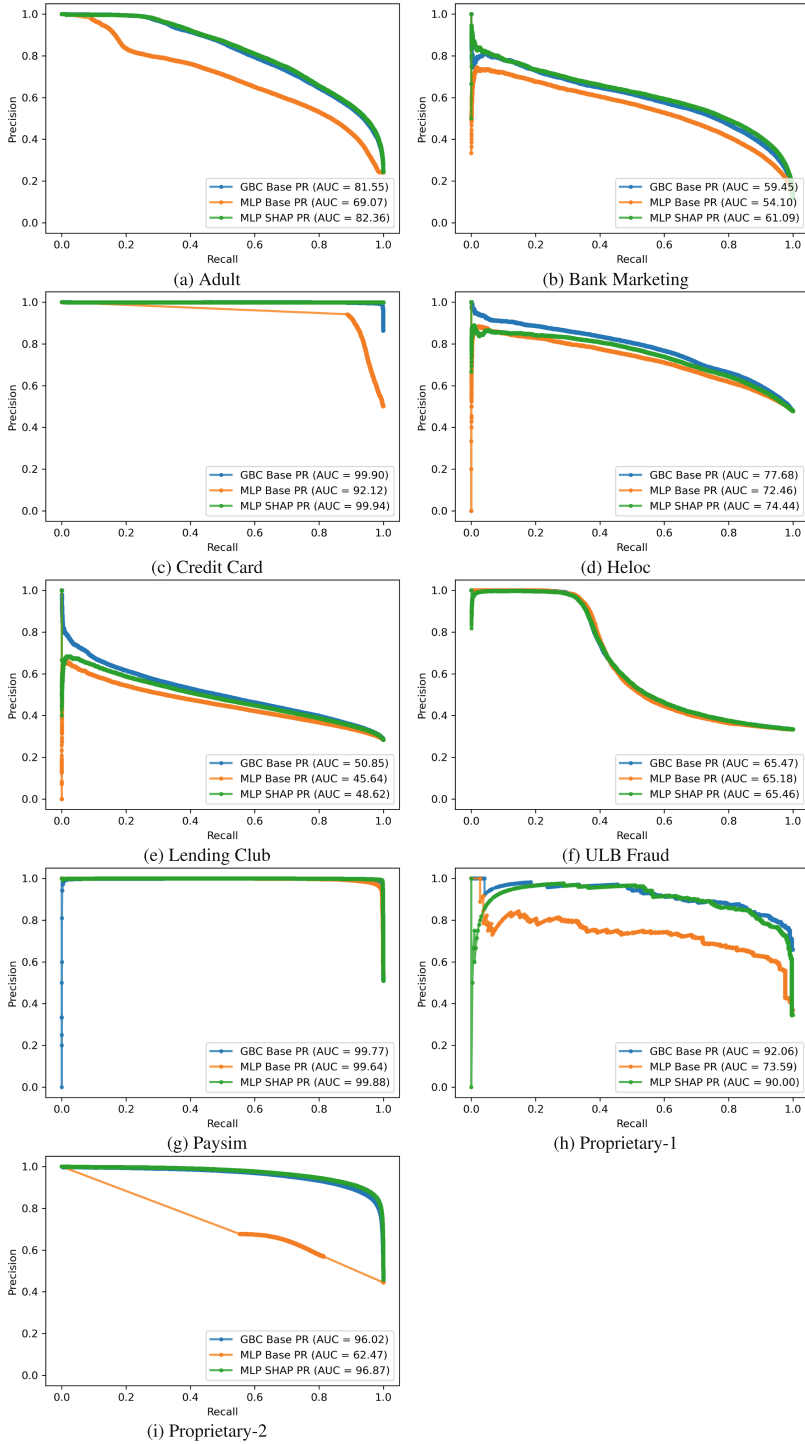


Fig. 4. PR curves comparison of GBC with base features vs. MLP with base features vs. MLP with SHAP features (5-fold cross validation + repeated 5 times) (Color figure online)

SHAP feature space, and an MLP classifier that uses SHAP features can work better in this space. Another reason could be related to the fact that more patterns come to the surface with SHAP features.

Answers of RQ2: We divided our analysis into two sub-research questions, RQ2.1 and RQ2.2, to report our analysis of positively and negatively classified samples, respectively.

RQ2.1 Answer: As described in Fig. 2-c), we compare a classical one-step GBC classifier with two two-step classifiers that focus on positively classified samples only. Both two-step classifiers rely on an MLP classifier, but one uses base features, and the other one uses SHAP features. A comparison of F1 and BCR scores can be seen in Table 1. The two-step (SHAP) obtains the best F1 score for 7 out of 9 datasets and the best BCR score for 6 out of 9 datasets. The one-step classifier obtains the best F1 score for 1 out of 9 datasets and the best BCR score for 2 out of 9 datasets. The two-step (Base) obtains the best F1 and BCR scores for only one out of 9 datasets. According to these findings, the two-step (SHAP) outperforms the other two classifiers on most of the datasets.

We rely on a Friedman/Nemenyi test [8] (with $\alpha = 0.1$) to confirm whether there is a statistically significant difference between the performance scores of the three classifiers. The test concludes that Two-step (SHAP) (The two-step classifier using SHAP features) is significantly better than Two-step (base) (The two-step classifier using base features). However, the test also concludes that there is no significant difference between the one-step classifier and the Two-step (SHAP).

RQ2.2 Answer: As described in Fig. 2-d), we follow an experimental process that is similar to the one used to answer RQ2.1, except that both two-step classifiers focus on negatively classified samples. A comparison of F1 and BCR scores can be seen in Table 2.

Two-step (SHAP) obtains the best F1 and BCR scores for 7 out of 9 datasets. The one-step classifier obtains the best F1 score for 1 out of 9 datasets and never obtains the best BCR score in any of the tested datasets. The two-step (Base) obtains the best F1 score for one out of 9 datasets and the best BCR score for two out of 9 datasets. According to these findings, the two-step (SHAP) obtains the best results overall.

We rely on a Friedman/Nemenyi test (with $\alpha = 0.1$) to confirm whether there is a statistically significant difference among the performance scores of the three classifiers. The test concludes that Two-step (SHAP) is superior to both Two-step (Base) and One-step.

Our findings show that a classifier with SHAP features can be applied to negatively or positively classified samples as a step to improve classification performance.

Comparing RQ2.1 and RQ2.2 Results: Two-step (SHAP) obtains better results on negatively classified samples than on positively classified samples. In all the datasets that we use, class distributions exhibit a slight to severe class imbalance, and positive samples are in minority class. Therefore, there is a relatively small amount of positively classified samples for some datasets. The higher number of negative samples, which means more data for training, can be the reason of better results.

Table 1. Best results of F1 and BCR for different thresholds on positively classified samples (5-fold cross validation + repeated 5 times).

F1 (positive)									
	Adult	Bank	Credit	Heloc	Lending	Paysim	ULB	Prop-1	Prop-2
One-step	71.40	60.04	99.19	72.56	52.93	98.97	52.44	87.65	90.38
Two-step (Base)	69.11	59.79	95.25	72.59	52.57	99.06	52.47	83.48	78.82
Two-step (SHAP)	72.62	61.89	99.91	72.47	53.03	99.31	52.60	87.63	91.61
BCR (positive)									
One-step	83.50	85.36	99.19	71.21	66.49	98.97	66.96	92.38	91.39
Two-step (Base)	81.49	83.85	95.51	72.25	66.28	99.06	67.35	87.98	83.59
Two-step (SHAP)	83.97	85.77	99.91	72.15	66.48	99.31	67.00	92.03	92.56

Table 2. Best results of F1 and BCR for different thresholds for negatively classified samples (5-fold cross validation + repeated 5 times).

F1 (negative)									
	Adult	Bank	Credit	Heloc	Lending	Paysim	ULB	Prop-1	Prop-2
One-step	71.40	60.04	99.19	72.56	52.93	98.97	51.95	87.65	90.38
Two-step (Base)	70.51	60.26	94.96	72.54	53.12	98.85	52.14	86.10	77.91
Two-step (SHAP)	72.62	62.11	99.89	72.55	53.31	99.33	52.13	87.82	91.62
BCR (negative)									
One-step	83.50	85.36	99.19	72.21	66.49	98.97	66.99	92.38	91.39
Two-step(Base)	83.13	85.37	94.05	72.23	66.54	98.84	67.29	91.69	76.56
Two-step(SHAP)	83.97	86.41	99.89	72.14	66.59	99.33	67.00	92.55	92.56

6 Conclusions

In this study, we leverage SHAP features to improve classification performance. Our experiments are performed on seven datasets from the literature and two datasets from our industrial partner. We start by showing that a classifier based on SHAP features can be as efficient as a classifier based on base features. We then show that a second-step classifier, based on the SHAP features, can easily be added to reduce both false-positives and false-negatives.

Our findings are important for several reasons. First, we detect that a classifier based on SHAP features is as powerful as a classifier based on base features. Second, our findings show that domain experts can infer from SHAP explanations comfortably, which is especially important when SHAP explanations offer better visualization. Third, the results reveal that it is possible to improve classification performance by the use of two-step classification.

As future work, we are planning to utilize SHAP explanations for detecting redundant samples in resampling strategies to tackle unbalanced datasets. Besides, it can be an interesting future work to use SHAP explanations in a positive-confidence classifier [12] in which SHAP values for each feature can be used instead of prediction probabilities.

References

1. Antwarg, L., Miller, R.M., Shapira, B., Rokach, L.: Explaining anomalies detected by autoencoders using Shapley Additive Explanations. *Expert Syst. Appl.* **186**, 115736 (2021)
2. Arslan, Y., et al.: On the suitability of SHAP explanations for refining classifications. In: *Proceedings of the 14th International Conference on Agents and Artificial Intelligence (ICAART 2022)* (2022)
3. Bank of England: Machine learning in UK financial services (2019). <https://www.bankofengland.co.uk/-/media/boe/files/report/2019/machine-learning-in-uk-financial-services.pdf>. Accessed Apr 2022
4. Becker, T.E., Robertson, M.M., Vandenberg, R.J.: Nonlinear transformations in organizational research: possible problems and potential solutions. *Organ. Res. Methods* **22**(4), 831–866 (2019)
5. Berger, C., Dohoon, K.: A two-step process for detecting fraud using ADW, oracle machine learning, APEX and oracle analytics cloud (2020). <https://blogs.oracle.com/machinelearning/a-two-step-process-for-detecting-fraud-using-oracle-machine-learning>. Accessed Apr 2022
6. Bradley, A.P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recogn.* **30**(7), 1145–1159 (1997)
7. Darwish, S.M.: A bio-inspired credit card fraud detection model based on user behavior analysis suitable for business management in electronic banking. *J. Ambient Intell. Human. Comput.* **11**, 4873–48871 (2020). <https://doi.org/10.1007/s12652-020-01759-9>
8. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **7**, 1–30 (2006)
9. Ghamizi, S., et al.: Search-based adversarial testing and improvement of constrained credit scoring systems. In: *28th ACM Joint Meeting on ESEC/FSE*, pp. 1089–1100 (2020)
10. Misheva, B.H., Hirsra, A., Osterrieder, J., Kulkarni, O., Lin, S.F.: Explainable AI in credit risk management. *Credit Risk Management*, 1 March 2021
11. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, vol. 2. Springer, New York (2009). <https://doi.org/10.1007/978-0-387-84858-7>
12. Ishida, T., Niu, G., Sugiyama, M.: Binary classification from positive-confidence data. In: *Advances in Neural Information Processing Systems*, vol. 31 (2018)
13. Jia, Y., Frank, E., Pfahringer, B., Bifet, A., Lim, N.: Studying and exploiting the relationship between model accuracy and explanation quality. In: Oliver, N., Pérez-Cruz, F., Kramer, S., Read, J., Lozano, J.A. (eds.) *ECML PKDD 2021*. LNCS (LNAI), vol. 12976, pp. 699–714. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86520-7_43
14. Khormuji, M.K., Bazrafkan, M., Sharifian, M., Mirabedini, S.J., Harounabadi, A.: Credit card fraud detection with a cascade artificial neural network and imperialist competitive algorithm. *IJCA* **96**(25), 1–9 (2014)
15. Komatsu, M., Takada, C., Neshi, C., Unoki, T., Shikida, M.: Feature extraction with SHAP value analysis for student performance evaluation in remote collaboration. In: *2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, pp. 1–5 (2020)
16. Le Borgne, Y.A., Siblini, W., Lebichot, B., Bontempi, G.: *Reproducible Machine Learning for Credit Card Fraud Detection - Practical Handbook*. Université Libre de Bruxelles (2022)
17. Li, R., et al.: Machine learning-based interpretation and visualization of nonlinear interactions in prostate cancer survival. *JCO Clin. Cancer Inform.* **4**, 637–646 (2020)
18. Lin, C.F.: Application-grounded evaluation of predictive model explanation methods. Master's thesis, Eindhoven University of Technology (2018)

19. Lopez-Rojas, E., Elmir, A., Axelsson, S.: PaySim: a financial mobile money simulator for fraud detection. In: 28th European Modeling and Simulation Symposium, EMSS, Larnaca, pp. 249–255. Dime University of Genoa (2016)
20. Lundberg, S.M., Lee, S.L.: A unified approach to interpreting model predictions. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 4768–4777 (2017)
21. Molnar, C.: Interpretable machine learning. Lulu.com (2020)
22. Moro, S., Cortez, P., Rita, P.: A data-driven approach to predict the success of bank telemarketing. *Decis. Support Syst.* **62**, 22–31 (2014)
23. Pascual, A., Marchini, K., Van Dyke, A.: Overcoming false positives: saving the sale and the customer relationship. White paper, Javelin strategy and research reports (2015). Accessed Apr 2022
24. Quigley, J., Walls, L.: Trading reliability targets within a supply chain using Shapley’s value. *Reliab. Eng. Syst. Saf.* **92**(10), 1448–1457 (2007)
25. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should I trust you?: Explaining the predictions of any classifier. In: ACM SIGKDD, pp. 1135–1144 (2016)
26. Saito, T., Rehmsmeier, M.: The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* **10**(3), e0118432 (2015)
27. Shachar, N., et al.: The importance of nonlinear transformations use in medical data analysis. *JMIR Med. Inform.* **6**(2), e27 (2018)
28. Shalev-Shwartz, S., Ben-David, S.: Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, Cambridge (2014)
29. Shapley, L.S.: A value for n-person games. In: Contributions to the Theory of Games, vol. 2, no. 28, pp. 307–317 (1953)
30. Sheng, H., Shi, H., et al.: Research on cost allocation model of telecom infrastructure co-construction based on value Shapley algorithm. *Int. J. Future Gener. Commun. Netw.* **9**(7), 165–172 (2016)
31. Song, C., Liu, F., Huang, Y., Wang, L., Tan, T.: Auto-encoder based data clustering. In: Ruiz-Shulcloper, J., Sanniti di Baja, G. (eds.) CIARP 2013. LNCS, vol. 8258, pp. 117–124. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41822-8_15
32. Tharwat, A.: Classification assessment methods. *New Engl. J. Entrep.* **17**(1), 168–192 (2020). <https://www.emerald.com/insight/content/doi/10.1016/j.aci.2018.08.003/full/html>
33. Thejas, G., Dheeshjith, S., Iyengar, S., Sunitha, N., Badrinath, P.: A hybrid and effective learning approach for click fraud detection. *Mach. Learn. Appl.* **3**, 100016 (2021)
34. Veiber, L., Allix, K., Arslan, Y., Bissyandé, T.F., Klein, J.: Challenges towards production-ready explainable machine learning. In: 2020 USENIX Conference on Operational Machine Learning (OpML 2020) (2020)
35. Wedge, R., Kanter, J.M., Veeramachaneni, K., Rubio, S.M., Perez, S.I.: Solving the false positives problem in fraud prediction using automated feature engineering. In: Brefeld, U., et al. (eds.) ECML PKDD 2018. LNCS (LNAI), vol. 11053, pp. 372–388. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-10997-4_23
36. Weerts, H.J.: Interpretable machine learning as decision support for processing fraud alerts. Ph.D. thesis, Master’s Thesis, Eindhoven University of Technology, 17 May 2019
37. Weerts, H.J., van Ipenburg, W., Pechenizkiy, M.: Case-based reasoning for assisting domain experts in processing fraud alerts of black-box machine learning models. In: KDD Workshop on Anomaly Detection in Finance (KDD-ADF 2019) (2019)